IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 1, January 2026

DOI: 10.17148/IARJSET.2026.13119

FIRE DETECTION USING AI

Ajay Kumar B R[1], Shafeeqa Banu[2], Syeda Asmi[3], Syeda Mariya[4], Shravan Kumar[5]

Assistant Professor, Department of ISE, Maharaja Institute of Technology Mysore, Mandya, India[1]

Student, Department of ISE, Maharaja Institute of Technology Mysore, Mandya, India[2]

Student, Department of ISE, Maharaja Institute of Technology Mysore, Mandya, India[3]

Student, Department of ISE, Maharaja Institute of Technology Mysore, Mandya, India[4]

Student, Department of ISE, Maharaja Institute of Technology Mysore, Mandya, India[5]

Abstract: Rapid, reliable detection of forest fires is critical to minimize ecological, economic, and human losses. This paper presents a practical AI-driven system for early forest-fire detection trained on the D-Fire dataset and deployed end-to-end with a Flask web service and HTML/CSS/JS front-end. The proposed pipeline detects both **fire** and **smoke** in single images, videos, and live streams, triggers configurable alerts, and exposes REST endpoints for easy integration. We report model and system design choices, training regimen, evaluation results, and field-deployment considerations. The solution supports live IP-camera feeds, user image/video upload, and dashboards for incident review. A modular microservice design enables scaling, audit logging, and integration with emergency notification channels.

Keywords: Forest fire detection, smoke detection, deep learning, YOLO, Flask, real-time inference, edge deployment, early warning.

I. INTRODUCTION

Wildfires are intensifying in frequency and magnitude, exacerbated by climate and land-use dynamics. Early, automated detection; especially of **incipient smoke** is essential to reduce response times. Traditional approaches (watchtowers, human patrols, point gas/heat sensors, and satellite-only monitoring) face limited coverage, latency, or high false alarms. Vision-based AI detectors close this gap by continuously analyzing camera feeds with learned spatio-temporal patterns for flames and smoke.

This work contributes:
(i) a trained object-detection model for **fire** and **smoke** on the D-Fire dataset;
(ii) an end-to-end web system with **live feed**, **upload**, and **alerting**;
(iii) a reproducible training recipe and deployment blueprint; and
(iv) quantitative evaluation aligned to practical constraints (latency, compute, bandwidth, lighting, camera noise).

II. MOTIVATION

Manual surveillance struggles with vast forested areas, night conditions, haze/cloud confusions, and operator fatigue. Fixed thresholding rules (e.g., color or motion alone) often underperform under varying illumination and backgrounds. Modern detectors learn discriminative features jointly for **fire** and **smoke** across diverse scenes, enabling robust early warnings and reduced false positives. Operationally, an integrated web-based solution lowers the barrier to adoption by forest agencies and communities: it can run in the cloud, on a control-room server, or on rugged edge boxes near towers.

III. LITERATURE SURVEY

Prior studies span **terrestrial**, **aerial**, and **satellite** sensing, each with distinct coverage, cost, and latency profiles. Terrestrial CCTV towers provide high spatial detail but limited range; UAVs improve vantage flexibility and rapid redeployment at the expense of flight time and maintenance; satellites (e.g., polar-orbiting and geostationary platforms) offer wide-area coverage with revisit delays and coarser spatial resolution. A recurring challenge across modalities is controlling false alarms under changing illumination and weather (sun glint, fog, low clouds, dust, haze), as well as camera artifacts (compression noise, lens flare) and scene dynamics (moving foliage, shadows).

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 1, January 2026

DOI: 10.17148/IARJSET.2026.13119

Image-based methods have evolved in three waves. First came rule-based cues using color and motion in RGB/YCbCr/HSV, together with background subtraction and simple temporal heuristics to capture the characteristic flicker and expansion of flames. A second wave used hand-crafted spatiotemporal descriptors—optical-flow turbulence, wavelet/LBP-TOP texture, and edge statistics—paired with classical classifiers (SVM/Random Forest). The current wave is dominated by deep learning, where single-stage detectors (YOLO family, SSD, EfficientDet) and two-stage variants (Faster R-CNN) learn discriminative features that generalize across backgrounds and lighting, enabling real-time performance on edge GPUs/CPUs.

The literature consistently highlights three practical lessons: [1]detect **smoke** as an early surrogate before flames become visible; [2]combine **smoke + flame** hypotheses in a single pipeline to improve robustness over different fire phases; and [1], [2]leverage multimodal sensing—especially optical + IR/LWIR—to disambiguate smoke from fog/clouds and to improve night performance. Beyond sensing, robust systems apply temporal consistency (track-level confirmation across frames), region priors (sky/horizon masks, exclusion of static hotspots), and hard-negative mining to suppress common confounders such as clouds, chimneys, and bright reflections. Domain-shift issues between training data and field deployments are typically mitigated with augmentation (color jitter, haze simulation, mosaic), calibration using diverse "none" negatives, and periodic model refresh via active learning.[3]

We follow these insights by jointly detecting smoke and flame, enforcing light-weight temporal smoothing during inference, and validating on diverse negatives that reflect real operating conditions.

IV. DATASET

Source: D-Fire (DFireDataset). The dataset contains >21K images across four categories (only smoke, only fire, both, none) with YOLO-format labels. Two classes: **fire**, and **smoke** are annotated via bounding boxes.

Splits: We use the provided train/val/test stratified splits. Non-fire ("none") images are retained to strengthen negative-class calibration.

Preprocessing: Images resized to 640×640 with letterbox padding; per-image normalization; automatic corrupt-JPEG recovery where applicable. Optional augmentations include **HSV jitter**, **horizontal flip**, **mosaic**, **copy-paste**, and random perspective.

Table 1. Dataset summary

Category	Images	Classes (bboxes)
Only fire	1,164	Fire bboxes contribute to total
Only smoke	5,867	Smoke bboxes contribute to total
Fire + smoke	4,658	Both classes present
None	9,838	—
Totals	**21,527**	**Fire: 14,692, Smoke: 11,865**

V. METHODOLOGY

A. Detection Model

We adopt a one-stage detector (YOLO-family) configured for two classes: **fire**, **smoke**. Input size is 640^2. We enable anchor-free heads, distribution-focal loss, and IoU-aware assignment as supported by the chosen release.

Training recipe (from notebook):

- **Backbone/Head**: lightweight YOLO variant suitable for real-time.
- **Epochs**: 100; **batch**: 64; **imgsz**: 640; **patience**: 20 (early stopping).
- **LR schedule**: lr0=1e-4, lrf=1e-5, cosine LR enabled; **optimizer**: default; **cos_lr**: True.
- **Augmentations**: flip 0.5, HSV, mosaic; erasing 0.4 (if available).
- **Best checkpoint**: selected by validation mAP.

B. Inference & Post-processing

- Confidence threshold (0.25); NMS with class-aware suppression.
- **Temporal smoothing** over successive frames: a short queue (8–12 frames) to enforce persistence on

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 1, January 2026

DOI: 10.17148/IARJSET.2026.13119

smoke; lower persistence for flames due to flicker dynamics.

- **Region filters**: optional masking for static hot sources and horizon/sky exclusion on specific cameras.

C. System Architecture (Deployment)

A modular web system supports **live stream**, **file upload**, and **alerts**. **Services**

1. **Inference Service** (Python): loads the ONNX/pt model once; exposes gRPC/REST /infer with JSON output (boxes, classes, confidences).
2. **Web/API Gateway** (Flask): user-facing endpoints, auth, rate limits; routes frames to inference; serves dashboards.
3. **Stream Worker**: subscribes to RTSP/HTTP camera feeds; decodes frames (FFmpeg), batches to inference; overlays boxes; publishes WebRTC/HLS tiles.
4. **Alerting**: threshold + persistence rules; notifies via email/SMS/webhooks; deduplicates events.
5. **Storage**: incident clips (MP4), key frames (JPEG), and JSON metadata in object store; relational DB for audit trails.

Front-end (HTML/CSS/JS)

Live player with bounding-box overlays; camera chooser; confidence/threshold slider; upload page (image/video) for instant predictions; incidents table with acknowledge/resolve actions.

Security & Ops

Flask behind uvicorn, envfor secrets, structured logging, health endpoints, Prometheus metrics.

D. Alert Logic

- **Smoke first**: if smoke persistence ≥ 10 frames in an ROI → **Early Warning**.
- **Flame now**: any high-confidence flame detection triggers **Immediate Alert**.
- **Cooldown** windows avoid alert storms; **dedup** by camera + region.

VI. EXPERIMENTS AND RESULTS

Validation metrics (from the trained notebook)

Class	P	R	mAP@0.5	mAP@0.5:0.95
Fire	0.943	0.914	0.963	0.732
Smoke	0.871	0.794	0.883	0.567
Mean (all)	**0.907**	**0.854**	**0.923**	**0.650**

Latency (per 640² image): preprocess **0.1 ms**, inference **2.9 ms**, post-process **1.2 ms** (≈ **4.2 ms** total per image).

Qualitative observations: early smoke is detected before visible flames in low-contrast scenes; temporal smoothing reduces flicker false positives. Hard cases include fog/cloud layers and distant thin plumes; per-camera region masks and horizon exclusion help.

VII. DISCUSSION

Generalization benefits from diverse “none” negatives, geographically varied smoke/plume textures, and temporal consistency checks. For field rollout, pair the visual detector with situational layers (wind, humidity, historical hotspots) and consider IR co-cameras where feasible. Privacy is addressed by avoiding face/plate retention and obfuscating non-forest scenes.

VIII. CONCLUSION AND FUTURE WORK

We built and deployed a practical, end-to-end AI system for **forest-fire detection** that operates in real time across images, videos, and live streams. The solution covers the full pipeline—from dataset preparation and model training (fire/smoke) to a Flask-based API, web UI, alerting logic, and storage/audit trails—so that agencies can plug cameras in and start receiving actionable notifications. In evaluations, the detector achieves strong precision/recall with efficient latency (≈4.2 ms per 640×640 frame on GPU), and the system-level design (temporal smoothing, region masking, cooldown/dedup) helps suppress false alarms in difficult scenes such as fog, haze, and fast illumination changes.

DOI: 10.17148/IARJSET.2026.13119

Operationally, the architecture is simple to run (Dockerized services behind Nginx/Gunicorn), easy to monitor (health checks, logs, metrics), and secure by default (HTTPS, auth tokens, CORS control), making it suitable for control rooms or edge gateways.

Beyond raw accuracy, a key outcome is **reliability**: the alert pipeline emphasizes persistence checks, human acknowledgment, and clear incident records (timestamps, camera IDs, snapshots/clips). This improves trust, reduces alert fatigue, and creates a feedback loop for data curation. The system is also **extensible**: it supports multiple camera types, batch or streaming inference, and clean REST endpoints so that dispatch centers or external dashboards can subscribe to events without changing field hardware.

Future work will focus on four directions and related extensions:

1. **Multi-sensor fusion**: Combine visible RGB with thermal/LWIR (and, where feasible, NIR) to better distinguish smoke from fog/clouds, improve night performance, and handle back-lit scenes.
2. **Active learning & continual improvement**: Mine hard negatives/positives from the field, route them for quick review, and auto-refresh the model on a rolling basis without service disruption.
3. **Geolocation & triangulation**: Use multi-camera geometry and simple range/bearing estimates to localize events on a map and prioritize dispatch. Integrate weather and terrain layers to provide operators with wind-aware spread context.
4. **Lightweight edge AI**: Deploy INT8/PTQ-compressed models or distilled backbones on low-power devices; add graceful degradation (adaptive frame-rate, dynamic resolution) for constrained links and solar-powered setups.
5. **Uncertainty & governance**: Calibrate confidence scores, expose clear alert states (warning vs immediate), and keep humans in the loop for edge cases—improving accountability and auditability.

In short, the system is **ready to use today** and designed to keep learning from real-world feedback. By fusing additional sensors, adding localization, and hardening the edge deployment, we can shorten detection-to-response time further and make early-stage fires easier to contain.

ACKNOWLEDGMENT

We thank the Department of ISE, MIT Mysore, for infrastructure and support during development and testing.

REFERENCES

[1]. K. Niu *et al.*, "Early Forest Fire Detection With UAV Image Fusion: A Novel Deep Learning Method Using Visible and Infrared Sensors," *IEEE J. Sel. Top. Appl. Earth Observations Remote Sensing*, vol. 18, pp. 6617–6629, 2025, doi: 10.1109/JSTARS.2025.3541205.

[2]. M. Trinath Basu, R. Karthik, J. Mahitha, and V. Lokesh Reddy, "IoT based forest fire detection system," *IJET*, vol. 7, no. 2.7, p. 124, Mar. 2018, doi: 10.14419/ijet.v7i2.7.10277.

[3]. P. Barmpoutis, P. Papaioannou, K. Dimitropoulos, and N. Grammalidis, "A Review on Early Forest Fire Detection Systems Using Optical Remote Sensing," *Sensors*, vol. 20, no. 22, p. 6442, Nov. 2020, doi: 10.3390/s20226442.

[4]. Y. Li, Z. Shen, J. Li, and Z. Xu, "A Deep Learning Method based on SRN-YOLO for Forest Fire Detection," in *2022 5th International Symposium on Autonomous Systems (ISAS)*, Apr. 2022, pp. 1–6. doi: 10.1109/ISAS55863.2022.9757300.

[5]. "Early Wildfire Smoke Detection Using Different YOLO Models." Accessed: Oct. 22, 2025. [Online]. Available: https://www.mdpi.com/2075-1702/11/2/246

[6]. L. Cao, Z. Shen, and S. Xu, "Efficient forest fire detection based on an improved YOLO model,"*Visual Intelligence*, vol. 2, no. 1, pp. 1–7, July 2024, doi: 10.1007/s44267-024-00053-y.

[7]. P. Zhang, X. Zhao, X. Yang, Z. Zhang, C. Bi, and L. Zhang, "F3-YOLO: A Robust and Fast Forest Fire Detection Model," *Forests*, vol. 16, no. 9, p. 1368, Aug. 2025, doi: 10.3390/f16091368.

[8]. Z. Wang, L. Xu, and Z. Chen, "FFD-YOLO: a modified YOLOv8 architecture for forest fire detection," *Signal, Image and Video Processing*, vol. 19, no. 3, pp. 1–8, Jan. 2025, doi: 10.1007/s11760-025-03821-5.

[9]. L. Zhao, L. Zhi, C. Zhao, and W. Zheng, "Fire-YOLO: A Small Target Object Detection Method for Fire Inspection," *Sustainability*, vol. 14, no. 9, p. 4930, Apr. 2022, doi: 10.3390/su14094930.

[10]. W. Yang, Z. Yang, M. Wu, G. Zhang, Y. Zhu, and Y. Sun, "SIMCB-Yolo: An Efficient Multi-Scale Network for Detecting Forest Fire Smoke," *Forests*, vol. 15, no. 7, p. 1137, June 2024, doi: 10.3390/f15071137.

[11]. C. Bahhar *et al.*, "Wildfire and Smoke Detection Using Staged YOLO Model and Ensemble

[12]. CNN," *Electronics*, vol. 12, no. 1, p. 228, Jan. 2023, doi: 10.3390/electronics12010228.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 1, January 2026

DOI: 10.17148/IARJSET.2026.13119

[13]. H. Wang, Y. Zhang, and C. Zhu, "YOLO-LFD: A Lightweight and Fast Model for Forest Fire Detection. | EBSCOhost", Accessed: Oct. 22, 2025. [Online]. Available: https://openurl.ebsco.com/EPDB%3Agcd%3A10%3A12506842/detailv2?sid=ebsco%3Aplink%3Ascholar&id=ebsco%3Agcd%3A183351114&crl=c&link_origin=scholar.google.com

[14]. X. Cao, Y. Su, X. Geng, and Y. Wang, "YOLO-SF: YOLO for Fire Segmentation Detection," *IEEE Access*, vol. 11, pp. 111079–111092, 2023, doi: 10.1109/ACCESS.2023.3322143.

