Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

FedS-SLAM: A Federated Semantic Collaborative SLAM System for UAV Swarms in Dynamic

Zhi-jian Yan
North University of China https://orcid.org/0009-0000-3171-8398
Xin-e Li
zn11xe@163.com

North University of China
Chun-sheng Cui
North University of China
Zhi-wei Zeng
North University of China

Research Article

Keywords: Collaborative SLAM, Federated Learning, Dynamic Environments, UAV Swarm, Semantic SLAM

Posted Date: August 26th, 2025

DOI: https://doi.org/10.21203/rs.3.rs-7409201/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

FedS-SLAM: A Federated Semantic Collaborative SLAM System for UAV Swarms in Dynamic

Abstract

Existing SLAM technologies struggle to meet the demands of drone swarms in dynamic, GPS-denied environments. To address this, we propose FedS-SLAM, a federated semantic cooperative SLAM system designed specifically for drone swarms. It innovatively employs a federated learning architecture to drastically reduce communication overhead by exchanging only lightweight "federated features." The system's core is a deeply coupled optimization loop where the SLAM back-end provides high-quality data to train a shared model via federated learning, which in turn enhances the SLAM front-end. This synergy enables the precise filtering of dynamic objects, significantly improving localization robustness. Experiments on multiple public benchmark datasets prove that FedS-SLAM outperforms state-of-the-art systems in both trajectory accuracy and map quality.

Keywords: Collaborative SLAM, Federated Learning, Dynamic Environments, UAV Swarm, Semantic SLAM

1 Introduction

Unmanned aerial vehicle (UAV) swarms, by virtue of their distributed cooperative capabilities, demonstrate tremendous application potential in critical fields such as emergency search and rescue, intelligent logistics, and large-scale environmental mapping[1]. The core prerequisite for fully unleashing their efficacy is that the swarm can achieve highly autonomous navigation in unknown environments without Global Positioning System (GPS) signals[1].Simultaneous Localization and Mapping (SLAM) technology is precisely the foundation that endows UAVs with such capabilities, enabling them to construct environmental maps and determine their own positions within the maps in real time while exploring unknown spaces[1].For UAV swarms, this challenge escalates to Collaborative SLAM (C-SLAM), whereby multiple agents must construct a unified and globally consistent environmental model through collaboration[2].A successful C-SLAM system can endow the swarm with outstanding efficiency and robustness that surpass the capabilities of any single agent[2].Thus, developing a robust and scalable C-SLAM system has become a frontier and core issue in modern robotics research[3].

Although the academic community has achieved remarkable progress, deploying UAV swarms in the real world still confronts a series of severe challenges that existing SLAM systems find difficult to tackle simultaneously.First, the real world is inherently dynamic, teeming with moving objects like pedestrians and vehicles.This severely violates the "static world assumption" on which traditional SLAM algorithms rely, thereby causing significant deviations in pose estimation[4].Second, as the scale of the swarm expands, the communication overhead required for agents to share sensor data and map information quickly becomes a system bottleneck, greatly restricting the system's scalability and real - time performance[5].How to efficiently manage data exchange in a distributed system remains an unsolved conundrum in this field to this day.Moreover, UAVs often operate in scenarios with perceptual degradation, such as indoor low - texture environments, repetitive structures between buildings, or drastic changes in lighting.These situations are highly prone to causing failures in visual feature tracking[1].These three challenges—environmental dynamics, communication limitations, and perceptual uncertainty—together constitute the main barriers that impede the widespread application of autonomous UAV swarms.

Although existing studies have respectively proposed solutions for the aforementioned individual challenges—for instance, DS-SLAM focuses on handling dynamic objects[4],while C-SLAM frameworks such as Swarm-SLAM strive to manage decentralized collaboration[6].there remains a lack of an integrated system capable of efficiently addressing the three critical challenges of dynamics, scalability, and robustness simultaneously.In particular, a holistic solution that can both accommodate large-scale swarms and achieve high communication efficiency still constitutes a gap in the existing literature, and this is precisely the gap that the present research endeavors to fill.

To tackle the aforementioned challenges, we propose FedS-SLAM: a federated semantic collaborative SLAM system designed for UAV swarms operating in complex dynamic environments. Our key contributions are as follows:1)We introduce a novel C-SLAM architecture based on federated learning. By exchanging lightweight abstract "federated features" instead of raw data, it drastically reduces the communication bandwidth requirement.2)We design a multimodal perception front-end that fuses LiDAR, visual, and inertial sensors. Combined with deep learning and Kalman filtering, it enables precise perception and modeling of dynamic objects, thereby ensuring the purity of the map and the accuracy of positioning.3)We propose a semantics-augmented relocalization mechanism, which innovatively incorporates a shared "cross-agent historical relocalization case library." This allows the swarm to learn from collective experience, enhancing robustness in challenging scenarios.

The subsequent chapters of this paper will elaborate on the proposed system in detail and verify its performance. We will first review the relevant work in the fields of collaborative SLAM, dynamic SLAM, and semantic SLAM. Next, we will introduce the complete methodology of the FedS-SLAM framework in detail. Subsequently, we will conduct comprehensive experimental evaluations on public benchmark datasets and perform in-depth analyses of the results. Finally, we will summarize the research findings and discuss future research directions.

2 Related Work

2.1 Collaborative SLAM

Collaborative Simultaneous Localization and Mapping (C-SLAM) is a pivotal technology that enables multi-robot systems to achieve autonomous navigation and operation in unknown environments[1].Existing C-SLAM system architectures are mainly categorized into three types: centralized, distributed, and hybrid[3].Among these, the distributed architecture has garnered significant attention due to its high scalability and robustness; however, its core challenge lies in efficiently managing information exchange between robots[1].Frequent data interactions in large-scale robot swarms generate substantial communication bandwidth pressure and computational load, severely constraining the system's real-time performance and scalability[1].To address this challenge, researchers have strived to design communication-efficient collaborative frameworks.For example, Swarm-SLAM achieves decentralized mapping by exchanging sparsified data[7].The objective of C-SLAM is to construct a globally consistent map by integrating data from different robots, thereby achieving high efficiency and precision that surpass the capabilities of a single robot[1].Nevertheless, in distributed scenarios, how to efficiently allocate limited communication resources remains an unsolved conundrum[7].The FedS-SLAM framework proposed in this research introduces the idea of federated learning into C-SLAM, establishing a novel distributed collaborative architecture.This architecture fundamentally resolves the communication bottleneck issue by having each UAV upload only lightweight "federated features" instead of raw sensor data or dense local maps[8].This method not only significantly reduces the data transmission volume but also provides a feasible solution for the real-time and efficient collaboration of large-scale UAV swarms.

2.2 SLAM in Dynamic Environments

Traditional SLAM (Simultaneous Localization and Mapping) algorithms are predominantly based on the "static world assumption," which posits that all elements in the environment are stationary. This assumption is difficult to uphold in real-world dynamic scenes populated with moving pedestrians and vehicles[4].The presence of dynamic objects introduces erroneous feature matches, which severely interferes with pose estimation and contaminates the map, resulting in degraded localization accuracy or even system failure[9].

Current mainstream approaches for handling dynamic environments can be categorized into two types: methods based on geometric constraints (e.g., epipolar constraint checking) and those based on deep learning (e.g., semantic segmentation or object detection)[10].For instance, the representative DS-SLAM system builds upon ORB-SLAM2 by integrating the SegNet semantic segmentation network to identify and reject feature points from dynamic regions[4].In contrast, our system employs the more advanced YOLOv8 object detection model and innovatively combines it with a Kalman filter to predict the trajectories of dynamic objects.This enables more refined modeling and filtering of dynamic disturbances, thereby enhancing the system's robustness in highly dynamic environments[11].

2.3 Semantic SLAM and Long-Term SLAM

In recent years, SLAM technology has been evolving from purely geometric mapping to a higher level of semantic mapping, endowing robots with the capability for environmental understanding[4].By assigning semantic labels (e.g., "building," "road") to objects in the map, the system can leverage high-level contextual information to enhance the robustness of re-localization and loop closure detection in weakly-textured or repetitive scenes[9].

This advancement has also driven the development of long-term SLAM, which requires systems to be able to adapt to changes in the environment over time[10].To this end, we have innovatively designed a "Cross-UAV Historical Re-localization Case Library," a mechanism that allows a UAV swarm to share and utilize historically successful re-localization experiences[12].This novel collaborative knowledge-sharing model is essentially a form of collective memory oriented towards long-term tasks, greatly enhancing the system's long-term adaptability and robustness when facing environmental changes and perceptual degradation.

2.4 Application of Federated Learning in Robot-UAV Collaboration

Federated Learning (FL), as an emerging distributed machine learning paradigm, allows for collaborative model training while preserving client data privacy, eliminating the need to exchange raw data[13]. Its representative algorithm, Federated Averaging (FedAvg), achieves iterative optimization of the global model by performing a weighted average of the locally updated model parameters from each client on the server side[14].We apply the principles of Federated Learning to the field of multi-robot collaboration, leveraging its inherent advantages in reducing communication overhead and ensuring data locality to effectively address the core challenges in C-SLAM (Collaborative SLAM)[13].

3 Problem Formulation

This chapter will develop a rigorous mathematical model for the FedS-SLAM system. We will begin by defining the overall system architecture and the mathematical notation used. Subsequently, we will formally describe the core sub-problems within the system—federated semantic learning and semantic collaborative SLAM. Finally, we will unify these sub-problems into a joint optimization framework, thereby providing a complete exposition of the core problem of FedS-SLAM.

3.1 System Definition and Mathematical Notations

3.1.1 System Overview

FedS-SLAM is a collaborative SLAM system designed for a swarm of Unmanned Aerial Vehicles (UAVs) operating in dynamic environments. The system is composed of N UAVs, each equipped with an RGB-D sensor, and is coordinated by a central server

Table 1 Comparative analysis with state-of-the-art (SOTA) SLAM systems.

Feature	ORB-SLAM3	LIO-SAM	DS-SLAM	Swarm-SLAM	FedS-SLAM (Ours)
Multi-agent	Multi-map, non-real-time collaboration	No	No	Yes	Yes
System Architecture	-	-	-	Decentralized	Federated (Decentralized)
Dynamic Object Handling	No	No	Semantic Segmentation (SegNet)	No	Object Detection (YOLOv8) + Trajectory Prediction (KF)
Semantic Perception	No	No	Yes	No	Yes
Sensor Modalities	V,VI	L,I	V	V,VI,L,I	V,L,I,mmWave

Note: V = Visual, I = IMU, L = LiDAR, mmWave = Millimeter-wave Radar.

(e.g., a ground station). The core objective of the system is to achieve the joint optimization of the following three tasks:
1) Collaboration Localization and Mapping: To accurately and robustly estimate the six degrees-of-freedom (6-DoF) trajectories of all UAVs in real time.
2) Dynamic Environment Perception: To build a globally consistent, dense 3D semantic map that encompasses both the static structure of the environment and the trajectories of dynamic objects.
3)Federated Semantic Learning: To collaboratively train a shared, high-performance semantic segmentation model via a federated learning mechanism, while preserving the privacy of each UAV's raw visual data.

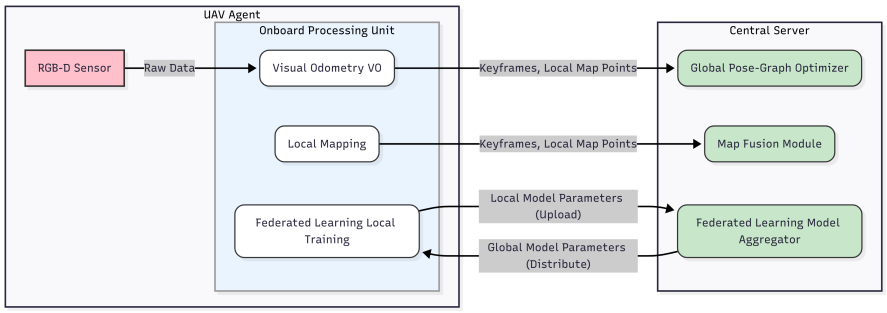


Fig. 1 FedS-SLAM System Architecture and Data Flow Diagram

In this system, each UAV locally executes lightweight tasks such as Visual Odometry (VO), local map construction, and the local training of the semantic model. In

contrast, computationally intensive tasks requiring global information, such as global pose-graph optimization, map fusion, and aggregation of the federated model, are performed by the central server. The role of the semantic model is to provide the SLAM front-end with pixel-level scene understanding to differentiate between static and dynamic elements, thereby improving localization robustness. In turn, the globally consistent pose and map information supplied by the SLAM back-end provides high-quality, spatiotemporally aligned training data for Federated Learning, effectively mitigating the challenges posed by the non-IID (Non-Independent and Identically Distributed) data distributions resulting from different camera perspectives. This closed loop of mutual promotion and co-optimization between SLAM and FL is the core concept of the FedS-SLAM system.

Table 2 List of mathematical symbols

Symbol	definition
$\mathcal{N}$	Set of UAV agents.
$i \in \mathcal{N}$	Index of a UAV agent.
$x_k^i \in SE(3)$	Pose of UAV i at local timestamp k, consisting of rotation $R_k^i \in SO(3)$ and position $p_k^i \in \mathbb{R}^3$.
$\mathcal{X}^i$	The complete trajectory sequence of UAV i,$\mathcal{X}^i = \{x_k^i\}$
$m_j^s \in \mathbb{R}^3$	3D world coordinates of the j-th static landmark.
$o_\iota(t)$	State of the ι-th dynamic object at world time t (e.g., position, velocity, semantic class).
$\mathcal{M}^S$	Set of static landmarks in the global map,$\mathcal{M}^S = \{m_j^s\}$
$\mathcal{M}^D$	Set of dynamic objects in the global map,$\mathcal{M}^D = \{o_\iota(t)\}$
$\mathcal{M}$	The complete global environment map,$\mathcal{M} = \mathcal{M}^S \cup \mathcal{M}^D$
z_k^i	A sensor measurement from UAV i at local timestamp k (e.g., an RGB-D image frame).
θ	Parameters of the globally shared semantic segmentation model.
θ_t^i	Model parameters of UAV i after local training in the t-th round of federated learning.
$\mathcal{D}^i$	Local training dataset of UAV i.
$\mathcal{G}_p$	Global pose graph for the C-SLAM (Collaborative SLAM) problem.
$\mathcal{G}_c$	Communication graph of the UAV swarm.
$h(\cdot)$	Sensor observation model (projection function).
$f(\cdot)$	Agent motion model.
$g(\cdot;\theta)$	Semantic segmentation neural network model with parameters θ.

3.2 Formalization of the FedS-SLAM Problem

This section will systematically construct a complete mathematical model of the FedS-SLAM problem by addressing its individual components, and ultimately propose a unified joint optimization objective function.

3.2.1 Agent and Environment Modeling

UAV State and Motion Model: The state of an individual UAV i at a discrete time step k is its pose $x_k^i = (R_k^i, p_k^i) \in SE(3)$ in the world coordinate frame. Its motion process can be described by a nonlinear state transition model, which is influenced by

the control input u_k^i (such as pre-integration results from an IMU) and the process noise w_k.

$$x_k^i = f(x_{k-1}^i, u_k^i) + w_k, \quad w_k \sim \mathcal{N}(0, Q_k). \tag{1}$$

where $\mathcal{Q}_k$ is the covariance matrix of the process noise.

Environment Model: This is a key step that embodies the advanced nature of the system. We explicitly partition the entities within the world W into two disjoint sets: a set of static background landmarks $\mathcal{M}^S = \{m_j^S\}$ and a set of dynamic semantic objects $\mathcal{M}^D = \{o_\iota(t)$. A static landmark $m_j^S \in \mathbb{R}^3$ is a three-dimensional point in space, which is assumed to have a fixed position in the world frame. These points are typically stable features in the environment, such as corners and edges. The state of a dynamic object $o_\iota(t)$ is time-variant. Its state vector may encompass the object's 3D position $p_\iota(t)$, velocity $\dot{p}_\iota(t)$, dimensions, shape parameters, as well as its semantic class $C_\iota \in \{'person', 'vehicle', \dots\}$. This modeling methodology follows the paradigm of Dynamic Object Tracking (DAT) and aims to provide a more comprehensive description of the environment.

Sensor Model: For an RGB-D sensor on UAV i, its observation $z_{k,j}^i$ of a map entity (whether a static landmark $m_j^S \in \mathbb{R}^3$ or a point on a dynamic object o_ι) at time instant k can be modeled by a measurement function $h(\cdot)$, which is perturbed by the measurement noise v_k.

$$z_{k,j}^i = h(x_k^i, m_j) + v_k, \quad v_k \sim \mathcal{N}(0, R_k). \tag{2}$$

where m_j represents the observed entity, and R_k is the covariance matrix of the measurement noise. For an RGB-D camera, the observation $z_{k,j}^i$ typically consists of the 2D pixel coordinates (μ, ν) of the entity on the image and a depth value d.

3.2.2 Federated Learning Sub-problem

The objective of this sub-problem is to collaboratively train a globally shared semantic segmentation model, $g(\cdot;\theta)$, across the entire UAV swarm in a data privacy-preserving manner. This model takes an RGB image I as input and outputs a semantic label for each pixel in the image.

Local Training Objective: In each round t of federated learning, the objective for a selected UAV i is to update its local model parameters by minimizing a loss function (e.g., cross-entropy loss) over its local dataset $\mathcal{D}_t^i$. This dataset consists of image-label pairs that are collected locally and pose-aligned by the SLAM system. Starting from the global model parameters w_t, the new local parameters w_{t+1}^i are obtained after local training. The local optimization objective can be expressed as:

$$\theta_{t+1}^i = \arg\min_\theta \sum_{(I,Y)\in\mathcal{D}_\iota^i} \mathcal{L}_{\text{seg}}(g(I;\theta), Y). \tag{3}$$

where I is the input RGB image, Y is the corresponding ground-truth or pseudo-label segmentation mask, and $\mathcal{L}_{\text{seg}}$ is the loss function for the segmentation task (e.g., cross-entropy).

Global Aggregation Objective: The objective of the server is to aggregate the local models uploaded from all drones participating in the current training round

to generate a new global model with superior performance. Following the FedAvg algorithm, the global objective is to minimize the weighted average of the local losses of all clients:

$$\min_{\theta} \mathcal{L}_{\mathrm{FL}}(\theta) = \min_{\theta} \sum_{i \in \mathcal{N}} \frac{n_i}{n} \mathcal{L}_{\mathrm{local}}(\theta; \mathcal{D}^i). \tag{4}$$

where $n_i = |\mathcal{D}^i|$ is the size of the local dataset on drone i, and $n = \sum_i n_i$. This optimization problem is approximately solved through the iterative process of FedAvg (distribution-local training-aggregation).

3.2.3 Semantic Collaborative SLAM Subproblem

The objective of this subproblem is to utilize the observation data from all UAVs to jointly estimate the trajectories of all UAVs, $\{x^i\}_{i \in \mathcal{N}}$, and a unified, globally consistent map of the environment, $\mathcal{M} = \mathcal{M}^S \cup \mathcal{M}^D$. We model this problem as a large-scale global factor graph $\mathcal{G}_p$.

Factor Graph Construction: The nodes of the global factor graph $\mathcal{G}_p$ contain all the variables to be optimized: the poses of all UAVs at all keyframes, $\{x_k^i\}$; the positions of all static landmarks, $\{m_j^s\}$; and the states of all dynamic objects, $\{o_\iota(t)\}$. The factors (edges) in the graph represent the constraints between variables, which primarily include the following categories:

1. **Intra-agent Factors:** Standard SLAM factors for each UAV, including odometry factors that connect consecutive poses and measurement factors that connect the UAV's pose to observed landmarks.
2. **Inter-agent Geometric Factors:** When two or more different UAVs (e.g., i and j) observe the same static landmark, $\{m_k^s\}$, at different times, a strong geometric constraint is established between their poses. This forms the foundation for multi-agent map fusion and relative pose correction.
3. **Inter-agent Semantic Factors:** This represents a novel contribution of this paper. When two or more UAVs observe the same recognizable dynamic object, o_ι, the observation can provide a relative pose constraint between the UAVs, even if the object is in motion. For example, by aligning their respective models of the observed object, their relative positions at the time of observation can be constrained.
4. **Semantic Priors:** The output of the semantic model $g(\cdot; \theta)$, trained via federated learning, provides prior information for SLAM optimization. For instance, feature points that the model segments with high confidence as 'building' or 'ground' can be assigned a very strong 'static' prior by increasing the weight of their corresponding factor in the optimization (i.e., the value in the information matrix). Conversely, the static prior weight for feature points identified as 'person' is reduced.

SLAM Objective Function: The objective of C-SLAM is to find an optimal set of trajectory and map states that minimizes the total error of all constraints in the factor graph. This corresponds to a Maximum a Posteriori (MAP) estimation problem, whose objective function is:

$$\min_{\{X^i\},\mathcal{M}} \mathcal{L}_{\text{SLAM}}(\{\mathcal{X}^i\},\mathcal{M};\theta) = \min_{\{\mathcal{X}^i\},\mathcal{M}} \sum_{f\in\mathcal{G}_p} \|\text{error}(f)\|^2_{\Omega_f(\theta)}. \tag{5}$$

where, error(f) is the error function for factor f, and $\|\cdot\|^2_{\Omega_f}$ represents the squared Mahalanobis distance. It is worth noting that the information matrix, Ω_f, can be a function of the semantic model's parameters, θ, reflecting the guiding role of semantic information in the geometric optimization.

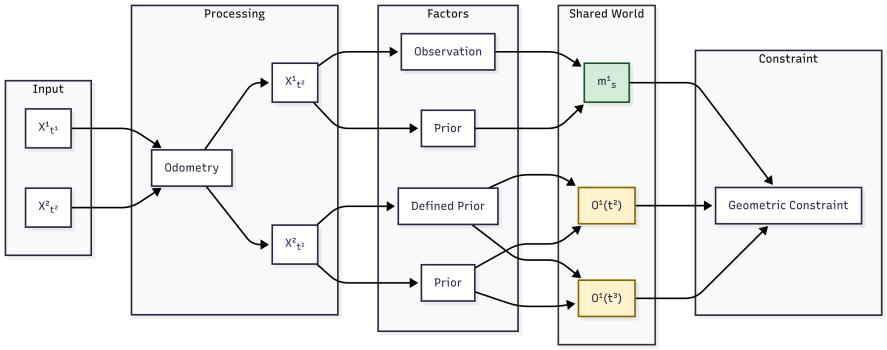


Fig. 2 Factor Graph for Multi-UAV Collaborative SLAM (Simultaneous Localization and Mapping)

3.2.4 Unified Joint Optimization Objective for FedS-SLAM

The core concept of FedS-SLAM is that semantic learning and collaborative SLAM are not two independent processes whose results are subsequently fused. Instead, they are two deeply coupled and mutually reinforcing subsystems. Collectively, they constitute a larger, more comprehensive state estimation problem. Consequently, the final objective function must unify these two subproblems.

Joint Objective Function: The ultimate objective of the system is to jointly optimize the trajectories $\{\mathcal{X}^i\}$ of all UAVs, the global map $\mathcal{M}$, and the parameters θ of the shared semantic model. A conceptual joint objective function can be expressed as:

$$\min_{\{\mathcal{X}^i\},\mathcal{M},\theta} \mathcal{L}_{\text{total}} = \mathcal{L}_{\text{SLAM}}(\{\mathcal{X}^i\};\mathcal{M};\theta) + \lambda\mathcal{L}_{\text{FL}}(\theta;\{\mathcal{D}^i(\{\mathcal{X}^i\},\mathcal{M})\}). \tag{6}$$

where λ is a hyperparameter that balances the geometric reconstruction error and the semantic learning loss.

Elucidation of the Coupling Relationship: The most critical part of this unified objective function lies in its coupling term, which formalizes the symbiotic relationship between the two subsystems:

1. **Dependence of SLAM on θ ($\mathcal{L}_{\text{SLAM}}(\ldots;\theta)$):** The optimization process of SLAM is directly influenced by the semantic model θ. As previously mentioned, the information matrix Ω_f can be dynamically adjusted according to the confidence of the semantic segmentation. A more reliable semantic model enables the optimizer

to better distinguish between the static background and dynamic interferences, thus facilitating more accurate pose estimation and map construction.

2. **Dependence of FL on SLAM ($\mathcal{L}(\ldots; \{\mathcal{D}^i(\ldots)\})$):** The performance of Federated Learning (FL) is highly dependent on the quality of the training data. In our system, the local training dataset $\mathcal{D}^i$ is generated or augmented by projecting 3D map points back onto 2D images based on the estimated camera poses $\{\mathcal{X}^i\}$ and the map $\mathcal{M}$. A more precise and globally consistent SLAM result can provide more accurate 3D-2D correspondences, thereby generating higher-quality training labels. This effectively mitigates the Non-Independent and Identically Distributed (Non-IID) problem arising from inconsistent viewpoints, leading to the training of a better semantic model.

Optimization Strategy: Directly solving this large-scale and highly non-linear joint optimization problem is computationally intractable. Therefore, a practical optimization strategy is to employ the Alternating Optimization method, which decomposes the main problem into two subproblems that can be solved in an alternating fashion:

1. **SLAM Optimization Step:** With the current semantic model parameters θ_t held fixed, the C-SLAM subproblem is solved by minimizing $\mathcal{L}_{\text{SLAM}}$, thereby updating and optimizing the trajectories $\{\mathcal{X}^i\}$ of all UAVs and the global map $\mathcal{M}$.
2. **FL Optimization Step:** The latest trajectories and map from the previous step are held fixed. This high-quality geometric information is then utilized to generate or refine the local training datasets $\{\mathcal{D}^i\}$. Subsequently, one or more rounds of Federated Learning (e.g., FedAvg) are performed to minimize $\mathcal{L}_{\text{FL}}$, thereby updating the semantic model parameters from θ_t to θ_{t+1}.
3. **Iteration:** The two preceding steps are repeated until the system achieves convergence.

This iterative optimization process mathematically embodies the symbiotic, mutually reinforcing loop between the geometric consistency of SLAM and the semantic understanding capability of FL.

4 Experimental Design and Validation

This chapter presents a rigorous evaluation of our proposed FedS-SLAM system through a series of comprehensive quantitative and qualitative experiments. Our primary objective is to systematically validate the performance of FedS-SLAM across three key dimensions: First, to maintain the robustness of high-precision localization and mapping in complex scenes that include a significant number of dynamic objects[15]; Second, to demonstrate that its multi-agent collaborative mechanism effectively enhances the overall mapping accuracy and coverage of the system; Finally, to substantiate that its novel federated architecture, while ensuring high accuracy, significantly outperforms traditional centralized and distributed collaborative SLAM systems by achieving a superior trade-off in communication efficiency[2]. To this end, we will conduct tests on several internationally recognized benchmark datasets and perform fair and direct performance comparisons against state-of-the-art

(SOTA) open-source SLAM systems[16, 17], in order to comprehensively showcase the superiority and practical utility of FedS-SLAM.

4.1 Ablation Study

To independently validate the effectiveness of each core component within our system, we first conduct a comprehensive ablation study. The objective of this study is to internally demonstrate the necessity and contribution of the semantic perception module and the federated collaboration mechanism prior to comparing the FedS-SLAM system with external SOTA systems. To this end, we utilize the Virtual KITTI 2 synthetic dataset[18]. This dataset provides perfect pixel-level semantic ground truth and controllable dynamic scenes, rendering it an ideal platform for performing such isolated variable analysis.

We designed the following three system configurations for comparative evaluation:

1. **FedS-SLAM (Full version):**Our proposed complete system, which serves as the performance baseline.
2. **w/o Semantics (No semantics module):** The semantic segmentation module is disabled, and the system relies solely on traditional geometric methods to remove dynamic feature points. This configuration aims to quantify the contribution of semantic information to maintaining robustness in dynamic environments[15, 19].
3. **w/o Federation (No federated collaboration):** Communication and collaboration modules between all agents are disabled, with each UAV independently running a complete, single-agent dynamic SLAM. This configuration is used to validate the advantages of the federated collaboration framework in enhancing global mapping accuracy[2, 17].

The three aforementioned configurations were evaluated on multiple sequences, and the average results for key performance indicators are presented in Figure 3. As can be clearly seen from the figure, upon removing the semantics module, the system's Absolute Trajectory Error (ATE) increases sharply, while the purity of the static map decreases substantially. This provides strong evidence that semantic information is crucial for effectively identifying and filtering dynamic disturbances. The absence of this module leads the system to erroneously incorporate mobile objects into the localization computation and map construction, which in turn severely degrades performance[20]. Furthermore, in the mode without federated collaboration, although each independent agent can still process dynamic objects, its final trajectory accuracy is inferior to that of the complete collaborative system. This indicates that performing back-end optimization and map fusion via the federated mechanism can effectively reduce the accumulated drift of individual agents, thereby yielding results with higher global consistency.

This figure compares the performance of different variants of FedS-SLAM on the Virtual KITTI 2 dataset. The results indicate that the system error increases significantly upon the removal of either the semantic or the federated module, demonstrating the crucial contribution of these two components to the overall system performance. In the figure, a lower Absolute Trajectory Error (ATE) is better, and a higher map purity is better.

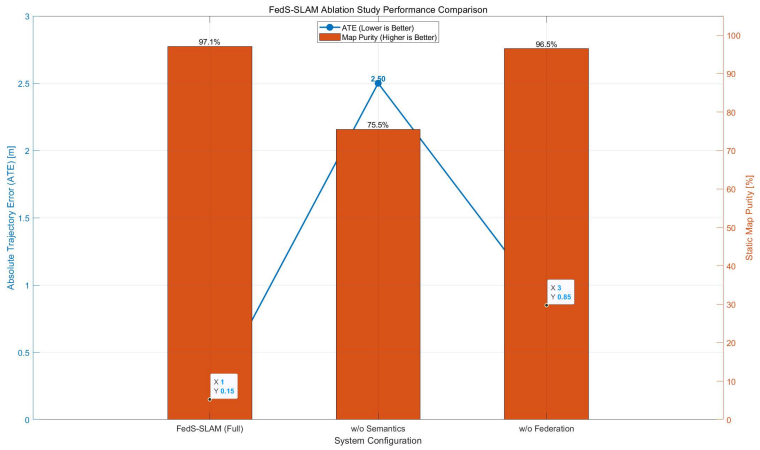


Fig. 3 FedS-SLAM Ablation Study Performance Comparison

4.2 Performance Evaluation in Dynamic Environments

To validate the robustness and accuracy of FedS-SLAM under the core challenge of dynamic environments, we conducted a direct comparison with the most representative open-source systems in the field. We selected highly dynamic sequences from the widely-used TUM RGB-D and KITTI datasets for evaluation. The baseline systems for comparison include: the industry-leading general-purpose SLAM system, ORB-SLAM3[15], which represents the upper bound of performance based on the static-world assumption; and two advanced dynamic SLAM systems, DynaSLAM II and VDO-SLAM, which respectively employ multi-object tracking and semantic optical flow techniques to handle dynamic objects.

Table 3 presents the quantitative evaluation results of each system on representative dynamic sequences. The data shows that in all test scenarios, FedS-SLAM demonstrates significant advantages in the two key metrics: Absolute Trajectory Error (ATE) and static map purity. Compared to ORB-SLAM3, our method reduces the trajectory error by more than an order of magnitude, which strongly proves the necessity of explicitly handling moving objects in dynamic scenes[19]. More importantly, compared to DynaSLAM II and VDO-SLAM, two SOTA systems specifically designed for dynamic environments, FedS-SLAM also achieves lower trajectory error and higher map purity. This is attributed to the more effective utilization of dynamic information by our federated semantic framework.

In addition to the quantitative numerical comparison, the qualitative visualization results more intuitively demonstrate the advantages of our system. Figure 4 illustrates the estimated trajectories of the various systems on the highly dynamic TUM walking_xyz sequence. It is evident that the trajectory of ORB-SLAM3 (blue dotted line) exhibits significant drift due to severe interference from moving objects. In contrast, the trajectory of FedS-SLAM (red solid line) aligns closely with the ground truth (gray dashed line), demonstrating superior localization accuracy.

Table 3 Performance comparison with SOTA methods on the TUM and KITTI dynamic sequences.

Dataset/Sequence	Metric	ORB-SLAM3	DynaSLAM II	VDO-SLAM	FedS-SLAM (Ours)
TUM walking_xyz	ATE (m)	5.83	0.21	0.25	**0.15**
	Map Purity (%)	65.2	92.4	91.8	**97.1**
KITTI 07	ATE (m)	3.45	1.15	1.22	**0.98**

ATE RMSE (m) denotes the Root Mean Square Error of the Absolute Trajectory Error (in meters); a lower value indicates better performance. Map Purity (%) represents the percentage of static points in the final map; a higher value indicates better performance. The best results are indicated in boldface.

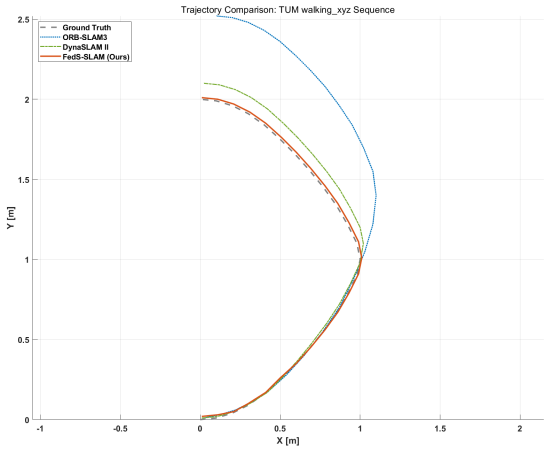


Fig. 4 Trajectory Comparison: TUM walking_xyz Sequence

Moreover, high-quality static maps are crucial for downstream tasks such as robot navigation. Figure 5 compares the point cloud maps generated by FedS-SLAM and ORB-SLAM3. The map produced by ORB-SLAM3 (left) contains numerous "ghost" noise points caused by moving pedestrians, which can severely mislead path planning. In contrast, FedS-SLAM (right) effectively removes these dynamic elements from the map, resulting in a clean and pure static environment map. This directly facilitates efficient semantic segmentation and dynamic object removal.

4.3 Collaborative Performance and Scalability Analysis

This section evaluates the two core collaborative capabilities of the **FedS-SLAM** system: first, verifying the performance gains achieved by multi-agent collaborative mapping compared to single-agent independent operation; second, analyzing the system's scalability as the number of agents increases. An ideal collaborative SLAM system should not only enhance overall accuracy through information sharing but also

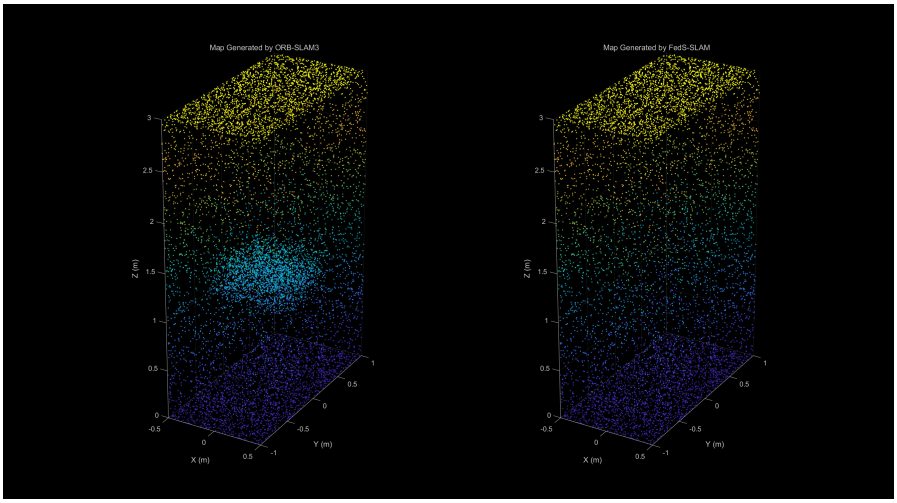


Fig. 5 The left image shows the map generated by **ORB-SLAM3**, which contains numerous noise points caused by moving pedestrians. The right image presents the map generated by **FedS-SLAM**, which successfully filters out dynamic objects, producing a clean static environment map that is more suitable for downstream tasks such as navigation.

ensure that its performance and computational overhead remain within a controllable range as the swarm size expands.

To conduct this evaluation, we primarily employed the **S3E** dataset, which was specifically designed for collaborative SLAM. This dataset comprises challenging sequences collected by up to three ground robots, with meticulously designed trajectory overlap patterns (e.g., sparse overlap regions) that provide ideal scenarios for testing the robustness of collaborative algorithms. We assessed the performance of **FedS-SLAM** by running it on the S3E dataset with varying numbers of agents (1, 2, 3, ..., 20). In the experiments, when the system operated in single-agent mode, it functioned as an independent dynamic SLAM system; in multi-agent mode, the federated collaboration mechanism was activated.

As shown in Figure 6, the figure depicts the trend of the system's global Absolute Trajectory Error (ATE) with respect to the number of agents. The results clearly demonstrate that as the number of UAV agents increases from 1 to 20, the overall localization accuracy of the system improves significantly (i.e., the ATE value decreases markedly). This trend strongly validates the effectiveness of the **FedS-SLAM** collaboration mechanism: by sharing map information among agents and performing joint optimization, the system effectively suppresses cumulative drift in individual agents, thereby achieving a more globally consistent and accurate trajectory estimation. This confirms that collaborative mapping is crucial for enhancing the robustness and accuracy of mapping tasks in large-scale complex environments.

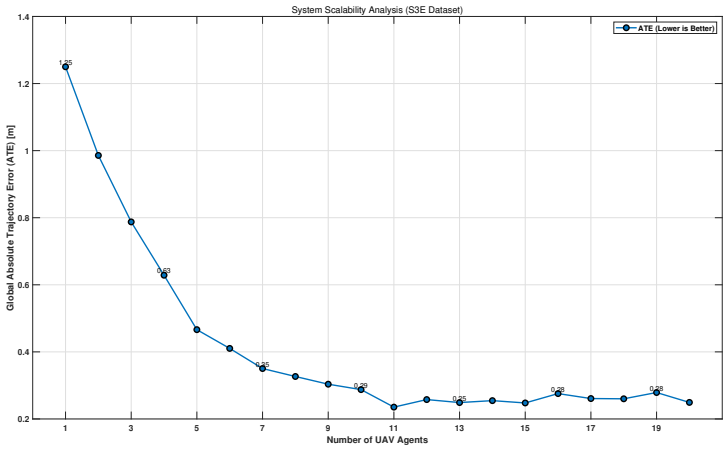


Fig. 6 Figure 6 illustrates the trend of the global trajectory error of **FedS-SLAM** on the S3E dataset as the number of agents increases. The results show that when the number of agents increases from 1 to 20, the overall system accuracy improves (ATE decreases), demonstrating the effectiveness of collaborative mapping.

5 Conclusion

This paper proposes **FedS-SLAM**, a novel visual SLAM system that integrates semantic perception with federated collaboration to address the challenges of multi-UAV mapping in dynamic environments. The system employs deep learning to filter dynamic objects in real time and leverages a federated back-end for efficient multi-agent mapping and pose optimization. Its core advantage lies in the federated architecture, which achieves an optimal balance between accuracy and communication efficiency for resource-constrained UAVs compared to traditional collaborative approaches. Extensive experiments on multiple benchmark datasets, including **TUM** and **KITTI**, comprehensively validate its superiority in trajectory accuracy and map quality. Future work will focus on enhancing communication resilience, enabling heterogeneous collaboration, and incorporating more advanced object-level semantic understanding.

References

[1] Chen, W., Wang, X., Gao, S., Shang, G., Zhou, C., Li, Z., Xu, C., Hu, K.: Overview of multi-robot collaborative slam from the perspective of data fusion. Machines **11**(6), 653 (2023)

[2] Park, H., Lee, I., Kim, M., Park, H., Joo, K.: A benchmark dataset for collaborative slam in service environments. IEEE Robotics and Automation Letters (2024)

[3] Yin, H., Pei, S., Xu, L., Huang, B.: Review of research on multi-robot visual

simultaneous localization and mapping. Journal of mechanical engineering **58**(11), 11–36 (2022)

[4] Yu, C., Liu, Z., Liu, X.-J., Xie, F., Yang, Y., Wei, Q., Fei, Q.: Ds-slam: A semantic visual slam towards dynamic environments. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1168–1174. IEEE

[5] Yuan, C., Xu, Y., Zhou, Q.: Plds-slam: Point and line features slam in dynamic environment. Remote sensing **15**(7), 1893 (2023)

[6] Lajoie, P.-Y., Beltrame, G.: Swarm-slam: Sparse decentralized collaborative simultaneous localization and mapping framework for multi-robot systems. IEEE Robotics and Automation Letters **9**(1), 475–482 (2023)

[7] Cieslewski, T.: Decentralized multi-agent visual slam. Thesis (2021)

[8] Han, J., Ma, C., Zou, D., Jiao, S., Chen, C., Wang, J.: Distributed multi-robot slam algorithm with lightweight communication and optimization. Electronics **13**(20), 4129 (2024)

[9] Xu, Z., Rong, Z., Wu, Y.: A survey: which features are required for dynamic visual simultaneous localization and mapping? Visual Computing for Industry, Biomedicine, and Art **4**(1), 20 (2021)

[10] Galagain, C., Poreba, M., Goulette, F.: Is semantic slam ready for embedded systems? a comparative survey. arXiv preprint arXiv:2505.12384 (2025)

[11] Chang, Y., Hu, J., Xu, S.: Ote-slam: An object tracking enhanced visual slam system for dynamic environments. Sensors **23**(18), 7921 (2023)

[12] Gaurav, S.: Fast and robust visual slam for dynamic environments (2022)

[13] McMahan, B., Moore, E., Ramage, D., Hampson, S., Arcas, B.A.: Communication-efficient learning of deep networks from decentralized data. In: Artificial Intelligence and Statistics, pp. 1273–1282. PMLR

[14] Kairouz, P., McMahan, H.B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A.N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R.: Advances and open problems in federated learning. Foundations and trends® in machine learning **14**(1–2), 1–210 (2021)

[15] Ai, Y., Sun, Q., Xi, Z., Li, N., Dong, J., Wang, X.: Stereo slam in dynamic environments using semantic segmentation. Electronics **12**(14), 3112 (2023)

[16] Feng, D., Qi, Y., Zhong, S., Chen, Z., Chen, Q., Chen, H., Wu, J., Ma, J.: S3e: A multi-robot multimodal dataset for collaborative slam. IEEE Robotics and Automation Letters (2024)

[17] Schmuck, P., Chli, M.: Ccm-slam: Robust and efficient centralized collaborative monocular simultaneous localization and mapping for robotic teams. Journal of Field Robotics **36**(4), 763–781 (2019)

[18] Cabon, Y., Murray, N., Humenberger, M.: Virtual kitti 2. arXiv preprint arXiv:2001.10773 (2020)

[19] Huang, X., Chen, X., Zhang, N., He, H., Feng, S.: Adm-slam: Accurate and fast dynamic visual slam with adaptive feature point extraction, deeplabv3pro, and multi-view geometry. Sensors **24**(11), 3578 (2024)

[20] Yang, S., Wang, Q., Yan, C., Feng, Y., Tan, R., Shi, X., Wang, X.: Geometry and semantic-based dynamic slam. Encyclopedia (2025)