Real-Time Wildfire Detection in Drone Imagery Using a CNN Ensemble

Patricia Cántaro[1], **Saymon Nicho**[1], **Carlos Zegarra**[1], **Rodrigo López**[1]
[1]Pontificia Universidad Católica del Perú
{patricia.cantaro,saymon.nicho,carlos.zegarra,rlopez}@pucp.edu.pe

Abstract

Wildfires increasingly threaten the environment and public health, and early detection is crucial to minimize their impact. Traditional methods, such as sensor- or satellite-based systems, are limited by coverage and timeliness. In this work, we propose a deep learning-based approach for real-time wildfire detection using drone-captured images. Our method ensembles three convolutional neural networks (Xception, DenseNet121, and ResNet152) fine-tuned on the FLAME dataset. We perform a systematic hyperparameter search and ensemble the best models, with the goal of balancing accuracy and computational efficiency for on-board deployment. DenseNet121 achieves the highest accuracy (86.5%), while the ensemble delivers robust performance (F1-score 0.7169, accuracy 79.4%), outperforming previous baselines. Our results demonstrate the viability of pure-CNN ensembles for rapid and efficient wildfire detection in aerial imagery, with further work directed toward deployment on resource-limited devices.

1 Introduction

Wildfires pose a serious and growing threat due to climate change and increased human activity in forested areas. Their rapid onset and spread cause significant damage to infrastructure, biodiversity, and the economy. Early detection is vital for effective intervention, but traditional methods relying on sensors or satellites often lack real-time capability or wide coverage. Deep learning, particularly convolutional neural networks (CNNs), has emerged as a promising tool for detecting wildfires from imagery with greater accuracy and speed. This project presents an ensemble of deep CNN models optimized for real-time wildfire detection in drone-captured images, aiming to improve both detection accuracy and deployment feasibility for aerial monitoring systems.

2 Problem Statement

Despite remarkable progress in wildfire detection using deep learning, many state-of-the-art architectures remain computationally intensive, making real-time deployment on drones difficult. While some optimized models for resource-limited devices exist, they may not utilize the latest detection strategies, which can impact both accuracy and efficiency. Our goal is to explore and validate deep learning models that strike a balance between precision and computational demands, enabling timely wildfire detection on embedded systems.

Preprint. Under review.

3 Related Work

3.1 Classic Methods and Sensors

Earlier wildfire detection methods rely on networks of ground-based sensors, such as optical, smoke, gas, and temperature sensors. These can provide early alerts but are constrained by limited spatial coverage and often fail to detect fires until they are well-developed. Similarly, classic computer vision methods that use color segmentation to identify flames or smoke tend to yield high false positives, mistaking clouds, fog, or sunlight for fire indicators [1].

3.2 Machine Learning and Deep Learning

Advances in computer vision have extended wildfire detection beyond simple color-based methods. Modern approaches incorporate features such as shape, texture, and brightness variations.

CNNs for Classification: Architectures like Inception, Xception, DenseNet, and ResNet have shown strong performance for binary fire/no-fire image classification tasks. Transfer learning (using pretrained ImageNet models) and data augmentation are widely used to improve results on smaller, specialized datasets [3].

Object Detection with YOLO: Unlike standard CNNs that classify whole images, YOLO (You Only Look Once) detects and localizes objects (such as fires) in real time. YOLOv8, in particular, adopts anchor-free detection and a lightweight backbone (CSPNet), boosting speed and accuracy for surveillance applications [2, 4].

Vision Transformers (ViTs): ViTs employ self-attention to model global relationships in images and have recently matched or exceeded CNN performance for fire detection, especially when large-scale pretraining is available. The integration of ViTs in ensembles has led to further improvements in accuracy [3].

4 Dataset

We use the **FLAME dataset**, which contains aerial images of wildfires captured by drones in northern Arizona:

- **Training/Validation:** 39,375 images (10% validation split)
- **Test:** 8,617 images
- **Classes:** Fire (63.55%), No Fire (36.45%)

The dataset is specifically designed for binary wildfire classification tasks.

5 Data Preprocessing

The FLAME dataset is already curated for wildfire classification, minimizing the need for additional filtering. All images are resized to 224×224 pixels to match the input requirements of popular CNN architectures, and pixel intensities are normalized to the $[0, 1]$ range. We analyzed the data distribution to ensure balanced evaluation, but did not employ additional augmentation for this study, focusing instead on rigorous model optimization.

6 Architecture

Our ensemble comprises three individually fine-tuned CNNs: Xception, DenseNet121, and ResNet152, each initialized with ImageNet-pretrained weights. Fine-tuning is tailored for the FLAME dataset, with selective unfreezing of layers and regularization.

6.1 Ensemble Fusion Strategies

We experimented with two main strategies:

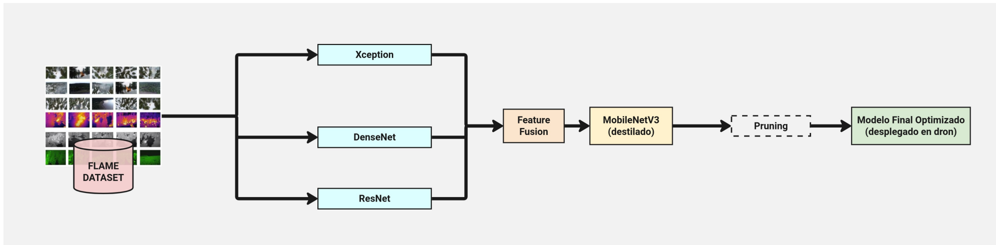

Figure 1: FLAME Dataset Sample

1. **Output Fusion:** Averaging or weighted combination of predicted probabilities for each class.
2. **Voting:** Each model votes for the predicted class; the class with the most votes is chosen (majority voting).

Voting provided the best trade-off between robustness and simplicity for our binary classification task. Additionally, feature-level fusion and stacking were considered but not pursued for deployment feasibility.

6.2 Knowledge Distillation and Lightweight Deployment

For deployment on drones, model size and inference time are critical. We propose to distill the ensemble into a MobileNetV3 architecture using soft-label supervision, aiming to retain performance while drastically reducing computational requirements. Pruning and quantization are future avenues to further accelerate inference on embedded hardware.

7 Hyperparameter Search

Each model's hyperparameters were optimized using `Keras Tuner` with random search. The key hyperparameters explored included learning rate, dropout, L2 regularization, and number of unfrozen layers. Up to 10 trials per model were conducted, with early stopping on validation loss to prevent overfitting.

Best configurations:

- **Xception:** 25 unfrozen layers, 0.45 dropout, 0.001 L2, 0.00541 learning rate
- **DenseNet121:** 20 unfrozen layers, 0.35 dropout, 0.001 L2, 0.00147 learning rate
- **ResNet152:** 45 unfrozen layers, 0.40 dropout, 0.0005 L2, 0.00093 learning rate

Each network was trained for up to 30 epochs with the Adam optimizer, checkpointing the best weights.

8 Results and Evaluation Metrics

Model performance was evaluated using accuracy, F1-score, and confusion matrices on the held-out test set. Table 1 compares our models with the FireSight baseline.

Table 1: Performance comparison on the FLAME test set

Model	F1-score (Baseline)	Accuracy (Baseline)	F1-score (Ours)	Accuracy (Ours)
Xception	0.58	49.09%	0.5087	70.72%
DenseNet121	0.53	70.35%	0.8324	86.50%
ResNet152	0.61	73.20%	0.6484	72.30%
Ensemble	0.75	81.94%	0.7169	79.40%

DenseNet121 achieved the highest accuracy (86.5%), outperforming both the ensemble and the baseline. The ensemble model provided robust, balanced results (F1-score 0.7169, accuracy 79.4%), with voting fusion. Confusion matrices showed that DenseNet had fewer false negatives, a key factor for early fire detection.

9 Conclusions and Future Work

Our experiments show that deep CNN ensembles can achieve state-of-the-art accuracy in wildfire detection from drone imagery. DenseNet121 performed best individually, while the ensemble provided stability. Hyperparameter tuning was crucial for optimal results. Future work includes:

- Distilling the ensemble into MobileNetV3 for lighter, faster inference on drones.
- Model pruning for further size reduction and speed improvement.
- Integrating Vision Transformers and more diverse datasets for greater generalization.

This research underlines the potential of deep learning to enable efficient, near-real-time wildfire monitoring from aerial platforms.

Acknowledgments and Disclosure of Funding

All CPSquad team members made equal contributions in this project. Code is available at `https://github.com/superflash41/isaFIRE`; trained models at `https://huggingface.co/superflash41/fire-chad-detector-v1.0`.

References

[1] Pablo Bot, Mauro Castelli, and Aleš Popovič. A systematic review of applications of machine learning techniques for wildfire management decision support. *International Journal of Disaster Risk Reduction*, 71:102989, 2022.

[2] Mohamed Chetoui and Moulay A. Akhloufi. Fire and smoke detection using fine-tuned yolov8 and yolov7 deep models. `https://www.mdpi.com/2571-6255/7/4/135`, 2024.

[3] Amrita Palaparthi and Sharmila Reddy Nangi. Firesight - wildfire detection through uav aerial image classification. In *Conference on Computer Vision Applications for Wildfire Monitoring*, 2023.

[4] Muhammad Yaseen. What is yolov8: An in-depth exploration of the internal features of the next-generation object detector. `https://arxiv.org/abs/2408.15857`, 2024.