electronics

MDPI

Article

A FL-Based Radio Map Reconstruction Approach for UAV-Aided Wireless Networks

Zhiqiang Tan [1,2], Limin Xiao [1], Xinyi Tang [1,2], Ming Zhao [1] and Yunzhou Li [1,*]

[1] Beijing National Research Center for Information Science and Technology, Beijing 100084, China; tzq21@mails.tsinghua.edu.cn (Z.T.); xiaolm@tsinghua.edu.cn (L.X.); tangxy20@mails.tsinghua.edu.cn (X.T.); zhaoming@tsinghua.edu.cn (M.Z.)
[2] Department of Electronic Engineering, Tsinghua University, Beijing 100084, China
* Correspondence: liyunzhou@tsinghua.edu.cn

Abstract: Radio maps, which can provide metrics for signal strength at any location in a geographic space, are useful for many applications of 6G technologies, including UAV-assisted communication, network planning, and resource allocation. However, current crowd-sourced reconstruction methods necessitate large amounts of privacy-sensitive user data and entail the training of all data with large models, especially in deep learning. This poses a threat to user privacy, reducing the willingness to provide data, and consuming significant server resources, rendering the reconstruction of radio maps on resource-constrained UAVs challenging. To address these limitations, a self-supervised federated learning model called RadioSRCNet is proposed. The model utilizes a super-resolution (SR)-based network and feedback training strategy to predict the pathloss for continuous positioning. In our proposition, users retain the original data locally for training, acting as clients, while the UAV functions as a server to aggregate non-sensitive data for radio map reconstruction in a federated learning (FL) manner. We have employed a feedback training strategy to accelerate convergence and alleviate training difficulty. In addition, we have introduced an arbitrary position prediction (APP) module to decrease resource consumption in clients. This innovative module struck a balance between spatial resolution and computational complexity. Our experimental results highlight the superiority of our proposed framework, as our model achieves higher accuracy while incurring less communication overheads in a computationally and storage-efficient manner as compared to other deep learning methods.

Keywords: radio map; federated learning; deep learning; UAV

Citation: Tan, Z.; Xiao, L.; Tang, X.; Zhao, M.; Li, Y. A FL-Based Radio Map Reconstruction Approach for UAV-Aided Wireless Networks. *Electronics* **2023**, *12*, 2817. https://doi.org/10.3390/electronics12132817

Academic Editor: Nurul I. Sarkar

Received: 30 May 2023
Revised: 19 June 2023
Accepted: 22 June 2023
Published: 26 June 2023

1. Introduction

The integration of UAVs in 6G wireless cellular communication networks has become an increasingly popular topic of research. In dense urban areas where wireless signals may be obstructed by buildings, UAVs can be utilized to carry wireless relays and enhance the link quality between base stations and users with high demand for data [1]. For example, UAVs can be deployed in a flexible manner to cover broadband gaps on the ocean that cannot be addressed by conventional shore-based Transmitter Base Stations (TBSs) or marine satellites [2–4]. Compared to ground-to-ground links, aerial-to-ground links with UAVs can provide greater line-of-sight probability [5]. To maximize the potential benefits of UAVs, researchers are studying optimal placement and trajectory design. Specifically, UAVs can be used as flying base stations to mitigate the effects of significant channel fading and provide enhanced coverage to ground users [6–8]. On the other hand, UAV mobility can enable them to fly over the served ground users, leading to improved energy efficiency and throughput by designing efficient trajectories [9–11].

However, various works have relied on the assumption of ideal pathloss models [12] or a statistical model for line-of-sight probability. These models have been criticized as inadequate for practical scenarios. As an alternative, the use of radio maps has been

recommended for achieving a more realistic design in recent literature. Radio maps are essential in accurately presenting channel qualities at every location between the transmitter (TX) and receiver (RX). These maps play a significant role in placement design [1,13], path planning [14], and other UAV applications [15].

1.1. Related Works

The radio map is the distribution of pathloss, which is a quantity that measures the loss of signal strength between a transmitter and a receiver with respect to geographic location. The spatial distribution of pathloss is affected by transmitter locations, building distribution, signal frequency, and other factors [16]. Many other important wireless communication applications also explicitly rely on the knowledge of the radio map, such as localization [17], physical-layer security [18], and frequency reuse [19]. Thus, the reconstruction of the radio map is a crucial task. However, the complex and realistic physical environment make it difficult to estimate path loss accurately.

Many approaches for estimating pathloss have been proposed in the literature. Most of them can be grouped into two categories, radio propagation models and data-driven models. The former have been proposed firstly to tackle the problem of predicting pathloss, which includes empirical models [20], ray-tracing models [21,22] and dominant path models [23] to name a few. The propagation-based approaches often assume a certain type of propagation. However, this general assumption may not always be accurate in specific areas. The latter includes traditional interpolation algorithms and recent developed deep learning models. Classical interpolation techniques, such as kriging [24], Gaussian process regression [25] and radial basis function [26], are fast and simple. However, they commonly rely only on distance information in the measured data and thus may not accurately predict the complex and heterogeneous spatial patterns often observed in real-world radio signals [27]. Recently, data-driven models, particularly deep learning, have received increasing attention due to the inherent limitations of radio propagation models that lack flexibility in adapting to various complex propagation environments. Deep learning is considered the state-of-the-art approach in this field [28]. The authors in [29] firstly described a two-step pathloss prediction by artificial neural network. In [16], RadioUNet has been proposed to estimate the propagation pathloss from city maps and other feature maps. In [30], for the real-time radio map, the authors proposed a fine-grained radio map reconstruction framework called Supreme, which explores spatial–temporal relationships within historical radio maps. In [31], the authors firstly employed a graph convolutional neural network for radio map prediction. For the reason that radio maps are functions of spatial coordinates, a convolutional autoencoder network was proposed for spectrum map interpolation in [32]. Recently, the Neural Architecture Search has been employed to optimize the Neural Network model, and additional side information (city plan, terrain height) has been used to enhance the accuracy of radio map reconstruction [27].

1.2. Motivations

The studies above suggest that certain deep learning models have the capacity to accurately and efficiently predict pathloss after the completion of comprehensive training, with the caveat being access to sufficient amounts of input data. However, data acquisition is a crucial factor that requires careful consideration in UAV-assisted communication areas. Firstly, it is difficult for UAVs to collect data from users centrally to train with deep learning models due to limited resources. Secondly, the data used for radio map reconstruction usually contain sensitive user positions. Therefore, users prefer to retain the data locally instead of uploading it to the UAVs due to privacy concerns. These factors make it challenging to reconstruct the radio map, which is essential for UAV-assisted communication. A potential solution to overcome these limitations is federated learning, which is an innovative approach that enhances data privacy in deep learning by utilizing a distributed learning system and centralized aggregation [33]. Recent studies have supported the FL-based approach as a popular and efficient way to facilitate privacy

preservation in UAV applications [34,35]. FL enables clients (users) to keep their private data and train the model locally in limited epochs before sending only the model parameters to the aggregation server (the UAV). This method eliminates the need for UAV's deep learning model training and consequent resource consumption.

1.3. Contribution

Storing user data locally in an FL is a viable method to ensure user privacy. However, it can lead to client drift [36], which occurs when the local dataset of a specific user does not represent the entire data distribution [37], resulting in lower model performance in predicting pathloss. To address this issue, our proposed FL architecture incorporates non-sensitive global information to improve information sharing among clients and ensure the model gradient is updated toward the global optimal direction. To reduce communication overhead, we incorporate training with feedback. This approach significantly reduces learning difficulty and allows us to design a lighter and more efficient model. Referring to the decoder structure of the autoencoder [32] and attention mechanism [38], a radio map reconstruct network in an image super-resolution (SR) manner (RadioSRNet) is firstly designed with fewer parameters and lower complexity. Then, considering the limited computing resources of clients, we design an arbitrary position prediction (APP) module based on the local implicit image function (LIIF) [39] to alleviate the trade-off between spatial resolution and computational complexity. Combined with the APP module, a network that predicts pathloss with a continuous position has been proposed in our paper, which is called RadioSRCNet. Additionally, we include a feature fusion module (FF) in our APP module that introduces external features such as the position of TX to enhance the model's scalability and accuracy. In brief, our contribution can be summarized as follows:

- We propose a federal learning architecture for radio map reconstruction in UAV-aided communication based on training with feedback. This architecture reduces the risk of privacy breach, mitigates client drift [36] by incorporating non-sensitive global information and speeds up the convergence.
- We propose a lightweight and efficient client model to reduce communication overhead between the FL server (UAVs) and clients (users) while maintaining a high level of accuracy in pathloss prediction despite the limited storage and computing resources.

2. System Model

2.1. Preliminaries

For a single sample at the RX baseband output, denoted as Y, the signal is affected by three factors: large-scale signal attenuation (P), small-scale fading (modeled as a normalized Gaussian random variable H with a unit second moment), and additive noise (Z). Therefore, we can express Y as shown below [16]:

$$Y = \sqrt{P}HX + Z. \tag{1}$$

Here, the symbol X represents the transmitted signal sample. The objective of radio map reconstruction is to estimate the pathloss (P), which refers to the large-scale signal attenuation in the area of interest.

2.2. Problem Formulation

This paper considers one UAV as a server and n users as clients, each actively operating in a corresponding area $A = \{A_0, A_1, \ldots, A_{n-1}\}$ with corresponding pathloss measurements consisting of datasets $D = \{D_0, D_1, \ldots, D_{n-1}\}$. The objective is to estimate pathloss in any RX location x across the entire area using FL methodology. To ensure data privacy and deal with limited resources in UAV, it is preferable for the server to not have access to user metadata or training models directly. Therefore, each individual client (represented by i) maintains their private data D_i locally for training purposes and subsequently uploads the trained model parameters to the server for aggregation. We try to find the best

reconstruction function F along with parameters θ to estimate pathloss $\hat{P}$ at any Rx location x, which is [31]:

$$\hat{P(x)} = F(\theta, x, D). \tag{2}$$

It is worth noting that deep learning methods typically require a substantial amount of data to train models and find the optimal parameter θ. For this reason, $N_x \times N_y$ rectangular grids are commonly used for the spatial discretization of radio maps in order to facilitate computation. Radio maps can be represented with tensors, and the estimated pathloss of RX is determined by the value of its nearest grid point on the radio map [16,30,32]. Prediction errors caused by spatial discretization are generally tolerable provided that the grids are fine enough (i.e., the spacing along the x- and y-axes, Δ_x and Δ_y, are sufficiently small). Discretization makes sense and can alleviate the effects of small-scale fading to some extent [32]. The information for radio map reconstruction, such as pathloss measurements or city maps, can also be represented as tensors. However, different regions may require various grid sizes, which should be carefully considered. Excessive grid spacing can result in low resolution (LR), which inevitably reduces the accuracy of pathloss prediction. Conversely, using overly fine grids with high resolution (HR) can be highly computationally expensive, particularly for CNN-based models.

3. Proposition

In this section, we present the methodology, the architectural design of FL, and the local models proposed.

3.1. Methodology and Architecture

As mentioned earlier, FL makes users keep their private data locally for model training, which significantly reduces the risk of privacy leakage. This paper proposes a method to enhance FL by integrating it with a feedback training strategy. This method reduces the learning complexity of the models, thereby making them more lightweight. Additionally, it introduces global radio maps, which stabilize the direction of gradient descent and improve the accuracy of FL.

3.1.1. FL Clients

Although our paper strictly limits the local training epochs for clients (e.g., 10), the process of radio map reconstruction still requires a significant amount of computing resources. This is due to the fact that the server's original training task is distributed among each client using FL. Therefore, in addition to data privacy concerns, it is important to minimize the number of parameters(params) and Floating-point Operations (FLOPs) in order to incentivize clients to participate in our task. However, this presents a challenge to the learning ability of models, making it necessary to design a lightweight and efficient model with FL.

3.1.2. FL Server

The FL server performs not just weight aggregation from clients but also generates non-sensitive global radio maps. This mechanism ensures user privacy protection while indirectly promoting client information sharing. Meanwhile, thanks to training with feedback, the global radio maps inferred from the last communication round are also used as the models' inputs in the current communication round, which retains historical learning results to some extent. Thus, the learning difficulty of local models is reduced, which makes models of clients lightweight and possibly efficient.

According to traditional methods, such as large-scale fading models in free space (as discussed in [40]), or historical radio maps, the FL server initializes global LR radio maps and model weights. These are then sent to selected clients who individually train local models with their own data and upload their weights to the server. Once received, the server combines the received weights and global LR radio maps to infer the models and update different regions of the global HR map corresponding to each client. An average is

taken if there is overlap. The global LR maps are updated by applying average pooling for the global HR radio map. After weight aggregation from selected clients, the FL server sends back the aggregated weights and updated global LR maps to clients in the next communication round. Figure 1 and Algorithm 1 illustrate the process.

Algorithm 1: FL-based radio map reconstruction with training feedback strategy

Input : The initial global model weights $\boldsymbol{\theta}_0$, the initial global LR radio map $\boldsymbol{RM}_0$, the local client datasets $\mathcal{D} = \{D_0, D_1, \cdots, D_{n-1}\}$ and the client activity areas $\mathcal{A} = \{A_0, A_1, \cdots, A_{n-1}\}$

Output: The trained global model weights $\boldsymbol{\theta}_T$ and global LR radio map $\boldsymbol{RM}_T$

Initialize $\boldsymbol{\theta}_0$, $\boldsymbol{RM}_0$;

for ***each*** *round* $t = 0, 1, \ldots, T-1$ **do**

  $S_M \leftarrow$ (randomly select M clients);

  `// The server download` $\boldsymbol{\theta}_t$ $\boldsymbol{RM}_t$ `to selected clients`

  for ***each*** *client* $i \in S_M$ ***in parallel*** **do**

    $\boldsymbol{\theta}_t^i \leftarrow \text{ClientTrain}(\boldsymbol{\theta_t}, \boldsymbol{RM_t}, D_i)$;

  end

  `// Clients upload` $\boldsymbol{\theta}_t^i$ `to the server`

  for $i \in S_M$ **do**

    $\boldsymbol{RM}_{\boldsymbol{t}}^{\boldsymbol{i}} \leftarrow \text{ServerInfer}(\boldsymbol{\theta}_{\boldsymbol{t}}^{\boldsymbol{i}}, \boldsymbol{RM_t}, A_i)$;

  end

  $\boldsymbol{RM_{t+1}} = \frac{1}{M}\sum_{i=1}^{M} \boldsymbol{RM}_t^i$;

  $\boldsymbol{\theta_{t+1}} = \frac{1}{M}\sum_{i=1}^{M} \boldsymbol{\theta}_t^i$;

end

Training with feedback is an important part of our proposal. After receiving the weights, the server performs inference on the model to reproduce the learning results of selected clients for the new global HR map. The updated global LR maps are generated using average pooling and are sent back to be shared with other clients. The global radio map is both the learning results of selected clients in the last round and the same input in the next round. Average pooling is necessary in the FL server because it reduces the size of the global HR radio map and makes global LR maps less related to any particular client's prediction, which can be seen as non-sensitive global information. The same input for all selected clients allows each one to indirectly make use of others' pathloss prediction, which alleviates the problem that the gradient is hard to decrease stably in FL, which is known as client drift [36]. The global LR maps represent the last round's learning results to some extent, giving our FL architecture some kind of memory, where users' models not only learn from local data but also use global LR maps as a reference. For FL with training feadback strategy, Equation (2) can be rewritten as:

$$\hat{P(x)} = F(\theta, x, \hat{RM}_t, D) \tag{3}$$

where $\hat{RM}_t$ is the estimated global radio map in round t. The new estimated radio map $\hat{RM}_{t+1}$ in round $t+1$ can be easily achieved with $x \in A$, where A is the area of interest. It allows users' models to pay attention to how to improve the performance on the basis of existing radio maps with local data. The models are then trained to reconstruct a better radio map $\hat{RM}_{t+1}$ from the previous map $\hat{RM}_t$. This approach greatly reduces the difficulty of learning and enables the users' models to require fewer parameters, making them more lightweight and efficient.

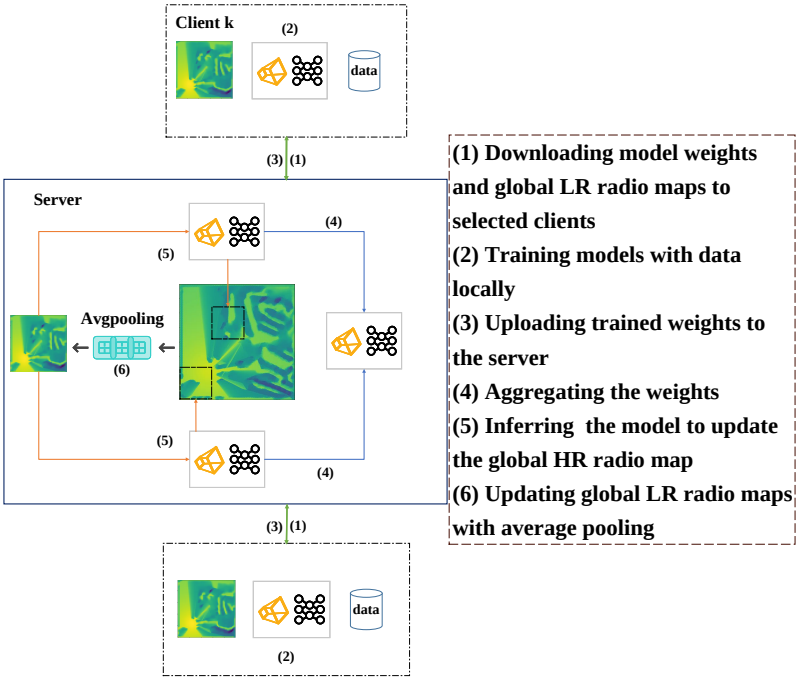

Figure 1. Overview of FL architecture.

3.2. Design of Models

In our proposal, we include average pooling in the FL server to encode the learning results from the previous round. The clients' models are used to decode the radio maps received from the FL server and train using local data. We begin by introducing RadioSRNet, which is based on the decoder architecture of the autoencoder [32]. We then enhance RadioSRNet by incorporating the Arbitrary Position Prediction (APP) module inspired by LIIF [39], resulting in RadioSRCNet, as proposed in this paper.

3.2.1. RadioSRNet

The architecture of RadioSRNet is illustrated in Figure 2a, which is composed of two fully connected (FC) layers, several denseSR blocks and the attention module. The denceSR block aims to extract features and enhance the spactial resolution, consisting of several convolution layers which are connected densely [41], and one convolution transpose layer [42] for upsampling with a scale $\times 2$. Thus, the upsampling scale of the models depends on the number of denseSR blocks n. Here, the first FC layer encodes the global LR radio maps to latent variables of which the code length is 12, and the other FC layer decodes it for recovery. Taking communication cost into consideration, instead of sending the weights of the first FC layers, the code is sent to the FL server to update the global HR radio map. The compression of FC layers is beneficial to lower noise, exchange spatial information and enhance the model's learning ability. In addition, an attention module is incorporated into our model to enhance accuracy by attending to significant features or grid points [38].

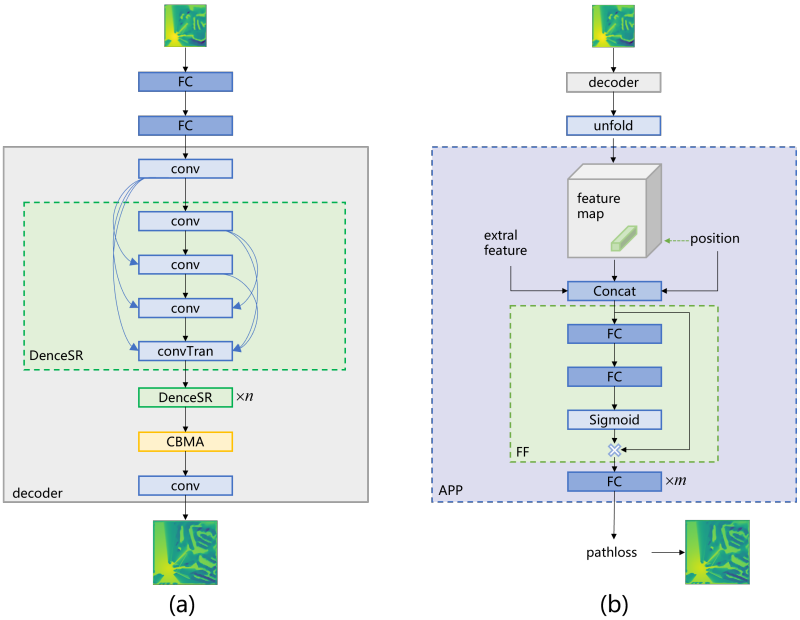

Figure 2. (**a**) RadioSRNet, based on the architecture of decoder in the autoencoder [32] and combined with attention mechanism [38]. (**b**) RadioSRCnet, predicting pathloss in a continuous manner with the proposed APP module based on RadioSRNet.

3.2.2. RadioSRCNet

As depicted in Figure 2b, the APP module plays a crucial role in RadioSRCNet, as it leverages the high-dimensional feature maps extracted by the aforementioned decoder module as well as RX positions and other relevant information. It utilizes these inputs to make continuous predictions of pathloss. The APP module is composed of two main components: the Feature Fusion (FF) module and the Multi-Layer Perceptron (MLP). The FF module is responsible for combining the feature maps and the additional information, while the MLP carries out the actual prediction process. The m in Figure 2b is the number of layers of MLP, which is 5 in our paper. The design of the APP module is based on the rationale that relying on the resolution of pre-defined grids for pathloss prediction accuracy is unreasonable. Additionally, estimating pathloss based on the value of the closest grid point is inadequate. Therefore, it is highly recommended to leverage the full information of RX positions and additional features to predict pathloss, regardless of the resolution of radio map grids. One approach, as described in [16,32], is to represent extra features as tensors and concatenate them with feature maps. However, this may be inefficient, particularly when the number of TX is small and the tensors are sparse. Furthermore, using RX positions in grids directly is challenging, as their coordinates are mostly non-integer values. To solve the problems above, MLP may be a more flexible choice, which is the basic module in APP. Specifically, when the high-dimensional feature maps are available, we select features around the RX positions, which are believed to be most related with RX pathloss considering radio propagation. The features, the relative coordinate distance between RX positions and grid points of selected features, and extra features (TX positions, etc.) are concatenated as local features for prediction. However, the local features from multiple sources are various in scale and importance, which need attention and scaling. Thus, a feature fusion module is introduced in our paper to adjust the weights of each feature adaptively. Then, an MLP after the FF module is used for pathloss prediction with the adjusted feature weights.

As shown in Figure 3, there are usually four grid points (top-left, top-right, bottom-left and bottom-right sub-spaces) around RX (padding zero-vectors if none). Each of them would predict pathloss in RX with its own features, relative distances and extra features as mentioned above. Inspired by [39], we employ local ensemble learning to aggregate predic-

tions based on their relative distance, thereby determining the ultimate result. Specifically, the longer the distance, the smaller the contribution. Let $c_t^*(t \in \{00, 01, 10, 11\})$ denote the features in the top-left, top-right, bottom-left, and bottom-right sub-spaces, respectively. Let x be the coordinate of RX, and let ex represent the extra features. The area of the rectangle bounded by the coordinate x and the point on the diagonal of the grid point c_t^* is represented by S_t. The formulation of pathloss prediction $\hat{P}$ with local ensemble learning can be defined as

$$\hat{P} = \sum_{t \in \{00,01,10,11\}} \frac{S_t}{S} f(\theta; x, ex, ; c_t^*). \tag{4}$$

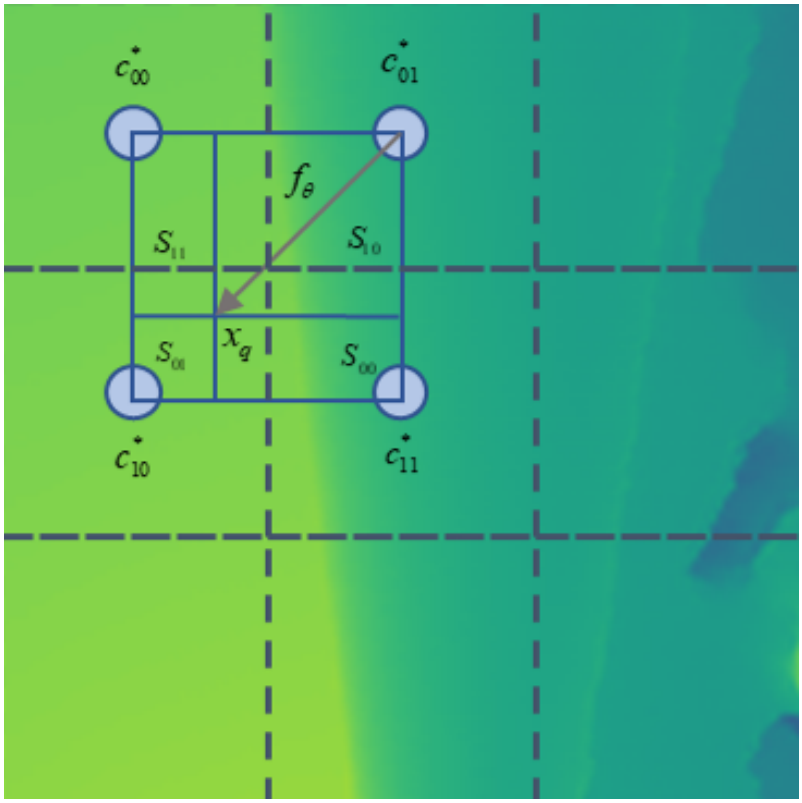

Figure 3. Pathloss prediction with local ensemble.

The weights are normalized by $S = \sum_t S_t$. The greater the value of S_t, the shorter the distance and the greater the contribution to $\hat{P}$. It is important to note that we use unfolding to enrich information and enlarge receptive fields as depicted in Figure 2b. Specifically, the feature unfolding is formally defined as [39]

$$c_{jk}^* = Concat(\{c_{j+l,k+m}\}_{l,m \in \{-1,0,1\}}), \tag{5}$$

where *Concat* is short for concatenation, and $c_{j+l,k+m}$ refers to decoder outputs without unfolding. Zero-vectors are padded if they are outside the border. The proposed APP module makes the model remain highly accurate with limited computing resources, which will be demonstrated in the following section.

4. Experiment and Results

In this section, the dataset used in the experiments and relevant experimental parameters are described. Subsequently, the baseline models and evaluation metrics are briefly introduced. Finally, the proposed approach is thoroughly examined in terms of its specific performance from multiple perspectives.

4.1. Dataset and Parameter Setting

Recently, an open-source dataset named as WAIR-D was proposed to support related research, which is used in our paper [43]. The dataset in our paper contains 4 base stations and 90 clients which are grouped into 9 groups. Each group includes 10 clients that are active in the same area with different data samples. The dataset consists of 90 client records, totaling 398,041 samples that include RX locations and corresponding pathloss. These records are associated with four base stations. The training and testing sets are divided at a ratio of 8:2. Specifically, a Cartesian coordinate system is established with the corner of the area as the origin. The RX position is represented as a two-dimensional coordinate, while the pathloss of each base station is determined by the signal strength received. The base stations do not interfere with each other in our experiments. To be detailed, Table 1 summarizes the parameters and their settings.

In our experiment, Pytorch was used as the model training framework, and the training and testing were completed on an NVIDIA 3090 GPU. The mean absolute error (MAE) was adopted as the loss function, and the Adam optimizer was utilized for local training. The initial learning rate was set to 0.005, and after 80 rounds, the learning rate was reduced to 75% of the previous value every 18 rounds. The number of local training epochs was set to 10, and the upper limit of total federated learning rounds was set to 180, with a batch size of 150 and early-stop rounds of 9. There are only differences in the grid sizes of the global HR radio map, and the grid size of the global LR radio map is identical (which is 50×50) if the tensors are used in models to represent radio maps, as depicted in Equation (3).

Table 1. Implementation parameters.

Dataset	
Size of area (m)	400×400
Coordinates of base stations (m)	$(200+j, 200+k)_{j,k\in\{-100,100\}}$
Size of area for each group (m)	240×240
Centroidal coordinates for each group (m)	$(200+j, 200+k)_{j,k\in\{-80,0,80\}}$
Training set	318,397
Testing set	79,644
Federated Learning	
FL server	1
Number of clients	90
Clients used in federated updates	5
Local training epochs	10
Communication rounds	180

4.2. Baseline and Metrics

MLP [29] and kriging [24] are classic models in deep learning and traditional interpolation, respectively. They are often compared in radio map construction [16,32]. The completion autoencoder [32] is one of the state-of-the-art models in this field and has been used as a comparison to our proposition, with a code length of latent variables equal to 32 in our paper. The model structure of the MLP is identical to the one described in [29]. Additionally, the ordinary kriging interpolation method is also compared in our paper [24]. The metric of accuracy in our paper is normalized mean squared error (NMSE), which is defined as follows

$$NMSE = E\{\frac{\|P-\hat{P}\|_2^2}{\|P\|_2^2}\}, \tag{6}$$

where P is the true pathloss and $\hat{P}$ is the estimated pathloss, both of which are in the unit of dB.

In FL, the uplink and downlink communication overheads are also important metrics [37]. The uplink communication overheads refer to the number of model parameters

to be sent to clients and other necessary information, so as to downlink communication overheads. The total communication overheads of all rounds are considered in our paper. Given that user resources are also limited, storage and computing resources (i.e., the parameters of the whole model and the FLOPs for training with one batch) are also considered as performance metrics of FL for radio map reconstruction.

4.3. Results

In this subsection, we investigate the effects of model design and FL architecture, which are APP modules and feedback training strategy, respectively. Additionally, we compare the performance of the proposed approach with FL and non-FL baseline models using the aforementioned metrics.

4.3.1. Impact of Client Model Architectures

The impact of the global HR radio map with different grid sizes on model performances is presented in Table 2. It can be seen that regardless of the proposed RadioSRCNet, RadioSRNet or the autoencoder [32], as the grid size increases, i.e., as the the spatial resolution increases, the FLOPs, params, and prediction accuracy all increase. It is worth noting that FLOPs are the ratio to the grid size for CNN-based models. Because we only use decoder-based networks (RadioSRCNet and RadioSRNet) in clients, these networks require approximately 50% fewer FLOPs compared to the autoencoder [32] implemented in the same grid size. Furthermore, both RadioSRCNet and RadioSRNet require significantly fewer parameters than the autoencoder [32]. This reduction in parameters is not solely attributed to the decoder-based architecture but also to the training process involving feedback, which helps alleviate learning difficulties and consequently requires fewer parameters for model design.

Table 2. The impact of different grid sizes on model performance.

Size of Grid	Model	NMSE (dB)	Complexity	
			Params (M)	FLOPs (G)
200×200	RadioSRCNet	−38.99	0.33	5.50
	RadioSRNet	−33.00	0.20	5.54
	Autoencoder [32]	−33.63	5.31	8.59
400×400	RadioSRCNet	−39.72	0.37	18.89
	RadioSRNet	−36.58	0.27	22.91
	Autoencoder [32]	−36.89	5.37	35.95
800×800	RadioSRCNet	−40.18	0.41	72.45
	RadioSRNet	−39.05	0.33	92.42
	Autoencoder [32]	−39.54	5.42	145.42

To verify the effectiveness of each component in our proposed model, we conduct the ablation investigation where RadioSRCNet-rf is RadioSRCNet without FF and RadioSRCNet-ra is RadioSRCNet without APP (i.e, RadioSRNet, where we put the FC layers after input layers rather than directly remove them with the APP module for fair comparison, which accounts for too many parameters). As depicted in Figure 4, for RadioSRCNet-ra, we can see that NMSE almost decreases by 3 dB as the grid area of the global HR radio map doubles, as shown in the autoencoder [32] in Table 2. Indeed, the trade-off between computational complexity (FLOPs) and prediction accuracy (NMSE) resulting from spatial resolution is determined by the gridding and convolution methods, as explained in Section 2.2. To address this issue, we propose the APP module, which aims to mitigate this trade-off. The goal of the APP module is to provide the model with a relatively high spatial resolution while keeping the computational complexity low. It can be seen that the accuracy of RadioSRCNet varies less with grid size compared to the RadioSRNet-ra and the autoencoder [32] in Figure 4 and Table 2, indicating a weak correlation between accuracy

and spatial resolution. Specifically, for a grid size of 200 × 200, RadioSRCNet achieves an NMSE estimation accuracy that is only about 0.6 dB worse than RadioSRNet-ra and the autoencoder [32] with a grid size of 800 × 800, while requiring FLOPs that are only about 5.95% and 3.78% of the latter two, respectively. This suggests that RadioSRCNet provides a favorable trade-off between accuracy and computational complexity, allowing for high accuracy with significantly reduced computational requirements. For the FF module, it is used to fuse features with extra useful information (TX location in our paper). Obviously, extra features are unrelated with the size of the grid, which further mitigates the trade-off. As shown in Figure 4, compared with the grid size of 800 × 800, the performace gap between RadioSRCNet and RadioSRCNet-rf is much larger in the grid size of 200 × 200, indicating that the FF module contributes more significantly to accuracy in low spatial resolution especially, which is beneficial for limited storage and computing resources.

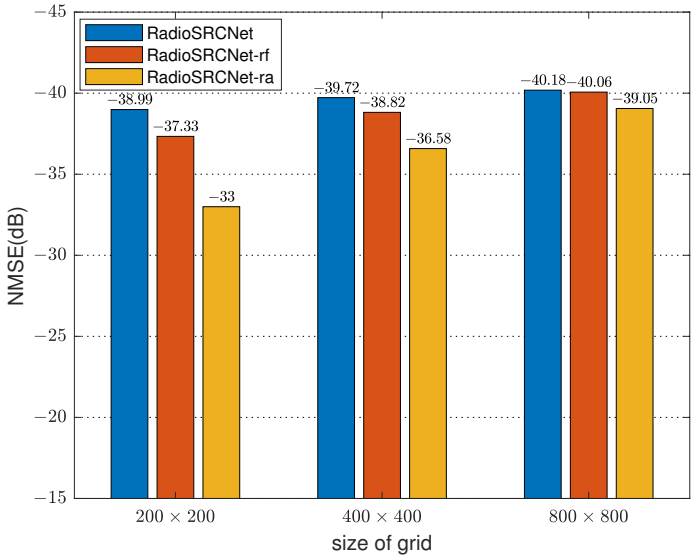

Figure 4. Performance comparisons over our proposed model and its variants.

4.3.2. Impact of Training with Feedback

We discuss the impact of the feedback training strategy on the prediction accuracy of FL learning for radio map construction. The experimental results are shown in Figure 5. We can see that the feedback training strategy has a certain degree of improvement on the accuracy of models. However, the effect of improvement varies among different models. Specifically, the improvement effect is significant for RadioSRCNet, which is 9.15 dB, while it is marginal for RadioSRNet and the autoencoder [32], which are 0.14 dB and 0.30 dB, respectively. Obviously, it is the architecture of models that determines the improvement effect of the feedback training strategy. Compared with RadioSRCNet, both of the other models will first compress the global LR radio map to latent variables (the code length is 12 for RadioSRNet and 32 for autoencoder [32]) and then recover it through FC layers and convolution layers. This coding compression structure has to some extent weakened the impact of the previous learning results on the training of this round. Without the compression of FC layers, RadioSRCNet can retain more information with the previous learning results. As inputs, the information can significantly enhance the high-dimensional features extracted by the decoder and thus fully leverage the advantages of the feedback training strategy. Therefore, RadioSRCNet benefits most significantly in terms of accuracy.

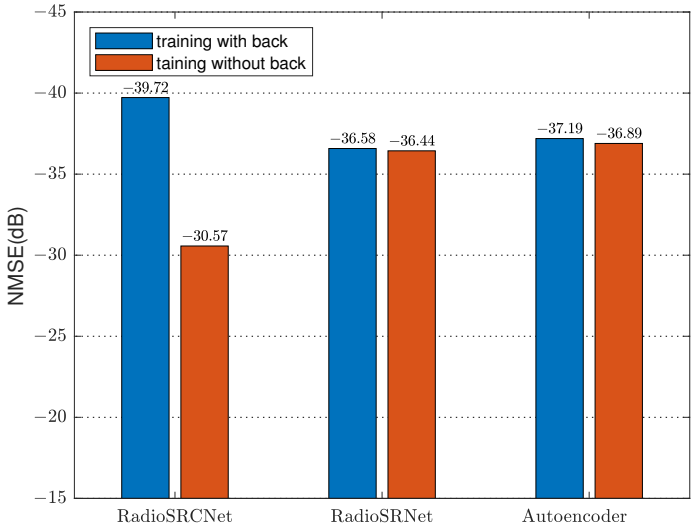

Figure 5. The impact of training with feedback on model performance where the size of the grid is 400 × 400, Autoencoder from [32].

4.3.3. Comparison with FL Models

In this section, we mainly compare the performance of various models in FL from two aspects: prediction accuracy of pathloss (measured by NMSE) and communication overheads (uplink and downlink). The global HR radio maps of kriging [24], autoencoder [32], RadioSRNet, and RadioSRCNet all utilize a grid size of 400 × 400. The corresponding global LR radio maps have a grid size of 50 × 50, resulting in a scale of 3 for the upsampling process in the decoder. In this experiment, each round of selected clients train their local models and then upload the weights to the FL server. In addition to the differences in model structure, the FL architecture is consistent with Section 3.1.2. Only the two models designed in this paper adapt a feedback training strategy. The experiment results are shown in Table 3.

Table 3. The performance of our model against other models in FL.

Models	DL Overheads (G)	UL Overheads (G)	NMSE (dB)	Epochs
RadioSRCNet	0.34	0.33	−39.72	180
RadioSRNet	0.25	0.24	−36.58	180
Autoencoder [32]	5.41	5.41	−36.89	180
MLP [29]	0.12	0.12	−22.11	180
Kriging [24]	0	0.058	−17.12	18

Compared to traditional kriging interpolation [24], deep learning models generally exhibit higher estimation accuracy, they but come with significantly larger communication overheads. There is a notable distinction between the classical deep learning model, namely MLP [29], and other CNN-based models. This discrepancy highlights the critical importance of a well-designed model structure. Comparing RadioSRCNet, RadioSRNet, and the autoencoder [32], it is evident that model lightweightness and efficiency are essential for reducing parameters and minimizing communication overheads. Specifically, the average uplink and downlink communication overheads of RadioSRCNet and RadioSRNet are approximately 6.28% and 4.62% of the autoencoder [32] respectively, while still achieving superior or comparable estimation accuracy. These ratios closely resemble their respective numbers of parameters. It can be observed that RadioSRCNet boasts the highest prediction accuracy. Moreover, as depicted in Table 2 and Figure 4, with a grid size of 200 × 200,

RadioSRCNet improves the NMSE compared to RadioSRNet and the autoencoder [32] with a 400×400 grid size by approximately 2.41 dB and 2.1 dB, respectively, while the computational complexity measured in FLOPs of RadioSRCNet is only 18.94% and 6.15% of the latter two methods. This may be attributed to the feedback training strategy and the APP module, which have been discussed in the details above.

Considering the limited resources on the clients and the server, it is preferable to achieve high prediction accuracy with as few communication rounds as possible. This requires the model to converge relatively quickly. Figure 6 depicts the impact of communication rounds on the performance of the FL model. In this figure, RadioSRCNet-rb refers to RadioSRCNet without the training feedback strategy. It can be observed that RadioSRCNet can achieve a decrease of -30 dB within approximately 30 rounds and a decrease of -35 dB within about 70 rounds. This indicates faster convergence compared to other deep learning baselines, such as the autoencoder [32] and MLP [29]. We can also see that the performance of RadioSRCNet is greatly influenced by the feedback training strategy, which not only improves accuracy but also accelerates convergence. This is because the strategy reduces learning difficulty and mitigates client drift [36], as explained in more detail earlier.

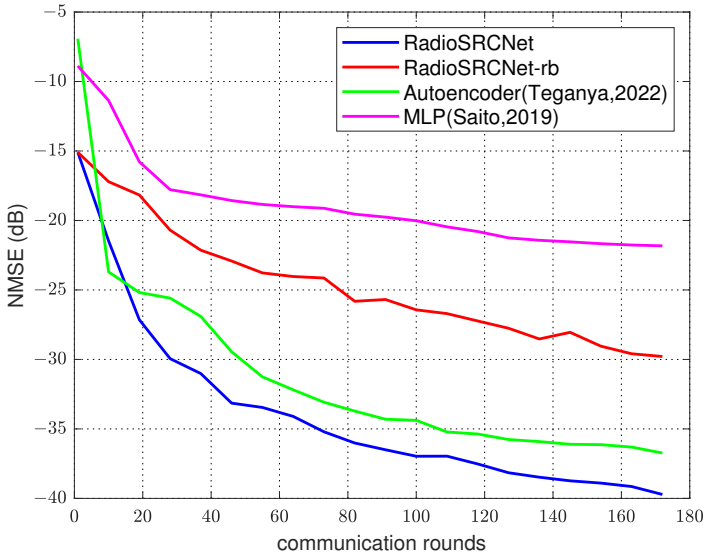

Figure 6. Effect of communication rounds on the FL model performance [29,32].

4.3.4. Comparison with Non-FL Model

As we can see in Figure 7, RadioSRCNet achieves the best accuracy no matter whether FL or Non-FL. It is worth noticing that the performance gap between FL and Non-FL caused by client drift [36] varies with different models. As mentioned before, to alleviate the problem, we use a feedback training strategy to narrow the performance gap by introducing global information (global radio maps) shared by all users. In our experiment, compared to other models without this training strategy, RadioSRCNet and RadioSRNet both have the smallest performace gaps, which are 0.23 dB and 1.25 dB, respectively. However, as we have discussed in detail in Section 4.3.2, the feedback training strategy has a greater impact on RadioSRCNet due to the differences in model structure, which is reflected not only in the improvement of accuracy but also in the reduction of performance gap between FL and Non-FL.

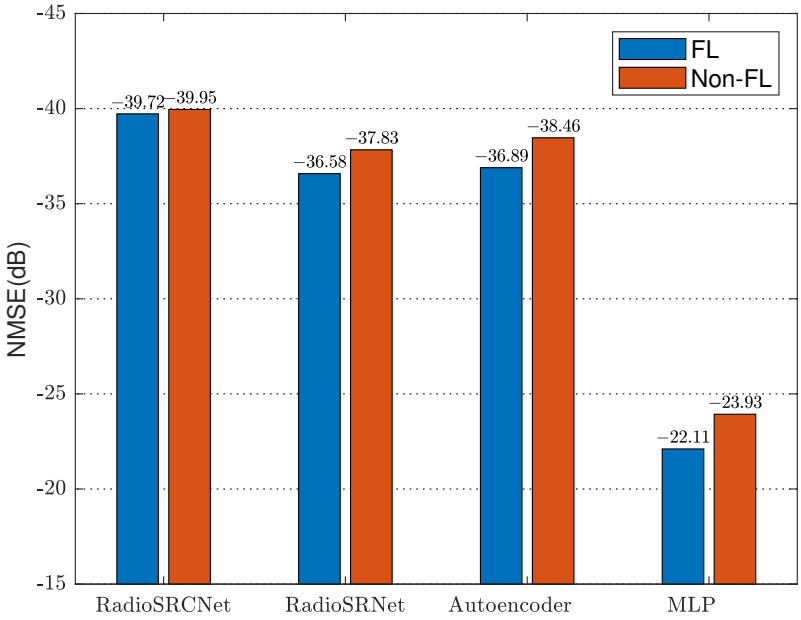

Figure 7. The accuracy of our model against other models in FL and Non-FL, Autoencoder from [32], MLP from [29].

5. Conclusions

Radio maps play a crucial role in various applications of 6G, such as UAV-assisted communication, network planning, and resource allocation. However, existing methods for crowd-sourced reconstruction suffer from several drawbacks. First, they require vast amounts of privacy-sensitive user data, which raises concerns about user privacy and reduces the data contribution. Second, these methods heavily rely on training large models, particularly in the context of deep learning, which places significant demands on server resources. As a result, reconstructing radio maps on resource-constrained UAVs becomes a challenging task. In order to address these limitations, we propose a self-supervised federated learning model called RadioSRCNet. This model utilizes a super-resolution-based network and feedback training to accurately predict continuous position pathloss. In our approach, users retain their original data locally and act as clients for training, while the UAV serves as a server to aggregate non-sensitive data and perform radio map reconstruction using federated learning. To expedite convergence and ease training difficulties, we adopt a feedback training strategy. Additionally, we introduce an Arbitrary Position Prediction (APP) module that reduces resource consumption in clients. This innovative module strikes a balance between spatial resolution and computational complexity. Experimental results demonstrate the superiority of our proposed framework, as our model achieves higher accuracy with lower communication overheads, all while being computationally and storage-efficient compared to other deep learning methods.

Author Contributions: Conceptualization, Z.T. and Y.L.; methodology, Z.T.; software, Z.T.; validation, Z.T., X.T. and Y.L.; formal analysis, Z.T. and X.T.; investigation, Z.T.; writing—original draft preparation, Z.T.; writing—review and editing, L.X., M.Z. and Y.L.; supervision, Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Key Research and Development Program of China under Grant 2020YFB1806603.

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

Abbreviations

The following abbreviations are used in this manuscript:

UAV	Unmanned aerial vehicle
SR	Super-resolution
APP	Arbitrary position prediction
TX	Transmitter
RX	Receiver
FL	Federated learning
LIIF	Local implicit image function
FLOPs	Floating-point Operations
FF	Feature fusion
HR	High-resolution
LR	Low-resolution
CNN	Convolutional neural network
FC	Fully connected
MLP	Multi-layer perceptron
MAE	Mean absolute error
NMSE	Normalized mean squared error

References

1. Yuan, X.; Hu, Y.; Gross, J.; Schmeink, A. Radio-Map-Based UAV Placement Design for UAV-Assisted Relaying Networks. In Proceedings of the 2021 IEEE Statistical Signal Processing Workshop (SSP), Rio de Janeiro, Brazil, 11–14 July 2021; pp. 286–290. [CrossRef]
2. Wang, Y.; Feng, W.; Wang, J.; Quek, T.Q.S. Hybrid Satellite-UAV-Terrestrial Networks for 6G Ubiquitous Coverage: A Maritime Communications Perspective. *IEEE J. Sel. Areas Commun.* **2021**, *39*, 3475–3490. [CrossRef]
3. Li, X.; Feng, W.; Wang, J.; Chen, Y.; Ge, N.; Wang, C.X. Enabling 5G on the Ocean: A Hybrid Satellite-UAV-Terrestrial Network Solution. *IEEE Wirel. Commun.* **2020**, *27*, 116–121. [CrossRef]
4. Fang, X.; Feng, W.; Wang, Y.; Chen, Y.; Ge, N.; Ding, Z.; Zhu, H. NOMA-Based Hybrid Satellite-UAV-Terrestrial Networks for 6G Maritime Coverage. *IEEE Trans. Wirel. Commun.* **2023**, *22*, 138–152. [CrossRef]
5. Zeng, Y.; Wu, Q.; Zhang, R. Accessing From the Sky: A Tutorial on UAV Communications for 5G and Beyond. *Proc. IEEE* **2019**, *107*, 2327–2375. [CrossRef]
6. Fan, R.; Cui, J.; Jin, S.; Yang, K.; An, J. Optimal Node Placement and Resource Allocation for UAV Relaying Network. *IEEE Commun. Lett.* **2018**, *22*, 808–811. [CrossRef]
7. Ouamri, M.A.; Oteşteanu, M.E.; Barb, G.; Gueguen, C. Coverage Analysis and Efficient Placement of Drone-BSs in 5G Networks. *Eng. Proc.* **2022**, *14*, 18. [CrossRef]
8. Ouamri, M.A.; Singh, D.; Muthanna, M.A.; Bounceur, A.; Li, X. Performance Analysis of UAV Multiple Antenna-Assisted Small Cell Network with Clustered Users. *Wirel. Netw.* **2023**, *29*, 1859–1872. [CrossRef]
9. Li, P.; Xu, J. Fundamental Rate Limits of UAV-Enabled Multiple Access Channel With Trajectory Optimization. *IEEE Trans. Wirel. Commun.* **2020**, *19*, 458–474. [CrossRef]
10. Zeng, Y.; Xu, J.; Zhang, R. Energy Minimization for Wireless Communication With Rotary-Wing UAV. *IEEE Trans. Wirel. Commun.* **2019**, *18*, 2329–2345. [CrossRef]
11. Ouamri, M.A.; Alkanhel, R.; Singh, D.; El-kenawy, E.S.; Ghoneim, S. Double Deep Q-Network Method for Energy Efficiency and Throughput in a UAV-Assisted Terrestrial Network. *Comput. Syst. Sci. Eng.* **2022**, *46*, 73–92. [CrossRef]
12. Chen, J.; Mitra, U.; Gesbert, D. Optimal UAV Relay Placement for Single User Capacity Maximization over Terrain with Obstacles. In Proceedings of the 2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Cannes, France, 2–5 July 2019; pp. 1–5. [CrossRef]
13. Li, H.; Li, P.; Xu, J.; Chen, J.; Zeng, Y. Derivative-Free Placement Optimization for Multi-UAV Wireless Networks with Channel Knowledge Map. In Proceedings of the 2022 IEEE International Conference on Communications Workshops (ICC Workshops), Seoul, Republic of Korea, 16–20 May 2022; pp. 1029–1034. [CrossRef]
14. Zhang, S.; Zhang, R. Radio Map-Based 3D Path Planning for Cellular-Connected UAV. *IEEE Trans. Wirel. Commun.* **2021**, *20*, 1975–1989. [CrossRef]
15. Chen, J.; Yatnalli, U.; Gesbert, D. Learning radio maps for UAV-aided wireless networks: A segmented regression approach. In Proceedings of the 2017 IEEE International Conference on Communications (ICC), Paris, France, 21–25 May 2017; pp. 1–6. [CrossRef]
16. Levie, R.; Yapar, Ç.; Kutyniok, G.; Caire, G. RadioUNet: Fast Radio Map Estimation With Convolutional Neural Networks. *IEEE Trans. Wirel. Commun.* **2021**, *20*, 4001–4015. [CrossRef]
17. Mo, X.; Huang, Y.; Xu, J. Radio-Map-Based Robust Positioning Optimization for UAV-Enabled Wireless Power Transfer. *IEEE Wirel. Commun. Lett.* **2020**, *9*, 179–183. [CrossRef]

18. Utkovski, Z.; Agostini, P.; Frey, M.; Bjelakovic, I.; Stanczak, S. Learning Radio Maps for Physical-Layer Security in the Radio Access. In Proceedings of the 2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Cannes, France, 2–5 July 2019; pp. 1–5. [CrossRef]
19. Grimoud, S.; Ben Jemaa, S.; Sayrac, B.; Moulines, E. A REM enabled soft frequency reuse scheme. In Proceedings of the 2010 IEEE Globecom Workshops, Seoul, Republic of Korea, 16–20 May 2010; pp. 819–823. [CrossRef]
20. Zugno, T.; Drago, M.; Giordani, M.; Polese, M.; Zorzi, M. Toward Standardization of Millimeter-Wave Vehicle-to-Vehicle Networks: Open Challenges and Performance Evaluation. *IEEE Commun. Mag.* **2020**, *58*, 79–85. [CrossRef]
21. Rizk, K.; Wagen, J.F.; Gardiol, F. Two-dimensional ray-tracing modeling for propagation prediction in microcellular environments. *IEEE Trans. Veh. Technol.* **1997**, *46*, 508–518. [CrossRef]
22. Kermani, M.; Kamarei, M. A ray-tracing method for predicting delay spread in tunnel environments. In Proceedings of the 2000 IEEE International Conference on Personal Wireless Communications, Conference Proceedings (Cat. No.00TH8488), Hyderabad, India, 17–20 December 2000; pp. 538–542. [CrossRef]
23. Wahl, R.; Wölfle, G.; Wertz, P.; Wildbolz, P.; Landstorfer, F.M. Dominant Path Prediction Model for Urban Scenarios. In Proceedings of the IST Mobile and Wireless Communications Summit, Dresden, Germany, 19–23 June 2005.
24. Boccolini, G.; Hernández-Peñaloza, G.; Beferull-Lozano, B. Wireless sensor network for Spectrum Cartography based on Kriging interpolation. In Proceedings of the 2012 IEEE 23rd International Symposium on Personal, Indoor and Mobile Radio Communications—(PIMRC), Sydney, NSW, Australia, 9–12 September 2012; pp. 1565–1570. [CrossRef]
25. Li, J.; Heap, A.D. A review of comparative studies of spatial interpolation methods in environmental sciences: Performance and impact factors. *Ecol. Inform.* **2011**, *6*, 228–241. [CrossRef]
26. Redondi, A.E.C. Radio Map Interpolation Using Graph Signal Processing. *IEEE Commun. Lett.* **2018**, *22*, 153–156. [CrossRef]
27. Malkova, A.; Amini, M.R.; Denis, B.; Villien, C. Deep Learning with Partially Labeled Data for Radio Map Reconstruction. *arXiv* **2023**, arXiv:2306.05294. [CrossRef]
28. Suto, K.; Bannai, S.; Sato, K.; Inage, K.; Adachi, K.; Fujii, T. Image-Driven Spatial Interpolation With Deep Learning for Radio Map Construction. *IEEE Wirel. Commun. Lett.* **2021**, *10*, 1222–1226. [CrossRef]
29. Saito, K.; Jin, Y.; Kang, C.; Takada, J.I.; Leu, J.S. Two-step path loss prediction by artificial neural network for wireless service area planning. *IEICE Commun. Express* **2019**, *8*, 611–616. [CrossRef]
30. Li, K.; Chen, J.; Yu, B.; Shen, Z.; Li, C.; He, S. Supreme: Fine-grained Radio Map Reconstruction via Spatial-Temporal Fusion Network. In Proceedings of the 2020 19th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN), Sydney, NSW, Australia, 21–24 April 2020; pp. 1–12. [CrossRef]
31. Tonchev, K.; Ivanov, A.; Neshov, N.; Manolova, A.; Poulkov, V. Learning Graph Convolutional Neural Networks to Predict Radio Environment Maps. In Proceedings of the 2022 25th International Symposium on Wireless Personal Multimedia Communications (WPMC), Herning, Denmark, 30 October–2 November 2022; pp. 392–395. [CrossRef]
32. Teganya, Y.; Romero, D. Deep Completion Autoencoders for Radio Map Estimation. *IEEE Trans. Wirel. Commun.* **2022**, *21*, 1710–1724. [CrossRef]
33. Liu, Y.; Yuan, X.; Xiong, Z.; Kang, J.; Wang, X.; Niyato, D. Federated learning for 6G communications: Challenges, methods, and future directions. *China Commun.* **2020**, *17*, 105–118. [CrossRef]
34. Islam, A.; Al Amin, A.; Shin, S.Y. FBI: A Federated Learning-Based Blockchain-Embedded Data Accumulation Scheme Using Drones for Internet of Things. *IEEE Wirel. Commun. Lett.* **2022**, *11*, 972–976. [CrossRef]
35. Do, Q.V.; Pham, Q.V.; Hwang, W.J. Deep Reinforcement Learning for Energy-Efficient Federated Learning in UAV-Enabled Wireless Powered Networks. *IEEE Commun. Lett.* **2022**, *26*, 99–103. [CrossRef]
36. Karimireddy, S.P.; Kale, S.; Mohri, M.; Reddi, S.; Stich, S.; Suresh, A.T. SCAFFOLD: Stochastic Controlled Averaging for Federated Learning. In Proceedings of the 37th International Conference on Machine Learning, PMLR, Vienna, Austria, 13–18 July 2020; Volume 119, pp. 5132–5143.
37. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; Arcas, B.A.Y. Communication-Efficient Learning of Deep Networks from Decentralized Data. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, PMLR, Jakarta, Indonesia, 20–22 April 2017; Volume 54, pp. 1273–1282.
38. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the Computer Vision—ECCV 2018, Munich, Germany, 8–14 September 2018; pp. 3–19.
39. Chen, Y.; Liu, S.; Wang, X. Learning Continuous Image Representation with Local Implicit Image Function. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 8624–8634. [CrossRef]
40. Graziosi, F.; Santucci, F. A general correlation model for shadow fading in mobile radio systems. *IEEE Commun. Lett.* **2002**, *6*, 102–104. [CrossRef]
41. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely Connected Convolutional Networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2261–2269. [CrossRef]

42. Dumoulin, V.; Visin, F. A guide to convolution arithmetic for deep learning. *arXiv* **2016**, arXiv:1603.07285. [CrossRef]
43. Huangfu, Y.; Wang, J.; Dai, S.; Li, R.; Wang, J.; Huang, C.; Zhang, Z. WAIR-D: Wireless AI Research Dataset. *arXiv* **2022**, arXiv:2212.02159.