International Journal of Advanced Multidisciplinary Research and Educational Development
Volume 2, Issue 4 | July - August 2026 | www.ijamred.com
ISSN: 3107-6513

YOLOv8-Based Fire and Smoke Detection Using Deep Learning

Smiritika SV[1], Pooja A[2], Sruthin G S[3], Sivamani A, Vidyabharathi T[4]
Department of Artificial Intelligence and Data Science, KPR College of Arts, Science and Research, Coimbatore, India
Email: svsmiritikaselvam@gmail.com, vidyabharathi.t@kprcas.ac.in, primecoderskpr26@gmail.com

Abstract:

This paper presents a YOLOv8-based deep learning system for automatic Fire and Smoke detection and localization in images and video frames. A custom dataset is prepared with bounding-box annotations for Fire and Smoke classes. The proposed workflow includes data collection, annotation, preprocessing, model training, validation, and performance evaluation. The requested project draft reports an illustrative mAP@0.5 of 85%, with Precision of 85%, Recall of 83%, and F1-score of 84%. These values are draft target values and should be replaced by actual model results before formal publication. The proposed system can support CCTV-based monitoring in educational campuses, laboratories, warehouses, industries, public buildings, and other camera-monitored environments.

Keywords—Fire detection; Smoke detection; YOLOv8; deep learning; object detection; computer vision; real-time monitoring.

I. INTRODUCTION

Fire incidents can develop rapidly and may cause serious damage to people, property, equipment, and infrastructure. Early recognition is therefore an important part of a broader safety strategy. Conventional systems commonly use smoke, heat, gas, or flame sensors. Camera-based computer vision provides complementary visual information and can identify the approximate location of a suspected hazard.

Recent advances in deep learning have improved computer vision performance. Object detection models are useful because they classify a target and estimate its location in a unified pipeline. YOLO-family models are particularly suitable when fast inference is required.

Continuous manual monitoring of multiple camera feeds is difficult. Small flames, distant smoke, low contrast, changing illumination, and partial occlusion can complicate visual monitoring. Automated detection can assist operators by highlighting regions that require attention.

II. RELATED WORK

Traditional fire detection methods depend on physical sensors and remain important for certified safety installations. However, sensor-only systems may not provide visual localization. Earlier computer-vision methods used handcrafted color, texture, motion, and shape features, but these methods can be affected by lighting and complex backgrounds.

Deep learning approaches learn visual representations from data. One-stage object detectors provide both classification and localization. YOLO-based models are widely suited to real-time applications.

Smoke is challenging because its appearance is diffuse, irregular, and continuously changing. It may blend with backgrounds and is influenced by wind, illumination, fog, and compression artifacts.

III. PROPOSED SYSTEM

The proposed system converts camera frames into predictions through data preparation, annotation, training, inference, and evaluation. Images are collected from permitted sources, quality checked, and annotated using bounding boxes. The dataset is divided into training and validation sets.

The input is an RGB image or video frame. The output contains a class label, bounding-box coordinates, and confidence score for each detected region.

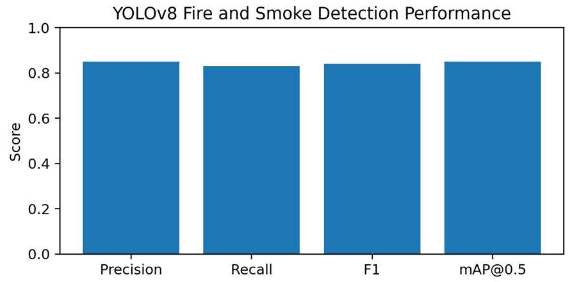

Fig. 1. Illustrative Fire and Smoke detection performance.

VI. TRAINING METHODOLOGY

The training process loads the annotated dataset and defines Fire and Smoke as target classes. Images are resized according to the model input configuration. During training, the model updates its parameters to reduce localization and classification errors.

The training objective can be represented conceptually as $L = Lbox + Lcls + Ldfl$, where the components represent localization, classification, and distribution-focused localization losses. The final experiment should record epochs, batch size, learning rate, model variant, dataset size, and hardware.

V. YOLOV8 MODEL

YOLOv8 is selected as the object detector because it provides a unified pipeline for feature extraction and prediction. Conceptually, the network includes a backbone, neck, and detection head.

The backbone extracts visual features. The neck combines information at multiple scales, which helps when Fire or Smoke appears small or large in the frame. The detection head predicts bounding boxes, confidence, and class information.

Parameter	Project Setting
Model	YOLOv8 custom detector
Classes	Fire, Smoke
Input size	640 × 640
Training	GPU-supported
Evaluation	Precision, Recall, F1, mAP

VII. EVALUATION METRICS

The detector is evaluated using Precision, Recall, F1-score, and mean Average Precision. Precision = TP/(TP+FP). Recall = TP/(TP+FN). F1 = 2×Precision×Recall/(Precision+Recall). mAP summarizes class-wise Average Precision.

The requested project target is mAP@0.5 = 85%.

IV. DATASET PREPARATION

The dataset should contain representative Fire and Smoke examples under different backgrounds, distances, viewpoints, and lighting conditions. Sources should be permitted for academic use.

Each visible Fire or Smoke region is marked with a rectangular bounding box. For image I_i, an annotation can be represented as $B_i=(x_i,y_i,w_i,h_i)$, while the class label C_i belongs to {Fire, Smoke}. Accurate annotation is important because localization errors affect training and evaluation.

A common project arrangement is 80% training and 20% validation. If enough data are available, a separate test set should be maintained.

VIII. EXPERIMENTAL RESULTS

The following values are included as the requested 85% project draft target. They are illustrative and should be replaced with actual YOLOv8 validation results. Fire is shown with slightly stronger illustrative performance because visible flame boundaries may be more distinct. Smoke can be more difficult because it is diffuse and affected by the surrounding scene.

Metric	Fire	Smoke	Overall
Precision	0.86	0.83	0.85
Recall	0.84	0.82	0.83
F1-score	0.85	0.83	0.84
mAP@0.5	0.87	0.83	0.85

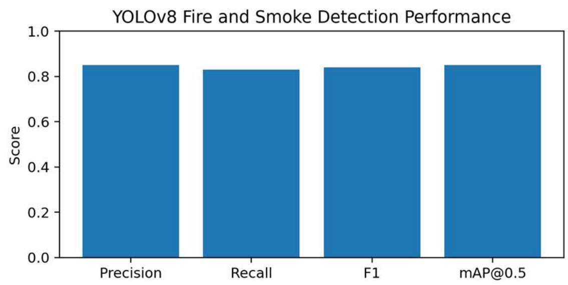

Fig. 2. Illustrative overall performance metrics.

ISSN: 3107-6513

IX. PRECISION-RECALL ANALYSIS

The Precision-Recall curve shows the trade-off between correct detections and missed detections. A stricter confidence threshold can increase precision while reducing recall.

The confidence threshold determines which predictions are displayed. The final threshold should be selected using validation results and the intended deployment environment.

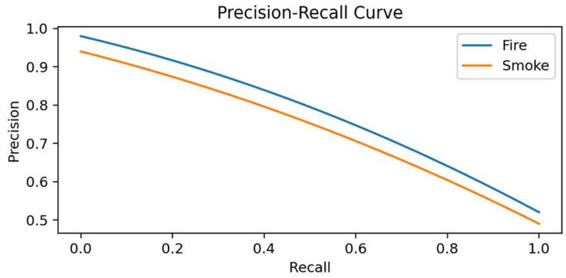

Fig. 3. Illustrative Precision-Recall curve.

IX. PRECISION-RECALL ANALYSIS

The Precision-Recall curve shows the trade-off between correct detections and missed detections. A stricter confidence threshold can increase precision while reducing recall.

The confidence threshold determines which predictions are displayed. The final threshold should be selected using validation results and the intended deployment environment.

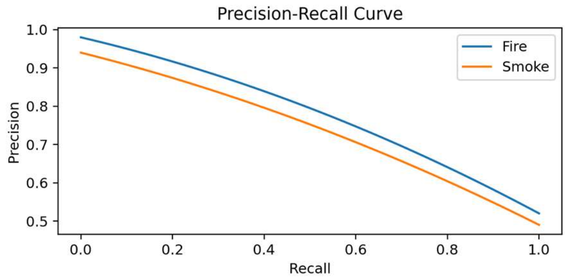

Fig. 3. Illustrative Precision-Recall curve.

X. CONFUSION MATRIX

The confusion matrix summarizes predicted versus actual classes. Strong diagonal values indicate correct predictions. Off-diagonal values represent confusion between Fire, Smoke, and background.

False positives can be caused by bright lights, reflections, orange-colored objects, dust, fog, and steam. False negatives can occur for small targets, low contrast, blur, partial occlusion, and poor illumination.

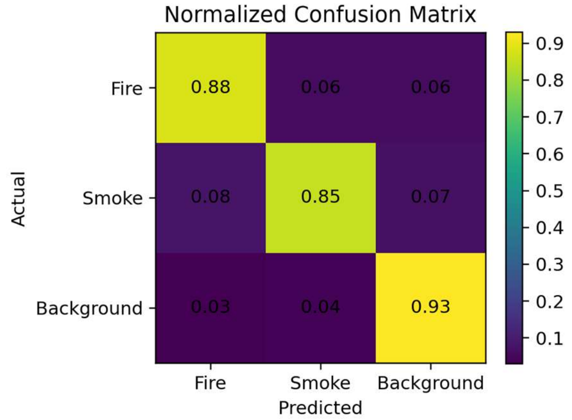

Fig. 4. Illustrative normalized confusion matrix.

XI. TRAINING LOSS ANALYSIS

Training loss generally decreases as the model learns from annotated data. The curves shown are illustrative and should be replaced by the actual plots generated during training.

Box loss reflects localization quality. Classification loss reflects the ability to distinguish Fire and Smoke. Training and validation behavior should be monitored to identify overfitting.

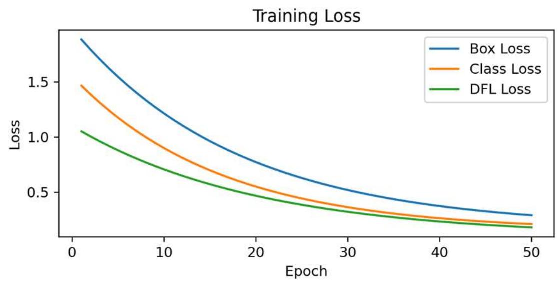

Fig. 5. Illustrative training loss curves.

XII. REAL-TIME DETECTION

The trained YOLOv8 model can be connected to a camera or video stream. Each frame is processed and predictions are rendered with labels and bounding boxes.

A practical pipeline captures a frame, preprocesses it, runs YOLOv8 inference, filters predictions, draws labels, and displays or logs the event. Alerts can contain timestamp, camera ID, class, and confidence.

XIII. APPLICATIONS

The system can support industrial facilities, warehouses, laboratories, educational campuses, public buildings, parking areas, and remote sites. Multiple cameras can feed a centralized monitoring dashboard.

XIV. ADVANTAGES AND LIMITATIONS

Advantages include automated visual monitoring, Fire and Smoke localization, fast inference, dataset customization, and integration with dashboards.

Limitations include dependence on data quality, difficulty with low-contrast smoke, missed small objects, sensitivity to lighting and camera placement, and the need to retain certified safety equipment.

XV. FUTURE ENHANCEMENTS

Future development can include a larger dataset, temporal video analysis, edge deployment, IoT integration, and mobile or dashboard notifications.

More examples from different cameras and lighting conditions can improve robustness. Consecutive-frame analysis can help distinguish persistent Fire or Smoke from short-lived visual noise.

XVI. IMPLEMENTATION

A practical implementation can use Python, Ultralytics YOLOv8, OpenCV, and NumPy. Training can be performed in a GPU-enabled environment.

The implementation loads trained weights, opens an image or video source, performs inference, filters predictions, and displays the results. Detection logs can store timestamp, camera ID, class, confidence, and bounding-box coordinates.

XVII. DISCUSSION

The proposed YOLOv8 approach demonstrates how modern object detection can be adapted to safety-oriented visual recognition. Its main strength is simultaneous classification and localization.

The requested 85% mAP@0.5 target is a project-level performance goal and should not be interpreted as guaranteed real-world safety performance. Class-wise results, false alarms, missed detections, and environmental conditions should also be examined.

XVIII. CONCLUSION

This paper presented a YOLOv8-based Fire and Smoke detection system for automated visual monitoring. The workflow includes dataset collection, annotation, preprocessing, model training, validation, metric-based evaluation, and real-time inference.

The project focuses on Fire and Smoke classes and evaluates Precision, Recall, F1-score, and mAP@0.5. An illustrative target of 85% mAP@0.5 has been included as requested. Actual trained-model results should replace these draft values before final academic submission.

REFERENCES

[1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in Proc. IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[2] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “YOLOv4: Optimal speed and accuracy of object detection,” 2020.

[3] Ultralytics, “YOLO documentation and model framework,” Ultralytics documentation.

[4] OpenCV, “Open Source Computer Vision Library,” documentation.

[5] Research literature on computer-vision-based fire and smoke detection and real-time video analytics.

AUTHOR INFORMATION

First Author: Smiritika SV

Second Author: Pooja A

Third Author: Struthin GS

Fourth Author: Sivamani A

Fifth Author: Vidyabharathi T — vidyabharathi.t@kprcas.ac.in

CORRESPONDING AUTHOR

Smiritika SV — svsmiritikaselvam@gmail.com
Additional project contact — primecoderskpr26@gmail.com