Revista de Informática Teórica e Aplicada - RITA - ISSN 2175-2745
Vol. 33, Num. 2 (2026) 318-325

RESEARCH ARTICLE

Federated Learning on Non-IID Environmental Images for Enhanced Wildfire Surveillance

Aprendizado Federado com Dados Ambientais Não-IID para Vigilância Aprimorada de Incêndios Florestais

Elisa Ribeiro Gonçalves[1]*, Emanuel Teixeira Martins[1], Rodrigo Moreira[2], Larissa Ferreira Rodrigues Moreira[1]

Abstract: Wildfires pose serious threats to ecosystems and human safety, requiring accurate monitoring systems. This study proposes a Federated Learning (FL) approach with Convolutional Neural Networks (CNNs) for wildfire detection using two heterogeneous image datasets, keeping the data locally on each client. The federated setup simulates non-Independent and identically distributed (IID) conditions, where each client trains locally and updates are aggregated its weights to a remote server. To ensure effective performance, hyperparameter optimization for each architecture was conducted using the Tree of Parzen Estimators (TPE), allowing efficient exploration of the best training configurations. Results demonstrate that FL can handle data heterogeneity while preserving privacy, with deeper CNN architectures achieving superior performance. The findings highlight the feasibility of FL for wildfire surveillance and the ability of optimized CNNs to generalize effectively across diverse environmental conditions, supporting collaborative model training without sharing raw data.
Keywords: federated learning — convolutional neural networks — wildfire detection — hyperparameter optimization (TPE)

Resumo: Incêndios florestais representam sérias ameaças aos ecossistemas e à segurança humana, exigindo sistemas de monitoramento precisos. Este estudo propõe uma abordagem de Aprendizado Federado (AF) com Redes Neurais Convolucionais (CNNs) para a detecção de incêndios florestais utilizando dois conjuntos de imagens heterogêneos, mantendo os dados localmente em cada cliente. A configuração federada simula condições não independentes e identicamente distribuídas (não-IID), nas quais cada cliente treina localmente e as atualizações de pesos são agregadas em um servidor remoto. Para garantir um desempenho eficaz, a otimização de hiperparâmetros para cada arquitetura foi realizada utilizando o método Tree of Parzen Estimators (TPE), possibilitando a exploração eficiente das melhores configurações de treinamento. Os resultados demonstram que o AF é capaz de lidar com a heterogeneidade dos dados ao mesmo tempo em que preserva a privacidade, com arquiteturas de CNN mais profundas alcançando desempenho superior. As descobertas ressaltam a viabilidade do AF para a vigilância de incêndios florestais e a capacidade de CNNs otimizadas de generalizar de forma eficaz em diferentes condições ambientais, apoiando o treinamento colaborativo de modelos sem a necessidade de compartilhar dados brutos.
Palavras-Chave: aprendizado federado — redes neurais convolucionais — detecção de incêndios — otimização de hiperparâmetros (TPE)

[1] *Instituto de Ciências Exatas e Tecnológicas, Universidade Federal de Viçosa (UFV), Brazil*
***Corresponding authors**: rodrigo@ufv.br, larissa.f.rodrigues@ufv.br
DOI: http://dx.doi.org/10.22456/2175-2745.150636 • **Received:** 01/10/2025 • **Accepted:** 26/10/2025

1. Introduction

Forest fires pose significant threats to biodiversity and the integrity of ecosystems globally [1, 2]. In Brazil, in particular, data from the Brazilian Institute of Geography and Statistics (IBGE) indicate that between 2000 and 2018, the country's biomes experienced a loss of approximately 500,000 km^2 of natural cover. This underscores the magnitude of anthropogenic pressure and environmental degradation, which exacerbate the vulnerability of forests to large-scale fires [3].

Despite the implementation of more stringent initiatives and legislation, as well as the deployment of technological monitoring systems, fires continue to occur on a substantial scale [1]. The tasks of monitoring, predicting, and responding to these incidents have emerged as significant contemporary environmental challenges, particularly in light of Brazil's expansive territory and the heterogeneity of data collected from various regions [4]. A multitude of initiatives, both preventive and reactive, have been employed to address fires. In this context, technology, particularly Artificial Intelligence (AI), has been extensively utilized to address environmental issues [5].

Prior studies have specifically focused on identifying fire outbreaks through image analysis from unique sources, yielding noteworthy results [3]. To address the challenges posed by data heterogeneity, the application of Federated Learning (FL) [6] seems suitable and promising. In contrast to centralized learning, which necessitates the transmission of all data to a single server, federated learning facilitates decentralized model training. This approach preserves privacy and capitalizes on the diversity of the data collected locally. This feature is particularly pertinent to environmental monitoring, where data are non-Independent and Identically Distributed (IID) and are distributed unevenly and heterogeneously across various sensors, regions, and institutions [7].

This work sheds light on the relevant issue of training Convolutional Neural Networks (CNNs) in a federated manner to extract hidden and valuable patterns from images from different sources and enhance the potential of image-based identification. In our approach, we employ non-IID data trained in a distributed fashion and aggregated by the FedAvg algorithm [8]. To maximize performance, we performed hyperparameter optimization for each architecture evaluated and found significant gains in applying the training model with non-IID data.

The structure of this paper is as follows: Section 2 reviews the related literature; Section 3 outlines the proposed methodology; Section 4 presents the findings and discussion; and Section 5 offers the conclusions and future directions.

2. Related Work

Recent studies on wildfire classification have explored UAV imagery, public datasets, and FL, yet most have assumed IID data [9]. Sathishkumar et al. [10] explored forest fire and smoke detection using transfer learning using different pretrained CNNs, such as VGG16, InceptionV3, and Xception. Biswas et al. [11] proposed an early fire detection and alert system based on a modified Inception-V3 model.

Shamta and Demir [12] proposed a Unmanned Aerial Vehicle (UAV)-based surveillance system with onboard deep learning for early detection of wildfires. Li et al. [13] proposed FICNet, a ResNet50-based framework for wildfire classification and detection. The model was trained on the FlameVision dataset and leveraged transfer learning, data augmentation, and visualization techniques.

Siddique et al. [6] proposed a framework for forest fire classification using FL. Edge devices with cameras were used to capture fire-prone areas, and local Convolutional Neural Network (CNN) models were trained with the results aggregated in the cloud. The authors employed an early version of the FlameVision dataset, which was distributed across nodes in an IID manner with data augmentation to expand the samples.

Sharma et al. [14] proposed a multimodal fire detection system that integrates thermal imagery with data from chemical sensors under a FL setup. Panneerselvam et al. [15] proposed IOFireNet, a FL framework using MobileNet with preprocessing (bilateral filtering) and segmentation. Moreira et al. [16] introduced a framework that combines FL with edge-based orchestration for disaster management. Using the AIDER aerial image dataset, which includes fire and smoke among other disaster classes, CNN models were collaboratively trained and served through separate training and inference pipelines. Chettiar [17] proposed an FL approach using the Kaggle Fire Dataset with synthetic metadata. The system addressed non-Independent and Identically Distributed (non-IID) settings, edge deployment, and explainability; however, its evaluation relied on a small dataset with artificial annotations, limiting generalization.

In contrast, this study applies FL to two public datasets under non-IID settings, capturing the heterogeneity of wildfire data. The framework goes beyond single-dataset and centralized approaches by enabling collaborative training across diverse sources, thereby improving model generalization and robustness for wildfire detection.

Table 1 compares wildfire identification studies along six axes: dataset source; Embedded (edge/IoT deployment); use of FL; evaluation under heterogeneous client distributions (non-IID); use of multiple datasets (Multi-DS); and Real-time claims/evaluation (● = Yes, ○ = No). Most prior works rely on centralized, single-dataset pipelines without FL, non-IID analysis, and real-time is often claimed but seldom validated. In contrast, our study jointly evaluates FL under non-IID partitions across two datasets (Multi-DS), while not targeting embedded or real-time execution in this iteration.

Table 1. Summary of wildfire identification studies.

Work	Dataset	Embedded	FL	Non-IID	Multi-DS	Real-time
[10]	Satellites, Google/Kaggle, BoWFire	○	○	○	○	●
[11]	Fire Dataset	○	○	○	○	●
[12]	Forest Fire	●	○	○	○	●
[13]	Flame Vision	○	○	○	○	●
[6]	Flame Vision	○	●	○	○	○
[14]	Authors' dataset	○	●	○	○	○
[15]	Fire Dataset	○	●	○	○	○
[16]	AIDER	○	●	○	○	●
[17]	Fire Dataset	○	○	●	○	●
Ours	FlameVision & Forest Fire	○	●	●	●	○

3. Proposed Method

We propose an approach based on FL with CNNs to train a unified fire detection model using two distinct datasets, FlameVision [18] and Forest Fire [12], without requiring data centralization.

As shown in Fig. 1, the FL setup involves two clients, C_1 and C_2, each trained locally on FlameVision and Forest Fire, respectively, thus creating a non-IID scenario. The updated weights (W_1 and W_2) are sent to a federated server, which aggregates them into the global model (W'). This process is repeated over multiple rounds, progressively refining the model while preserving the data privacy.

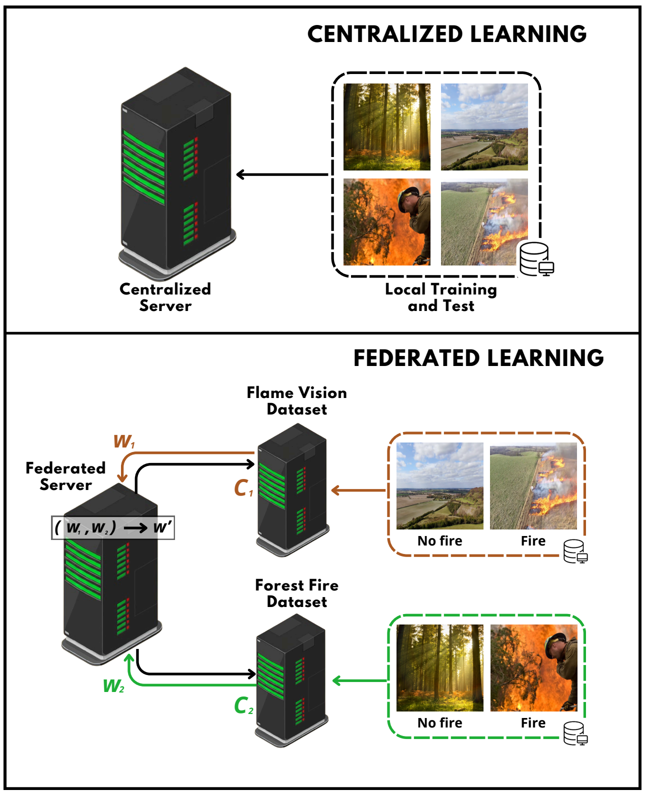
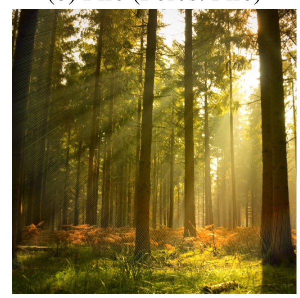


Figure 1. Proposed method.

(a) Fire (FlameVision)

(b) Fire (Forest Fire)

(c) No-fire (FlameVision)



(d) No-fire (Forest Fire)

Figure 2. Examples of fire (top row) and no-fire (bottom row) images from each dataset.

3.1 Image Datasets

In this study, we considered two wildfire image datasets: the FlameVision[1] and the Forest Fire[2]. Both contain RGB images and were used for training and evaluation of the classification models.

The FlameVision dataset consists of 8,600 high-resolution images, including 5,000 fire and 3,600 non-fire scenes (Fig. 2(a) and (c). The Forest Fire dataset includes 2,917 images, with 1,672 depicting active wildfires and 1,245 showing intact forest environments without fires (Fig.2(b) and (d)). To ensure balance, all 2,917 images from the Forest Fire dataset were used, and an equal number of samples (2,917) were randomly selected from FlameVision, resulting in two datasets of the same size for training and evaluation.

[1] ⟨https://www.kaggle.com/datasets/anamibnjafar0/flamevision⟩
[2] ⟨https://data.mendeley.com/datasets/fcsjwd9gr6/2⟩

3.2 Convolutional Neural Networks

CNN are Deep Learning (DL) models widely used in computer vision because of their ability to automatically extract hierarchical features from images for tasks such as classification and object detection [19, 20]. In this study, several representative architectures were evaluated, including AlexNet, ResNet (with three variants: ResNet-18, ResNet-34, and ResNet-50), EfficientNet, and SqueezeNet architectures. All models were initialized with pretrained weights on the ImageNet [21].

- **AlexNet:** introduced key innovations such as the ReLU activation function for faster training and Dropout for overfitting prevention. Its 8-layer architecture (five convolutional and three fully connected) also incorporated overlapping pooling, which reduced error rates, and local response normalization, which improved generalization [21].

- **ResNet:** residual networks enable the training of very deep models through shortcut connections that implement residual learning, mitigating the gradient degradation problem [22]. In this study, we used ResNet-18, ResNet-34, and ResNet-50, which mainly differ in their modular-block design. ResNet-18 and ResNet-34 use basic residual blocks with two 3×3 convolutions, whereas ResNet-50 and deeper variants adopt a more efficient bottleneck block composed of a 1×1, 3×3, and 1×1 convolution sequence, allowing significant depth with a controlled computational cost [22].

- **EfficientNet:** introduced compound scaling, a method that uniformly scales network depth, width, and resolution using a predefined coefficient. The baseline model, EfficientNet-B0, employed in this study served as the foundation for larger variants (B1–B7). Its architecture is built on inverted residual convolutional blocks but optimized with squeeze-and-excitation attention mechanisms [23].
- **SqueezeNet:** uses "Fire modules" consisting of a squeeze layer with 1×1 filters followed by an expand layer with a mix of 1×1 and 3×3 filters. This achieves AlexNet-level accuracy with significantly fewer parameters [24].

3.3 Hyperparameter Optimization

Hyperparameter optimization is an essential process for maximizing the performance of machine learning models, such as CNNs. An optimal configuration of these parameters can prevent the model from underfitting or overfitting [25]. In this study, optimization was performed using the Tree-structured Parzen Estimator (TPE) algorithm, a Bayesian optimization technique that intelligently searches for the best combinations of hyperparameters based on the results of previous trials [26].

The TPE approach models the distribution of hyperparameters $p(x|y)$ and the probability of an outcome $p(y)$. It defines $p(x|y)$ with two densities: $l(x)$ for hyperparameter configurations with the best results (where the loss is $y < y^*$) and $g(x)$ for the others. The next configuration to be tested is chosen by maximizing the ratio of the "expected improvement", which is proportional to $l(x)/g(x)$ [25].

The search space for hyperparameter optimization was defined and explored by TPE, as detailed in Table 2.

Table 2. Hyperparameter Search Space.

Hyperparameter	Distribution	Search Space
Learning Rate (LR)	Log-Uniform	$1\times10^{-5}, 1\times10^{-3}$
Batch Size	Categorical	$\{16, 32, 64\}$
Optimizer	Categorical	{Adam, SGD}

The optimized parameters include:

- **Learning Rate:** This parameter controls the step size taken by the optimizer to minimize the loss function during training [27]. The search space for the LR was defined on a logarithmic scale, which is a common practice for hyperparameters operating across widely varying ranges. This allowed the optimization algorithm to efficiently explore the model's sensitivity to changes in the LR [28].
- **Batch Size:** Defines the number of training samples processed before updating the model's weights [29]. This parameter, which influences both training stability and speed, was optimized as a discrete value among the tested options.
- **Optimizer:** The algorithm responsible for adapting the model's weights to minimize the loss function [30] was treated as a categorical parameter, with TPE identifying the most suitable optimizer.

3.4 Federated Learning

FL is a decentralized approach in which multiple clients train a shared model without exchanging raw data. Each client i trains locally on its dataset D_i and sends the updated parameters θ_i to a central server. The server aggregates these updates to form a new global model using Federated Averaging (FedAvg) [31], as defined in Eq. 1:

$$\theta^{(t+1)} = \sum_{i=1}^{N} \frac{|D_i|}{\sum_{j=1}^{N} |D_j|} \theta_i^{(t+1)} \tag{1}$$

where $\theta^{(t+1)}$ is the global model after round t.

A central challenge in FL is the statistical heterogeneity (non-IID) across clients. Under IID, samples $\{(x_i, y_i)\}_{i=1}^{n}$ are independent draws from a single distribution $\mathscr{D}$; in non-IID settings, client k follows its own process $(x_i^{(k)}, y_i^{(k)}) \sim \mathscr{D}_k$ with $\mathscr{D}_k \neq \mathscr{D}_\ell$ for $k \neq \ell$ (and possible within-client dependencies). Equivalently, centralized learning presumes a single data distribution $P(x,y)$, whereas FL admits client-specific $P_k(x,y)$, inducing covariate/label/concept shifts. This heterogeneity complicates optimization and aggregation but mirrors practices such as wildfire imagery collected across regions, sensors, and seasons [32].

3.5 Evaluation Metrics

The performance of the models was assessed using four standard metrics, where TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives.

- **Accuracy:** proportion of correctly classified samples (Eq. 2).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{2}$$

- **Precision:** proportion of correctly predicted positive samples (Eq. 3).

$$Precision = \frac{TP}{TP+FP} \tag{3}$$

- **Recall:** ability to identify positive samples (Eq. 4).

$$Recall = \frac{TP}{TP+FN} \tag{4}$$

- **F1-Score:** harmonic mean of precision and recall (Eq. 5).

$$F1\text{-}Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{5}$$

4. Results and Discussion

For the experimental setup, all models were trained and evaluated on a Dell Precision 5860 workstation, equipped with an Intel Xeon w3-2435 processor 8 Central Processing Unit (CPU) 3.1 GHz, and 64 GB of DDR5 memory and an NVIDIA RTX 4000 Ada Generation GPU with 20 GB;

Initially, we built a training, validation, and testing pipeline using the Flower [33] library. The evaluation of our method consists of measuring the gain in accuracy when conducting training, validation, and testing of CNNs with heterogeneous wildfire data, specifically using the non-IID format.

The weight aggregation method in our approach is based on the FedAvg [34] algorithm for a setup consisting of 2 clients, C1 and C2, and an aggregation server (S), each with a subset of non-IID data. In each training round, each client trained its model locally with its own data; the clients work with different data from each other, ensuring privacy without compromising the quality of pattern extraction. At the end of the training round, the clients report their weights to the server, which aggregates the weights by averaging and then redistributes the updated weights to the clients, enabling them to build a model that is aware of the burned area image context from another domain.

For each model, the dataset was randomly partitioned, with 80% of the images used for training and 20% for testing. To replicate the heterogeneity found in real-world environments, the non-IID configuration was implemented by assigning a unique dataset to each client. Thus, C1 was trained exclusively with the *FlameVision* [18] dataset, while C2 used only the *ForestFire* [12] dataset.

This intentionally uneven distribution allowed us to assess the model's generalization ability in a context where client data distributions are fundamentally different. By keeping the data local and separate, the approach ensures privacy and collaborative training through the FedAvg algorithm, which aggregates updates from both models on the central server to form a global model at each round.

The hyperparameter optimization for each CNN was conducted using the *Hyperopt* [35] library, which enabled the identification of the best parameter combination for each CNN. This step was important to ensure that each CNN was evaluated in its highest-performing configuration, providing a fair and robust comparison between the different architectures. The search was configured for 50 evaluations per model, with each trial consisting of 10 training epochs. The search space for the Batch Size, LR, and Optimizer parameters is detailed in Table 2.

Considering that the optimization experiments were conducted on two distinct datasets, the best hyperparameters for each model were selected based on the lowest loss (best loss) obtained on the Flame Vision dataset, which showed the most promising results. The search space for the LR was defined on a logarithmic scale, from 10^{-5} to 10^{-3}. This is a common practice for hyperparameters that operate over very different value ranges. This allowed *Hyperopt* to efficiently explore the model's sensitivity to changes in the learning rate.

The optimization results, which indicate the best combination of hyperparameters for each model, are presented in Table 3. These values were used in the subsequent training and evaluation experiments.

Table 3. Results of Hyperparameter Optimization.

Architecture	LR	Batch Size	Optimizer
AlexNet	0.0005984	16	SGD
ResNet50	0.0000425	16	Adam
ResNet34	0.0000172	32	Adam
ResNet18	0.0001124	32	Adam
SqueezeNet	0.0000742	16	Adam
EfficientNet	0.0003586	16	Adam

This stage was crucial to ensure that each model was evaluated in its best-performing configuration, providing a fair and robust comparison between the different architectures.

As Fig. 3 and Fig. 4 present the accuracy and loss curves of the client models, C1 and C2, over the training rounds in a Federated Learning scenario.

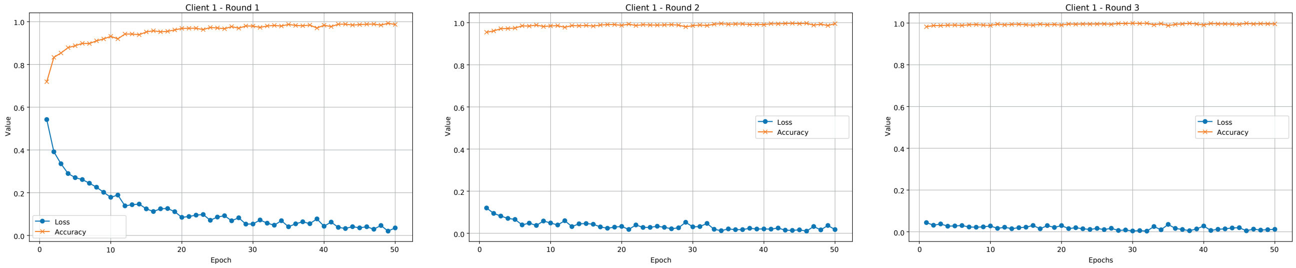
The speed at which the models reach a state of convergence is remarkable. The loss decreases drastically and the accuracy increases within just a few training epochs, a strong indication that the models adjust to their respective local datasets in a highly efficient manner. This behavior demonstrates the intrinsic ability of the models to quickly adapt to each client's local datasets, effectively absorbing the relevant patterns.

In the subsequent rounds, rounds 2 and 3, the stability of the performance metrics suggests that the aggregated model updates on the central server have efficiently consolidated the knowledge that was initially dispersed across heterogeneous datasets. This stability validates the effectiveness of the Federated Learning aggregation process, especially with non-IID databases, which combines the information efficiently and allows the model to generalize robustly from each client's local data.

The analysis of the classification model results, as shown in Table 4, provides a quantitative assessment of the performance of different models. It allows for a direct comparison of accuracy, precision, recall, and F1-Score, as well as training time.

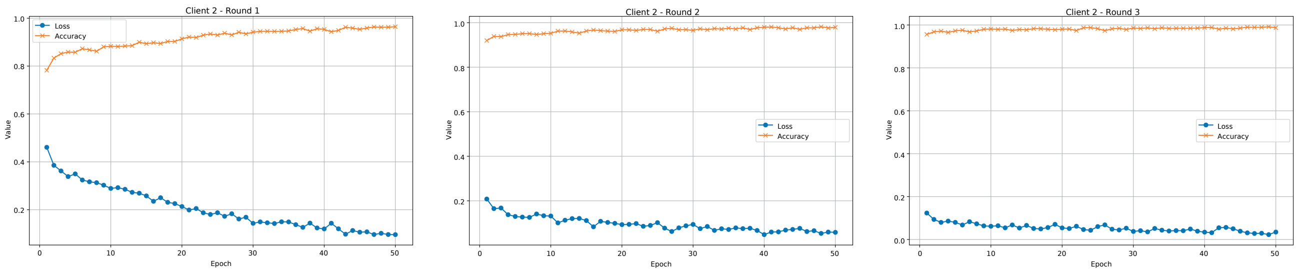
The Resnet50 model stands out as the best overall performer, achieving an accuracy of 96.42%, along with equally high scores in precision, recall, and F1-Score. Although its training time was 7,132.45 seconds, one of the longest, the superior performance across all metrics suggests that the complexity of its architecture is the most suitable for the classification task. This may be related to the optimization of its hyperparameters, which used the Adam optimizer with a low learning rate (0.0000425), allowing the model to fine-tune its weights effectively.

The Resnet34 and Resnet18 models also delivered solid results, with an identical accuracy of 95.83%. Both used the Adam optimizer and a batch size of 32, but with differ-



(a) Round 1 (b) Round 2 (c) Round 3

Figure 3. Evolution of C_1 during federated training on FlameVision.



(a) Round 1 (b) Round 2 (c) Round 3

Figure 4. Evolution of C_2 during federated training on Forest Fire.

Table 4. Summary of results by model

Model	Training Time	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
AlexNet	6468,60	92.08%	92.41%	92.08%	92.12%
Resnet50	**7132,45**	**96.42%**	**96.43%**	**96.42%**	**96.42%**
Resnet34	6392,22	95.83%	95.87%	95.83%	95.84%
Resnet18	6229,97	95.83%	95.88%	95.83%	95.84%
SqueezeNet	6655,89	94.08%	94.51%	94.08%	94.11%
Efficientnet	6493,94	79.83%	86.22%	79.83%	79.63%

ent learning rates. This robust performance positions them as viable alternatives, offering performance comparable to Resnet50 with shorter training times of 6392.22 and 6229.97 seconds, respectively.

On the other hand, EfficientNet showed the lowest performance, with an accuracy of 79.83%, suggesting that its architecture, even with optimized hyperparameters (Adam optimizer, batch size 16), may not have been the most suitable for the data in question. SqueezeNet and AlexNet showed intermediate results. The performance of AlexNet, which used the SGD optimizer, was the lowest among the models that achieved results above 90%, indicating the possible superiority of Adam-based optimizers for this specific problem.

Embracing heterogeneity rather than enforcing IID pooling proved beneficial: aggregating models trained on disjoint wildfire domains (FlameVision for C_1 and ForestFire for C_2) via FedAvg consistently enhanced the generalization of all CNN backbones, yielding the top accuracy (96.42% with ResNet50; Table 4) and fast, stable convergence within a few rounds (Figs. 3–4). By allowing each client to specialize in its local distribution and exchange only weights, the global model assimilates complementary cues (e.g., background, sensor, and context shifts) that single-domain training misses while preserving privacy. In our view, this is the main contribution: non-IID federated training, coupled with principled hyperparameter tuning, improves CNN accuracy for wildfire imagery without data centralization, offering a practical path to stronger detectors in real-world deployments.

5. Concluding Remarks

The present study evaluated the performance of various convolutional neural network (CNN) architectures in a Federated Learning (FL) scenario, applied to the task of forest fire detection. We contributed a relevant assessment of how heterogeneous databases organized in the non-IID format enrich and improve accuracy metrics. This highlights the potential of model training techniques in distributed environments as a robust and viable approach. Hyperparameter optimization allowed each model to operate at its peak performance, ensuring a solid foundation for comparative analysis.

In this study, we delineate specific characteristics of CNNs applicable to heterogeneous data scenarios, emphasizing potential strategies to enhance performance in contexts extending beyond wildfires. Furthermore, our findings demonstrate competitiveness with the current state of the art by achieving reduced training times, thereby establishing them as efficient alternatives in scenarios where the balance between performance, computational cost, and effective feature extraction is paramount. This research advances the field of fire detection systems by presenting a methodology that safeguards data privacy while optimizing model training in distributed environments.

For future research, we plan to assess the impact of noise introduced by heterogeneous datasets on established state-of-the-art CNN architectures. In addition, we intend to incorporate early fire (smoke-dominant) datasets and micro-client approaches into a unified multi-dataset benchmark.

Acknowledgments

The authors gratefully acknowledge the financial support from FAPEMIG (Grant #APQ00923-24) and the Institutional Program for Scientific Initiation Scholarships (PIBIC) at the Federal University of Viçosa (UFV). This study was financed in part by CAPES - Finance Code 001.

Author Contributions

All authors contributed equally to the development of this study.

References

[1] SHIVAPRASAD, K. et al. Chapter 4 - forest fire and its impact on forest biodiversity. In: SAIKIA, P. et al. (Ed.). *Forests for Inclusive and Sustainable Economic Growth*. Elsevier, 2025. p. 37–53. ISBN 978-0-443-31406-3. Disponível em: ⟨https://www.sciencedirect.com/science/article/pii/B9780443314063000047⟩.

[2] ALAM, G. M. I. et al. Real-time detection of forest fires using firenet-cnn and explainable ai techniques. *IEEE Access*, v. 13, p. 51150–51181, 2025.

[3] ELEUTÉRIO, C. L. et al. Identifying wildfires with convolutional neural networks and remote sensing: application to amazon rainforest. *International Journal of Remote Sensing*, Taylor & Francis, v. 46, n. 7, p. 2665–2688, Apr 2025. ISSN 0143-1161. Disponível em: ⟨https://doi.org/10.1080/01431161.2024.2425119⟩.

[4] SHIMABUKURO, Y. E. et al. Assessment of burned areas during the pantanal fire crisis in 2020 using sentinel-2 images. *Fire*, v. 6, n. 7, 2023. ISSN 2571-6255. Disponível em: ⟨https://www.mdpi.com/2571-6255/6/7/277⟩.

[5] CAMPS-VALLS, G. et al. Artificial intelligence for modeling and understanding extreme weather and climate events. *Nature Communications*, v. 16, n. 1, p. 1919, Feb 2025. ISSN 2041-1723. Disponível em: ⟨https://doi.org/10.1038/s41467-025-56573-8⟩.

[6] SIDDIQUE, A. A. et al. Sustainable collaboration: Federated learning for environmentally conscious forest fire classification in green internet of things (iot). *Internet of Things*, v. 25, p. 101013, 2024. ISSN 2542-6605. Disponível em: ⟨https://www.sciencedirect.com/science/article/pii/S2542660523003360⟩.

[7] LI, X. et al. Two-tier submodel partition framework for enhancing uav swarm robustness in forest fire detection. *IEEE Transactions on Mobile Computing*, p. 1–15, 2025.

[8] MCMAHAN, H. B. et al. Communication-efficient learning of deep networks from decentralized data. In: *Proceedings of AISTATS*. [s.n.], 2017. Disponível em: ⟨https://proceedings.mlr.press/v54/mcmahan17a.html⟩.

[9] CHENG, G. et al. Visual fire detection using deep learning: A survey. *Neurocomputing*, v. 596, p. 127975, 2024. ISSN 0925-2312. Disponível em: ⟨https://www.sciencedirect.com/science/article/pii/S092523122400746X⟩.

[10] SATHISHKUMAR, V. E. et al. Forest fire and smoke detection using deep learning-based learning without forgetting. *Fire Ecology*, v. 19, n. 1, p. 9, Feb 2023. ISSN 1933-9747. Disponível em: ⟨https://doi.org/10.1186/s42408-022-00165-0⟩.

[11] BISWAS, A.; GHOSH, S. K.; GHOSH, A. Early fire detection and alert system using modified inception-v3 under deep learning framework. *Procedia Computer Science*, v. 218, p. 2243–2252, 2023. ISSN 1877-0509. International Conference on Machine Learning and Data Engineering. Disponível em: ⟨https://www.sciencedirect.com/science/article/pii/S1877050923002004⟩.

[12] SHAMTA, I.; DEMIR, B. E. Development of a deep learning-based surveillance system for forest fire detection and monitoring using uav. *PLOS ONE*, Public Library of Science, v. 19, n. 3, p. 1–20, 03 2024. Disponível em: ⟨https://doi.org/10.1371/journal.pone.0299058⟩.

[13] LI, S. et al. Research on fire classification and detection models based on deep learning. In: *Proceedings of the 2024 International Symposium on AI and Cybersecurity*. New York, NY, USA: Association for Computing Machinery, 2025. (ISAICS '24), p. 85–90. ISBN 9798400714429. Disponível em: ⟨https://doi-org.ez35.periodicos.capes.gov.br/10.1145/3744103.3744121⟩.

[14] SHARMA, A. et al. Fire detection in urban areas using multimodal data and federated learning. *Fire*, v. 7, n. 4, 2024. ISSN 2571-6255. Disponível em: ⟨https://www.mdpi.com/2571-6255/7/4/104⟩.

[15] PANNEERSELVAM, S. et al. Federated learning based fire detection method using local mobilenet. *Scientific Reports*, v. 14, n. 1, p. 30388, Dec 2024. ISSN 2045-2322. Disponível em: ⟨https://doi.org/10.1038/s41598-024-82001-w⟩.

[16] MOREIRA, R.; MOREIRA, L. F. R.; SILVA, F. D. O. Unleashing ai-empowered slices on mobile networks for natively cognitive service delivery. *IEEE Access*, p. 1–1, 2025.

[17] CHETTIAR, F. Federated multi-task ai for fire detection and risk assessment with edge deployment and explainability. In: *2025 7th Global Power, Energy and Communication Conference (GPECOM)*. Bochum, Germany: IEEE, 2025. p. 985–990.

[18] JAFAR, A. I. et al. *FlameVision: A new dataset for wildfire classification and detection using aerial imagery*. Mendeley Data, 2023. Disponível em: ⟨https://data.mendeley.com/datasets/fgvscdjsmt/4⟩.

[19] KHAN, A. et al. A survey of the recent architectures of deep convolutional neural networks. *Artificial Intelligence Review*, v. 53, n. 8, p. 5455–5516, Dec 2020. ISSN 1573-7462. Disponível em: ⟨https://doi.org/10.1007/s10462-020-09825-6⟩.

[20] RODRIGUES MOREIRA, L. F. et al. Deep learning based image classification for embedded devices: A systematic review. *Neurocomputing*, v. 623, p. 129402, 2025. ISSN 0925-2312. Disponível em: ⟨https://www.sciencedirect.com/science/article/pii/S0925231225000748⟩.

[21] KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. ImageNet Classification with Deep Convolutional Neural Networks. In: PEREIRA, F. et al. (Ed.). *Advances in Neural Information Processing Systems 25*. Curran Associates, Inc., 2012. p. 1097–1105. Disponível em: ⟨http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf⟩.

[22] HE, K. et al. Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. Las Vegas, NV, USA: IEEE, 2016. p. 770–778.

[23] TAN, M.; LE, Q. EfficientNet: Rethinking model scaling for convolutional neural networks. In: CHAUDHURI, K.; SALAKHUTDINOV, R. (Ed.). *Proceedings of the 36th International Conference on Machine Learning*. PMLR, 2019. (Proceedings of Machine Learning Research, v. 97), p. 6105–6114. Disponível em: ⟨https://proceedings.mlr.press/v97/tan19a.html⟩.

[24] IANDOLA, F. N. et al. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and $<$0.5mb model size. *arXiv:1602.07360*, 2016.

[25] RODRIGUES, L. F.; NALDI, M. C.; MARI, J. F. Comparing convolutional neural networks and preprocessing techniques for hep-2 cell classification in immunofluorescence images. *Computers in Biology and Medicine*, v. 116, p. 103542, 2020. ISSN 0010-4825. Disponível em: ⟨https://www.sciencedirect.com/science/article/pii/S0010482519303993⟩.

[26] BERGSTRA, J. et al. Algorithms for hyper-parameter optimization. In: SHAWE-TAYLOR, J. et al. (Ed.). *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2011. v. 24, p. 4. Disponível em: ⟨https://proceedings.neurips.cc/paper_files/paper/2011/file/86e8f7ab32cfd12577bc2619bc635690-Paper.pdf⟩.

[27] GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. Deep Learning. Book in preparation for MIT Press. 2016. Disponível em: ⟨http://www.deeplearningbook.org⟩.

[28] BENGIO, Y. Practical recommendations for gradient-based training of deep architectures. In: *Neural Networks: Tricks of the Trade: Second Edition*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. p. 437–478. ISBN 978-3-642-35289-8. Disponível em: ⟨https://doi.org/10.1007/978-3-642-35289-8_26⟩.

[29] MASTERS, D.; LUSCHI, C. Revisiting small batch training for deep neural networks. *International Conference on Learning Representations (ICLR)*, 2018. Disponível em: ⟨https://arxiv.org/abs/1804.07612⟩.

[30] RUDER, S. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04746*, 2016. Disponível em: ⟨https://arxiv.org/abs/1609.04746⟩.

[31] MCMAHAN, B. et al. Communication-Efficient Learning of Deep Networks from Decentralized Data. In: SINGH, A.; ZHU, J. (Ed.). *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*. PMLR, 2017. (Proceedings of Machine Learning Research, v. 54), p. 1273–1282. Disponível em: ⟨https://proceedings.mlr.press/v54/mcmahan17a.html⟩.

[32] RODRIGUES, L. G. F. et al. Medical image classification with privacy: Centralized and federated learning comparison. *Revista de Informática Teórica e Aplicada*, v. 32, n. 1, p. 180–187, Feb. 2025. Disponível em: ⟨https://seer.ufrgs.br/index.php/rita/article/view/143478⟩.

[33] BEUTEL, D. J. et al. Flower: A friendly federated learning research framework. *arXiv preprint arXiv:2007.14390*, 2020.

[34] NILSSON, A. et al. A performance evaluation of federated learning algorithms. In: *Proceedings of the Second Workshop on Distributed Infrastructures for Deep Learning*. New York, NY, USA: Association for Computing Machinery, 2018. (DIDL '18), p. 1–8. ISBN 9781450361194. Disponível em: ⟨https://doi.org/10.1145/3286490.3286559⟩.

[35] BERGSTRA, J. et al. Hyperopt: a python library for model selection and hyperparameter optimization. *Computational Science & Discovery*, IOP Publishing, v. 8, n. 1, p. 014008, jul 2015. Disponível em: ⟨https://dx.doi.org/10.1088/1749-4699/8/1/014008⟩.