An Efficient Federated Transfer Learning Approach for Multi-UAV Systems

Vedang Bhupesh Shenvi Nadkarni⋆, Sandeep Joshi†, and L. Rajya Lakshmi⋆
⋆Department of Computer Science and Information Systems
†Department of Electrical and Electronics Engineering
Birla Institute of Technology and Science Pilani, Rajasthan 333031, India
f20200897@pilani.bits-pilani.ac.in, {sandeep85joshi, rajya99}@gmail.com

***Abstract*—Recent advances in multi-unmanned aerial vehicle (UAV) based federated learning do not take into consideration the massive computational requirements of modern deep learning models on mobile UAVs. Additionally, there has been significant progress that shows that the information transmitted between the federated agent and the central hub can be attacked to undermine the privacy of the data. We propose a novel multi-UAV-based federated transfer learning system that drastically reduces the computational burden overall, shifts it from UAVs to the ground fusion center, and reduces the bandwidth requirements while enhancing its secure nature. The proposed system makes multi-UAV learning significantly fast, reliable, power efficient, and practically feasible. Furthermore, we provide simulation and experimental results to demonstrate the effectiveness of the proposed system.**

***Index Terms*—Deep learning, federated learning, image classification, transfer learning, unmanned aerial vehicle.**

I. Introduction

Recent developments in applied federated learning have proposed systems that use multiple unmanned aerial vehicles (UAVs) which are equipped with a sensor for data collection, for example, a camera, and onboard computers to process this data. After the data is processed, it is transmitted to a ground fusion center (GFC) for further processing. In the case of federated learning, there is also a re-transmission of data back to the UAVs from the GFC thereafter. Such systems have vast practical applications such as aerial surveying, particularly for military and civil uses [1], [2].

The seminal works on federated learning [3] proposed a purely theoretical model for decentralized machine learning. It addressed critical issues concerning machine learning and the usage of gathered data, such as data privacy, data security, and access rights. Further work on bringing federated learning to multi-UAV networks focused on adapting the same system as initially proposed to a multi-UAV system, which needs to transmit data wirelessly [1]. However, this adaptation may not be well optimized for wireless communication networks due to the large model sizes of state-of-the-art deep learning networks [4].

Transfer learning [5] is a machine learning technique that re-uses knowledge from a task to boost performance on a similar task, typically in the same domain. With the open availability of high-quality trained deep learning model weights [6], transfer learning has been used widely in the machine learning community both in academia and industry [7] for tasks ranging from image classification [8] to natural language generation [9]. The key benefit of using transfer learning is that it saves vast amounts of computation by initializing the neural network weights to values that can be adapted to local data after some training. Such data is not used (and not available) during the initial training (e.g., private medical image data) but is from a similar domain (*i.e.,* a similar class of images). In this way, one can re-use knowledge learned from large-scale public datasets, such as ImageNet [10] or LAION-5B [11], which require training on extremely expensive, often widely inaccessible compute infrastructure. The model's weights (henceforth *weights* shall refer to the weights and biases of neural networks) are openly available on the internet [6], [12]. Since such models and their weights are typically optimized for the objective task (in our case, image classification), much of the model does not need to be modified.

A. Related works

In [1], the authors propose a federated learning-based approach in multi-UAV networks where multiple UAVs can communicate with a GFC. The system collects data from onboard sensors, performs local updates, and aggregates the weights at the GFC, which transmits the aggregate weights back to each UAV. This loop is repeated until satisfactory model performance is achieved. The proposed work adapts this system but crucially reduces the computations involved and the amount of data that needs to be transmitted. It also shifts the compute-heavy workload to the GFC, making it possible for UAVs to be power efficient. Transfer learning was proposed as a consequence of observations in the similarity of features learned by neural networks in similar tasks across moderately diverse domains of data [5]. By relaxing the requirement of the training data being independent and identically distributed (i.i.d.) with the test data, transfer learning allows models to adapt to a domain even with relatively insufficient data to train large models [7]. This has two important effects, as described below, which are exploited in the proposed work.

- The model weights can be initialized from pre-trained models. Thus, we avoid having to perform vast amounts of computation on large-scale datasets.

- The weights used to initialize the model may be frozen and still perform favorably. They may even make the model more generalizable, *i.e.,* less prone to overfitting.

While there are different specific approaches to applying transfer learning for different tasks, for the sake of simplicity, we freeze most weights of the model as initialized from pre-trained weights, only training the network head (output layers of the model). This results in a simple, low-compute, and communication-efficient model for learning.

B. Contributions

While UAVs are an emerging technology finding applications in real-time federated learning, it is important to understand that the cost can limit the number of UAVs being used. Further, since these UAVs do not have direct access to unlimited power, it is crucial that all the processing onboard is as power-efficient as possible. On the other hand, it is likely in realistic scenarios, that GFCs will have a relatively larger power supply since the GFC is a critical and central equipment which is storing and aggregating crucial learning information from the UAVs. Providing a greater power supply to a central entity than all the UAVs also makes the system more scalable, feasible, and cost-efficient to implement. Additionally, any modifications to such a system that reduce the amount of data required to be transmitted would clearly benefit the system.

With these objectives in mind, we summarize the contributions of our work as follows.

- We propose a multi-UAV-based federated transfer learning system for performing image classification, a benchmark for machine learning and other broad applications. In the proposed system, multiple UAVs coordinate with a GFC to communicate cooperatively to optimize data delivery forming a multi-UAV network. Each UAV is equipped with a computing-caching-communication device and a camera to collect ground images.
- We propose a method for adapting transfer learning to a federated learning-based multi-UAV system. We show that the nature of the proposed method naturally results in a significantly large reduction of computation, convergence time, communication, and power required. This makes it more suitable than traditional federated learning for multi-UAV networks.
- The proposed method eliminates the possibility of member inference attacks [13], which can be used to infer input data and violate the inherent privacy benefits of federated learning.

The remainder of the paper is organized as follows. Section II describes the system model and the proposed multi-UAV federated transfer learning-based algorithm. The simulation and results are presented in Section III, and finally, Section IV concludes the paper.

II. Multi-UAV Communication System

In the following subsections, we describe the multi-UAV communication system model and the proposed federated transfer learning model for improved UAV-to-GFC communication.

A. System Model

The considered multi-UAV communication system model is derived from [1]. The channel during a single transmission is assumed to be time-invariant. The channel between UAV and GFC is modeled by the air-to-ground (A2G) channel mode [14]. The A2G channel operates in line-of-sight (LoS) mode with probability expressed as

$$p_{LoS} = \left\{ 1 + \rho_1 \exp\left(-\rho_2 \left[\frac{180 \arctan(\frac{H_B}{d})}{\pi} - \rho_1 \right] \right) \right\}^{-1}, \tag{1}$$

where ρ_1 and ρ_2 are constants that characterize the environment (rural, urban, dense urban, or others), d is the distance between the UAV and the GFC, H_B is the building height in meters which follows the Rayleigh distribution and is expressed as [14]

$$f_{H_B}(x) = \frac{x}{\gamma^2} \exp\left(-\frac{x}{2\,\gamma^2} \right), \tag{2}$$

where γ is an environment-dependent parameter. The non-LoS (NLoS) probability is expressed as $p_{NLoS} = 1 - p_{LoS}$. The channel gain between the uth UAV and the GFC at time instant t is expressed as

$$g_u(t) = \left(\frac{4\pi f_c}{c} \right)^{-2} (d_u(t))^{-\eta} [p_{LoS}\mu_{LoS} + p_{NLoS}\mu_{NLoS}]^{-1}, \tag{3}$$

where f_c denotes the carrier frequency, c denotes the speed of light, $d_u(t)$ denotes the distance between the uth UAV and the GFC at time instant t, η is the path loss exponent, and μ_{LoS} and μ_{NLoS} denote the attenuation factors of the LoS and NLoS links, respectively. The discrete-time narrowband complex channel gain of the communication link between the uth UAV and the GFC at time t is expressed as [15], [16]

$$h_u(t) = \sum_{n=0}^{N_p - 1} \sqrt{g_{u,n}(t)}\, \exp(-j\phi_{u,n}(t)), \tag{4}$$

where N_p denotes the total number of multipath components, $g_{u,n}(t)$ denotes the power gain of the nth multipath component, as given by (3), and $\phi_n(t)$ denotes the phase at time instant t of the nth multipath component. The complex baseband symbols received at the GFC, therefore, are given as

$$r = \sum_{u=1}^{U} h_u\, x_u + n, \tag{5}$$

where x_u denotes the transmitted symbol, h_u is the sampled channel gain of the link between the uth UAV and the GFC, and $n \sim CN(0, \sigma_n^2)$ denotes the complex additive white Gaussian noise (AWGN), with noise variance σ_n^2.

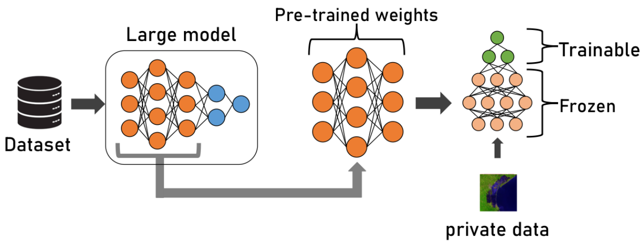

Fig. 1. Typical transfer learning model with frozen weights.

B. Transfer learning

Transfer learning is a technique wherein a portion of the weights of a large deep learning model obtained by training on a very large scale (often public) dataset is used to bootstrap (*i.e.,* initialize) the weights of local models as depicted in Fig.1. The layers thus initialized, called the backbone, are *frozen i.e.,* only used during the forward pass and left untouched during the backpropagation step of model training. Typically, the head (*i.e.,* a few layers at the end of the inference pipeline in a model) is *trainable*. This prevents excess massive computations of the backpropagation of the whole network. Thus, the model utilizes the learned weights *i.e.,* information from a similar but public dataset and *adapts or **transfers*** by fine-tuning to the local (often private) dataset. This is possible because of the similarities in the domain of the public and private datasets. This has been shown to be effective for fine-tuning models on low-resource systems, maintaining model stability and fast model convergence.

C. Federated learning

Federated learning in multi-UAV systems involves four steps, as detailed below.

1) The data collected by the UAV undergoes a training step on the UAV, using the locally available weights.
2) The weights obtained after the local backpropagation are transmitted to the GFC.
3) The weights are aggregated at the GFC.
4) These weights are re-transmitted to the UAV, and they overwrite the local weights of all the UAVs.

In these steps, the first step is the most computationally intensive, the third is the least computationally intensive, and the second and fourth are communication intensive. It is important to understand that typical UAVs are likely to have low onboard power because they use power storage as a power source. However, the GFC is likely to be situated in a secure location and will have better access to continuous and abundant power. It logically follows that any modifications that transfer the computational load from the UAV to a GFC will improve the feasibility of the model. It can further reduce the cost of the UAVs, as they would require significantly less power for computation and simpler computing systems onboard. Another important motivation for federated learning is the increased privacy in processing the data since only the model weights are transmitted to the GFC and not the image data itself. However, the possibility of membership inference attacks [13] may lead to a breach of data privacy. By using only very high-level feature vectors, the proposed technique makes it impossible to infer which input data was used to train the model, thus enhancing the privacy of the model.

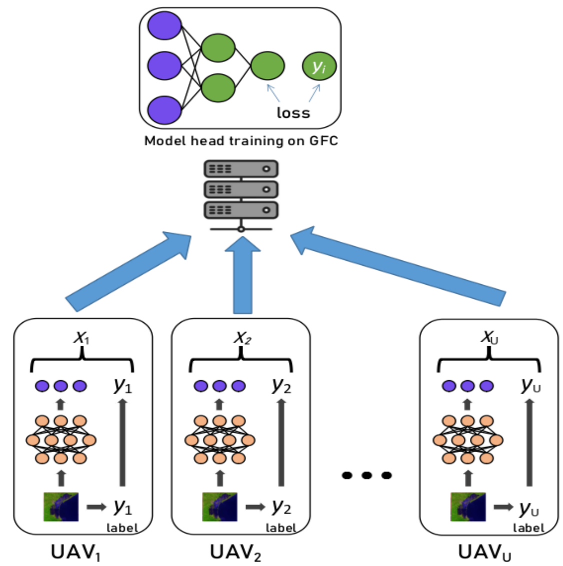

Fig. 2. Federated transfer learning with UAVs.

Algorithm 1: Multi-UAV federated transfer learning.
U UAVs are indexed by u, B is local batch size

Local Processing
  Inputs: Images $x_{i,u}$
  Outputs: Image features $f_{i,u}$ and image labels $y_{i,u}$
  for $u \leftarrow 1$ **to** U **do**
    for $i \leftarrow 1$ **to** B **do**
      compute features $f_{i,u}$ for images $x_{i,u}$, and aggregate them in tensor f'_u
      store the feature with the corresponding label $y_{i,u}$

Single upstream data transmission
  Inputs: Agrregated feature tensors f'_u and ground truth labels y_u
  Outputs: Transmitted feature vectors F_u and y^*_u
  for $u \leftarrow 1$ **to** n **do**
    Transmit f'_u and y_u from UAV_u to GFC
    $F_u = \frac{\hat{h}^*_u \hat{h}_u}{|\hat{h}^2_u|} f'_u$
    $y^*_u = \frac{\hat{h}^*_u \hat{h}_u}{|\hat{h}^2_u|} y_u$

GFC aggregation
  Inputs: F_u, y^*_u at GFC
  Output: Updated weights of deep learning model
  Update model head by backpropagation.

D. Federated Transfer learning

We combine aspects of both federated learning and transfer learning to propose a federated transfer learning model, as depicted in Fig.2. This process involves the following steps.

- The data collected by the UAV undergoes only the forward pass through the neural network layers initialized with the pre-trained weights. The output of this step is a set of features.
- The features and the labels of the data are transmitted to the GFC.
- The features and labels are used to train **only** the trainable layers of the model on the GFC.

These steps modify and improve the federated learning system in the following ways. Firstly, the UAVs only perform a forward pass of the network. In the next step, it is to be noted that we transfer the features *along with the labels*. The proposed model transmits labels and features (which are compressed representations of the local data), and this information would not be adequate to violate the privacy of the data. In fact, the transmission of only features may help improve the privacy and security of the proposed model, for example, by preventing potential membership inference attacks [13]. Since features and labels are vastly smaller in comparison to model weights, this step reduces the transmission burden. In the final step, we train the final layers using the features and labels centrally on the GFC. While this increases the computational burden on the GFC, it is still far lower than performing the training step on the entire network. We have not included a back-transfer of weights from GFC to the UAVs. This is because the processing on UAVs in the first step only uses the same set of initially loaded pre-trained weights. Hence, the proposed learning model reduces the computational burden, shifts it from the UAVs to the GFC, and reduces the communication burden. The data processing, transmission, and aggregation scheme is described by Algorithm1.

III. Simulation and Results

We use the PyTorch [17] library for both the deep learning model training as well as communication systems simulation, allowing us to perform all the operations on a GPU or a CPU without moving around the data between devices. In all our experiments, we use an Nvidia RTX A5000 to simulate the communication system and train the deep learning model. We use two state-of-the-art deep-learning image classifiers:

- Vision Transformer [18] (variant `ViT_B_16`)
- ResNet50 [19].

We have chosen to demonstrate our method with these two models, the first being a more recent vision transformer and the second being a standard convolutional neural network (CNN), which has been used to benchmark various computer vision tasks. These are models with openly available weights pre-trained on 1000 classes of the ImageNet [10] dataset (`Imagenet1k`), accessed from `torchvision` [6]. For transfer learning, we freeze the backbone, leaving only the head trainable. The simulations are performed using the CIFAR-100 dataset [20] with a learning rate of `0.0001` and a batch size of `8`. This dataset has 100 classes and is typically more challenging than MNIST-10, Fashion-MNIST, and CIFAR-10 datasets due to its use of color images and 100 classes. We show the results of training by varying these parameters in the form of the Top-1 accuracy measured on the validation dataset. Further, we use $\rho_1 = 4.88$, $\rho_2 = 0.43$, $\gamma = 20$ (assuming a dense urban environment) [1], $f_c = 2.4$ GHz, $\eta = 2.2$, $\mu_{LoS} = 0.1$, and $\mu_{NLoS} = 21$. We show the training results in different scenarios, which outline the nature of the proposed system under different training and transmission parameters.

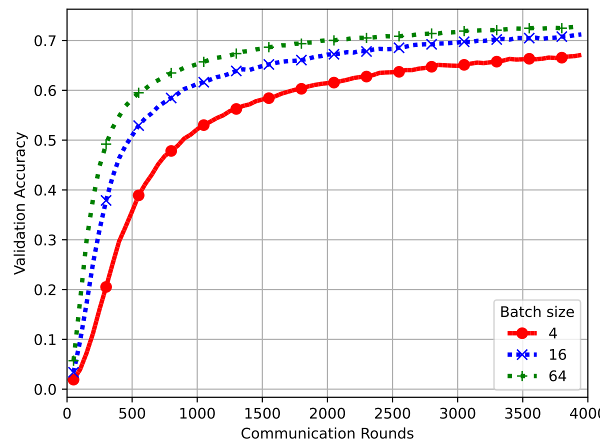

Fig. 3. Validation accuracy versus the communication rounds with varying batch size.

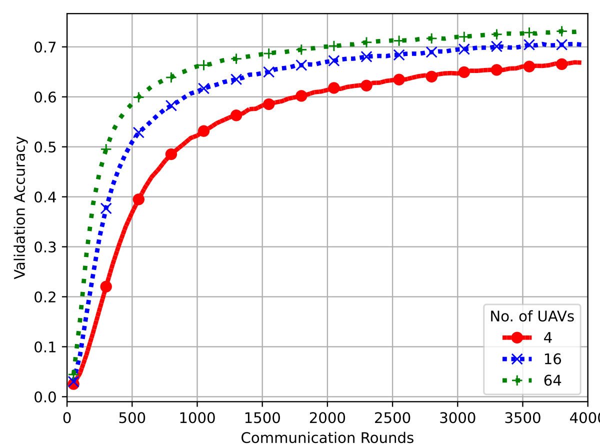

Fig. 4. Validation accuracy versus the communication rounds with varying number of UAVs.

Fig.3 gives a plot of the validation accuracy with communication rounds with a variation in the batch size. It can be observed that on increasing batch size on each UAV, the model attains higher accuracy after fewer transmissions.

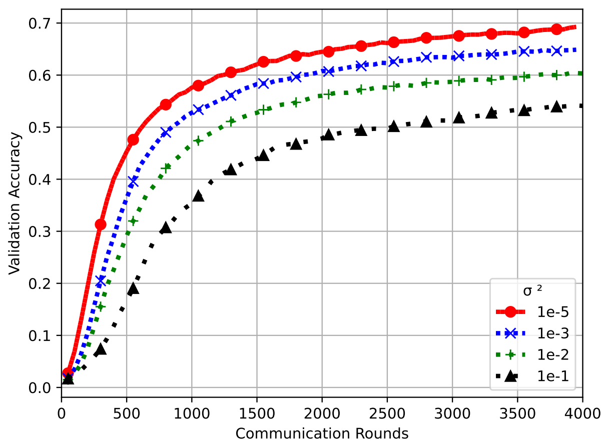

Fig. 5. Validation accuracy versus the communication rounds with the variation of the error magnitude in the communication channel.

Fig.4 shows the variation of the validation accuracy with communication rounds with different numbers of UAVs. It can be observed that on increasing the number of UAVs, the model attains higher accuracy after fewer transmissions.

Fig.5 gives the plot of the validation accuracy with communication rounds with the variation of the magnitude of the error in the communication channel. We see that on increasing the magnitude of the error term up to 10^{-2}, the model converges. Beyond this, the model fails to converge to a high accuracy due to the errors becoming significant.

TABLE I
TIME TAKEN PER COMMUNICATION ROUND.

Dataset	Federated learning	Proposed
ViT_B_16	28.849 ms	11.975 ms
ResNet50	9.333 ms	5.524 ms

TABLE II
DATA TRANSMITTED PER COMMUNICATION ROUND PER IMAGE.

Dataset	Federated learning	Proposed
ViT_B_16	660.459 Mb	0.0029 Mb
ResNet50	179.914 Mb	0.0029 Mb

It is important to note that it is not unusual for the validation accuracy to begin its initial saturation at 0.7. Since the dataset contains 100 classes, this problem is much more challenging than a dataset with 10 classes. For practical use, one may run the model for more epochs, with better hyper-parameter tuning and other techniques, such as data augmentation and learning rate schedulers, to improve accuracy and training times.

We discuss the efficiencies of using the proposed transfer learning-based fine-tuning approach to adapting the model to previously unseen private, local data.

1) **Reduction in compute on the UAV**: TableI shows the time taken per communication round, and it can be seen that the proposed model reduces the computational burden on the UAVs. Since we do not perform backpropagation over the entire model graph, we can use an optimizer such as Adam, with roughly 66% fewer FLOPS, resulting in up to 2.5x time reduction for the same amount of data. However, since we are only inferring the model, we may use other optimization techniques such as quantization, model pruning, and knowledge distillation. These may likely make the model more runtime efficient.
2) **Reduction in data to be transmitted**: TableII shows the data transmitted per communication round per image, and it can be seen that the proposed model drastically reduces the communication burden on the UAVs. Since we only transmit computed feature vectors and labels to the GFC, the proposed model significantly reduces the communication overhead. This method changes the way we communicate in the UAV swarm and the amount of data transmitted. This makes our system low latency, efficient, and uses much lower power.

While the proposed model increases the burden on the GFC, it reduces the burden on the UAVs, making the system more feasible in realistic scenarios.

IV. CONCLUSION

In this work, we have proposed a novel mechanism for federated transfer learning in UAV swarms for efficient data transmission. The results clearly demonstrate the efficacy of the proposed system. The proposed method is particularly useful when large models are readily available, and when we must rapidly adapt to the local data conditions. The proposed method uses Kilobyte scale transmission, therefore, we do not require advanced or sophisticated transmission equipment or a high power supply in UAVs. This work can further be extended to other problems in machine learning, such as object detection and image segmentation.

REFERENCES

[1] H. Zhang, H. Zhang, and L. Hanzo, "Federated learning assisted multi-uav networks," *IEEE Trans. Veh. Tech.*, vol. 69, no. 11, pp. 14 104–14 109, 2020.

[2] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless communications with unmanned aerial vehicles: opportunities and challenges," *IEEE Commun. Mag.*, vol. 54, no. 5, pp. 36–42, 2016.

[3] J. Konecný, H. B. McMahan, D. Ramage, and P. Richtárik, "Federated optimization: Distributed machine learning for on-device intelligence," *ArXiv*, 2016.

[4] P. V. et. al., "Machine learning model sizes and the parameter gap," *ArXiv*, 2022.

[5] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Proc. 27th Int. Conf. Neural Info. Processing Systems - vol. 2*, 2014, pp. 3320–3328.

[6] T. maintainers and contributors. (2016) Torchvision: Pytorch's computer vision library. https://github.com/pytorch/vision.

[7] C. Tan et. al., “A survey on deep transfer learning,” in *Artificial Neural Networks and Machine Learning – ICANN*, 2018, pp. 270–279.

[8] M. Hussain, J. J. Bird, and D. R. Faria, “A study on cnn transfer learning for image classification,” in *Advances in Computational Intelligence Systems*, 2019, pp. 191–202.

[9] S. Ruder, M. E. Peters, S. Swayamdipta, and T. Wolf, “Transfer learning in natural language processing,” in *North American Chapter of the Association for Computational Linguistics*, 2019.

[10] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “Imagenet: A large-scale hierarchical image database,” in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[11] C. Schuhmann et. al., “Laion-5b: An open large-scale dataset for training next generation image-text models,” in *Advances in Neural Information Processing Systems*, vol. 35. Curran Associates, Inc., 2022, pp. 25 278–25 294.

[12] R. Wightman, “Pytorch image models,” https://github.com/rwightman/pytorch-image-models, 2019.

[13] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, “Membership inference attacks against machine learning models,” in *2017 IEEE Symposium on Security and Privacy (SP)*, 2017, pp. 3–18.

[14] A. Al-Hourani, S. Kandeepan, and S. Lardner, “Optimal lap altitude for maximum coverage,” *IEEE Wireless Commun. Lett.*, vol. 3, no. 6, pp. 569–572, 2014.

[15] D. W. Matolak and R. Sun, “Air–ground channel characterization for unmanned aircraft systems—part i: Methods, measurements, and models for over-water settings,” *IEEE Trans. Veh. Tech.*, vol. 66, no. 1, pp. 26–44, 2017.

[16] R. Rajashekar, M. Di Renzo, K. Hari, and L. Hanzo, “A beamforming-aided full-diversity scheme for low-altitude air-to-ground communication systems operating with limited feedback,” *IEEE Trans. Commun.*, vol. 66, no. 12, pp. 6602–6613, 2018.

[17] A. Paszke et. al., “Pytorch: An imperative style, high-performance deep learning library,” in *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[18] A. Dosovitskiy et. al., “An image is worth 16x16 words: Transformers for image recognition at scale,” *ICLR*, 2021.

[19] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[20] A. Krizhevsky, V. Nair, and G. Hinton. (2009) Cifar-10 and cifar-100 datasets. [Online]. Available: https://www.cs.toronto.edu/~kriz/cifar.html