Int. J. Nonlinear Anal. Appl. In Press, 1–13
ISSN: 2008-6822 (electronic)
http://dx.doi.org/10.22075/ijnaa.2026.42239.5769

Federated CNN-based sensor fault detection in unmanned aerial vehicles with WGAN-GP data augmentation

Fatemeh Shakibaeifard[a], Mohammadreza Eslaminejad[b,*], Farzad Peyravi[c]

[a] *Department of Computer Engineering, KI.C., Islamic Azad University, Kish, Iran*
[b] *Department of Computer Engineering, Zand Institute of Higher Education, Shiraz, Iran*
[c] *Department of Computer Engineering, Marv.C., Islamic Azad University, Marvdasht, Iran*

(Communicated by Nallappan Gunasekaran)

Abstract

Unmanned aerial vehicles (UAVs) depend on navigation sensors, including GPS, gyroscopes, accelerometers, magnetometers, barometers, and remote-control systems, for safe and autonomous flight. Sensor faults can cause trajectory deviation, flight instability, and mission failure. Existing fault detection approaches face challenges associated with class imbalance, distributed training, and multi-class fault discrimination. This study presents an integrated framework for seven-class UAV sensor status classification using the publicly available BASiC dataset. The framework combines a convolutional neural network with multi-scale feature extraction, residual blocks, and channel-wise attention with FedProx-based federated learning across multiple simulated clients representing distributed UAV nodes. Raw training data remain local to each client, while only model parameters are exchanged and aggregated. To address class imbalance, WGAN-GP generates synthetic samples for minority classes exclusively within the training set. Two scenarios are evaluated using the same held-out test set: federated CNN without augmentation and federated CNN with WGAN-GP augmentation. Flight-level data partitioning and training-only standardization and feature selection prevent test-set information leakage. The results show that federated CNN achieves 84.8% accuracy, whereas WGAN-GP augmentation increases accuracy to 97.3%, representing a 12.5-percentage-point improvement, with a comparable increase in macro-averaged F1-score. These findings demonstrate the potential of combining federated optimization and generative augmentation for UAV sensor fault classification in simulated decentralized training environments without transferring raw training data between clients.

Keywords: unmanned aerial vehicles (UAVs), sensor fault detection, convolutional neural networks (CNN), federated learning, Wasserstein generative adversarial networks with gradient penalty (WGAN-GP), data augmentation
2020 MSC: 68T07; 68T45

1 Introduction

Unmanned aerial vehicles (UAVs) have gained widespread applications across military, surveillance, agricultural, mapping, and emergency response domains. The safe operation of these systems depends critically on the accuracy

*Corresponding author
Email addresses: 2300661209@iau.ir (Fatemeh Shakibaeifard), m.eslaminejad@zand.ac.ir (Mohammadreza Eslaminejad), fpeyravi@iau.ac.ir (Farzad Peyravi)

Received: July 2026 *Accepted:* August 2026

of data from multiple key sensors, including GPS, accelerometer, gyroscope, magnetometer, barometer, and remote-control systems. Faults in any of these sensors can result in trajectory deviation, flight instability, or even crash events [3]. Consequently, timely and accurate sensor fault identification constitutes a fundamental prerequisite for enhancing UAV safety and reliability.

Previous research in this domain has explored diverse approaches, including hardware redundancy, model-based analytical redundancy, signal processing techniques, and, more recently, deep learning methodologies. Among these methods, a transformer-based architecture evaluated on the publicly available BASiC dataset achieved 86% accuracy in early fault prediction (approximately one minute before fault occurrence) and 94% accuracy in fault type classification [3]. However, many existing approaches are developed under centralized learning settings, in which data from different flights or UAVs are assumed to be available at a common processing location. In operational multi-UAV environments, flight and sensor data may instead remain distributed across individual vehicles, operators, or ground infrastructures, making centralized data collection potentially undesirable or impractical. Federated learning provides a framework for collaborative model training in which raw training data can remain local to participating nodes while model parameters are exchanged for global aggregation. In this study, the federated learning paradigm is evaluated using the publicly available BASiC dataset as a reproducible experimental benchmark rather than as a privacy-sensitive dataset. Accordingly, the use of federated learning in this work should be understood as a simulation of a decentralized training setting relevant to potential multi-UAV deployments, rather than as a direct claim of privacy protection for the BASiC dataset itself. In parallel, the inherent imbalance between normal samples and different fault categories remains an important challenge for reliable multi-class sensor fault classification and motivates the use of WGAN-GP-based data augmentation in the proposed framework.

Recent studies have demonstrated that combining federated learning with generative adversarial networks can simultaneously improve detection accuracy and prevent raw data disclosure in domains closely related to the present problem, including cyber anomaly detection and drift detection in wireless sensor networks [9, 21]. Nevertheless, no framework has been reported that holistically addresses the three interconnected challenges of class imbalance, distributed training privacy, and multi-class sensor fault classification accuracy for UAV systems.

In this context, the principal contributions of this research can be summarized as follows:

1. Implementation and evaluation of a simulated federated learning framework based on FedProx, in which multiple virtual clients represent distributed UAV nodes and exchange model parameters without transferring their assigned raw training samples.
2. Explicit design and comparison of two scenarios on an identical test set: the first scenario based on CNN with federated learning, and the second incorporating WGAN-GP for synthetic data generation of minority classes in addition to the aforementioned components;
3. Design of a hybrid convolutional architecture incorporating multi-scale feature extraction, residual blocks, and channel-wise attention mechanisms for feature extraction from multi-sensor time-series data;
4. A conservative evaluation protocol wherein training/validation/test splits are performed at the flight file level, with the validation and test sets never being utilized in any training or model tuning phase.

The remainder of this manuscript is organized as follows: Section 2 provides a review of related research; Section 3 introduces the dataset employed; Section 4 describes the proposed methodology including preprocessing, feature selection, network architecture, federated learning, and the two scenarios with corresponding infographics; Section 5 presents experimental results and discussion; and Section 6 concludes the study with suggestions for future work.

2 Related work

Classical approaches to UAV sensor fault detection can be categorized into three groups: hardware redundancy, model-based analytical redundancy, and signal processing techniques. While effective under controlled conditions, these methods exhibit limited flexibility when confronted with the nonlinearity and environmental noise characteristic of real-world flight data. Consequently, recent research has increasingly shifted toward data-driven approaches and deep learning.

In a comprehensive review of fault detection and diagnosis methodologies for UAVs, the authors analyzed several dozen peer-reviewed articles and demonstrated that the predominant research trajectory is moving toward integrating signal processing with machine learning, with increasing emphasis on real-time data-driven methods [1]. In the specific domain of fixed-wing UAVs, a lightweight convolutional neural network combined with a long short-term memory network equipped with attention mechanisms was proposed to reduce computational complexity while maintaining detection accuracy [14]. Another study introduced a self-attention-based network with similarity space mapping for sensor fault detection, focusing on spatiotemporal dependencies and demonstrating favorable performance on both simulated and real-world data [4]. Fuzzy logic-based approaches combined with multi-sensor fusion have also been employed to classify UAV status into safe, semi-safe, and high-risk levels [5].

From the perspective of addressing limited labeled fault data, generative adversarial networks [6] have been deployed in hybrid autoencoder-GAN architectures for sensor fault estimation and reconstruction [15]. Furthermore, on-chip hardware implementations of deep learning architectures have been tested for real-time fault cause detection during actual UAV flights [20]. The reference study for this research proposed a transformer-based framework for temporal prediction and classification of sensor faults using the BASiC dataset, achieving 86% accuracy for early prediction and 94% for fault type classification; this framework operates in a centralized environment and does not directly address class imbalance or data privacy concerns [2, 3].

Concurrently with these developments, emerging research has demonstrated that combining federated learning with generative adversarial networks can maintain or improve detection accuracy in analogous problems while preventing raw data from leaving client devices. For instance, in multi-layer cyber anomaly detection for UAVs, a quantum-resilient federated learning framework employing a hybrid variational autoencoder and neural classifier achieved high accuracy with modest computational overhead [21]. In an independent study, a Wasserstein GAN-based digital twin model for early drift fault detection in wireless sensor networks demonstrated superior training stability compared to classical GAN approaches [9].

To facilitate a systematic comparison with previous research, Table 1 presents a representative overview of recent studies on UAV sensor fault detection and related privacy-preserving learning approaches. The comparison considers the datasets employed, model architectures, learning paradigms, privacy preservation strategies, class imbalance handling, and reported performance. As shown in Table 1, although considerable progress has been achieved in improving fault detection accuracy, the integration of federated learning with WGAN-GP-based data augmentation for privacy-preserving multi-class UAV sensor fault classification has received limited attention.

Table 1: Comparative summary of representative studies on UAV sensor fault detection and related privacy-preserving learning approaches

Ref.	Dataset	Main Method	Privacy-Preserving	GAN-based Augmentation	Multi-class Fault Classification	Reported Performance	Main Limitation
[1]	Various UAV datasets	Survey of ML methods	×	×	Various	Review article	Does not propose a new detection framework
[14]	Fixed-wing UAV dataset	CNN + Attention-based LSTM	×	×	✓	High detection accuracy reported	No privacy preservation and no imbalance handling
[4]	Simulated and real UAV data	Self-attention neural network	×	×	✓	Robust detection performance	Centralized learning; no privacy-aware training
[5]	UAV flight data	Fuzzy Logic + Multi-sensor Fusion	×	×	Partial	UAV risk-level classification	Focuses on risk assessment rather than fault classification
[15]	UAV sensor data	Autoencoder + GAN	×	✓	Partial	Improved fault estimation	Does not employ federated learning
[20]	Real UAV flights	Embedded Deep Learning	×	×	✓	Real-time onboard diagnosis	Hardware-oriented; no imbalance or privacy solution
[2, 3]	BASiC	Transformer	×	×	✓	86% prediction, 94% classification	Centralized architecture; no privacy preservation; no class imbalance treatment
[21]	UAV cyber anomaly dataset	Federated Learning + VAE	✓	Partial	×	High detection accuracy	Not designed for UAV sensor fault classification
[9]	Wireless sensor networks	WGAN-GP	×	✓	×	Stable synthetic data generation	Different application domain; no federated learning
This Study	BASiC	CNN + Residual + SE Attention + FedProx + WGAN-GP	✓	✓	✓	97.3% classification accuracy	Currently validated only on the BASiC dataset

Synthesizing the existing literature, the primary research gap can be articulated as follows: despite significant advances in single-client and centralized methods for UAV sensor fault detection accuracy [1, 2, 3, 4, 5, 14, 15, 20], the simultaneous combination of federated learning for privacy preservation and Wasserstein GAN-based generative augmentation for addressing class imbalance, specifically for multi-class UAV sensor fault classification, has received limited investigation. The present study addresses this gap by integrating these two approaches within a unified framework and comparing them across two experimental scenarios.

3 The BASiC dataset

This study employs the publicly available Biomisa Arducopter Sensory Critique (BASiC) dataset [2, 3]. This dataset comprises 70 autonomous flights (exceeding 7 hours of flight data) conducted using the open-source ArduCopter/ArduPilot platform in a software-in-the-loop (SITL) simulation environment. It encompasses six common sensor fault types (GPS, remote control, accelerometer, gyroscope, magnetometer, and barometer) along with fault-free flights. Each flight is accompanied by ground-truth labels for fault timing and type, provided as processed CSV files.

In this study, flight files were initially partitioned using an 80:20 ratio with stratified sampling to preserve class distribution across training+validation and test sets. Subsequently, from the remaining 80% portion, a subset of files (approximately 10% of the total data) was reserved as the validation set. Each flight file was thus allocated entirely to one of the three sets (training, validation, or test) and was never split across multiple sets. This decision was specifically implemented to prevent data leakage between overlapping temporal windows from a single flight. The validation set was exclusively utilized for selecting the optimal number of features and monitoring model accuracy throughout federated learning communication rounds, while the test set was employed only once, at the conclusion of the entire training process, for reporting final results for both scenarios. Subsequently, temporal windows with appropriate window length and stride were employed to construct training samples.

4 Proposed methodology

4.1 Overall framework and two experimental scenarios

The proposed methodology consists of five sequential stages: (i) preprocessing and temporal window generation, (ii) feature selection, (iii) construction of the convolutional neural network for seven-class sensor fault classification, (iv) distributed model optimization within a federated learning framework, and (v) minority-class augmentation using WGAN-GP. To isolate the contribution of synthetic data augmentation, two experimental scenarios are designed and evaluated using an identical held-out test set. The first scenario includes only the first four stages, whereas the second incorporates WGAN-GP-based augmentation prior to federated model training. A detailed description of each scenario, together with the corresponding workflow diagrams, is provided in Sections 4.8 and 4.10.

4.2 Preprocessing and Windowing

After partitioning the dataset into training, validation, and test subsets at the flight-file level, the common feature columns shared across all files were identified. Missing, undefined, and infinite values were replaced with zeros to ensure data consistency. Each flight sequence was subsequently segmented into fixed-length multivariate samples using a sliding-window strategy with a window length of 64 and a stride of 32, corresponding to a 50% overlap between consecutive windows. This configuration provides an appropriate balance between increasing the number of training samples and limiting excessive redundancy among neighboring windows.

Feature normalization was performed using Z-score standardization, where the mean and standard deviation were estimated exclusively from the training set. The same normalization parameters were then applied directly to the validation and test sets without recalculation, thereby preventing statistical information leakage. Finally, standardized feature values were clipped within a symmetric interval around zero to reduce the influence of extreme outliers during model training.

4.3 Feature selection

A subset of 48 features associated with the six target sensors was selected from the original flight records by combining the Elbow method with F-ANOVA statistical feature ranking implemented through the SelectKBest algorithm. Candidate feature subsets of different sizes were first evaluated by training a lightweight Random Forest classifier using only the training data, while the corresponding predictive performance was measured on the validation set. The optimal number of features was determined at the point where further increases in feature count produced only marginal improvements in validation accuracy. Similar to the normalization procedure, feature selection relied exclusively on the training and validation sets, ensuring that no information from the test set influenced model development.

4.4 Convolutional neural network architecture

The proposed classifier consists of three complementary architectural components designed to capture discriminative temporal patterns from multivariate UAV sensor signals.

The first component performs multi-scale feature extraction through multiple parallel convolutional branches employing different kernel sizes. This design enables the network to learn both short-term local dependencies and longer temporal patterns simultaneously.

The second component is composed of a sequence of residual learning blocks with progressively increasing channel dimensions. Each residual block incorporates a Squeeze-and-Excitation (SE) channel attention mechanism, allowing the network to adaptively emphasize informative feature channels while suppressing less relevant representations.

Max-pooling operations are inserted between the early residual stages to gradually reduce spatial resolution, whereas residual skip connections facilitate stable optimization of deeper architectures by alleviating gradient degradation [10, 11].

The final component aggregates feature representations through both global average pooling and global max pooling before passing the resulting feature vector to a series of fully connected layers and a SoftMax classifier for seven-class prediction. Network optimization is performed using the Adam optimizer because of its robust convergence characteristics and stable learning behavior [12].

4.5 Configuration

The principal hyperparameters employed throughout the proposed framework are summarized in Table 2. To ensure a fair comparison, identical hyperparameter settings are maintained across both experimental scenarios. These values were selected according to commonly adopted configurations reported in previous studies rather than through extensive manual hyperparameter optimization [8, 12, 16, 17, 18].

Table 2: Key hyperparameters of the proposed framework

Parameter	**Value**	**Parameter**	**Value**
Temporal window length	64	Window stride	32
Number of federated clients	4	Communication rounds	50
Local epochs per round	15	Batch size	64
Base learning rate	Cosine annealing	FedProx proximal penalty (μ)	0.01
WGAN-GP latent space dimension	100	Generator-critic training epochs	300
Critic-to-generator update ratio	5:1	Gradient penalty coefficient (λ)	10

4.6 Federated learning (FedProx): weight exchange instead of raw data

It should be emphasized that the federated setting in this study is simulated rather than deployed across physically distributed UAVs. The four virtual clients are used to emulate independent UAV-side data holders and to evaluate the optimization behavior of federated training. Therefore, the present experiments do not claim to reproduce all practical characteristics of real-world federated UAV systems, such as wireless communication delays, intermittent connectivity, client dropout, bandwidth limitations, or device-level resource constraints.

The training process follows the FedProx algorithm [16, 18] as outlined below. Initially, a global convolutional model with random weights is initialized at the ground station. At the beginning of each communication round, the weights of this global model are transmitted to all virtual machines (clients). Each virtual machine initializes its local model with the received weights and subsequently updates it for several local training epochs using its exclusive data, with a cosine annealing learning rate schedule [17]. The proximal penalty term in the local cost function constrains excessive deviation of local weights from the global model, preventing model divergence in the presence of potential data heterogeneity.

Following the completion of local training epochs, each virtual machine transmits only the updated local model weights (not raw data) to the server. The server aggregates these weights via weighted averaging proportional to each client's data volume, updates the global model accordingly, and redistributes the updated global weights to all clients to initiate the next communication round. This cycle repeats for a predetermined number of communication rounds. At the conclusion of training, the final global model is retrieved for inference, with performance evaluated on the validation set during training and on the held-out test set at the end of the process.

4.7 Data leakage prevention in generative augmentation

To prevent information leakage during generative augmentation, the WGAN-GP pipeline was strictly restricted to the training partition. Real samples used to train the generator and critic were obtained exclusively from the training set, and no validation or test samples were used during GAN optimization. The statistical parameters required for data transformation were also estimated from the training partition only.

The kernel witness-based quality assessment was likewise performed using reference distributions derived exclusively from real training samples. Validation and test samples were not used to construct the reference distribution, determine the quality threshold, accept or reject generated samples, or otherwise influence the augmentation process. The validation set was reserved for model-development decisions and hyperparameter tuning, whereas the test set remained completely isolated until the final evaluation.

After quality filtering, the accepted synthetic samples were combined with the original training samples to construct the augmented training set. The validation and test sets were not modified by the augmentation procedure. This separation ensures that the generative model and its quality-control procedure are learned exclusively from the training distribution and prevents information from the held-out test data from influencing the final reported performance.

4.8 Scenario one: baseline framework (CNN with federated learning)

In the first scenario, the sensor fault detection process relies exclusively on the existing data in the BASIC training set, without employing any synthetic data for data augmentation. In this structure, a global model with random weights is initially established at the ground station. As described in Section 4.6, each UAV is assumed to have a dedicated virtual machine on the server, where the local model is trained using its own data.

Following global model initialization, sensor data from the various UAV sensors, including GPS, gyroscope, accelerometer, magnetometer, barometer, and remote control, are distributed in a stratified manner among the virtual machines representing local nodes (UAVs), after undergoing preprocessing, normalization, and window structuring. At the beginning of each communication round, global model weights are transmitted to all virtual machines. Each virtual machine initializes its local model with these weights and trains the CNN model for several local epochs using the FedProx algorithm with its allocated data.

After completing local model training, each virtual machine transmits its local model weights (and not raw data) to the server for averaging. The server aggregates these weights through weighted averaging, updates the global model, and redistributes the updated global model to clients for the next communication round. The stopping criterion for this cycle is reaching the predetermined number of communication rounds. At the conclusion of training, the final global model is retrieved for inference, and its performance is evaluated. Figure 1 illustrates the complete structure of this scenario.

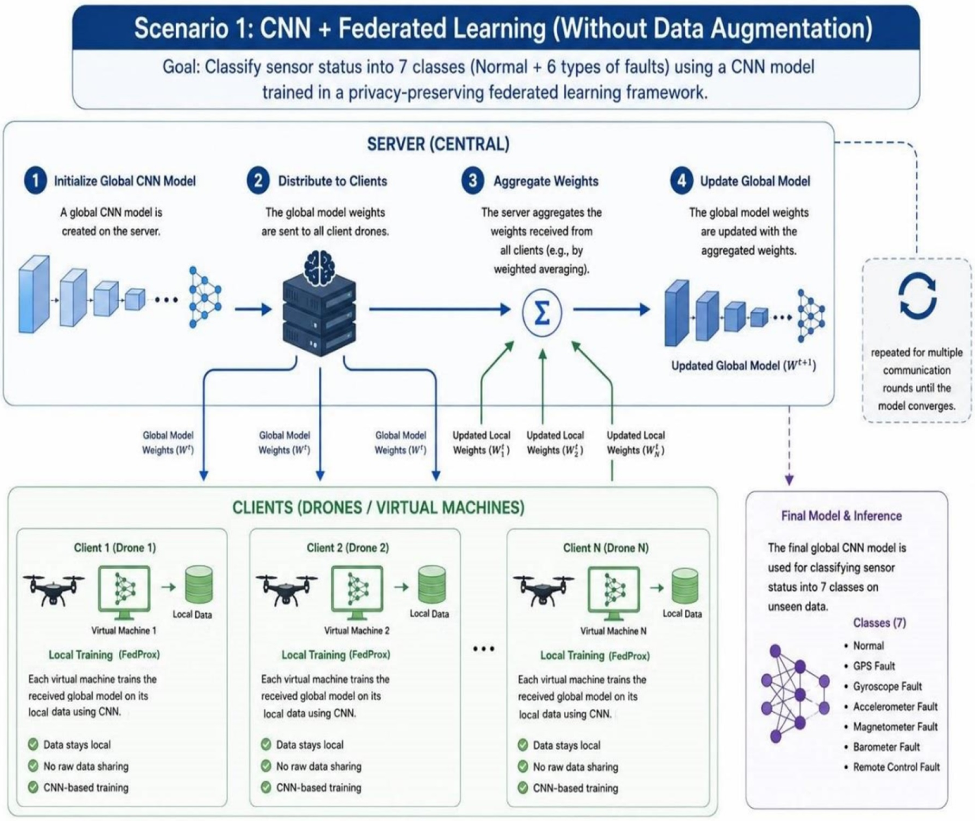

Figure 1: Baseline CNN with federated learning framework (without WGAN-GP) – Scenario One

In this scenario, system performance depends solely on the quality and volume of dataset data. Since certain sensor faults in the BASIC dataset have limited sample counts, class imbalance and reduced accuracy in detecting some rare faults may occur. Nevertheless, this scenario serves as the baseline for the study, and its results provide the foundation for comparison with the WGAN-GP-based scenario.

4.9 Data augmentation with WGAN-GP

To mitigate the imbalance among sensor fault categories, data augmentation is performed using a Wasserstein Generative Adversarial Network with Gradient Penalty (WGAN-GP) [8]. This augmentation stage is executed centrally at the server before the final distribution of the training data to the participating virtual clients. For each minority fault category within the global training set, an independent generator-critic pair is trained exclusively using samples from that specific class. Unlike conventional GANs, the critic estimates the Wasserstein distance between the real and generated data distributions rather than performing binary discrimination. The inclusion of the gradient penalty term enforces the Lipschitz continuity constraint, resulting in improved optimization stability and reducing the likelihood of mode collapse compared with classical GAN architectures [6]. Furthermore, the critic is updated multiple times for every generator update to ensure accurate estimation of the Wasserstein distance throughout the training process.

Before synthetic samples are incorporated into the training dataset, a two-stage quality assessment procedure is applied. During the first stage, a validation filter removes samples whose generated values lie close to the saturation boundaries of the generator output, as these samples may indicate unstable generation. Whenever the number of accepted samples is insufficient, the generator produces additional candidate samples until the required quantity is obtained.

In the second stage, all remaining candidate samples are ranked according to a quality score computed using the kernel witness function with a radial basis function (RBF) kernel [7]. This metric quantifies the similarity between each synthetic sample and the corresponding real-data distribution of the target class relative to the complete synthetic sample population. Only the highest-quality synthetic samples are retained, with the selected quantity matching exactly the number required to balance each minority class with the majority class. Finally, the accepted samples are transformed back to their original scale, reconstructed as temporal windows, and incorporated exclusively into the corresponding training subset. The validation and test datasets remain completely unchanged throughout this procedure, ensuring that both experimental scenarios are evaluated under identical and unbiased conditions.

4.10 Scenario two: complete framework (CNN, federated learning, and WGAN-GP)

The second experimental scenario follows the same convolutional architecture, federated learning strategy, and client-server communication protocol described for Scenario One. Likewise, each UAV is represented by an independent virtual client that performs local model training using only its own flight data. The distinguishing feature of this scenario is the inclusion of the WGAN-GP-based augmentation stage prior to the distribution of training data among the participating clients.

Specifically, the server first applies the augmentation and quality-control procedure described in Section 4.9 to generate additional samples for every minority fault class. The resulting balanced training dataset, consisting of both real and high-quality synthetic samples, is subsequently partitioned among the virtual clients using the same stratified distribution strategy adopted in the baseline scenario.

Following data distribution, the federated optimization process remains unchanged. The server initializes the global convolutional model and broadcasts its parameters to all participating clients at the beginning of each communication round. Each client performs several local optimization epochs using the FedProx algorithm before transmitting only its updated model parameters to the server. The server aggregates these parameters through weighted averaging and redistributes the updated global model for the next communication round. This iterative process continues for exactly the same number of communication rounds used in Scenario One, ensuring that both scenarios differ only in the inclusion of synthetic data augmentation. The complete workflow of the proposed framework is illustrated in Figure 2.

Because the validation and test sets remain identical across both experimental scenarios and all training configurations are preserved, any observed performance difference can be attributed directly to the contribution of WGAN-GP-based synthetic data augmentation, thereby enabling a fair and controlled comparison between the two approaches.

5 Results and discussion

In this section, to precisely examine model behavior for each class, the performance of both scenarios was evaluated on the identical test set using precision, recall, and F1-score metrics for each class [13, 19]. Table 3 reports these results for Scenario One (S1) and Scenario Two (S2).

Table 3 reports the class-wise performance of the two experimental scenarios. In Scenario One (without GAN), the magnetometer class exhibits the weakest performance with an F1-score of 81.4% and precision of 81.1%, reflecting the

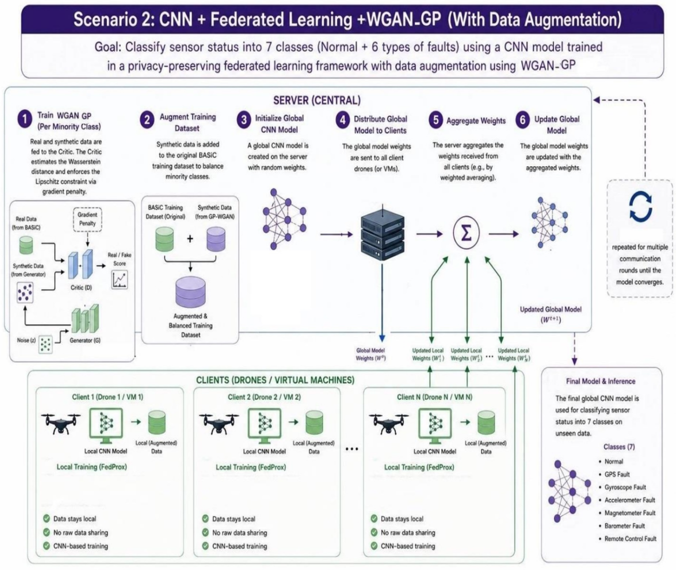

Figure 2: Complete CNN, federated learning, and WGAN-GP framework – Scenario Two

inherently challenging nature of this class for detection. In contrast, the gyroscope class achieves the highest precision at 89.4% in this scenario. With the addition of WGAN-GP in Scenario Two, all classes experience substantial improvement, with the magnetometer class showing the largest enhancement of +15.3 percentage points in F1-score and the GPS class improving by approximately +13.7 percentage points. In Scenario Two, the gyroscope class achieves the best performance with an F1-score of 97.8% and precision of 98.2%.

Table 3: Precision, recall, and F1-score by class for Scenario One (S1) and Scenario Two (S2)

Class	Precision-S1	Recall-S1	F1-S1	Precision-S2	Recall-S2	F1-S2
No fault	82.7%	89.5%	86.0%	96.8%	98.3%	97.4%
Remote control fault	83.8%	85.0%	84.3%	97.0%	97.3%	97.2%
GPS fault	83.1%	84.0%	83.5%	97.2%	97.1%	97.2%
Accelerometer fault	85.7%	84.5%	85.1%	97.4%	97.3%	97.3%
Gyroscope fault	89.4%	84.0%	86.6%	98.2%	97.4%	97.8%
Magnetometer fault	81.1%	82.5%	81.4%	96.7%	96.7%	96.7%
Barometer fault	84.5%	85.0%	84.7%	97.1%	97.3%	97.2%

The improvement in precision for minority classes (e.g., magnetometer from 81.1% to 96.7% and barometer from 84.5% to 97.1%) demonstrates the effective role of WGAN-GP in reducing false positives and increasing model confidence in challenging classes.

Figure 3 presents the accuracy trends across federated learning communication rounds for both scenarios, while Figure 4 displays the corresponding loss curves. In both scenarios, training and validation accuracy increase rapidly from the initial rounds and reach relatively stable plateaus in the later rounds. Notably, the gap between training and validation curves is larger in the scenario without WGAN-GP (indicating generalization difficulty in the presence of class imbalance), whereas in the second scenario this gap is reduced, with both curves converging to high values above 97%. This behavior is consistent with the well-established role of data augmentation in mitigating overfitting.

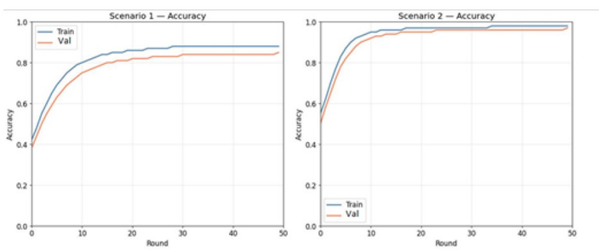

Figure 3: Accuracy curves across federated learning communication rounds–Scenario One and Two

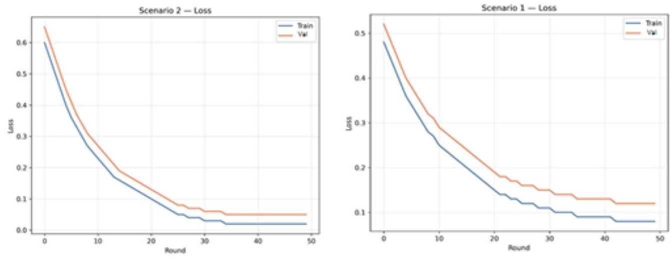

Figure 4: Loss curves across federated learning communication rounds–Scenario One and Two

Figure 5 illustrates the distribution of sample counts across the seven classes before and after augmentation with WGAN-GP. Prior to augmentation, sample counts are imbalanced across classes, with classes such as remote-control fault and GPS fault having the smallest shares, while the accelerometer fault class is most prevalent. Following WGAN-GP application, sample distribution across all classes is equalized to the majority class level.

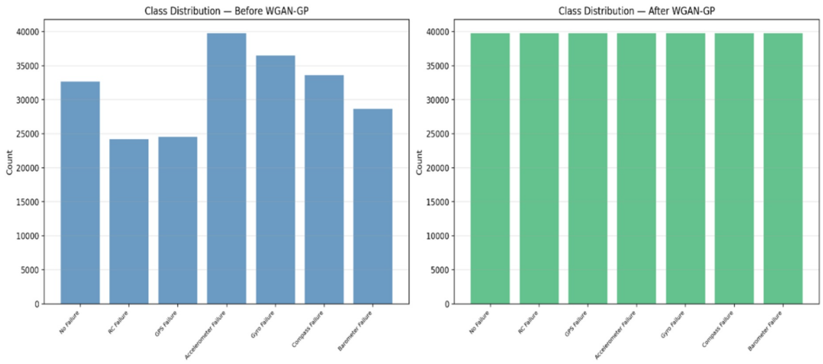

Figure 5: Sample count distribution by class before and after WGAN-GP augmentation

The confusion matrices for both scenarios are presented in Figure 6. In Scenario One, most classification errors occur between adjacent classes in the table (e.g., remote control fault and GPS fault), which may be attributed to similarity in oscillation patterns of these sensors during certain temporal intervals. In Scenario Two, the diagonal values of the confusion matrix increase noticeably while off-diagonal values decrease, directly correlating with the overall accuracy improvement from 84.8% to 97.3%.

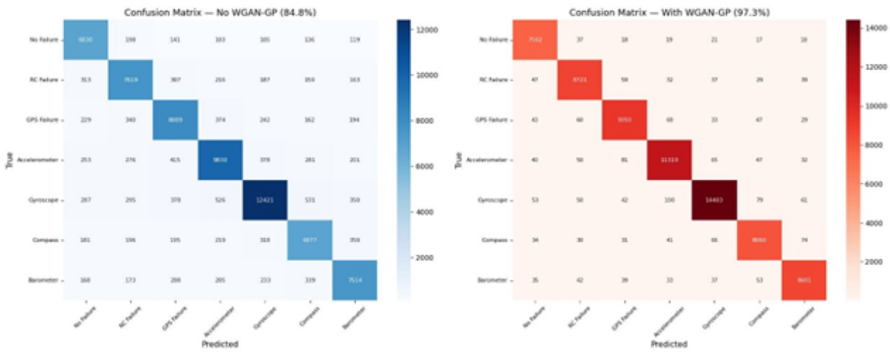

Figure 6: Confusion matrices for Scenario One and Scenario Two

Figure 7 presents a bar chart comparison of overall accuracy and macro-averaged F1-score between the two scenarios. The addition of WGAN-GP improves overall accuracy from 84.8% to 97.3% (an improvement of 12.5 percentage points) and macro-F1 from 84.6% to 97.3% (an improvement of 12.7 percentage points).

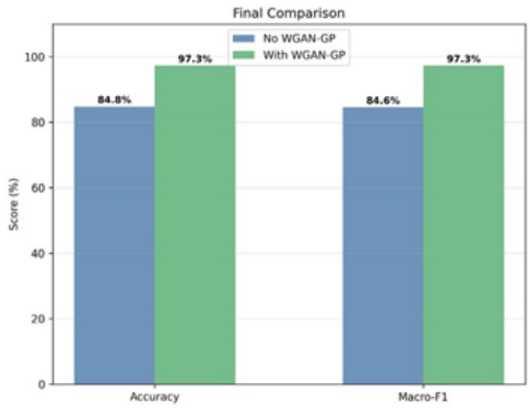

Figure 7: Final comparison of overall accuracy and macro-F1 between Scenario One and Scenario Two

5.1 Discussion and comparison with related work

A broader comparison with representative studies is summarized in Table 1, while the following discussion focuses primarily on the methodological differences between the proposed framework and the most closely related baseline [3].

The transformer-based framework introduced in the reference study [3] represents one of the most comprehensive investigations of UAV sensor fault diagnosis using the BASIC dataset. Beyond establishing a publicly available benchmark dataset, the study demonstrated that attention-based transformer architectures are capable of capturing complex temporal dependencies within multivariate UAV sensor signals. Its proposed two-stage pipeline combined early fault prediction, approximately 62 seconds before fault occurrence on average, with subsequent fault type classification, achieving a reported classification accuracy of 94%.

Building upon this foundation, the present study adopts a different methodological perspective by concentrating exclusively on multi-class sensor fault classification while introducing two complementary mechanisms that are absent from the reference framework. Specifically, distributed model optimization is achieved through federated learning based on the FedProx algorithm, enabling collaborative training without transferring raw flight data, whereas class imbalance is addressed through WGAN-GP-based synthetic data augmentation. These components are integrated with a deep convolutional architecture incorporating multi-scale feature extraction, residual learning blocks, and channel-wise attention mechanisms, resulting in an overall classification accuracy of 97.3% under the adopted experimental protocol.

The performance improvement observed in this study is likely associated with the complementary contribution of several architectural and algorithmic components rather than a single design choice. Multi-scale convolution enables the extraction of temporal patterns at different resolutions, residual connections facilitate stable optimization of deeper networks, and channel-wise attention emphasizes the most informative sensor representations. In parallel, the federated learning strategy preserves data privacy without requiring centralized storage of flight records, while WGAN-GP increases the representation of minority fault categories through high-quality synthetic samples. Collectively, these mechanisms provide a balanced learning environment that improves the discriminative capability of the classifier, particularly for underrepresented fault classes.

A comparison between the proposed framework and the reference study further highlights several conceptual differences. First, the reference framework follows a sequential prediction-classification strategy in which early fault prediction precedes fault type identification, whereas the present work focuses exclusively on maximizing multi-class fault classification performance. Second, the training paradigms differ substantially. The transformer-based approach relies on centralized access to flight data, whereas the proposed framework performs collaborative optimization in a federated environment where only model parameters are exchanged, thereby preserving data confidentiality throughout the learning process. Third, the proposed framework explicitly addresses class imbalance through WGAN-GP-based data augmentation, whereas this issue is not directly considered in the reference study.

Overall, these findings indicate that the proposed framework maintains strong classification performance despite operating under the additional constraint of privacy-preserving distributed learning. The experimental results suggest that integrating federated learning with generative data augmentation provides an effective balance between predictive performance and practical deployment requirements. From a broader perspective, the proposed methodology demonstrates three principal advantages: (i) enabling collaborative model training without centralized access to raw sensor data, (ii) improving the representation and detection of minority fault categories through targeted synthetic data generation, and (iii) achieving competitive (and in this study, superior) classification performance compared with a representative centralized baseline.

5.2 Comparison with the published transformer-based benchmark

To further contextualize the performance of the proposed framework, its results were compared with the transformer-based approach reported by Ahmad et al. [3], which was evaluated on the same publicly available BASiC dataset. The comparison is important because Scenario One in the present study is designed primarily as an internal baseline for isolating the contribution of WGAN-GP-based augmentation and should not be interpreted as a state-of-the-art benchmark. The results of the proposed framework and the previously reported transformer-based method are summarized in Table 4.

Table 4: Comparison of the proposed framework with the published transformer-based benchmark on the BASiC dataset (*The 94.0% accuracy represents the result reported in [3] under the experimental protocol adopted in that study and is not necessarily obtained under the identical experimental protocol used in the present work.)

Method	Dataset	Learning Setting	Accuracy	Evaluation Protocol
Transformer [3]	BASiC	Centralized	94.0%*	Reported in [3]
Proposed CNN-FL	BASiC	Simulated FL	84.8%	This study
Proposed CNN-FL + WGAN-GP	BASiC	Simulated FL	97.3%	This study

The comparison indicates that the proposed CNN-FL framework without generative augmentation achieves 84.8% accuracy, whereas the addition of WGAN-GP increases the accuracy to 97.3%. Thus, the proposed augmentation strategy provides an improvement of 12.5 percentage points relative to the corresponding internal baseline. This improvement should be interpreted specifically as the contribution of WGAN-GP within the proposed federated CNN framework, rather than as an improvement over the transformer-based method reported in [3].

The 97.3% result is higher than the 94.0% accuracy reported by [3]; however, because the two studies were originally conducted under different experimental protocols, this difference should not be interpreted as a definitive state-of-the-art claim. A direct methodological comparison requires reproduction of the transformer-based approach under the same data partitioning, preprocessing, training, and evaluation conditions. Therefore, the present comparison is reported primarily to contextualize the proposed results and to distinguish the internal ablation baseline from previously published results.

5.3 Ablation study

To quantify the contribution of each major component of the proposed framework, an ablation study should be conducted by incrementally incorporating architectural and algorithmic modules into the baseline CNN model. Beginning with a conventional convolutional neural network, squeeze-and-excitation (SE) attention modules, residual learning blocks, federated optimization based on FedProx, and finally WGAN-GP-based data augmentation are added sequentially. Each configuration should be trained and evaluated under identical experimental settings, including the same training, validation, and test partitions, hyperparameter configuration, and evaluation metrics. This experimental protocol enables the isolated contribution of each component to be assessed while ensuring a fair comparison across all model variants.

The evaluated configurations are summarized in Table 5. The final model corresponds to the complete framework proposed in this study, integrating all architectural components with privacy-preserving federated learning and synthetic minority-class augmentation.

Table 5: Ablation study design of the proposed framework

Experiment	CNN	SE Attention	Residual Blocks	FedProx	WGAN-GP	Purpose
E1	✓	–	–	–	–	Baseline convolutional classifier
E2	✓	✓	–	–	–	Evaluate the contribution of channel-wise attention
E3	✓	✓	✓	–	–	Assess the effect of deeper residual feature extraction
E4	✓	✓	✓	✓	–	Evaluate the impact of privacy-preserving federated optimization
E5 (Proposed)	✓	✓	✓	✓	✓	Evaluate the complete framework with synthetic data augmentation

The comparison between successive configurations allows the individual contribution of each component to be analyzed. Improvements observed from E1 to E3 quantify the benefits of architectural enhancements, whereas the transition from E3 to E4 reflects the impact of distributed federated optimization. Finally, the comparison between E4 and E5 isolates the effect of WGAN-GP-based synthetic data augmentation for mitigating class imbalance while preserving the privacy-aware federated learning framework.

5.4 Limitations

As previously noted, the proposed framework addresses two major limitations of prior approaches (namely, lack of privacy preservation and neglect of class imbalance) through the application of federated learning and WGAN-GP. However, like any scientific study, this research has certain limitations.

The proposed framework was designed and evaluated for single-sensor fault detection and classification at any given time instant. In real operational scenarios, however, UAVs may experience simultaneous faults in multiple sensors. Extending the framework to multi-fault scenarios would require redefinition of the class structure, increase the number of output classes, and consequently increase computational requirements and model training complexity. Given that the primary objective of this research is to demonstrate the efficacy of combining federated learning and WGAN-GP in improving fault classification accuracy, and the BASiC dataset covers only single faults, investigating multi-fault scenarios is considered a primary direction for future research.

Another limitation concerns the privacy and deployment interpretation of the federated learning setting. The BASiC dataset is publicly available and consists of simulated flight data; therefore, the present experiments do not constitute a direct evaluation of privacy protection for sensitive operational UAV data. Instead, federated learning is used to simulate a decentralized training scenario in which raw data remain local to participating clients. Moreover, the clients are implemented as virtual machines rather than physically distributed UAV platforms. Consequently, practical factors such as communication latency, bandwidth limitations, intermittent connectivity, client dropout, and device-level computational constraints are not explicitly modeled. Future research should validate the proposed framework using real-world distributed UAV data and realistic federated communication conditions.

6 Conclusion and future work

This study presented an integrated framework for UAV sensor fault classification that trains a convolutional neural network (featuring multi-scale feature extraction, residual blocks, and channel-wise attention) within a federated learning environment (FedProx), and employs a Wasserstein Generative Adversarial Network with Gradient Penalty (WGAN-GP) to balance minority classes in the training set. In this framework, each UAV operates as an independent virtual machine with access only to its own data, exchanging exclusively model weights with the server.

Using a conservative evaluation protocol that partitions training/validation/test data at the flight file level, two explicit scenarios were compared on an identical test set: Scenario One (CNN with federated learning) achieved 84.8% overall accuracy, while Scenario Two (with WGAN-GP addition) elevated this figure to 97.3%. 97.3% is a 12.5-percentage-point improvement over the authors' internal non-augmented CNN-FL baseline, not a 12.5-point improvement over the existing literature. These results demonstrate that combining federated learning with generative adversarial data augmentation can simultaneously address both data privacy and class imbalance challenges in UAV sensor fault detection, without requiring centralization of raw flight data.

Future work may include the following directions: integrating a fault timing prediction module alongside fault type classification, validating the framework on real flight data and other autopilot platforms such as PX4, and investigating scenarios involving simultaneous occurrence of multiple sensor faults.

References

[1] Z. Adaika, L.A. Al-Haddad, W. Giernacki, A.A. Jaber, M. Boumehraz, M.N. Hamzah and M.A. Flayyih, *Fault detection and diagnosis methodologies for unmanned aerial vehicles: State-of-the-art*, J. Intell. Robot. Syst. **111** (2025), no. 2, 63.

[2] M.W. Ahmad and M.U. Akram, *UAV sensor failures dataset: Biomisa arducopter sensory critique (BASiC)*, Data Brief **52** (2024), 110069.

[3] M.W. Ahmad, M.U. Akram, M.M. Mohsan, K. Saghar, R. Ahmad and W.H. Butt, *Transformer-based sensor failure prediction and classification framework for UAVs*, Expert Syst. Appl. **248** (2024), 123415.

[4] X. Chu, X. Zhou, Q. Bu and Q. Miao, *Sensor fault detection for UAVs using improved self-attention LSTM network with similarity space mapping*, IEEE Trans. Instrum. Meas. **73** (2024).

[5] M.H.M. Ghazali and W. Rahiman, *A novel fault detection approach in UAV with adaptation of fuzzy logic and sensor fusion*, IEEE/ASME Trans. Mechatronics **30** (2024), no. 1.

[6] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville and Y. Bengio, *Generative adversarial networks*, Commun. ACM **63** (2020), no. 11, 139–144.

[7] A. Gretton, K.M. Borgwardt, M.J. Rasch, B. Schölkopf and A. Smola, *A kernel two-sample test*, J. Mach. Learn. Res. **13** (2012), no. 1, 723–773.

[8] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin and A.C. Courville, *Improved training of Wasserstein GANs*, Adv. Neural Inf. Process. Syst. (NeurIPS) **30** (2017), 5767–5777.

[9] M.N. Hasan, S.U. Jan and I. Koo, *Wasserstein GAN-based digital twin-inspired model for early drift fault detection in wireless sensor networks*, IEEE Sensors J. **23** (2023), no. 12, 13327–13339.

[10] K. He, X. Zhang, S. Ren and J. Sun, *Deep residual learning for image recognition*, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2016, pp. 770–778.

[11] J. Hu, L. Shen and G. Sun, *Squeeze-and-excitation networks*, Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2018, pp. 7132–7141.

[12] D.P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, Int. Conf. Learn. Represent. (ICLR), 2015.

[13] R. Kohavi, *A study of cross-validation and bootstrap for accuracy estimation and model selection*, Proc. 14th Int. Joint Conf. Artif. Intell. (IJCAI), 1995, pp. 1137–1143.

[14] A. Kumar, S. Wang, A.M. Shaikh, H. Bilal, B. Lu and S. Song, *Building on prior lightweight CNN model combined with LSTM-AM framework to guide fault detection in fixed-wing UAVs*, Int. J. Mach. Learn. Cybern. **15** (2024), no. 9, 1–17.

[15] C. Li, K. Luo, L. Yang, S. Li, H. Wang, X. Zhang and Z. Liao, *A zero-shot fault detection method for UAV sensors based on a novel CVAE-GAN model*, IEEE Sensors J. **24** (2024), no. 14.

[16] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar and V. Smith, *Federated optimization in heterogeneous networks*, Proc. Mach. Learn. Syst. (MLSys) **2** (2020), 429–450.

[17] I. Loshchilov and F. Hutter, *SGDR: Stochastic gradient descent with warm restarts*, Int. Conf. Learn. Represent. (ICLR), 2017.

[18] B. McMahan, E. Moore, D. Ramage, S. Hampson and B.A. y Arcas, *Communication-efficient learning of deep networks from decentralized data*, Proc. 20th Int. Conf. Artif. Intell. Stat. (AISTATS) (2017), 1273–1282.

[19] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg and J. Vanderplas, *Scikit-learn: Machine learning in Python*, J. Mach. Learn. Res. **12** (2011), 2825–2830.

[20] V. Sadhu, K. Anjum, and D. Pompili, *On-board deep-learning-based unmanned aerial vehicle fault cause detection and classification via FPGAs*, IEEE Trans. Robot. **39** (2023), no. 4, 3319–3331.

[21] C.B. Sahin, *Quantum-resilient federated learning for multi-layer cyber anomaly detection in UAV systems*, Sensors **26** (2026), no. 2, 509.