

Exploring the Robustness: Hierarchical Federated Learning Framework for Object Detection of UAV Cluster

Xingyu Li, Wenzhe Zhang, Linfeng Liu, and Jia Xu,

Abstract—The deployment of Unmanned Aerial Vehicle (UAV) cluster is an available solution for object detection missions. In the harsh environment, UAV cluster could suffer from some significant threats (e.g., forest fire hazards, electromagnetic interference, and ground-to-air attacks), which could lead to the destruction of UAVs and loss of data. To this end, we propose a Hierarchical Federated Learning Framework for Object Detection (HFL-OD) to enhance the robustness of UAV cluster conducting object detection missions. In HFL-OD, UAVs are grouped through a Three-Dimensional (3D) graph coloring method, and an intragroup backup mechanism is provided to prevent the data loss caused by the destruction of UAVs. Besides, a dynamic server selection mechanism deals with the potential destruction of servers (cluster server and group servers) by adaptively reassigning the server roles. To further improve the robustness and mission efficiency of UAV cluster, a two-tier federated learning framework is introduced to make a proper trade-off between object detection accuracy and communication/computational overhead. This framework is built on the concept of hierarchical federated learning by implementing both intragroup parameter aggregation and global parameter aggregation. Extensive simulations and comparisons demonstrate the superior performance of our proposed HFL-OD, i.e., the robustness of UAV cluster conducting object detection missions can be significantly improved, and the communication/computational overhead is effectively reduced.

Index Terms—unmanned aerial vehicle cluster; mission robustness; object detection; hierarchical federated learning; 3D graph coloring.

I. INTRODUCTION

In recent years, with the rapid advancement of Internet of Things (IoT) technology and the widespread deployment of 5G networks, the applications of Unmanned Aerial Vehicles (UAVs) have been expanded largely. Among them, Unmanned Aerial Vehicles Object Detection (UAV-OD), as one of the fundamental applications of UAVs, has garnered considerable interest [1], [2], [3]. Several UAVs constitute a UAV cluster, and many sophisticated missions can be carried out.

However, when deploying the UAV-OD model into a new mission scenario, the generalization capability of the model is usually unsatisfactory due to the time-varying diversification of the mission scenarios and mission objectives [4]. Therefore, it becomes imperative to train the UAV-OD model during the

object detection missions of the UAV cluster, allowing it to adapt to the mission scenarios. With regard to a UAV cluster, a Federated Learning (FL) framework [5] is suitable for conducting the object detection missions, since each UAV can perform the gradient descents to train the model parameters locally based on its local dataset, thus significantly reducing the communication/computational overhead.

When the UAV cluster utilizes the real-time data to train the UAV-OD model, due to the harshness environment, the UAV cluster could suffer from some threats (e.g., forest fire hazards, electromagnetic interference, and ground-to-air attacks) seriously, and some UAVs in the UAV cluster could be destroyed. As illustrated in Fig. 1, the UAV cluster is confronted with some threats, and the destruction of some UAVs could lead to the failure of the object detection missions due to the data loss of these UAVs.

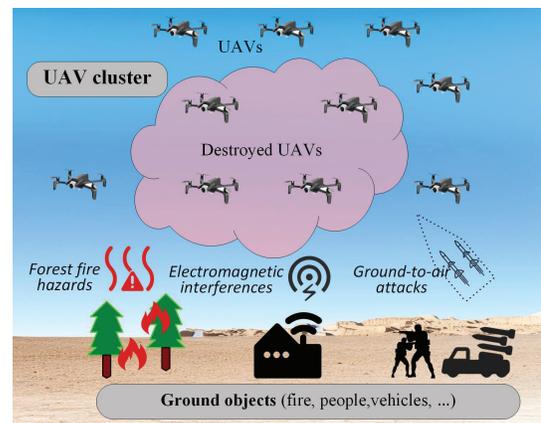


Fig. 1: UAV cluster confronted with some threats.

It is vital to prioritize the robustness of UAV cluster against the destruction of some UAVs, and several considerations are provided as follows:

(i) Intuitively, to enhance the robustness of UAV cluster, UAVs should be grouped, and the UAVs in the same groups share and backup the local data (local business data¹ and local model parameters) to avoid the data loss when some UAVs are destroyed and the performance decline of object detection missions.

¹The term "business data" refers to the data (e.g. the images of ground objects) collected from the visual coverage of the UAV cluster that is essential for model training.

X. Li, W. Zhang, L. Liu, and J. Xu are with the School of Computer Science and Technology, Nanjing University of Posts and Telecommunications, Nanjing 210023, China, and also with the Jiangsu Key Laboratory of Big Data Security and Intelligent Processing, Nanjing 210023, China ({2024040407,1223045409,liulf,xujia}@njupt.edu.cn).

Corresponding author: Linfeng Liu, email: liulf@njupt.edu.cn.

(i) Naturally, the groups can be formed based on the location distribution of UAVs, making the UAVs in the same groups spatially adjacent [6]. However, when an airspace inhabited by one or more groups is threatened, there is a significant risk of losing all UAVs in those groups and the associated data, and the business data collected from the visual coverage of the UAV cluster is incomplete, because the UAVs in the same groups typically collect and maintain the same/similar business data regarding the same/adjacent ground area. Thereby, the remaining data could be insufficient for conducting the object detection missions. To this end, we adopt a Three-Dimensional (3D) graph coloring method instead of the traditional location-based grouping method for the UAV cluster. As depicted in Fig. 2, the UAVs with the same color are classified into the same group to ensure that the UAVs in each group are dispersedly distributed to avoid the spatial aggregation of the UAVs in the same group. This mechanism ensures that when an airspace is threatened, the performance decline of object detection missions can be largely relieved, even though all UAVs falling into the airspace have been destroyed.

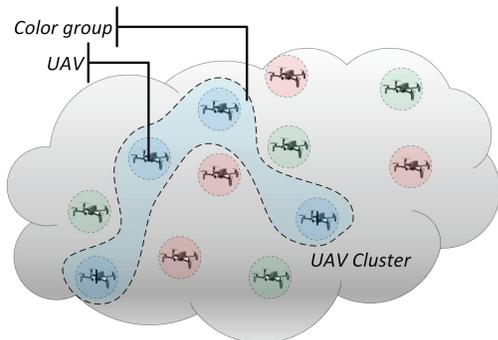


Fig. 2: Graph coloring method for grouping UAVs.

By the two mechanisms mentioned in (i) and (ii), the local data of the destroyed UAVs can be reserved by the surviving UAVs as much as possible, thus greatly enhancing the robustness of the UAV cluster.

(iii) In an FL framework, the cluster server and group servers are responsible for the global parameter aggregation and intragroup parameter aggregation, respectively, and a dynamic server selection mechanism is adopted to deal with the potential destruction of cluster server and group servers. The dynamic server selection mechanism periodically reselects the cluster server and group servers based on the reputations of UAVs. The reputations of UAVs are evaluated by some factors (data similarity, distribution uniformity deviation, residual battery electricity, and energy consumption), which can measure the impacts of the destruction of UAVs on the performance of object detection missions.

Moreover, considering the communication/computational overhead of UAVs conducting the object detection missions, we specially design a two-tier FL framework: (a) In the lower tier, each UAV trains a local model based on its local dataset, and then the intragroup parameter aggregation is implemented among the UAVs in the same group to expedite the iterative optimization of the local model, which facilitates the generation

of the optimal group model; (b) In the upper tier, the global parameter aggregation is implemented among all groups to implicitly share the group model parameters among different groups, collaboratively training the optimal global model. The local training manner can reduce the training complexity and expedite the training process on UAVs. Besides, the training performance can be guaranteed through implementing the global parameter aggregation (these model parameters are obtained by the local training on all UAVs).

The main contributions of this paper are summarized as follows: (a) We propose the HFL-OD, specifically designed to enhance the robustness of UAV cluster in harsh environment. HFL-OD enables the robust object detection missions conducted by a UAV cluster where some UAVs could be destroyed. (b) A 3D graph coloring method is developed to group UAVs, and this method disperses UAVs to avoid the severe situation where all UAVs in the same group are destroyed. We design an intragroup backup mechanism to ensure the redundancy and recovery of local datasets of UAVs when some UAVs are destroyed. (c) Against the destruction of some UAVs, a two-tier FL framework is introduced to preserve the performance of object detection missions and reduce the communication/computational overhead as much as possible. In this framework, a dynamic server selection mechanism is also adopted to address the potential destruction of servers, further improving the robustness of the UAV cluster.

The remainder of this paper is organized as follows: Section II briefly surveys some existing related studies. Section III provides a system model and problem formulation for the robustness of the UAV cluster. Section IV proposes the HFL-OD. Section V covers some further analyses on HFL-OD, including complexity, robustness, and group number. Simulation results for performance evaluation of HFL-OD are reported in Section VI. Finally, Section VII concludes the paper.

II. RELATED WORK

A. Grouping of UAV Cluster

Recently, the integration of UAVs with mobile edge computing has become a promising research topic. However, due to the limited computational power of UAVs, it is necessary to form a UAV cluster to complete the complex missions, and the UAV cluster can largely broaden the scope of mission scenarios of the unconnected UAVs. The UAV cluster enables the synergistic cooperations among UAVs to execute the complex missions efficiently and cost-effectively, e.g., the object detection missions in the realistic environment. The collaborative approach shows potential for advancing the UAV cluster technology in various applications [7]. For example, [8] proposes an improved YOLO algorithm that can be applied to UAV cluster for object detection.

[9] classifies the architectures of UAV cluster into four types: centralized form, distributed form, multi-group form, and multi-layer form. As shown in Fig. 3, the last three forms typically require some UAVs to act as the backbone to facilitate the communications among all UAVs. For the grouping and role assignments in the UAV cluster, [10] transforms the grouping problem into a multiple traveling salesman problem.

Likewise, [11] investigates the selection criteria for the role assignments in UAV cluster, including heuristic clustering, mobile ad hoc networks clustering, position-based clustering, weight-based clustering, destination-based clustering, and server selection.

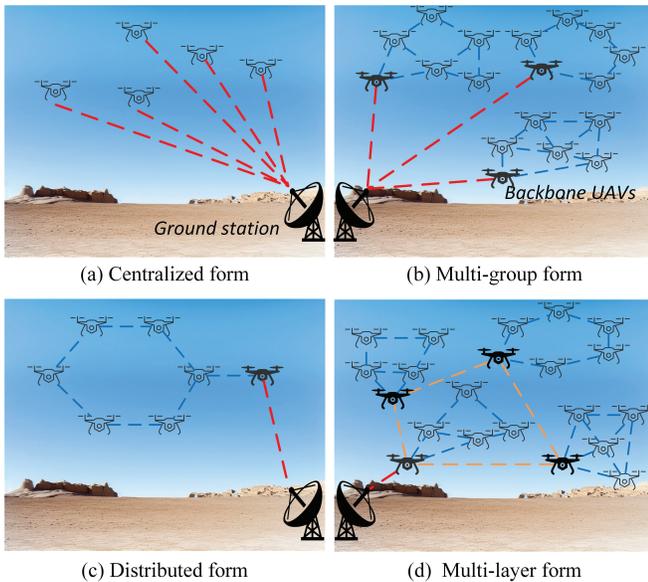


Fig. 3: Architectures of UAV cluster.

For the issue of server selection, [12] proposes an enhanced gray wolf algorithm to optimize the server selection in UAV cluster. UAVs are grouped according to the velocity and distance similarity, and the optimal server is selected according to the residual energy, UAV degree, and communication condition. In addition, [13] designs a multi-objective clustering model to accurately and reasonably select the server by considering the energy consumption, residual energy, packet loss rate, and transmission delay.

It is crucial to enhance the robustness of UAV cluster against the destruction of some UAVs. Furthermore, the server selection should be adaptive to the dynamic situation of UAV cluster.

B. Federated Learning Innovations

Distributed Machine Learning (DML) is initially designed for the computer cluster, and has proven to be highly effective in training the large-scale Machine Learning (ML) models. DML addresses the challenges of high computational complexity, massive training data, and large-scale model.

FL is a promising case of DML, and FL has been applied in some existing works. For instance, [14] proposes a Federated Meta-Learning (FML) framework, where a model is first trained across a set of source users, and then can be quickly adapted to achieve the real-time edge intelligence. To address the vulnerability of the meta-learning framework, an FML framework is further proposed based on distributed robust optimization. Furthermore, [15] introduces an algorithm that utilizes a non-uniform device selection scheme to accelerate the convergence. [15] integrates the user selection and resource

allocation, and employs two first-order approximation techniques [16] to reduce the computational complexity. As for the aggregation methods, [17] proposes a framework termed FedProx to tackle the heterogeneity in federated networks. FedProx provides the convergence guarantees when learning over non-Independent and Identically Distributed (non-IID) data. Additionally, [18] presents a new algorithm which uses the control variates (variance reduction) to correct for the client-drift in its local updates. [19] provides a novel FL method for training neural network models distributively, where the server orchestrates cooperations between a subset of randomly chosen devices.

[20] proposes a DML architecture that combines Split Learning (SL) and FL to jointly train the learning models deployed on UAVs. UAVs with satisfactory channel qualities and local model updates are selected to participate in the global model updates. This architecture can provide higher learning accuracy than FL and smaller communication overhead than SL under both IID dataset and non-IID dataset. Moreover, [21] provides SplitFed Learning (SFL) that combines the parallel processing mechanism in FL and the network splitting mechanism in SL. SFL can split the mission model and perform parallel processing between clients and the server.

To resolve the issue of high communication resource consumption which is associated with the parallel training, [22] proposes a client-edge-cloud Hierarchical Federated Learning (HFL) framework that allows multiple edge servers to perform the partial parameter aggregation, thus resulting in faster training and better communication-computation trade-off. Likewise, [23] develops a communication-efficient HFL framework. This framework employs an adaptive algorithm to determine the aggregation intervals. The client-edge aggregation interval decreases slowly, while the setting of edge-cloud aggregation interval adapts to the ratio between the client-edge propagation delay and edge-cloud propagation delay. Furthermore, [24] gives a hierarchical game framework to observe the dynamics of edge association and resource allocation in the HFL framework. The hierarchical game framework utilizes an evolutionary game to model the dynamics of edge server association. Then, a Stackelberg differential game is used to model the strategies of the optimal bandwidth allocation and reward allocation. In [25], an optimization-based communication resource constrained HFL framework is designed to minimize the generalization error of the autonomous driving model using hybrid data and model aggregation. [26] presents a three-fold FL framework for training deep learning models collaboratively, without the need of sharing local data among the construction robots. The proposed method can leverage the potential of big data while protecting the data privacy. [27] provides a rapid-converged heterogeneous HFL framework (FedRC) to address the inter-city data heterogeneity and accelerate the convergence rate.

[28] demonstrates the feasibility of implementing an FL framework over wireless networks. During the training process, some specially-designed methods can be employed to minimize the results of loss functions [29]. Additionally, there have been some precedents of conducting FL framework for the object detection missions of UAV cluster, such as [30].

Besides, [31] has verified that applying the FL into object detection missions can effectively address the challenge of building some object detection models on centrally-stored large-size training datasets.

The above studies focus on the practical applications of FL in IoT. There are two key considerations in these applications: minimizing the computational complexity during the parameter aggregation and enhancing the resource utilization of devices. When applying FL to UAV cluster and adopting the hierarchical Peer-to-Peer (P2P) architecture, the mission efficiency of the UAV cluster can be significantly enhanced, and the communication/computational overhead can be largely reduced.

C. UAV Cluster Robustness Assurance

Despite extensive research on UAV communications, path planning, and mission collaborations, the robustness of the UAV cluster remains a great challenge.

With regard to the robustness issue, [32] explores the biological robustness and designs a reliable UAV cluster to resist the UAV failures, thereby ensuring the reliable end-to-end communications. In addition, [33] investigates the effect of erasure codes on cost-effective data storage at the edges, aiming to minimize the storage cost while ensuring that all users can be served. The problem in [33] is mapped into an integer linear programming problem, which is NP-hard problem.

When some UAVs in the UAV cluster are destroyed, it is vital to improve the robustness of the UAV cluster and maintain the missions undertaken by the UAV cluster. Regarding the robustness of the UAV cluster, [34] proposes a self-healing mechanism that finds alternative links to bypass the destroyed UAVs. [35] proposes a self-healing trajectory planning algorithm that utilizes a monitoring mechanism and a graph convolutional neural network to identify the recovery topology of the UAV cluster.

However, the UAV cluster may suffer from some threats. To this end, our work introduces a novel approach by implementing a 3D graph coloring method to group UAVs in the UAV cluster. This method disperses UAVs to avoid the severe situation where all UAVs in the same groups are destroyed. Moreover, an intragroup backup mechanism is realized by the data fault tolerance method to ensure the redundancy and recovery of local datasets of UAVs. Thus, the data of destroyed UAVs could be restored during the object detection missions, thus greatly bolstering the robustness of the UAV cluster. Our work proposes a two-tier FL framework specially tailored for the object detection missions of the UAV cluster.

III. SYSTEM MODEL AND PROBLEM FORMULATION

We first describe the robustness problem of the UAV cluster. TABLE I provides an overview of the main notations. Time is divided into discrete time slots with an equal length of t_s , and some relevant definitions are given as follows:

The distribution of UAVs in the UAV cluster typically resembles a cloud-like structure. In the UAV cluster, each UAV can be considered as a node occupying different coordinates

TABLE I: Main notations

Parameter	Description
\mathcal{U}	UAV cluster
χ	Number of groups in \mathcal{U}
t^*	Update epoch (number of time slots)
G_k	The k -th group in \mathcal{U}
N_d	Number of destroyed UAVs
$V_d(G_k)$	Set of destroyed UAVs in group G_k
$p(v_i)^{(t)}$	Coordinate of UAV v_i at the t -th time slot
$D(v_i, t)$	Local dataset of UAV v_i at the t -th time slot
$D_B(v_i, t)$	Business data of UAV v_i in $D(v_i, t)$
$B(v_i)^{(t)}$	Business data collected from the visual coverage of UAV v_i at the t -th time slot
$D(G_k, t)$	Dataset of group G_k at the t -th time slot
$D_L(G_k, V_d(G_k))$	Data loss of group G_k due to the destroyed UAVs in $V_d(G_k)$
$OH(\mathcal{U}, \chi)$	Communication overhead and backup overhead of \mathcal{U}

in a 3D airspace. Therefore, we employ the 3D graph coloring method to group UAVs in the UAV cluster.

To evaluate the performance of the object detection missions, the effectiveness of object detection missions can be measured by the mean Average Precision (mAP) and detection accuracy of ground objects.

A. UAVs

Suppose there are N UAVs in a UAV cluster denoted by $\mathcal{U} = \{v_1, \dots, v_N\}$. The coordinate of a UAV v_i at the t -th time slot is denoted by $p(v_i)^{(t)} = \{x_i, y_i, z_i\}$. We assume that all UAVs in the UAV cluster are trustable and their cooperations are reliable, without any malicious attackers or data stealers. Due to the fact that the communication range of a UAV is typically large (e.g., the communication range of a UAV is 400 m in [36], and the maximum communication range of a UAV even reaches 200 km [37]), we assume that in the UAV cluster each UAV can directly communicate with others.

The local dataset of a UAV v_i at the t -th time slot is denoted by $D(v_i, t)$ which is comprised of two parts: (a) The business data collected from the visual coverage of v_i denoted by $D_B(v_i, t) = \{B(v_i)^{(0)}, \dots, B(v_i)^{(t)}\}$; (b) The business data backed up and shared with other UAVs in the same group (suppose v_i belongs to the group G_k), i.e., $\bigcup_{v_j \in G_k \setminus v_i} D_B(v_j, t)$.

B. UAV Groups

In our work, all UAVs are taken as the participants to collaboratively train the object detection model of the UAV cluster \mathcal{U} . \mathcal{U} is divided into χ groups by a graph coloring method.

The dataset of each group is updated every t^* time slots, through each UAV sharing its local dataset with the group server (every t^* time slots). In G_k , the group server g_k obtains the group dataset $D(G_k, t)$ by consolidating the local datasets received from all UAVs in G_k . Then, the group dataset is shared among the UAVs in G_k .

$$D(G_k, t) = \bigcup_{v_i \in G_k} D(v_i, t). \quad (1)$$

Assuming that the UAV cluster suffers from some threats, making N_d UAVs destroyed, and the set of destroyed UAVs in the group G_k is denoted by $V_d(G_k)$.

C. Objective Functions

To evaluate the robustness of the UAV cluster, the problem objectives are given as follows:

$$\begin{cases} \max mAP', \\ \min \sum_{k=1}^X D_L(G_k, V_d(G_k)), \\ \min OH(\mathbf{U}, \chi), \end{cases} \quad (2)$$

where mAP denotes the mAP of object detection missions, and mAP' denotes the mAP under the destruction of some UAVs. $\frac{mAP'}{mAP}$ represents the performance maintenance of the object detection missions under the destruction of some UAVs. $D_L(G_k, V_d(G_k))$ denotes the data loss. $OH(\mathbf{U}, \chi)$ denotes the sum of communication overhead and backup overhead.

To maximize $\frac{mAP'}{mAP}$, a two-tier FL framework is introduced to preserve the performance of object detection missions and reduce the communication/computational overhead as much as possible. Additionally, an intragroup backup mechanism ensures the redundancy and restoration of local datasets of UAVs, and helps to output the superior training results. To minimize $\sum_{k=1}^X D_L(G_k, V_d(G_k))$, we will develop a 3D graph coloring method to group the UAVs to avoid the severe situation (all UAVs in the same group are destroyed). Furthermore, the intragroup backup mechanism can improve the robustness of UAV cluster and relieve the negative impacts of destroyed UAVs on the performance of object detection missions. As for $\min OH(\mathbf{U}, \chi)$, HFL-OD makes a proper trade-off between object detection accuracy and communication/computational overhead by properly setting the number of groups. The decrease of the number of groups results in higher communication/computational overhead of UAVs, which can confine the number of global epochs.

In the next section, we specify the design of HFL-OD, where UAVs are grouped by a 3D graph coloring method. UAVs in the same groups share and backup the local business data to avoid the data loss and the performance decline of object detection missions when some UAVs are destroyed. Moreover, each UAV can perform the gradient descents for the training of local model to minimize the training loss based on the local dataset.

IV. ROBUSTNESS FRAMEWORK FOR OBJECT DETECTION MISSIONS OF UAV CLUSTER

As outlined in Section I, it is crucial to prioritize the robustness of the UAV cluster confronted with the potential destruction of some UAVs. In response to this imperative, the UAV cluster is first grouped by a balanced graph coloring method, where the UAVs in the same groups share and backup the local business data. Moreover, we specially design a dynamic server selection mechanism and a two-tier FL framework for the object detection missions. As shown

in Fig. 4, a detailed overview of our proposed HFL-OD is provided.

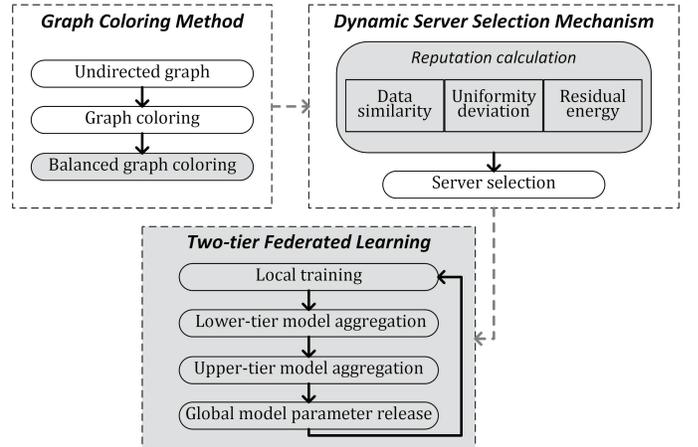


Fig. 4: Overview of HFL-OD.

A. Graph Coloring Method

A balanced graph coloring method is adopted to achieve a uniform distribution of UAVs in the same group. This method ensures that when an airspace is threatened, the performance decline of object detection missions can be largely relieved, even though all UAVs falling into the airspace have been destroyed. In the UAV cluster \mathbf{U} , if the euclidean distance $L_2^{(t)}(v_i, v_j)$ between two UAVs v_i and v_j satisfies that $L_2^{(t)}(v_i, v_j) \leq d_{max}$ (d_{max} denotes the maximum distance for establishing the edge between two UAVs), and then there exists an edge $e^{(t)}(v_i, v_j)$. The coordinates of all UAVs constitute the vertex set $P^{(t)}$, and all edges between UAVs constitute the edge set $E^{(t)}$. An undirected graph regarding the UAV cluster is expressed as $(P^{(t)}, E^{(t)})$.

The graph coloring denotes the mapping of each vertex to a color such that the adjacent vertices are assigned with different colors [38]. The set of vertices with the same color is taken as a color group. The total number of colors is termed coloring number or chromatic number [39], denoted by χ .

In our work, the grouping of the UAV cluster by assigning colors to UAVs is completed by the graph coloring method during χ iterations. In the beginning, all UAVs in \mathbf{U} are not assigned to any color group, and the set of uncolored UAVs $U_0^{(0)} = \mathbf{U}$. During the i -th iteration (suppose at the t -th time slot), the set of uncolored UAVs is denoted by $U_i^{(t)}$, and a new color is given to establish a new color group $CG_i^{(t)}$ that differs from the colors marked in previous iterations.

An uncolored UAV v_θ is randomly selected from $U_i^{(t)}$. We examine the edge set $E^{(t)}$ to identify the adjacent UAVs of v_θ , denoted by $V_\theta^{(t)}$, where each UAV v_ε ($v_\varepsilon \in V_\theta^{(t)}$) satisfies that $e^{(t)}(v_\theta, v_\varepsilon) \in E^{(t)}$. If none of the UAVs in $V_\theta^{(t)}$ falls into $CG_i^{(t)}$, and then v_θ is colored by the new color (v_θ is assigned into $CG_i^{(t)}$). Then, the set of uncolored UAVs is updated as: $U_i^{(t)} \leftarrow U_i^{(t)} \setminus v_\theta$. The above process will be repeated until the set of uncolored UAVs is empty and each UAV in \mathbf{U} has been assigned to a color group.

After completing the graph coloring, we further balance these color groups. Different from the equitable graph coloring [40] which requires that each color group has the same size (the number of UAVs in each color group is equal to γ , where $\gamma = \frac{N}{\chi}$), our proposed HFL-OD allows the slight difference in the size of color groups. The color groups whose size is greater than γ are referred to as the over-size groups, while those whose size is smaller than γ are referred to as the under-size groups. In the over-size groups, some UAVs could be recolored by the colors associated with the under-size groups. Thus, each color group is approximately assigned γ UAVs.

An example is given in Fig. 5, where the number of UAVs is 100 ($N = 100$), and χ is set to 5. Note that UAVs in the same color groups are dispersedly distributed to avoid the spatial aggregation. The balanced graph coloring ensures that when an airspace including some UAVs is attacked, the performance decline of the object detection missions can be relieved, even though all UAVs in the airspace are destroyed.

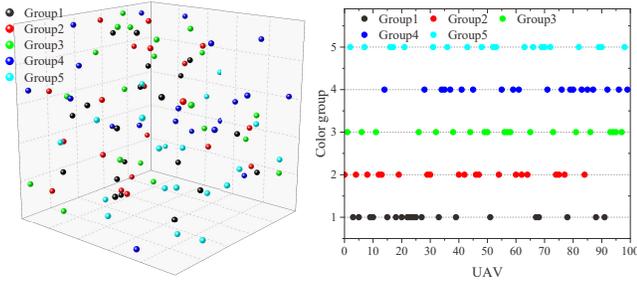


Fig. 5: Example of balanced graph coloring.

The overhead of conducting the 3D graph coloring method is tolerable, because the 3D graph coloring process is conducted before the object detection missions of the UAV cluster. The 3D graph coloring is calculated by a designated UAV, and the graph coloring results are then sent back to all UAVs to complete the grouping.

B. Dynamic Server Selection

After grouping UAVs, one UAV in each color group is selected as the group server responsible for the intragroup parameter aggregation. To mitigate the risk of the destruction of group servers, we employ a dynamic server selection mechanism to update the group servers periodically. The group servers are selected on basis of the reputations of UAVs (measured by data similarity, distribution uniformity deviation, residual battery electricity, and energy consumption).

(i) Data similarity: Data similarity is taken to quantify the alignment between the business data collected by different UAVs. A larger data similarity of a UAV contributes to a larger reputation, indicating that the business data of the UAV matches the standard dataset more closely. For example, the data similarity of a UAV v_i is expressed as:

$$DS(v_i) = \frac{cov(\vec{d}_i, \vec{d}_s)}{\sigma_i \cdot \sigma_s}, \quad (3)$$

where $cov(\vec{d}_i, \vec{d}_s)$ denotes the covariance between the data vector \vec{d}_i of v_i and the data vector \vec{d}_s of standard dataset. σ_i denotes the standard deviation of the data vector of v_i , and σ_s denotes the standard deviation of the data vector of standard dataset. These data vectors are obtained by applying the feature extraction method to the business data, which is converted into high-dimensional numerical vector. $cov(\vec{d}_i, \vec{d}_s)$ is computed as:

$$cov(\vec{d}_i, \vec{d}_s) = \frac{1}{n-1} \sum_{j=1}^n (d_{i,j} - \mu_i)(d_{s,j} - \mu_s), \quad (4)$$

where $d_{i,j}$ and $d_{s,j}$ denote the j -th elements of vectors \vec{d}_i and \vec{d}_s , respectively. μ_i and μ_s denote the mean values of the respective data vectors. n represents the dimension of the feature vectors.

The value of data similarity falls into the numerical interval [0,1]. A larger data similarity implies that the UAV is more valuable and reputable for the object detection missions.

(ii) Distribution uniformity deviation: To enhance the robustness of UAV cluster, the spatial aggregation of group servers should be prevented as well, i.e., the group servers of different groups should also be uniformly distributed as much as possible. The reputation of each UAV is also evaluated by the distribution uniformity deviation (the spatial uniformity of group servers). For example, the uniformity deviation of a UAV v_i is calculated by:

$$UD(v_i) = \left| \frac{N_i}{\chi} - \frac{V(v_i)}{V(U)} \right|, \quad (5)$$

where $V(v_i)$ denotes the volume of the cube centered on v_i , $V(U)$ denotes the volume of the 3D space where UAV cluster is located. N_i denotes the number of group servers in the cube. If the group servers are uniformly distributed in the UAV cluster, the ratio of the volume of the cube centered on v_i to $V(U)$ should be close to the ratio of the number of group servers in the cube to the total number of group servers. Therefore, the difference between these two ratios can be used to measure the distribution uniformity of group servers. The value of distribution uniformity deviation falls into the numerical interval [0,1]. Note that the reputation of a UAV is inversely related to its distribution uniformity deviation, implying that the reputation of the UAV decreases as the distribution uniformity deviation increases.

(iii) Residual battery energy: Most of the battery energy of UAVs is spent on flights. The propulsion power consumption of a UAV for the flight movement is given by [41], [42], [43]:

$$P_{move} = \sqrt{\frac{Q^3}{2\pi \cdot r_p^2 \cdot n_p \cdot \alpha}} + \frac{\tilde{P} - \bar{P}}{\tilde{\nu}} \cdot \nu_f + \bar{P}, \quad (6)$$

where Q , r_p , n_p , and α denote the gravity of UAV, propeller radius, number of propellers, and air density, respectively. ν_f denotes the flight speed of the UAV. \tilde{P} and \bar{P} denote the hardware power levels when the UAV is moving at full speed $\tilde{\nu}$ and when the UAV is hovering, respectively. When the UAV hovers around a point to collect the business data (e.g. the images of ground objects) in the visual coverage, the power consumption P_{hover} is calculated by substituting $\nu_f = 0$ in

(6). Besides, the energy consumption for moving from a point c to another point c' is written as:

$$E_{move}^{c,c'} = \frac{\|c - c'\|}{\nu_f} \cdot P_{move}. \quad (7)$$

To conduct the collaborative object detection missions in the UAV cluster, UAVs transmit the business data when they are hovering. We assume that the hovering time of each UAV is equal to the time spent on the data transmission. Hence, the energy consumption of the UAV v_i hovering around a hovering point (e.g. the point c) is given by:

$$E_{hover}^c = \frac{D_{i,c}}{\zeta} \cdot (P_{hover} + P_{com}), \quad (8)$$

where $D_{i,c}$ denotes the amount of business data that is collected and transmitted by v_i , ζ denotes the average transmission rate of UAVs, and P_{com} denotes the communication power of each UAV.

Hence, by integrating (6) into (7), the total energy consumption of a flying UAV v_i is written as:

$$\begin{aligned} E_i &= \sum_{k=0}^K E_{hover}^{c_k} + \sum_{k=0}^K \sum_{\substack{j=0 \\ j \neq k}}^K E_{move}^{c_k, c_j} \\ &= \sum_{k=0}^K \frac{D_{i,c}}{\zeta} \cdot \left(\sqrt{\frac{Q^3}{2\pi \cdot r_p^2 \cdot n_p \cdot \alpha}} + \bar{P} + P_{com} \right) \\ &\quad + \sum_{k=0}^K \sum_{\substack{j=0 \\ j \neq k}}^K \frac{\|c_k - c_j\| \cdot (\bar{P} - \bar{P})}{\tilde{\nu}} \\ &\quad + \sum_{k=0}^K \sum_{\substack{j=0 \\ j \neq k}}^K \frac{\|c_k - c_j\|}{\nu_f} \cdot \left(\sqrt{\frac{Q^3}{2\pi \cdot r_p^2 \cdot n_p \cdot \alpha}} + \bar{P} \right), \\ &\quad \forall c_k, c_j \in \mathcal{C} \end{aligned} \quad (9)$$

where $\mathcal{C} = \{c, \dots, c_k, \dots, c_K\}$ denotes the set of flight trajectory points of v_i . (9) indicates that E_i is inversely proportional to the flight speed ν_f [41]. We assume that the initial battery energy of all UAVs is the same, denoted by E_{init} . Then, the residual battery energy of each UAV (e.g. v_i) is normalized into State of Charge (SOC) to calculate the reputation, i.e., $SOC(v_i) = \frac{E_{init} - E_i}{E_{init}}$.

Combining the three aforementioned factors, the reputation $RP(v_i)$ of UAV v_i is given by:

$$RP(v_i) = DS(v_i) - UD(v_i) + SOC(v_i). \quad (10)$$

Each group server is responsible for aggregating the local model parameters from UAVs in the same color group and will be periodically reselected. In addition, the cluster server is selected from the group servers according to their reputations. Likewise, the cluster server is responsible for aggregating the group model parameters uploaded by group servers. The dynamic server selection mechanism avoids the vulnerability caused by the destruction of group servers or cluster server and greatly enhances the robustness of the UAV cluster.

During the process of dynamic server selection, each UAV calculates and sends the obtained reputation to the current group server. With regard to each color group, the UAV with the largest reputation is selected as the new group server. After that, the new group server informs the intragroup UAVs of the

update of group server. Likewise, the group server with the largest reputation is selected as the new cluster server.

Note that during the object detection missions of UAV cluster, the network topology of the UAV cluster could continuously change. To enhance the robustness, the UAV cluster needs to be regularly updated and maintained. Every update epoch (t^* time slots), the reputations of UAVs are recalculated for the update of the two-tier FL framework (both the group servers and cluster server are reselected), and the local dataset of each UAV is shared with the group server and other UAVs in the same group.

To further enhance the robustness and fairness of the dynamic server selection, the current servers (group servers and the cluster server) are excluded from participating in the next selection epoch. This ensures dynamic selection by altering the server roles among UAVs and preventing any single UAV from acting as a server multiple times, except in unavoidable cases, e.g., when the group size γ is smaller than the number of global epochs κ_3 , it is inevitable that some UAVs could act as servers multiple times.

C. Two-tier Federated Learning

With regard to the model training for the object detection missions, the primary objective is to identify an optimal mapping function $\mathcal{H}_w : \mathcal{X} \rightarrow \mathcal{Y}$, where \mathcal{X} denotes the set of training samples, \mathcal{Y} denotes the corresponding ground-truth labels, and w denotes the model parameters. By minimizing the value of the sample-wise loss function $l(\mathcal{H}_w(\mathcal{X}), \mathcal{Y})$, the optimal model parameters w^* can be obtained. The model training is expressed as:

$$w^* = \arg \min_w F(w) = \arg \min_w \frac{\sum_{x \in \mathcal{X}, y \in \mathcal{Y}} l(\mathcal{H}_w(x), y)}{|\mathcal{X}|}. \quad (11)$$

We adopt a two-tier FL framework (Fig. 6), where the object detection model is trained in a hierarchical and collaborative manner. Each UAV independently updates the local model parameters using the Stochastic Gradient Descent (SGD) method. The model parameter aggregation (using the FedAvg algorithm) is comprised of two stages: lower-tier parameter aggregation and upper-tier parameter aggregation, as shown in Fig. 7. Besides, the UAVs in the same color group exchange the local datasets and model parameters with each other. To reduce the frequent communications among UAVs, each UAV uploads the local model parameters to the group server every update epoch.

(i) Lower-tier parameter aggregation: Each UAV trains a local model based on its local dataset, and the intragroup parameter aggregation is implemented among the UAVs in the same color group to expedite the convergence of the local model. For example, UAV v_i trains the local model based on the local dataset $D(v_i, t)$, and then sends the parameters of the trained local model w_i^* to the group server g_k . The group server g_k aggregates the local model parameters received from all UAVs in the color group G_k , and yields the group model parameters.

(ii) Upper-tier parameter aggregation: This mechanism facilitates the aggregation of group model parameters uploaded by all group servers, providing a mechanism for the implicit

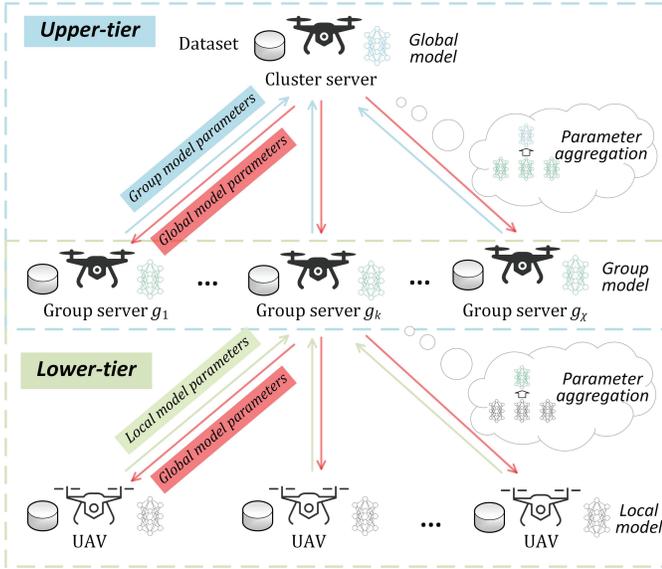


Fig. 6: Two-tier FL framework.

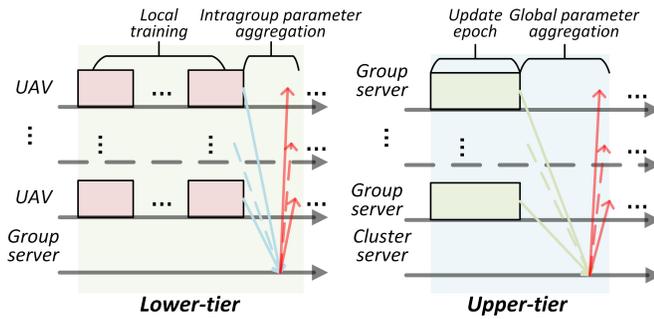


Fig. 7: Training in two-tier FL framework.

data sharing and collaborative enhancement of the performance of the global model. Each group server periodically uploads the group model parameters to the cluster server, and the cluster server implements the parameter aggregation, and finally yields the global model parameters. Then, the cluster server releases the global model parameters to all group servers.

Specifically, at the t -th time slot, the local loss function for UAV v_i is expressed as:

$$F_i(w^{(t)}) = \frac{\sum_{(x_j, y_j) \in D(v_i, t)} l(\mathcal{H}_w(x_j), y_j)}{|D(v_i, t)|}, \quad (12)$$

where $l(\mathcal{H}_w(x_j), y_j)$ denotes the sample-wise loss function quantifying the prediction error of the local model with model parameters w on the training samples x_j and the corresponding labels y_j . The global loss function based on all the datasets of UAVs at the t -th time slot is represented as:

$$F(w^{(t)}) = \frac{\sum_{(x_j, y_j) \in \bigcup_i D(v_i, t)} l(\mathcal{H}_w(x_j), y_j)}{|\bigcup_i D(v_i, t)|} = \sum_{i=1}^N \varsigma_i \cdot F_i(w^{(t)}), \quad (13)$$

where ς_i denotes the model weight of UAV v_i , which is set according to the deviation of the local model parameters deviated from the global model parameters. The learning process

is to minimize the output of $F(w^{(t)})$, i.e., the optimal global model parameters are obtained by: $w^* = \arg \min F(w^{(t)})$.

Based on the received global model parameters $w^{(t)}$, each UAV v_i uses the SGD method to compute the gradient $\nabla F_i(w^{(t)})$ based on its local dataset to update the local model parameters: $w^{(t+1)} = w^{(t)} - \eta \cdot \nabla F_i(w^{(t)})$, where η denotes a learning rate.

The obtained local model parameters will be transmitted to the group server for intragroup parameter aggregation. For example, the group model parameters aggregated by the k -th group server is expressed as:

$$\mathcal{P}_k^{(t)} = \sum_{i \in G_k^{(t)}} \nabla_{\varsigma_i} F_i(w^{(t)}), \quad (14)$$

where ∇_{ς_i} denotes the aggregation weights for UAV v_i ($v_i \in G_k$), and ∇_{ς_i} is defined as:

$$\nabla_{\varsigma_i} = \frac{|D(v_i, t)|}{|D(G_k, t)|}. \quad (15)$$

Then, each group server uploads the group model parameters to the cluster server for global parameter aggregation:

$$\mathcal{P}^{(t)} = \sum_{k=1}^X \nabla_{\varsigma_k} \mathcal{P}_k^{(t)}, \quad (16)$$

where ∇_{ς_k} denotes the aggregation weights of group G_k , and it is defined as:

$$\nabla_{\varsigma_k} = \frac{|D(G_k, t)|}{|\bigcup_{k=1}^X D(G_k, t)|}. \quad (17)$$

Note that the transmissions of all UAVs are carried out in a synchronized manner. The cluster server updates the global model parameters by the FedAvg algorithm. The above process is repeated until the global model has converged.

D. Intragroup Backup Mechanism

In the harsh environment, the UAV cluster is susceptible to some threats which could cause the destruction of some UAVs. To deal with this issue, an intragroup backup mechanism is specially designed to bolster the robustness of the UAV cluster, as shown in Fig. 8. The intragroup UAVs share and backup the local data (local business data and local model parameters) to avoid the data loss and the performance decline of object detection missions when some UAVs are destroyed. The local data of destroyed UAVs can be easily restored through the data backup of surviving UAVs in the same color group.

When one or some UAVs are destroyed, if they are not the servers (group servers or cluster server), the object detection accuracy of HFL-OD will not be affected due to the implement of the intragroup backup mechanism. If they are servers, the object detection accuracy could be slightly affected in the current update epoch, because the servers are periodically reselected (every update epoch) according to the dynamic server selection mechanism. The pseudo-code of HFL-OD is given in Algorithm 1.

Essentially, the intragroup backup mechanism ensures the dataset redundancy, and thus avoids the performance decline of HFL-OD caused by the destruction of some UAVs.

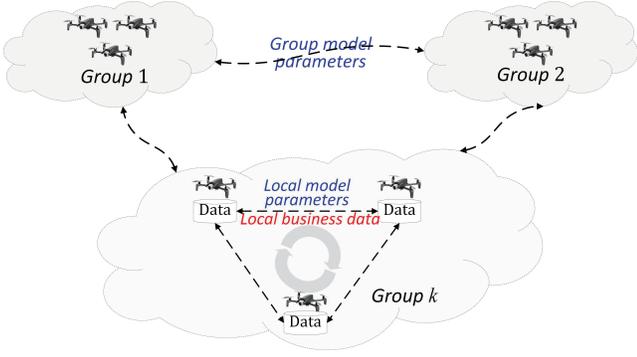


Fig. 8: Intragroup backup in a UAV cluster.

V. THEORETICAL ANALYSIS OF HFL-OD

A. Complexity

TABLE II shows the communication complexity and computational complexity of our proposed HFL-OD.

TABLE II: Complexity of HFL-OD

Module	Communication complexity	Computational complexity
Graph coloring	$O(N + \frac{N^2}{\chi})$	$O(N \cdot m)$
Server selection	$O(\frac{N}{\chi} + \chi^2)$	$O(N)$
Two-tier FL	$O(\kappa_3 \cdot (N + \chi))$	$O(L \cdot K_s^2 \cdot C_{in} \cdot C_{out} \cdot W \cdot H)$
Total	$O(N + \frac{N^2}{\chi} + \frac{N}{\chi} + \chi^2 + \kappa_3 \cdot (N + \chi))$	$O(L \cdot K_s^2 \cdot C_{in} \cdot C_{out} \cdot W \cdot H + N \cdot m + N)$

1) *Communication complexity*: In the graph coloring process, UAVs send their current coordinates to a designated UAV, which is responsible for calculating the graph coloring results and sending the results back to all UAVs, which incurs a communication complexity of $O(N)$. Moreover, UAVs in the same color groups could exchange the local business data with each other, leading to a communication complexity of $O(\frac{N^2}{\chi})$. Consequently, the communication complexity for the graph coloring is written as $O(N + \frac{N^2}{\chi})$.

In the dynamic server selection, each UAV calculates the reputation and sends it to the current group server. The new group server informs the intragroup UAVs of the group server update. Thus, the communication complexity for information exchange reaches $O(\chi \cdot (\frac{N}{\chi^2} + \chi))$, where $O(\frac{N}{\chi^2} + \chi)$ denotes the communication complexity of information exchange in each color group.

In the two-tier FL framework, each UAV uploads its local model parameters to the group server for the intragroup parameter aggregation, and the group model parameters are then uploaded to the cluster server. The cluster server releases the global model parameters to the group servers and then to all UAVs. The communication complexity for this process is up to $O(N + \chi)$. Assuming κ_3 epochs are required for the model training, the communication complexity reaches $O(\kappa_3 \cdot (N + \chi))$.

Therefore, the total communication complexity of HFL-OD is of $O(N + \frac{N^2}{\chi} + \frac{N}{\chi} + \chi^2 + \kappa_3 \cdot (N + \chi))$.

Algorithm 1 Pseudo-code of HFL-OD.

Require: UAV cluster, number of groups, training epoch, update epoch.

- 1: **UAV Grouping**
- 2: UAVs are grouped by a balanced graph coloring method.
- 3: **Dynamic Server Selection**
- 4: **while** Every update epoch (t^* time slots) **do**
- 5: The reputation of each UAV is recalculated based on data similarity, distribution uniformity deviation, and residual battery energy.
- 6: UAVs with the highest reputation are selected as group servers and/or cluster server.
- 7: **end while**
- 8: **Intragroup Data Backup**
- 9: **while** Every update epoch (t^* time slots) **do**
- 10: **for** Each UAV **do**
- 11: Local data is shared among group server and UAVs in the same group.
- 12: **end for**
- 13: **end while**
- 14: **Two-Tier Federated Learning**
- 15: **while** Every training epoch **do**
- 16: **for** Each UAV **do**
- 17: Local model is trained.
- 18: **end for**
- 19: **end while**
- 20: **while** Every update epoch (t^* time slots) **do**
- 21: **for** Each UAV **do**
- 22: Local model parameters are uploaded to group server.
- 23: **end for**
- 24: **for** Each group server **do**
- 25: Local model parameters are aggregated to update group model.
- 26: Group model is uploaded to cluster server.
- 27: **end for**
- 28: Cluster server aggregates group models to update global model.
- 29: Global model is released to group servers and UAVs.
- 30: **end while**

2) *Computational complexity*: The graph coloring is implemented through a sequential greedy scheme, and thus the computational complexity is of $O(N \cdot m)$, where m denotes the maximum degree in the UAV cluster. The additional computational overhead required for achieving the balanced graph coloring does not surpass the upper bound of the original graph coloring. Thus, the computational complexity of the graph coloring is written as $O(N \cdot m)$.

In the dynamic server selection, UAVs calculate the reputations for the selection of group servers. The reputation calculation incurs a computational complexity of $O(N)$.

In the two-tier FL, each UAV trains a local object detection model. Supposing the input image dimension is of $W \times H$, and the convolution kernel size is of $K_s \times K_s$. The number of input/output channel in each layer is denoted by C_{in} and C_{out} , respectively. There are L layers, A anchor boxes, ξ classes,

and the predictions made across S scales with the reduced dimension $W' \times H'$ for the feature maps. Thus, the complexity contribution of the convolutional layers is approximated as $O(L \cdot K_s^2 \cdot C_{in} \cdot C_{out} \cdot W \cdot H)$. For the prediction layers, where each anchor box predicts a bounding box (center coordinates and dimensions), a confidence score, and class probabilities. The complexity is approximatively written as $O(S \cdot A \cdot (5 + \xi) \cdot W' \cdot H')$. Therefore, the computational complexity for training the object detection model is approximated as:

$$O(L \cdot K_s^2 \cdot C_{in} \cdot C_{out} \cdot W \cdot H + S \cdot A \cdot (5 + \xi) \cdot W' \cdot H'), \quad (18)$$

where W' and H' are typically smaller than W and H , which depend on the network structure and input dimension. Therefore, by training the object detection model in the two-tier FL framework, the total computational complexity is of $O(L \cdot K_s^2 \cdot C_{in} \cdot C_{out} \cdot W \cdot H + N \cdot m + N)$.

B. Model Convergence

Each UAV executes κ_1 training epochs of the local model parameters before uploading them to the group server. Then, the group server aggregates the local model parameters. Every κ_2 aggregations, the group model parameters are uploaded to the cluster server. This procedure ensures that the global model parameters are updated every $\kappa_1 \cdot \kappa_2$ training epochs. For the convergence analysis, we focus on the discrepancy between the global model parameters aggregated at the K -th epoch (denoted by \mathcal{P}_K), and the optimal model parameters \mathcal{P}_K^* . Assuming that \mathcal{P}_K^* is obtained after κ_3 parameter aggregations on the cluster server. For any UAV (e.g. v_i), the loss function $F_i(w^{(t)})$ is ρ -continuous, β -smooth, and non-convex, and there is:

$$\|w^{(K)} - w^*\| \leq H(\kappa_1 \cdot \kappa_2, \eta), \quad (19)$$

where

$$H(\kappa_1 \cdot \kappa_2, \eta) = h(\kappa_1 \cdot \kappa_2, \Delta, \eta) + h(\kappa_1, \delta, \eta) + \kappa_1 \cdot \kappa_2 \cdot \frac{(1 + \eta \cdot \beta)^{\kappa_1 \cdot \kappa_2} - 1}{(1 + \eta \cdot \beta)^{\kappa_1} - 1} \cdot h(\kappa_1, \delta, \eta). \quad (20)$$

For example, $h(\kappa_1, \delta, \eta)$ is defined as:

$$h(\kappa_1, \delta, \eta) = \frac{\delta}{\beta} \cdot [(\eta \cdot \beta + 1)^{\kappa_1} - 1] - \eta \cdot \beta \cdot \kappa_1. \quad (21)$$

In (20), δ and Δ denote the gradient divergence at the UAV level and the group level, respectively. δ and Δ can measure the non-IIDness of the data distribution. Essentially, a larger gradient divergence indicates a more pronounced deviation away from the ideal IID data distribution, highlighting the heterogeneity inherent in the business data collected or processed by different UAVs or groups. Specifically, when the business data is IID ($\delta = \Delta = 0$), there exists $H(\kappa_1 \cdot \kappa_2, \eta) = 0$, implying that the global model parameters can converge [22], [44].

In the situation of non-IID data, the hierarchical aggregation in HFL-OD helps reduce the heterogeneity in business data by the intragroup parameter aggregation. Considering the delay-sensitive requirement of object detection missions in HFL-OD begin with deploying a pre-trained model, i.e., HFL-OD

is initialized from $w^{(0)}$, $F_{inf} = F(w^*)$, and \mathbf{U} is divided into χ color groups. After K ($\kappa_1 \cdot \kappa_2 \cdot \kappa_3 \leq K$) local updates, the expected average-squared gradients of $F(w^{(K)})$ is bounded by:

$$\begin{aligned} & \frac{\sum_{k=1}^K \eta \cdot \|\nabla F(w^{(k)})\|^2}{K \cdot \eta} \\ & \leq \frac{[\frac{\chi-1}{N} + (1-\eta)^{\kappa_1 \cdot \kappa_2}]^{\kappa_3} \cdot [F(w^{(0)}) - F(w^*)]}{K \cdot \eta} \\ & + \frac{(1 - [\frac{\chi-1}{N} + (1-\eta)^{\kappa_1 \cdot \kappa_2}]^{\kappa_3}) \cdot \rho \cdot \kappa_3 \cdot H(\kappa_1 \cdot \kappa_2, \eta)}{K \cdot \eta} \\ & + \frac{\beta^2 \cdot \kappa_1 \cdot \kappa_2 \cdot \kappa_3 \cdot \|H(\kappa_1 \cdot \kappa_2, \eta)\|^2}{K \cdot \eta} \\ & - \frac{[\frac{\chi-1}{N} + (1-\eta)^{\kappa_1 \cdot \kappa_2}]^{\kappa_3} \cdot \beta^2 \cdot \kappa_1 \cdot \kappa_2 \cdot \kappa_3 \cdot \|H(\kappa_1 \cdot \kappa_2, \eta)\|^2}{K \cdot \eta}. \end{aligned} \quad (22)$$

As $K \rightarrow \infty$, (22) converges to 0.

C. Setting of Group Number

After performing the balanced graph coloring, the number of UAVs in each color group is approximately the same. Note that the number of color groups (group number) χ is strongly related to the group size and the communications among UAVs. In addition, since we adopt an intragroup backup mechanism in each color group, the variation of χ leads to the non-IIDness of group datasets, thus affecting the training effect of the object detection model. To obtain the optimal setting of χ , the objective function is rewritten as:

$$\min_{\kappa_3, \chi} R \cdot [F(w^{(0)}) - F(w^*)] + (1 - R) \cdot (\rho \cdot \kappa_3 \cdot H(\kappa_1 \cdot \kappa_2, \eta) + \beta^2 \cdot \kappa_1 \cdot \kappa_2 \cdot \kappa_3 \cdot \|H(\kappa_1 \cdot \kappa_2, \eta)\|^2), \quad (23)$$

where $R = [\frac{\chi-1}{N} + (1-\eta)^{\kappa_1 \cdot \kappa_2}]^{\kappa_3} < 1$, and R is related to κ_3 and χ . For setting the group number χ and the update epoch ($\kappa_1 \cdot \kappa_2$ time slots), it is vital to consider the budget of communication overhead ℓ_c and the backup overhead ℓ_b . Each UAV in the color group G_k is assumed to spend c_k units of resource on communications and b_k units on data backup during each epoch of global parameter aggregation. Accordingly, the total resource consumption over κ_3 global epochs across all color groups is expressed as $OH(\mathbf{U}, \chi) = \sum_{k=1}^{\chi} \frac{\kappa_3 \cdot N \cdot b_k \cdot c_k}{\chi}$.

Moreover, the interval between two global parameter aggregations of the color group G_k incurs a time overhead ι_k against a given time budget T . The number of training epochs within the budget, denoted by $\frac{T}{\iota_k}$, is assumed to follow a Gaussian distribution with the mean of T_{ce} [45].

Based on the above assumptions, the constraints for κ_3 are established as follows: κ_3 must satisfy that $\kappa_3 \leq \frac{\ell_b + \ell_c}{\frac{N}{\chi} \cdot (b_k + c_k)}$ and $\kappa_3 \leq T_{ce} \cdot \chi$. Hence, the optimal value of κ_3 is obtained by:

$$\kappa_3 = \min \left\{ \frac{(\ell_b + \ell_c) \cdot \chi}{N \cdot (b_k + c_k)}, T_{ce} \cdot \chi \right\}. \quad (24)$$

Despite of the value attributed to κ_3 , κ_3 can be expressed in the form of $\kappa_3 = \lambda \cdot \chi$, where λ is a finite real number. For the simplicity, κ_3 is taken into R , and hence there is:

$$Z(\chi) = \left[\frac{\chi - 1}{N} \cdot (1 - (1 - \eta)^{\kappa_1 \cdot \kappa_2}) + (1 - \eta)^{\kappa_1 \cdot \kappa_2} \right]^{\lambda \cdot \chi}. \quad (25)$$

To analyze the effects of χ in (24) under the resource constraints, we observe the monotonicity of (25). By defining $\epsilon = \frac{1}{N} \cdot (1 - (1 - \eta)^{\kappa_1 \cdot \kappa_2})$, (25) is rewritten as:

$$Z(\chi) = [\epsilon \cdot \chi + 1 - \epsilon \cdot (N + 1)]^{\lambda \cdot \chi}, \quad (26)$$

where $\epsilon \cdot \chi + 1 - \epsilon \cdot (N + 1) > 0$ and $\epsilon \in (0, \frac{1}{N})$. We have that $\frac{\partial^2 R(\chi)}{\partial \chi^2} > 0$, indicating that $\frac{\partial R(\chi)}{\partial \chi}$ is monotonically increased with the increase of χ . Consequently, we obtain the following formula:

$$Z(\chi, \epsilon) = \lambda \cdot \ln(1 + \chi \cdot \epsilon - (N + 1) \cdot \epsilon) + \frac{\lambda \cdot \chi \cdot \epsilon}{1 + \chi \cdot \epsilon - (N + 1) \cdot \epsilon}, \quad (27)$$

where $Z(\chi, 0) = 0$ ($\epsilon = 0$), and $\frac{\partial Z(\chi, \epsilon)}{\partial \epsilon} > 0$. According to [45], there is $\chi \in \{1, \dots, \lfloor \frac{N+1}{2} \rfloor\}$.

In the practical applications, the value of $Z(\chi)$ is related to the initial estimations of b_k , c_k , and T_{ce} , which can be obtained by offline calculations during the early training stages, thus facilitating the search of the optimal setting of χ with a logarithmic time complexity of $O(\log \frac{N+1}{2})$.

(24) indicates that the decrease of χ results in higher communication/computational overhead of UAVs, which can confine the value of κ_3 (number of global epochs). Considering the potential destruction of some UAVs, κ_3 should be set small, and we let $\kappa_3 \leq 10$. For the graph coloring process, at least 4 colors are required, and the five-color theorem has been proven as a weaker version [46]. In Section VI.C, we observe the effect of group number on HFL-OD by varying χ from 5 to 100.

VI. PERFORMANCE EVALUATIONS

In this section, we provide comprehensive performance evaluations for our proposed HFL-OD, along with comparisons with other training methods or in different scenarios. Considering the evaluation cost, execution cost, and uncontrollable conditions in the real-world deployment of UAV cluster, we adopt the manner of simulations for performance evaluations. The following simulations are conducted on VisDrone dataset released by Tianjin University (<http://aiskyeye.com/home/>). This dataset is comprised of 10,209 static images captured by cameras installed on UAVs. VisDrone dataset is collected using multiple UAV platforms in different scenarios (e.g., urban and country scenarios), under various weather and lighting conditions. The object detection boxes are manually annotated and defined by the bounding boxes of over 2.6 million common objects, such as pedestrians, cars, bicycles, and tricycles. This dataset also provides some important attributes, including scenario visibility, object classes, and occlusion. Based on this dataset, we simulate the missions of UAVs detecting various ground objects. The main parameter settings for simulations are presented in TABLE III.

TABLE III: Simulation Parameters

Parameter	Description	Value
N	Number of UAVs	100
χ	Number of groups in U	5
d_{max}	Maximum distance in graph coloring	80 m
κ_1	Training epoch (number of time slots)	5
κ_2	Upload interval of group model parameters (number of time slots)	1
t^*	Update epoch (number of time slots)	5
N_d	Number of destroyed UAVs	50
N_s	Number of surviving UAVs	50
η	Learning rate	0.0001
B_s	Batch size	64

In our proposed HFL-OD, YOLOv5 model is employed for the object detection missions. Considering the delay-sensitive requirement of object detection missions in the harsh environment, YOLOv5 model is first pre-trained. During the pre-training phase, the simulation results obtained by YOLOv5 model are observed as follows: (i) Fig. 9 shows the confusion matrix of the pre-trained YOLOv5 model. (ii) Fig. 10 shows the loss value and mAP value of the pre-trained YOLOv5 model. The mAP value reaches 0.329, and the object detection accuracy for cars reaches 0.70. The above results demonstrate that YOLOv5 model is capable of achieving preferable object detection outcomes on the VisDrone dataset.

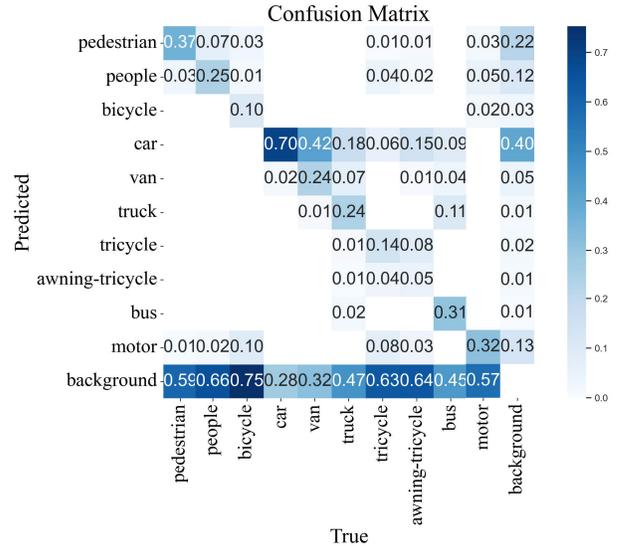


Fig. 9: Confusion matrix (with IoU threshold of 0.6).

In addition, Fig. 9 indicates that cars are easier to be detected by UAVs, due to the following reasons: VisDrone dataset provides more car instances for the model training, and the cars typically have larger size and more distinct shapes/reflecting surfaces compared with other objects in VisDrone dataset, making them easier to be detected from the perspective of UAVs (top-down perspective).

Fig. 11 illustrates that the numbers of objects belonging to different classes are quite different, i.e., the non-IID property is evident. To this end, we use YOLOv5n, a variant of YOLOv5 designed for edge devices such as NVIDIA Jetson Nano,

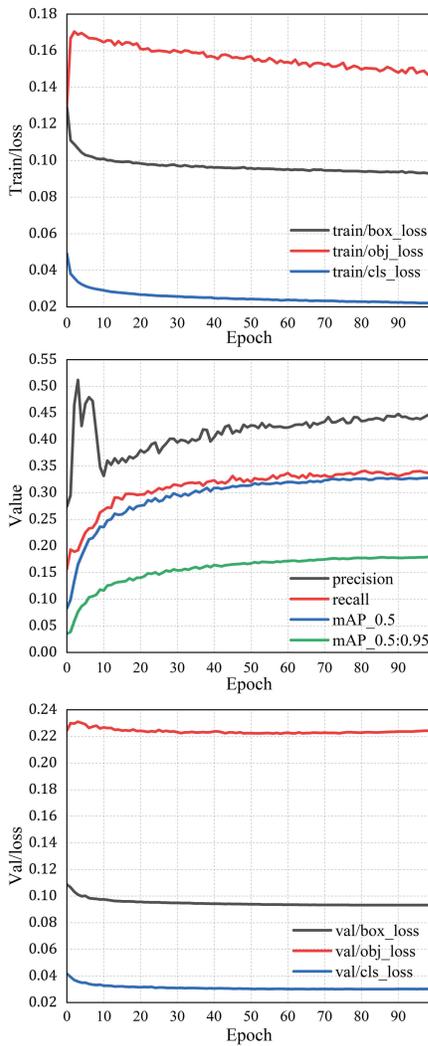


Fig. 10: Loss value and mAP value (This figure shows the curves of training loss and validation loss for bounding box, objectness, and classification, as well as the two metrics precision and recall).

which especially performs well in mobile solutions. YOLOv5n makes a balance between performance and efficiency, which is well-suited to UAV-OD missions, and the low computational requirement of YOLOv5n also aligns well with the constraints of FL.

A. Comparisons among Different Training Methods

To analyze the merits of HFL-OD, we compare HFL-OD with ML, FL, and DML in terms of mAP value, loss value, and training time. ML refers to a centralized learning paradigm where UAVs transmit data to a central server for training the model. DML refers to “Distributed Machine Learning”, a method where the model is locally trained by UAVs in a distributed manner. With DML, each UAV trains a local model without any information exchanges. The key distinction between DML and FL is that DML does not aggregate the model parameters. HierFL [22] is taken as a representative

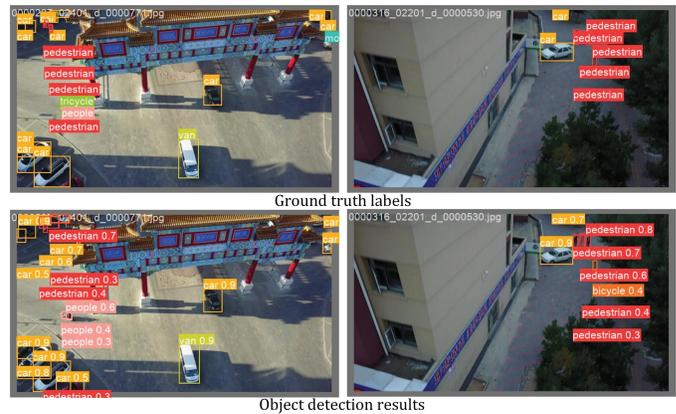


Fig. 11: Examples of object detection results.

HFL framework that allows multiple edge servers to perform the partial parameter aggregation.

The metric mAP value is taken to assess the object detection accuracy. We use mAP50 and mAP50:95 to provide a comprehensive understanding of the object detection accuracy of HFL-OD. mAP50 denotes the mAP value calculated at an IoU threshold of 0.50, and mAP50:95 denotes the average of mAP value calculated at the IoU thresholds ranging from 0.50 to 0.95 with the step size of 0.05.

Fig. 12 indicates that the mAP value obtained by ML is greater than that obtained by HFL-OD, FL, and HierFL, while the loss value obtained by ML is smaller than that obtained by HFL-OD, FL, and HierFL. These phenomena indicate that ML can achieve the highest object detection accuracy among these training methods, because ML is trained based on the complete dataset. Note that the loss value of FL and HierFL is considerably larger than others due to the client drift [47], which inevitably decelerates the model convergence and reduces the object detection accuracy of the models trained based on non-IID datasets.

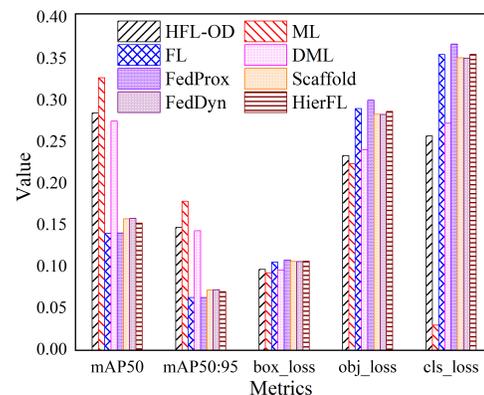


Fig. 12: Comparisons among different methods.

Moreover, FedProx and FL exhibit similar performance, while Scaffold and FedDyn achieve better performance, since that both Scaffold and FedDyn incorporate some measures

to tackle the heterogeneity in FL, and thus they improve the performance and mitigate the impact of non-IID data and insufficient local data. However, HFL-OD outperforms these methods, due to the following mechanisms adopted in HFL-OD: (i) the adoption of HFL framework enhances the robustness of the UAV cluster significantly, and (ii) the utilization of an intragroup backup mechanism helps output the superior training results, making HFL-OD more suitable for the object detection missions of UAV cluster.

As illustrated in Fig. 13, the communication delay of ML is slightly smaller than that of HFL-OD, because HFL-OD involves the transmissions of the model parameters while ML does not. The training time of DML is approximately equal to that of FL, since the communication delay for transmitting the local model parameters is much shorter than the training time. Moreover, the training time of ML is evidently longer than the other two methods, as ML is based on the local datasets of all UAVs. Therefore, ML is not an available solution for the object detection missions of the UAV cluster.

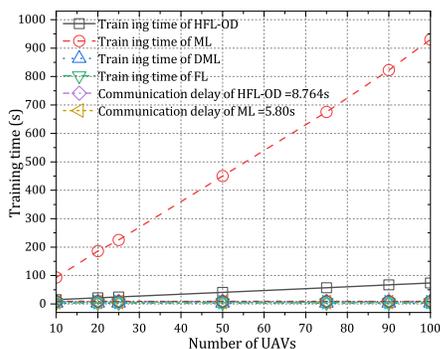


Fig. 13: Comparisons among different frameworks on training time and communication delay (transmission speed of data is set to 20Mbps).

The simulation results presented in Fig. 12 and Fig. 13 indicate that our proposed HFL-OD can make a preferable trade-off between the object detection accuracy and training time (training time is related to the communication/computational overhead).

B. Impact of Intragroup Backup Mechanism on HFL-OD

We consider the destroyed UAVs are randomly selected in the following evaluations due to the unpredictable threats. In HFL-OD, UAVs are grouped through a 3D graph coloring method, and an intragroup backup mechanism is provided to prevent the data loss caused by the destruction of UAVs. Besides, a dynamic server selection mechanism deals with the potential destruction of servers.

Specifically, the intragroup backup mechanism is implemented through a data fault tolerance method to ensure the redundancy and restoration of local datasets of destroyed UAVs. By the intragroup backup mechanism, the local datasets of some destroyed UAVs could be restored, thus significantly enhancing the robustness of the object detection missions.

The intragroup backup mechanism is applied to FL and DML to validate the effectiveness as well. Fig. 14 illustrates the performance of HFL-OD, FL, DML (without the

aggregations of model parameters), FL-G (FL with the intragroup backup mechanism), DML-G (DML with the intragroup backup mechanism), and HierFL [22]. The simulation results in Fig. 14 indicate that FL-G and DML-G perform significantly better than FL and DML, respectively. Additionally, HierFL performs better than FL, indicating that the HFL helps achieve superior training results, and HFL-OD performs better than HierFL due to the intragroup backup mechanism.

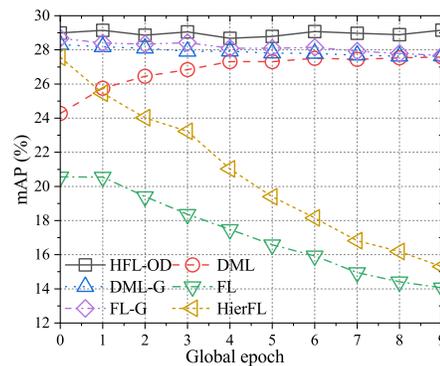


Fig. 14: Effect of intragroup backup mechanism.

We also conduct an ablation experiment to validate the effects of coloring grouping, intragroup backup, and dynamic server selection, respectively. Fig. 15 indicates that each of the three components: coloring grouping, intragroup backup, and dynamic server selection exerts a unique and indispensable effect on the object detection accuracy.

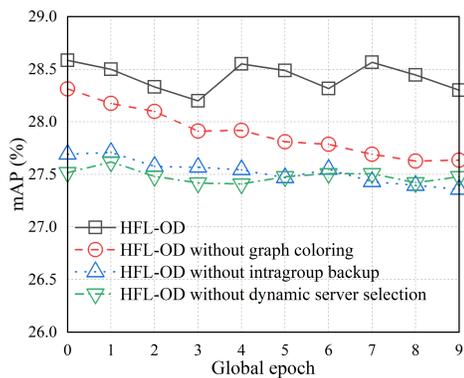


Fig. 15: Ablation experiment.

Furthermore, Fig. 16 demonstrates the impact of destruction of UAVs on these training methods. With the intragroup backup mechanism, HFL-OD, FL-G, and DML-G can restore the data of destroyed UAVs, while FL, DML, and HierF suffer the data loss due to the destruction of UAVs seriously. Fig. 17 shows the variations in the training effect when half of the UAVs have been destroyed. The training effect of FL and HierFL fluctuates wildly due to the data loss, while the training effect of HFL-OD and FL-G almost remains unchanged, implying that the intragroup backup mechanism can largely improve the robustness of UAV cluster and relieve

the negative impacts of destroyed UAVs on the performance of object detection missions.

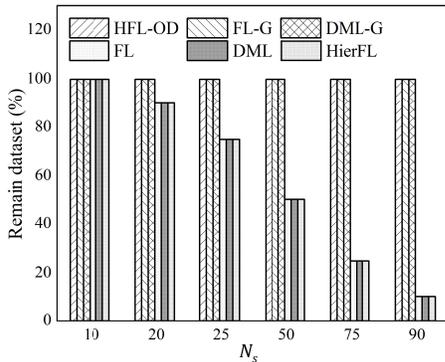


Fig. 16: Data loss among different methods under different N_s (the number of surviving UAVs).

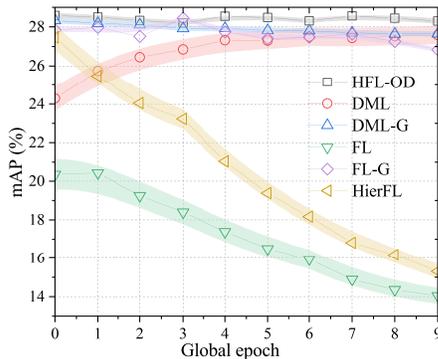


Fig. 17: Training effect of surviving UAVs among different methods.

C. Effect of Group Number on HFL-OD

Two-tier FL is employed in the UAV cluster to reduce the communication overhead. From Fig. 18, we can observe that the mAP value decreases with the increase of χ . Note that χ is increased with the decrease of group size (the increase of heterogeneity among different group datasets). This is because when χ increases, the impact of the non-IID datasets during the training process becomes more pronounced, which degrades the training performance of HFL-OD. Fig. 18 also indicates that the performance of HFL-OD can be enhanced by properly setting the value of χ . The maximum mAP value in Fig. 18 reaches 0.286 when $\chi = 5$.

Moreover, Fig. 19 and Fig. 20 illustrate the results of the object detection missions by varying the value of χ . The results include the object detection accuracy (measured by mAP50 and mAP50:95 in Fig. 19), and the value of three types of loss functions (classification loss (cls_loss), object loss (obj_loss), and box regression loss (box_loss) in Fig. 20). The best performance of object detection missions can be obtained when $\chi = 5$, implying that the object detection

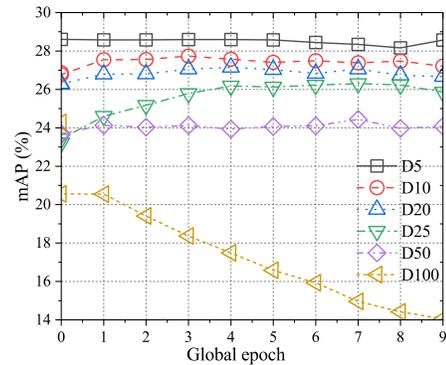


Fig. 18: Impact of χ on mAP value. (The number following D represents χ , which is the number of color groups in the UAV cluster)

accuracy and communication/computational overhead can be balanced by properly setting the value of χ , also enabling the UAV cluster to conduct the object detection missions more efficiently and cost-effectively.

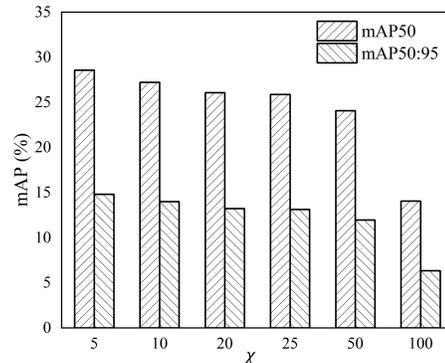


Fig. 19: χ vs. mAP value.

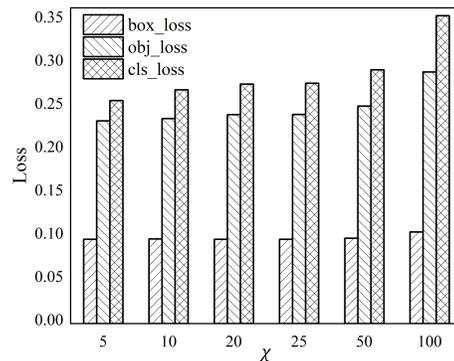


Fig. 20: χ vs. loss value.

D. Robustness

Fig. 21 indicates that the surviving UAVs can maintain the object detection missions after the UAV cluster suffers from the threats (some UAVs are destroyed). As the number of surviving UAVs decreases, the performance of object detection missions declines. Nevertheless, the performance decline of the object detection missions can be obviously relieved by HFL-OD, even though when more UAVs are destroyed. Fig. 21 demonstrates that HFL-OD effectively mitigates the impact of data loss on object detection accuracy. The training effect is well maintained even when the proportion of destroyed UAVs reaches 75%, which verifies the strong robustness provided by HFL-OD.

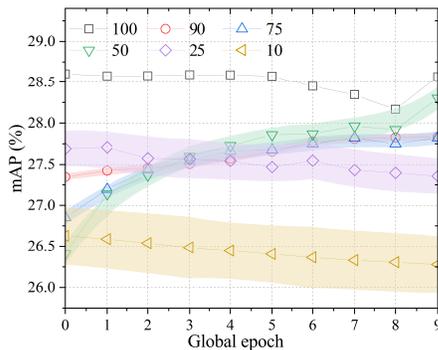


Fig. 21: Impact of surviving UAVs number on mAP value.

By Fig. 22 and Fig. 23, we can observe the effects of the number of surviving UAVs on the object detection accuracy and model loss. As illustrated in Fig. 22 and Fig. 23, the robustness of UAV cluster can be enhanced by HFL-OD, thereby the number of surviving UAVs does not obviously affect the object detection accuracy and model loss. In Fig. 23, the model loss is quite small when the number of surviving UAVs is 75 or 90, compared with the model loss when the UAV cluster remains unaffected. This phenomenon also indicates the potential for a more optimal group dataset configuration, which could further enhance the training effect and model performance on non-IID datasets.

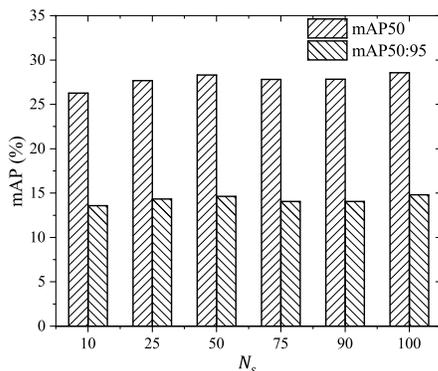


Fig. 22: Robustness of UAV cluster on mAP value.

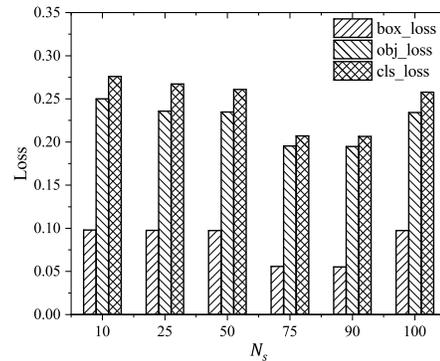


Fig. 23: Robustness of UAV cluster on loss value.

E. Evaluation across Different Datasets

As shown in Fig. 24, we evaluate the generalization capability of HFL-OD across different datasets. Specifically, DOTA dataset is a dataset for object detection in aerial images, which contains 2,806 aerial images with 188,282 instances. UAVDT dataset serves as a challenging UAV detection and tracking benchmark for three fundamental missions in UAV-based vision, i.e., object detection, single-object tracking, and multiple-object tracking. To provide a more intuitive demonstration of the performance, we validate the effectiveness in single-object detection on UAVDT. Additionally, we conduct some evaluations on COCO, which is a widely used benchmark dataset for the general object detection missions. The simulation results demonstrate that HFL-OD consistently exhibits robustness and strong performance across different datasets.

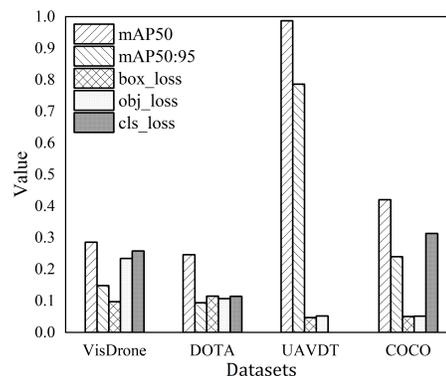


Fig. 24: Evaluations across different datasets.

VII. CONCLUSION

We have studied the robustness of UAV cluster conducting object detection missions, and a Hierarchical Federated Learning Framework for Object Detection (HFL-OD) of UAV cluster has been proposed. In HFL-OD, UAVs are first grouped through a balanced graph coloring method, and an intragroup backup mechanism is provided to avoid the data loss due to the destruction of some UAVs. Specially, a dynamic server selection mechanism is employed to deal with the destruction

of cluster server and group servers. Furthermore, considering the communication/computational overhead of UAVs conducting the object detection missions, a two-tier federated learning framework is proposed to preserve the performance of the object detection missions as much as possible, through enhancing the robustness of UAV cluster.

Some practical issues need to be considered in future when applying our proposed HFL-OD: (i) HFL-OD does not consider the intergroup backup mechanism. The intergroup backup mechanism can be designed by considering the communication/computational overhead and the requirement of privacy/security jointly. (ii) In the harsh environment, when some UAVs are destroyed, it is crucial that the UAV cluster should swiftly relocate to another safe area and promptly reconfigure the cluster topology, which necessitates that the UAV cluster can make the reasonable decisions regarding the flight path planning and topology reconfiguration. (iii) For the protection of data privacy of different UAVs, an erasure coding method or alternative backup strategy could be adopted to avoid the privacy disclosure. (iv) Although HFL-OD primarily leverages the hierarchical aggregation and grouping method to handle the non-IID data, we could further enhance the adaptivity to different non-IID datasets by integrating some advanced FL methods such as Scaffold or FedDyn. (v) The number of color groups should be properly set to make a reasonable tradeoff between object detection accuracy and communication/computational overhead, and it can be adjusted according to different mission scenarios.

ACKNOWLEDGMENTS

This research is supported by National Natural Science Foundation of China under Grant Nos. 62272237, 62372249, 61872191.

REFERENCES

- [1] P. Mittal, R. Singh, and A. Sharma, "Deep learning-based object detection in low-altitude UAV datasets: A survey," *Image and Vision Computing*, vol. 104, 2020.
- [2] X. Wu, W. Li, D. Hong, et al., "Deep learning for unmanned aerial vehicle-based object detection and tracking: A survey," *IEEE Geoscience and Remote Sensing Magazine*, vol. 10, no. 1, pp. 91–124, 2021.
- [3] H. Yu, G. Li, W. Zhang, et al., "The unmanned aerial vehicle benchmark: Object detection, tracking and baseline," *International Journal of Computer Vision*, vol. 128, pp. 1141–1159, 2020.
- [4] K. Wang, X. Fu, Y. Huang, et al., "Generalized UAV Object Detection via Frequency Domain Disentanglement," *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1064–1073, 2023.
- [5] Q. Yang, Y. Liu, T. Chen, et al., "Federated machine learning: concept and applications," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, no. 2, pp. 1–19, 2019.
- [6] O. Senouci, S. Harous, Z. Aliouat, "Survey on vehicular ad hoc networks clustering algorithms: Overview, taxonomy, challenges, and open research issues," *International Journal of Communication Systems*, vol. 33, no. 11, 2020.
- [7] W. Chen, J. Liu, H. Guo, et al., "Toward robust and intelligent drone swarm: Challenges and future directions," *IEEE Network*, vol. 34, no. 4, pp. 278–283, 2020.
- [8] X. Zhu, S. Lyu, X. Wang, et al., "TPH-YOLOv5: Improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios," *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pp. 2778–2788, 2021.
- [9] J. Li, Y. Zhou, L. Lamont, "Communication architectures and protocols for networking unmanned aerial vehicles," *2013 IEEE Globecom Workshops (GC Wkshps)*, pp. 1415–1420, 2013.
- [10] W. Huang, H. Guo, J. Liu, "Task offloading in UAV swarm-based edge computing: Grouping and role division," *2021 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, 2021.
- [11] O. Senouci, S. Harous, Z. Aliouat, "Survey on vehicular ad hoc networks clustering algorithms: Overview, taxonomy, challenges, and open research issues," *International Journal of Communication Systems*, vol. 33, no. 11, pp. e4402, 2020.
- [12] R. Zhang, Y. Gao, Y. Ding, "Research on clustering optimization algorithm for UAV cluster network," *2021 6th International Symposium on Computer and Information Processing Technology (ISCIPT)*, pp. 88–92, 2021.
- [13] Y. Gong, C. Li, F. Wang, et al., "MHCF-CECSO: A novel high-performance clustering framework for industrial IoT," *IEEE Internet of Things Journal*, vol. 11, no. 3, pp. 4942–4955, 2024.
- [14] S. Lin, G. Yang, and J. Zhang, "A collaborative learning framework via federated meta-learning," *2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS)*, pp. 289–299, 2020.
- [15] S. Yue, J. Ren, J. Xin, et al., "Efficient federated meta-learning over multi-access wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 5, pp. 1556–1570, 2022.
- [16] A. Fallah, A. Mokhtari, and A. Ozdaglar, "Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach," *Advances in Neural Information Processing Systems*, vol. 33, pp. 3557–3568, 2020.
- [17] T. Li, A. Sahu, M. Zaheer, et al., "Federated optimization in heterogeneous networks," *Proceedings of Machine learning and systems*, vol. 2, pp. 429–450, 2020.
- [18] S. P. Karimireddy, S. Kale, M. Mohri, et al., "SCAFFOLD: Stochastic controlled averaging for federated learning," *37th International Conference on Machine Learning*, vol. 119, pp. 5132–5143, 2020.
- [19] D. A. E. Acar, Y. Zhao, R. M. Navarro, et al., "Federated learning based on dynamic regularization," *International Conference on Learning Representations*, 2021.
- [20] X. Liu, Y. Deng, and T. Mahmoodi, "Wireless distributed learning: A new hybrid split and federated learning approach," *IEEE Transactions on Wireless Communications*, vol. 22, no. 4, pp. 2650–2665, 2022.
- [21] C. Thapa, P. C. M. Arachchige, S. Camtepe, et al., "Splitfed: When federated learning meets split learning," *AAAI Conference on Artificial Intelligence*, vol. 36, no. 8, pp. 8485–8493, 2022.
- [22] L. Liu, J. Zhang, S. H. Song, et al., "Client-edge-cloud hierarchical federated learning," *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, pp. 1–6, 2020.
- [23] L. Liu, J. Zhang, S. H. Song, et al., "Hierarchical federated learning with quantization: Convergence analysis and system design," *IEEE Transactions on Wireless Communications*, vol. 22, no. 1, pp. 2–18, 2022.
- [24] W. Y. B. Lim, J. S. Ng, Z. Xiong, et al., "Dynamic edge association and resource allocation in self-organizing hierarchical federated learning networks," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 12, pp. 3640–3653, 2021.
- [25] W. B. Kou, S. Wang, G. Zhu, et al., "Communication resources constrained hierarchical federated learning for end-to-end autonomous driving," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Detroit, USA, 2023.
- [26] H. T. Wu, H. Li, H. L. Chi, et al., "A hierarchical federated learning framework for collaborative quality defect inspection in construction," *Engineering Applications of Artificial Intelligence*, vol. 133, 2024.
- [27] W. B. Kou, Q. Lin, M. Tang, et al., "FedRC: A rapid-converged hierarchical federated learning framework in street scene semantic understanding," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Abu Dhabi, 2024.
- [28] N. H. Tran, W. Bao, A. Zomaya, et al., "Federated learning over wireless networks: Optimization model design and analysis," *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*, pp. 1387–1395, 2019.
- [29] S. Wang, T. Tuor, T. Salonidis, et al., "Adaptive federated learning in resource constrained edge computing systems," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 6, pp. 1205–1221, 2019.
- [30] F. Wu, C. Dong, Y. Qu, et al., "CIOFL: Collaborative Inference-Based Online Federated Learning for UAV Object Detection," *2022 IEEE 19th International Conference on Mobile Ad Hoc and Smart Systems (MASS)*, pp. 258–259, 2022.
- [31] Y. Liu, A. Huang, Y. Luo, et al., "Fedvision: An online visual object detection platform powered by federated learning," *AAAI Conference on Artificial Intelligence*, vol. 34, no. 08, pp. 13172–13179, 2020.

- [32] K. Hazra, V. K. Shah, S. Roy, *et al.*, "Exploring biological robustness for reliable multi-UAV networks," *IEEE Transactions on Network and Service Management*, vol. 18, no. 3, pp. 2776–2788, 2021.
- [33] H. Jin, R. Luo, Q. He, *et al.*, "Cost-effective data placement in edge storage systems with erasure code," *IEEE Transactions on Services Computing*, vol. 16, no. 2, pp. 1039–1050, 2022.
- [34] P. Zhou, X. Fang, Y. Fang, *et al.*, "Beam management and self-healing for mmWave UAV mesh networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 2, pp. 1718–1732, 2018.
- [35] Z. Mou, F. Gao, J. Liu, *et al.*, "Resilient UAV swarm communications with graph convolutional neural network," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 1, pp. 393–411, 2021.
- [36] L. Hong, H. Guo, J. Liu, *et al.*, "Toward swarm coordination: Topology-aware inter-uav routing optimization," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 9, pp. 10177–10187, 2020.
- [37] X. Fan, H. Zhou, K. Sun, *et al.*, "Channel assignment and power allocation utilizing NOMA in long-distance UAV wireless communication," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 10, pp. 12970–12982, 2023.
- [38] A. Schrijver, "Combinatorial optimization: Polyhedra and efficiency," *Springer*, vol. 24, no. 2, 2003.
- [39] W.-J. van Hoeve, "Graph coloring with decision diagrams," *Mathematical Programming*, vol. 192, no. 1-2, pp. 631–674, 2022.
- [40] H. Li, B. Zhang, S. Qin, *et al.*, "UAV-Clustering: Cluster head selection and update for UAV swarms searching with unknown target location," *2022 IEEE 23rd International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pp. 483–488, 2022.
- [41] B. Zhu, E. Bedeer, H. H. Nguyen, *et al.*, "Joint cluster head selection and trajectory planning in UAV-aided IoT networks by reinforcement learning with sequential model," *IEEE Internet of Things Journal*, vol. 9, no. 14, pp. 12071–12084, 2021.
- [42] J. Yao, N. Ansari, "QoS-aware power control in internet of drones for data collection service," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 7, pp. 6649–6656, 2019.
- [43] M. B. Ghorbel, D. Rodriguez-Duarte, H. Ghazzai, *et al.*, "Joint position and travel path optimization for energy efficient wireless data gathering using unmanned aerial vehicles," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 2165–2175, 2019.
- [44] L. Liu, Z. Xi, K. Zhu, *et al.*, "Mobile charging station placements in internet of electric vehicles: A federated learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 24561–24577, 2022.
- [45] Z. Wang, H. Xu, J. Liu, *et al.*, "Accelerating federated learning with cluster construction and hierarchical aggregation," *IEEE Transactions on Mobile Computing*, vol. 22, no. 7, pp. 3805–3822, 2023.
- [46] J. Harris, J. L. Hirst, M. Mossinghoff, "Combinatorics and graph theory," *Springer*, 2008.
- [47] S. P. Karimireddy, S. Kale, M. Mohri, *et al.*, "Scaffold: Stochastic controlled averaging for federated learning," *International Conference on Machine Learning*, pp. 5132–5143, 2020.

AUTHOR BIOGRAPHY

Xingyu Li received the B. S. degree in information security from the Nanjing University of Posts and Telecommunications in 2023, where he is currently pursuing the Ph. D. degree in cyberspace security. His current research interest includes vehicular ad-hoc networks and UAV networks.

Wenzhe Zhang received the B. S. degree in information security from the Nanjing University of Posts and Telecommunications in 2023, where he is currently pursuing the M. S. degree in electronic information. His current research interest includes UAV networks and topological repair.

Linfeng Liu received the B. S. and Ph. D. degrees in computer science from the Southeast University, Nanjing, China, in 2003 and 2008, respectively. At present, he is a Professor in the School of Computer Science and Technology, Nanjing University of Posts and Telecommunications, China.

His main research interests include the areas of vehicular ad hoc networks, wireless sensor networks and multi-hop mobile wireless networks. He has published more than 120 peer-reviewed papers in some technical journals or conference proceedings, such as IEEE TMC, IEEE TPDS, IEEE TIFS, IEEE TITS, IEEE TVT, IEEE TSC, ACM TAAS, ACM TOIT, Computer Networks, Elsevier JPDC. He has served as the TPC member of Globecom, ICONIP, VTC, WCSP.

Jia Xu received the Ph. D. Degree in School of Computer Science and Engineering from Nanjing University of Science and Technology, Jiangsu, China, in 2010. He is currently a professor in Jiangsu Key Laboratory of Big Data Security and Intelligent Processing at Nanjing University of Posts and Telecommunications. His main research interests include crowdsourcing, edge computing and wireless sensor networks. Prof. Xu has served as the PC Co-Chair of SciSec 2019, Organizing Chair of ISKE 2017, TPC member of Globecom, ICC, MASS, ICNC, EDGE, and has served as the Publicity Co-Chair of SciSec 2021.