Computer Modeling in Engineering & Sciences

Tech Science Press

https://doi.org/10.32604/cmes.2026.083503

ARTICLE

Multi-Class Severity-Aware Fire and Smoke Detection Using YOLOv12 for Sustainable Intelligent Real-Time Monitoring

Aminah Almehmadi[1], Ayman Noor[1], Aziza I. Noor[2], Hanan Almukhalfi[1] and Talal H. Noor[1,*]

[1]Department of Computer Science, College of Computer Science and Engineering, Taibah University, Madinah, Saudi Arabia
[2]Department of Environmental Health Sciences, Fielding School of Public Health, University of California-Los Angeles (UCLA), Los Angeles, CA, USA

*Corresponding Author: Talal H. Noor. Email: tnoor@taibahu.edu.sa

Received: 05 April 2026; Accepted: 05 June 2026; Published: 30 June 2026

ABSTRACT: Fire emergencies have long posed a serious threat to people's lives, real estate assets, and environmental sustainability in civilized societies, especially when combustible events are detected at late stages of development. Recent advancements in computer vision–based fire detection have enabled automated real-time monitoring; however, most solutions either detect the existence of fire/smoke or employ binary decision-making, which limits visual monitoring systems from being risk-aware. This work introduces a severity-aware fire/smoke detection model that supports intelligent monitoring systems in detecting visual hazards. The goal is to identify varying levels of fire intensity and smoke density and to detect humans in real time. We design a system capable of monitoring environments using components such as sensing devices, network communication buses, cloud data centers, and computer vision–based detectors. The fire/smoke detection model comprises modern deep learning–based object detectors, with the YOLOv12 model serving as the detection backbone. Moreover, our work proposes a two-stage training method that first learns coarse representations of fire, smoke, and humans, and then adapts the detector for fine-grained, severity-aware classification, thereby enhancing severity discrimination and reducing inter-class confusion. We gathered our dataset to comprise approximately 6500 annotated images, split between coarse-grained and severity-aware detection models. The dataset consists of seven classes indicating human presence, three classes indicating varying levels of fire intensity, and three classes indicating varying levels of smoke density. We conducted experiments comparing three baseline object detection architectures (i.e., YOLOv12s, RT-DETR-L, and SSDLite320-MobileNetV3) using identical training/testing configurations. YOLOv12 has outperformed other baseline object detection architectures, achieving 0.929 mAP@50, 0.884 precision, 0.876 recall, 0.880 F1-score, and 2.48 ms per-image latency, providing the best balance between detection performance and real-time processing capability. Our results indicate that severity-aware detection can improve the early-stage detection of intelligent monitoring systems.

KEYWORDS: Fire detection; smoke detection; person detection; severity-aware detection; YOLOv12; real-time monitoring

1 Introduction

Fire incidents are among the most dangerous disasters that affect human lives, buildings, safety features, and emergency response efforts worldwide. There are millions of dollars lost yearly to property damage caused by residential, industrial, and public fires, not including the hundreds of thousands of injuries and deaths that occur every year [1]. In several scenarios, the degree of fire damage suffered correlates with how quickly the fire was detected. Fast detection of fire and smoke is one of the essential elements required

to reduce the risk of increased fire damage, allow time for evacuation plans to take place, and enable quicker responses from emergency teams to fire accidents [2,3]. Smoke is sometimes considered one of the earliest signs of a developing fire. Furthermore, the speed at which smoke propagates throughout a certain environment can be alarming and hazardous to one's health and vision before the fire even occurs [4,5]. As such, being able to sense your surroundings and identify early signs of fire or smoke can play an important role in increasing fire hazard awareness [6].

Traditional fire detection systems have focused on sensor-based systems, such as smoke detectors, heat sensors, and flame detectors, in homes and industrial areas. These sensors have been widely used for the past few decades because they are easy to implement and can detect the presence of a fire when specific conditions are met in the surrounding environment [7,8]. Sensor-based fire detection systems can be constrained by several limitations, reducing their effectiveness and real-world accuracy. For example, conventional smoke detectors are designed to detect fire events based on particulate matter concentrations in the environment and may trigger false alarms due to non-hazardous factors such as cooking smoke, dust, and/or steam [9]. Similarly, conventional heat sensors are designed to detect fire only when the ambient temperature exceeds specific thresholds, which may occur during the later stages of the fire, when the overall fire hazard has already escalated [10]. In addition, conventional fire detection systems are typically implemented for specific environments and may lack sufficient knowledge of the overall environment, thereby complicating the identification of the fire event's location/scale [11]. Thus, the need for more intelligent systems to monitor the environment and provide timely, accurate, and context-aware fire detection is emphasized.

Significant advancements have been made in deep learning, especially in deep learning-based computer vision. As a result, the computer systems' ability to interpret the information accessible to them has been significantly improved. In particular, the application of the Convolutional Neural Network (CNN) model has demonstrated promising results for computer vision-based computer systems, especially for image classification, object detection, etc. [12,13].

This has motivated the research community to explore developing a computer vision-based approach to detect fires and smoke using visual data from cameras [14]. Vision-based detection systems can efficiently interpret the spatial characteristics of fire or smoke, such as color, texture, motion patterns, and shape changes, to detect hazardous situations. This can be done efficiently over a wider area than conventional sensor-based detection systems [14,15]. There are many Deep Learning (DL)-based object detection architectures, such as Faster Region-Based Convolutional Neural Networks (R-CNN), Single Shot MultiBox Detector (SSD), and You Only Look Once (YOLO), that serve the same purpose. They enable efficient simultaneous detection of multiple objects, such as fire and smoke, in image frames, as well as localizing and classifying the objects [16–18]. These techniques enable the automation system to accurately detect fire and smoke in images and are compatible with surveillance cameras [19,20].

Even though significant advances have been made in the development of deep learning-based fire detection systems, most existing studies have focused primarily on detecting the presence or absence of a fire or smoke hazard. In many of the proposed methods for fire detection, the problem is treated as a classification problem where the system is only concerned with the detection of the presence or absence of fire or smoke in the given image [21–23]. Though formulating the problem in this way is helpful for initiating fire or smoke alarm systems, in many real-life situations in safety or disaster management, the severity of the hazard is critical in determining the urgency of response actions [24]. Consistent with the varying fire dynamics and smoke behavior reported in recent studies [25,26], different hazard intensities may require distinct response mechanisms. For example, a small local flame may require a local intervention measure, while a spreading, high-intensity flame may require emergency intervention measures. In addition, local smoke diffusion may signal an early-stage hazard, while local smoke accumulation may signal a more critical

hazard situation in confined spaces. Because of a lack of severity awareness in hazard detection, current fire detection systems may not be effective in providing adequate support for informed decision-making or risk-aware monitoring. This is because current detection systems can only detect the presence of flames or smoke, but cannot distinguish their intensity or density.

To address the limitations of conventional fire detection methods, this research introduces a novel, severity-aware multi-class fire and smoke detection system that provides more informative fire and smoke recognition for effective fire hazard monitoring. To achieve the proposed goal, the research aims to employ deep learning-based computer vision techniques to effectively analyze images captured by surveillance cameras for the recognition of fire hazards. Unlike conventional fire detection systems that can detect the presence of fire or smoke hazards, the proposed system aims to achieve multi-class fire and smoke detection by effectively distinguishing among different levels of fire or smoke occurrence. Specifically, the system aims to detect the presence of a fire hazard with a human while distinguishing between Small Fire, Medium Fire, High Fire, Light Smoke, Medium Smoke, and Dense Smoke. This severity-aware approach will enable the system to provide more context about the identified hazard, thereby improving risk assessment and decision-making in a safe environment. Integrating real-time monitoring through visual observation with multi-class hazard identification can enhance situation awareness and enable quicker responses to accidents.

It's worth pointing out that our proposed system architecture does not introduce a completely new detection backbone. Instead, we leveraged YOLOv12 and extended it to perform a severity-aware multi-class detection paradigm. The novelty of this work is three-fold: (i) redefine the fire/smoke detection task to a fine-grained severity-aware detection problem, (ii) propose a structured two-stage training paradigm that decouples generic feature learning from severity-specific adaptation, and (iii) create a task-specific dataset with hierarchical severity annotations. The contributions in this paper are:

- Developed a real-time human-fire-smoke detector based on the YOLOv12 model, detecting humans, fire events, and smoke in a scene.
- Proposed a two-stage training strategy in which coarse-grained representations of fire, smoke, and human are learned first, followed by fine-tuning the detector for fine-grained severity-aware categorization, leading to better severity discrimination performance with less inter-class confusion.
- A layered system architecture for smart fire monitoring that employs sensor data collection, intelligent DL-based detection, and alarming modules for seamless monitoring and alerting.
- A novel severity-aware labeling protocol that reduces subjective labeling bias and improves annotation consistency.
- A dataset containing 6500 labeled images, each for a coarse-grained and severity-aware model. The dataset includes seven classes for human, three for fire with varying intensity levels, and three for smoke with varying density levels.
- Experimental validation and benchmarking of various object detection models for performance analysis of fire/smoke detection with severity-level awareness, including accuracy, precision, recall, etc.

The remainder of this paper is organized as follows. Section 2 reviews the related work on fire and smoke detection using computer vision and deep learning techniques. Section 3 presents the architecture of the proposed hazard monitoring system, describing the data-acquisition, detection, and response layers. Section 4 explains the severity-aware fire and smoke detection approach and the underlying detection model used in this study. Section 5 describes the implementation details and experimental setup used to train and evaluate the proposed system. Section 6 presents the experimental results and performance evaluation. Finally, Section 7 concludes the paper and outlines possible directions for future research.

2 Related Work

Over the past few years, multiple attempts have been made to develop Artificial intelligence (AI)-based vision systems for forest fire detection and smoke detection/infiltration in infrastructure and industrial settings. Researchers have used powerful DL approaches, such as Convolutional Neural Networks (CNNs) and recent object detection models, to accurately detect fire/smoke instances in real time from imagery. Additionally, there has been work focusing on lightweight versions of fire detection models, reducing false alarms/activations, and implementing fire detection models in cloud-edge collaborative environments. This subsection reviews recent, fundamental works on vision-based fire detection, their unique attributes, and the challenges faced when building effective real-time fire detection models.

Al-Smadi et al. [27] present a DL-based object detection framework for early wildfire smoke detection. The authors compare the inference speed and accuracy of multiple YOLO variants, such as YOLOv3, five versions of YOLOv5, and YOLOv7, against traditional detectors like Fast R-CNN and Faster R-CNN. For this purpose, a Kaggle dataset on wildfire smoke detection was used, and the proposed approach's detection performance was evaluated at three observation distances: near, medium, and far. To overcome the limitation of a small number of images in this dataset, data augmentation techniques were applied, increasing the dataset size from 737 to 1723 images. Experimental results showed that YOLOv5x performed better, with an mAP@50 of 96.8%, followed by YOLOv7 at 95.08%, indicating the superiority of YOLO variants over earlier detectors. However, in visual fire detection systems, a major limitation is the generation of false alarms due to the visual similarity between wildfire smoke and environmental factors, such as bright clouds.

YOLOv8 was compared with recent versions of YOLO detectors (YOLOv9, YOLOv10, YOLOv11) done by Ramos et al. [28] experimentally for the forest fire/smoke detection use case. The experiments were done on a dataset of 9796 images. Images were captured from both top-view and ground-level angles. Each Detector was trained for 100 epochs, using similar hyperparameters to keep benchmarking fair. Based on experiments, YOLOv8 showed promising trade-offs between accuracy, inference time, and training time. For instance, YOLOv8l achieved a higher mAP@50 of 87.2% than YOLOv10, which achieved 86.3%. However, newly developed detectors such as YOLOv9 and YOLOv11 showed relatively low detection accuracy given their model complexity. It was also observed that none of these detectors detected thin smoke at night.

A lightweight fire detection model, DSS-YOLO, was proposed by Wang et al. [29] for fire monitoring. The model addressed challenges in detecting small fires, scattered smoke, and partially occluded fire targets. To achieve a good balance between accuracy and efficiency, three modifications were made to the YOLOv8n model. These modifications include the DynamicConv module to reduce the model's computational complexity and size, the Self-Supervised Equivariant Attention Mechanism (SEAM) attention mechanism to detect small or partially occluded fire targets, and the Spatial Pyramid Pooling Efficient Layer Aggregation Network (SPPELAN) module to improve the extraction of multi-scale features. Experimental results showed that the DSS-YOLO model achieved superior performance with 89.5% mAP50 and 85.4% recall in detecting fire targets, while reducing model size by 3.4% and computational cost by 12.3% compared to the baseline model.

Geng et al. [30] introduced an improved, high-precision model, YOLOv9-CBM, for fire and smoke detection. The paper's proposed model aimed to solve the problems of false alarms and missed detections. Training data issues were also addressed by creating a new custom dataset, CBM-Fire, containing 2000 images of fire and smoke. The paper implemented several new architectural changes to the original YOLOv9 model. These changes include adding the C3 module and the Squeeze-and-Excitation (SE) attention mechanism, which enhance the model's response. The Bi-Functional Feature Pyramid Network (BiFPN) was another improvement on top of the original YOLOv9 model. Lastly, implementing Minimum Point Distance Intersection over Union (MPDIoU) as the loss function was another improvement made to the

baseline model. The paper's proposed model achieved higher accuracy in fire detection than other well-known models. This includes YOLOv5, YOLOv8, and Faster R-CNN. The proposed model achieved a 7.6% increase in Recall and a 3.8% increase in mAP compared to the baseline model.

Lin et al. [31] introduced LD-YOLO, a lightweight deep learning model that builds on YOLOv8 to detect early forest fires and smoke with higher accuracy and faster speeds in resource-constrained environments. To improve the detection of small fires and distinguish them from visually similar backgrounds, such as clouds, certain architectural modifications have been proposed. These modifications include Ghost Convolution (GhostConv) and C2f-Ghost-DynamicConv to reduce model size and computation cost while maintaining high feature extraction capabilities; Dynamic Sample (DySample) to enhance the resolution of small target features; Spatial Context Awareness Module (SCAM) to eliminate interference from the background; and Shape-IoU to enhance the accuracy of bounding box location. To improve the model's accuracy, it has been trained on a dataset of 3603 images, including additional negative cloud images to reduce false detections. From the experiment, the proposed model improved accuracy to 86.3% (mAP@0.5), increasing it by 4.2% over the baseline model, while reducing model size by 36.79% and computation cost by 48.24%, and improving FPS by 15.99%.

Xue et al. [32] presented a new improved model for the detection of fire and smoke. This model was named YOLOv11-DH3. This model was intended to improve detection performance in complex scenes with diverse backgrounds. To address the limitations of the original YOLOv11 model in detecting irregular patterns of fire and smoke, two major modifications were made. First, the Deformable Convolutional Networks (DCNv2) module was replaced with the DCNv3 module to improve the detection performance. Second, the Complete Intersection-over-Union (CIOU) loss function was replaced with the Intersection-over-Union (IOU) loss function to reduce model complexity. This model was validated using two different datasets. One dataset was the close-range monitoring scene (Baidu Paddle), while the other was the long-range monitoring scene (i.e., YOLO Monitoring). The results showed that the model's performance improved by 1.4% in the complex scene. However, the original model performed well in the long-range scene with small smoke targets.

In the paper by Zhang et al. [33], the authors proposed an improved fire and smoke detection system for smart factories based on the improved YOLOv8n model. The paper aims to overcome the limitations of most current fire and smoke detection systems, which primarily focus on forests and cities but ignore the smart factory scenario. To improve the proposed model's performance, the authors built a dataset of over 5000 images for training, including fire and smoke scenarios inside and outside the factory area. To improve the model's detection ability, some improvements were proposed, including the addition of the ConvNeXtV2 module to the model to improve the model's feature extraction ability, the use of the Re-parameterization Block (RepBlock) and Simplified Convolution (SimConv) modules to improve the model's computational efficiency and processing speed, and the replacement of the original loss function with the MPDIoU loss function to improve the model's bounding box location ability. The experiment showed that the model's precision, recall, F1-score, and mAP were all above 95%, with mAP@50 reaching 95.6%, improving over the baseline by 4.5% and achieving processing speeds above 250 FPS.

In the study by Zhou and Jiang [34], the authors proposed a lightweight forest fire detection model, FEDS-YOLOv11n, to improve the efficiency of the original YOLOv11n model for forest fire monitoring. This model aims to overcome the limitations of the conventional model, which are high computational cost, slow processing speed, and poor ability to detect small fires. To achieve the best trade-off between model accuracy and efficiency, the model was improved in several architectural areas. These improvements are the application of the FasterBlock module to decrease the computational complexity and improve the processing speed, the Efficient Multi-Scale Attention (EMA) mechanism to improve the ability to detect slight small-target features,

the DySample Factor, and the Separated and Enhancement Attention Module Head (SEAMHead) to reduce the interference of the background and improve the ability to detect occluded fires. From the experiment, the proposed model reduced the number of model parameters by 21.32% and the computational cost by 26.98% while achieving 71.8 FPS, achieving 79.2% mAP@50 and 73.8% recall on the dataset, outperforming other models such as YOLOv5n, YOLOv8n, and Faster R-CNN.

An integrated fire and smoke detection system was proposed by Yang et al. in [35] to reduce false alarm rates and improve detection accuracy in complex industrial environments. This system is based on a collaborative cloud-edge framework and is an improved version of the YOLOv8 model. To improve the robustness of the proposed system, the authors employed various data augmentation techniques, including image merging, cropping, blurring, and illumination adjustment. Moreover, a spatial attention mechanism called Cross Stage Partial with Two Fusion + Coordinate Attention (C2FCA) is utilized to improve feature extraction and focus on critical visual details. In addition, an iterative transfer learning approach is proposed, in which false alarm cases verified by humans and captured by surveillance cameras are sent to the cloud for continuous retraining and model optimization. Experimental results obtained using a proprietary dataset of 14,380 images showed that the improved model achieves 95.4% accuracy with a processing speed exceeding 25 Frames Per Second (FPS) on NVIDIA Jetson edge devices. After three iterations of retraining in a real-world scenario, the proposed model achieves 87.4% mAP@0.5:0.95, reducing false alarm rates from 200 to fewer than 10 per week.

Yang et al. [36] proposed an integrated intelligent fire detection and alarm system to improve fire monitoring efficiency in buildings and urban areas. In the proposed framework, the fire detection system is based on the cloud-edge collaborative framework. To mitigate latency caused by processing large volumes of visual information, the proposed framework uses Forward Looking Infrared (FLIR) thermal cameras that simultaneously capture RGB and infrared images. Furthermore, the proposed framework uses a Raspberry Pi for image processing with the YOLOv8 model, enabling rapid fire detection in just 80 ms. On the other hand, the proposed framework uses the cloud layer, based on the AWS/Kafka architecture, to ensure stable operation of the fire detection system. Moreover, the linear fusion method is used for image fusion, thereby improving fire detection accuracy. The experiment evaluated the fire detection accuracy of the proposed framework at approximately 89.7% and was conducted at the University of British Columbia (UBC Okanagan).

Borges et al. [37] introduced a system, SEMFOGO-DF, to detect wildfires in Brazilian savannahs (Cerrado) using distributed processing and DL techniques. To train the model, the authors generated two datasets from scratch due to the lack of available data. These datasets consist of sequences of panoramic images labeled with contour masks to locate smoke and zoomed images to classify smoke plumes. The detection task consists of two steps. First, SmokeyNet is used to extract movement information and suspicious regions; this information is sent to the automatic Pan-Tilt-Zoom (PTZ) camera to acquire the focused view of the area. Second, an alert detection algorithm based on InceptionResNetV2 was used to confirm the detection with higher accuracy. The overall precision achieved was 86.97% with an average response time of 11 s. Employing both steps reduced false alarms significantly compared to using only step one. A natural improvement would be to reduce false positives even further at the cost of increasing false negatives, meaning some fires might go unnoticed. Another problem is that there are multiple natural phenomena that appear like smoke, including fog, clouds, and dust.

El-Madafri et al. [38] presented a DL framework, Hierarchical Domain-Adaptive Learning (HDAL), that accurately detects forest fires while reducing false alarms caused by fog/fire interactions or sunlight. The authors have used a dual dataset training approach to minimize the effects of data scarcity and improve the generality of the proposed approach. The proposed approach is trained on two datasets: the first contains

forest fire images, while the second contains images of other non-forest fire scenarios. These images include both urban and vehicle fires. The proposed approach is based on a modified EfficientNetB0 architecture. The proposed approach is designed to identify common features of forest fires and environment-specific features in forest fire images. The proposed approach is effective at preventing overfitting. The effectiveness of the proposed approach is validated through experiments. The proposed approach achieves 95.36% accuracy, 96.11% precision, and 93.83% specificity, thereby minimizing false alarms. The proposed approach is efficient since the time required to make predictions is 5.46 ms per image.

Sultan et al. [39] proposed a deep learning-based multistage fire detection and classification system to overcome the drawbacks of the conventional binary-based approach, which only detects fire occurrence but fails to identify its type or location. For the development of the proposed system, the authors created a dataset by integrating five different datasets. The dataset consists of 17,837 images divided into four different classes: natural images without fire, smoke images, apartment fires, and forest fires. The proposed system is based on a DL approach using the DenseNet201 architecture. The proposed DenseNet architecture is modified by adding more layers to improve the system's feature extraction capability. Moreover, the proposed system leverages the explainability of AI-based approaches via Grad-CAM++ and SmoothGrad. The proposed system is found to perform better at distinguishing different types of fires than other well-known architectures, such as YOLOv8 and ResNet. The proposed system achieves 97% accuracy in fire detection, with precision and recall of 94%.

Suh [40] presented a computer vision-based framework that can monitor the dynamic behavior of fire events rather than only detecting the existence of fire or flames. In the proposed framework, a deep learning-based model, YOLOv5, is used to detect and track three fire-related elements simultaneously: fire or flames, gray smoke, and black smoke. By continuously analyzing the size and position of objects, eight vision-based patterns are proposed to classify fire events and determine risk levels. In the proposed model, 5941 images were used for training, achieving precision values greater than 90% across all classes, including 95% for fire or flames, 93% for black smoke, and 91% for gray smoke. Moreover, the developed framework was tested with an actual video of the Seomun Market fire in Korea and successfully traced the fire progression in real time. Further, the framework detected the moment to take early action before it propagated.

Sun and Cheng [41] presented Smoke-DETR, a real-time fire smoke detection model using a transformer-based object detection method with RT-DETR. The proposed model aims to deliver real-time fire warning systems to take action before it happens. There are various challenges in detecting fire smoke, including irregular smoke shape, confusing background images such as clouds or fog, and the high computational cost of previous detection models. A novel dataset of 4874 images, including negative images without smoke to reduce false alarms during detection, was collected, thereby increasing the proposed model's robustness. Various architectural advancements were also incorporated into the model to improve its performance. For example, Enhanced Partial Convolution (ECPConv) is applied to decrease computation costs by using only relevant image channels, the Effective Attention Module (EMA) is applied to capture features of smoke with irregular shapes, and the Multi-scale Fusion Feature Pyramid Network (MFFPN) is applied to improve separation performance between foreground and background objects. Experimental results showed that "Smoke-DETR" outperformed existing detection models such as YOLOv8 and YOLOv11, achieving 86.8% detection accuracy and mAP@50 of 86.2%, while reducing parameter count by 17% and computation cost by 23.9%.

There has been emerging research showing intelligent diagnosis and risk prediction methods for monitoring disasters and predicting early warnings. For example, a review paper on explainable artificial intelligence in disaster risk management by Ghaffarian et al. [42] which identifies different hazard and disaster types, risk components and elements, AI and Explainable Artificial Intelligence (XAI) methods being

used, discusses the explainability issues and limitations of these methods, and offers consolidated recommendations to improve explainability of these methods for disaster decision-making. Mustafa et al. [43] presented explainable deep learning methods based on convolutional neural networks and Grad-CAM visualization techniques for natural disaster classification toward improved diagnosis transparency of disaster monitoring systems and explainable risk-aware decision making for wildfire, flood, and earthquake monitoring applications.

While considerable progress has been made in vision-based fire and smoke detection, most studies to date primarily focus on detecting the presence of fire or smoke and/or improving detection accuracy through architectural modifications, lightweight models, or deployment strategies. However, these studies generally considered only one aspect of fire or smoke, without accounting for its varying levels of severity within a structured detection model. Some studies improved detection accuracy through architectural optimization, lightweight model detection, or deployment strategies. Some studies extended the detection problem beyond binary classification to contextual categorization (i.e., distinguishing between forest and apartment fires), while a few studies considered post-detection intensity estimation based on flame area or fire or smoke characteristics. However, none of the studies considered a detection model that first detects general objects such as Person, Fire, Smoke, etc., and then refines the detection of Fire or Smoke to different levels of severity within the same model (i.e., using a structured two-stage training strategy). Furthermore, most current works fail to incorporate the concept of human presence alongside fire severity levels within a single object detection model. However, the proposed approach incorporates a well-organized fire and smoke detection model, enabling the system to progress from fire detection to severity levels and produce more useful outputs rather than just raising an alarm. This design bridges the gap between conventional binary fire detection systems and practical risk-aware monitoring solutions as shown in Table 1, most previous works conduct either binary classification or coarse multi-class detection for fires or smoke without considering severity-aware detection. Also, the two-stage methods were designed to refine features or cascade detectors, but not for staged adaptation from datasets with different labeling settings. Meanwhile, our approach explicitly models severity-level annotations for the fire and smoke detection task and devises a two-stage learning scheme that promotes fine-grained classification.

Table 1: Comparison of recent deep learning-based fire and smoke detection approaches.

Ref.	Model/Approach	Dataset	Key Contribution	Performance
[27]	YOLOv3/YOLOv5/YOLOv7 comparison	Kaggle wildfire dataset (737→1723 images)	Comparative evaluation of YOLO detectors at different observation distances	YOLOv5x achieved mAP@50 = 96.8%
[28]	YOLOv8–YOLOv11 comparison	9796 images	Evaluation of recent YOLO detectors for wildfire monitoring	YOLOv8l achieved mAP@50 = 87.2%
[29]	DSS-YOLO (YOLOv8n based)	Fire monitoring dataset	DynamicConv, SEAM attention, SPPELAN modules for lightweight detection	mAP@50 = 89.5%, Recall = 85.4%
[30]	YOLOv9-CBM	CBM-Fire dataset (2000 images)	C3 + SE attention + BiFPN architecture improvements	mAP improved by 3.8%
[31]	LD-YOLO (YOLOv8 based)	3603 images	Lightweight architecture with GhostConv, SCAM, DySample	mAP@0.5 = 86.3%

(Continued)

Table 1 (continued)

Ref.	Model/Approach	Dataset	Key Contribution	Performance
[32]	YOLOv11-DH3	Baidu Paddle & YOLO Monitoring datasets	DCNv3 module and improved loss function	Performance improved by 1.4%
[33]	Improved YOLOv8n	>5000 images	ConvNeXtV2, RepBlock, MPDIoU for smart factory detection	mAP@50 = 95.6%
[34]	FEDS-YOLOv11n	Forest fire dataset	FasterBlock, EMA attention, DySample	mAP@50 = 79.2%, 71.8 FPS
[35]	Cloud-edge YOLOv8 system	14,380 images	Cloud-edge framework with iterative retraining	mAP@0.5:0.95 = 87.4%
[36]	Cloud-edge fire monitoring system	RGB + thermal dataset	Fusion of RGB and infrared images	Accuracy = 89.7%
[37]	SEMFOGO-DF system	Custom wildfire datasets	Two-stage monitoring system with PTZ camera	Precision = 86.97%
[38]	HDAL (EfficientNetB0 based)	Dual dataset	Domain-adaptive learning for reducing false alarms	Accuracy = 95.36%
[39]	DenseNet201 classification	17,837 images	Multi-stage fire classification with explainability	Accuracy = 97%
[40]	YOLOv5 detection & tracking	5941 images	Fire behavior monitoring using vision patterns	Precision up to 95%
[41]	Smoke-DETR (RT-DETR)	4874 images	Transformer-based smoke detection	mAP@50 = 86.2%
[43]	XAI-driven disaster diagnosis and risk prediction framework	Multi-disaster image datasets	Explainable deep learning framework using Vision Transformers, ResNet50, and Grad-CAM for intelligent disaster diagnosis and risk prediction assessment	Accuracy = 94.3%, improved interpretability
This Work	Severity-aware (YOLOv12, RT-DETR, SSD comparison)	Custom Fire Smoke Person dataset (~6500 images, 7 classes)	Integrates person detection with graded fire/smoke severity in a unified severity-aware pipeline	mAP@50" YOLO = 0.929; RT-DETR = 0.931, SSD = 0.882

3 System Architecture

The proposed system for monitoring fire and smoke will be based on a layered architecture capable of detecting them in real time. As shown in Fig. 1, the system architecture comprises three main layers: the Data Acquisition and Sensing Layer, the Detection Layer, and the Response and Decision Layer. These three layers are combined so that the system can capture visual information from the environment, intelligently analyze the severity of the fire and smoke, and generate appropriate responses. Layered system architecture helps develop the system because each layer serves its own purpose and interacts with the other layers. Integrating the system with devices, the cloud, and emergency responders helps develop it by enabling it to act as a single unit for hazard detection.

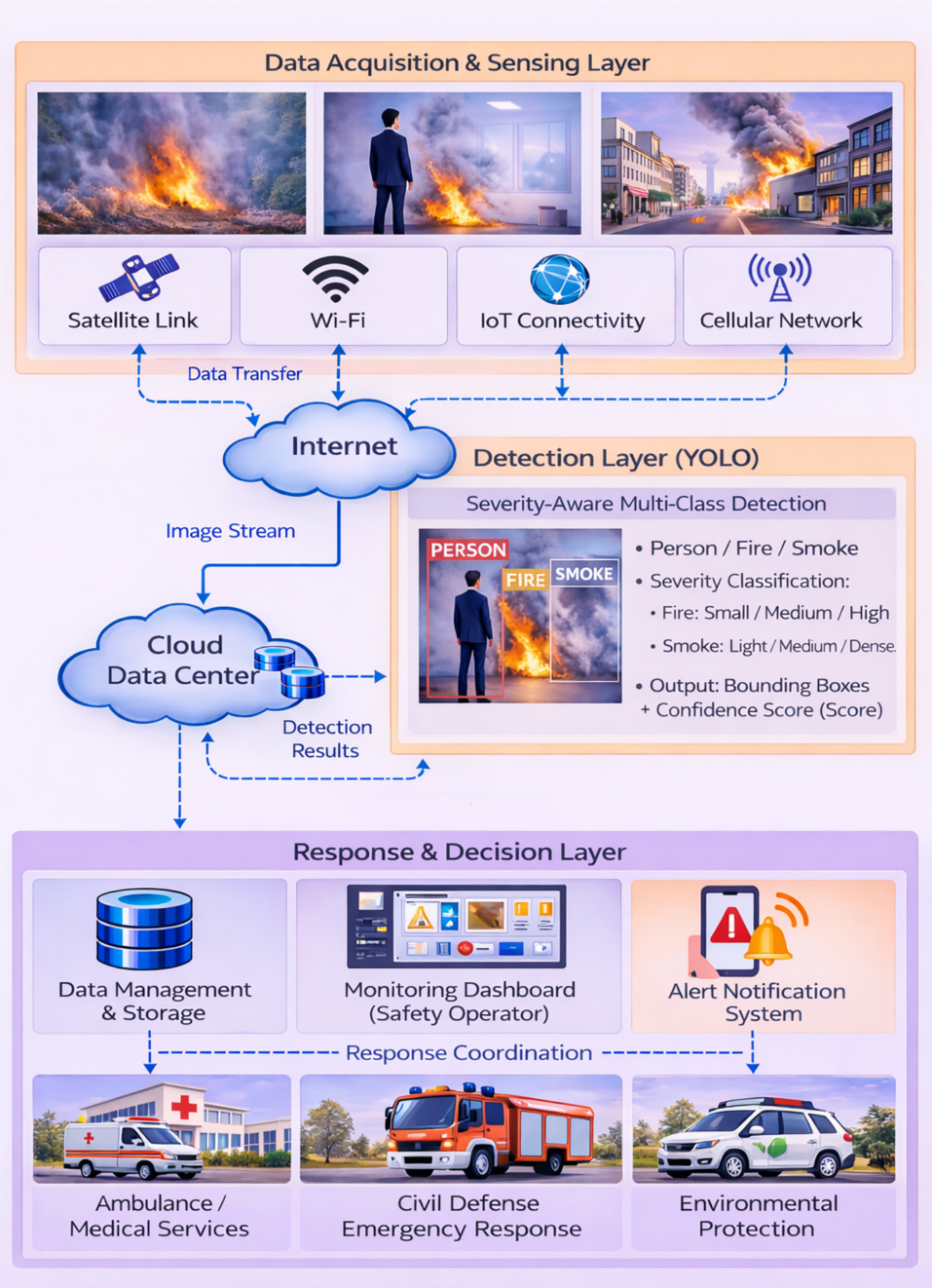

Figure 1: Architecture of the severity-aware fire, smoke, and person detection system.

3.1 Data Acquisition and Sensing Layer

The Data Acquisition and Sensing Layer captures imagery data from the physical environment being monitored. The image sources that serve as components in this layer include surveillance cameras, Internet of Things (IoT) cameras, and other image-capturing devices that take RGB pictures of the environment around them. These cameras monitor the environment and generate streams of imagery on various conditions, with and without fire and smoke present. Images can be obtained from any type of surveillance video feed. This includes both indoor locations, such as buildings and industrial complexes, and outdoor locations, such

as roads, populated areas, or forest-surveillance cameras. Ongoing visual surveillance of the environment allows the system to detect any signs of fire starting or smoke appearing, catching it before it has a chance to grow into a wildfire threat. The visual sensing system acquires images at regular intervals to provide a solid environmental context for analyzing the images. Sending video streams over various communication channels enables the reliable transfer of imagery data. This can be done via Wi-Fi, cellular, satellite, and IoT networks. These networks allow visual data to be transmitted through the internet to the environment where it will be processed. By using these varying networks, the system can function in various settings, whether that be in big cities or remote environments. Once it passes through the communications channel, the captured image data is uploaded to the cloud, where hazard detection occurs.

3.2 Detection Layer

The Detection Layer is the system's most intelligent component and interprets incoming visual information to identify dangerous occurrences. The Detection Layer analyses the incoming image stream with a severity-aware multi-class object detection algorithm to detect fire, smoke, and humans in the scene. This detection model can localize and classify objects in a single inference. The object detection model is based on YOLOv12. The received image frames undergo preprocessing before being passed to the object detection model for evaluation. The preprocessing stage ensures that the received images are properly formatted for evaluation using the object detection model. The preprocessing stage may include image resizing, normalization, and standardization, among other preprocessing steps. The preprocessing ensures that the received images are properly formatted for evaluation with the object detection model and that the evaluation remains stable across various environmental conditions, such as lighting, background, or smoke.

While conventional fire detection systems can only indicate the presence or absence of a fire hazard, the proposed model can provide a more precise assessment of the hazard. The detection model identifies the presence of a human (i.e., as a single class) and classifies fire and smoke hazards into different severity levels. For instance, the model identifies fire hazards as small, medium, and high, and smoke hazards as light, medium, and dense (i.e., as six classes). This will enable the model to identify a wider range of contextual information related to the detected fire hazard. The detection model will generate a set of results, including the detected fire hazard location, the class label, and the confidence level. The detection computation will be performed in a cloud computing environment, enabling real-time DL computation. This will enable the model to handle a wider range of images while maintaining high detection accuracy. Once the computation is performed, the results will be sent to the next architectural layer for further interpretation and response coordination.

3.3 Response and Decision Layer

The Response and Decision Layer is responsible for converting detection results into effective responses that support emergency management and awareness. The Response and Decision Layer comprises modules such as the data management and storage module, monitoring interfaces, and alert notification systems that collectively facilitate effective hazard monitoring and response coordination. The detection results are first stored in the data management and storage module. The module is responsible for storing historical records of detected events and their corresponding metadata. Additionally, the detection's confidence level is stored in the module. The detection results are stored in this module to allow long-term monitoring and evaluation of the detection system. The stored information is also helpful for analyzing past hazard events and providing insights that could inform further research and development of the detection system. Detection results can also be displayed on the monitoring interface, allowing safety operators to track detected hazards as they

occur. By showing operators where something has been detected, what was detected, and at what level, quick situational awareness can be achieved and the necessary response determined.

Once alerted to a hazardous event, the system's alert notification module will broadcast messages to all relevant parties (e.g., emergency assistance groups, other responsible agencies). Alerts may be sent to the appropriate level of authority or operational team, depending on the severity of the detected issue. The proposed system could assist in coordinating several emergency response teams, such as emergency medical services, civil defense, and environmental protection departments, with time-critical events. For example, if a person is detected in a fire and smoke area, an alarm may be sent not only to civil defense groups but also to emergency medical services, as human life may be in danger. Also, if dense smoke is detected, environmental protection departments may be notified as well due to possible air-quality and environmental damage. With connections from the detection feed to the monitoring interfaces and to alert emergency parties, the Response and Decision layer would implement the full workflow for the intended system. This layered approach provides continuous hazard surveillance, the generation of early warnings, and coordinated emergency response, and promotes safety and situational awareness in areas where there could be a fire/smoke hazard. We should clarify that our experiments and evaluations are conducted on the severity-aware detection model in Section 6, where other parts of the system are presented as part of the intended deployment-oriented system architecture demonstration on how our proposed detection model may operate on a deployable intelligent fire monitoring system. These include the data acquisition layer, communication infrastructure, cloud processing services, storage module, monitoring interface, and alerting system.

4 Severity-Aware Multi-Class Detection Model

Fire incidents pose a considerable threat to life safety, property conservation, and environmental sustainability. Predictably, fire events are most devastating when detected late in development. Fire surveillance mechanisms usually involve physical detection sensors, such as smoke or heat detectors. Although sufficient in closed-system deployments, these systems are unable to comprehend the spatial arrangement of objects in their visual field or to contextualize the scene around them. Computer vision–enabled fire detection has enabled intelligent monitoring systems to react autonomously to dangerous incidents in their vicinity. However, most computer vision approaches frame fire/smoke detection as a binary classification problem that outputs only hazard-existence predictions. Thresholding decisions based on binary classification disable intelligent surveillance systems from being risk-aware and taking necessary actions based on the level of emergency.

In this work, we propose a novel fire and smoke detection model that equips monitoring systems with the ability to assess the severity of a potential hazard and its presence. Specifically, our model should predict varying degrees of fire intensity/smoke density and detect humans in the scene. The components of our system include sensing elements (e.g., security cameras), a network bus, a cloud data center, and a deep learning-based computer vision detector.

The pipeline of our proposed fire/smoke detector ingests raw RGB inputs and outputs structured hazard predictions for the scene. These structured hazard detections contain information about object locations, hazard categories, and their relative severity levels. Given an input image captured by a Closed-Circuit Television (CCTV) camera, the resulting embedded image is computed by resizing and normalizing the image using the detector's pre-processing block. The resulting image is then evaluated by the detection network, which simultaneously localizes and classifies objects in the scene. Lastly, post-processing steps are applied to the resulting image to filter low-confidence detections and suppress overlapping bounding boxes.

Traditional fire detection techniques can only indicate whether fire or smoke is present in the scene. In contrast, our model makes predictions over several categories that implicitly annotate the severity of the

observed hazards. Label space is defined as $\mathcal{C} = \{\text{Person}, \text{Fire}_S, \text{Fire}_M, \text{Fire}_H, \text{Smoke}_L, \text{Smoke}_M, \text{Smoke}_D\}$ where Fire_S, Fire_M and Fire_H correspond to small, medium and high fire intensities, and Smoke_L, Smoke_M and Smoke_D correspond to light, medium and dense smoke densities. This novel multi-class formulation enables monitoring systems to predict the severity of the hazards and provides rich contextual information for risk-aware decision-making.

Formally, the detection model takes as input an image and produces a set of predicted detections as shown in Eq. (1):

$$\mathcal{D} = \{(\hat{\mathbf{b}}_j, \hat{c}_j, \hat{s}_j)\}_{j=1}^{M} \tag{1}$$

where $\hat{\mathbf{b}}_j = (\hat{x}_j, \hat{y}_j, \hat{w}_j, \hat{h}_j)$ are the predicted coordinates for the jth bounding box, $\hat{c}_j$ is the predicted class label, and $\hat{s}_j$ is a confidence score.

YOLOv12-based detectors implement the detection head such that class logits and bounding-box distributions are predicted for each candidate spatial location as shown in Eq. (2):

$$\mathbf{z}_j \in \mathbb{R}^K \tag{2}$$

represent the vector of class logits predicted for location j, where K is the number of classes. The class probability vector is obtained using the sigmoid activation function:

Where $\mathbf{z}_j$ are the vector of class logits predicted at location j, and K is the number of foreground classes. The predicted class probability vector $\mathbf{p}_j$ is then computed as follows in Eq. (3) using the sigmoid activation:

$$\mathbf{p}_j = \sigma(\mathbf{z}_j) \tag{3}$$

We can then define the predicted class label $\hat{c}_j$ and confidence score $\hat{q}_j$ as shown in Eqs. (4) and (5):

$$\hat{c}_j = \arg\max_{k \in \mathcal{C}} p_{j,k} \tag{4}$$

$$\hat{s}_j = \max_{k \in \mathcal{C}} p_{j,k} \tag{5}$$

For bounding-box regression, we use a distribution-based representation in which the four sides of each bounding box are predicted as independent discrete probability distributions using the following Eq. (6):

$$\mathbf{r}_j \in \mathbb{R}^{4R} \tag{6}$$

$\mathbf{r}_j$ denote the predicted bounding-box distributions at location j, where each side of the box is discretized into R bins. The predicted continuous box coordinates $\mathbf{b}_j$ are computed by applying the following projection operation as shown in Eq. (7):

$$\hat{\mathbf{b}}_j = \text{dist2bbox}\big(\text{Proj}(\mathbf{r}_j), \mathbf{a}_j\big) \tag{7}$$

where $\mathbf{a}_j$ denotes the anchor point corresponding to location j, and $\text{Proj}(\cdot)$ represents the operation that projects the discrete distribution to continuous offsets. The Intersection-over-Union (IoU) between predicted bounding box $\hat{B}$ and a ground-truth box B^* is defined as shown in Eq. (8):

$$IoU = \frac{\text{Area}(\hat{B} \cap B^*)}{\text{Area}(\hat{B} \cup B^*)} \tag{8}$$

The model learns its parameters by minimizing a joint loss consisting of bounding-box regression loss $\mathcal{L}_b$, classification loss $\mathcal{L}_c$, and distribution focal loss $\mathcal{L}_d$, shown in Eq. (9):

$$\mathcal{L} = \lambda_b \mathcal{L}_b + \lambda_c \mathcal{L}_c + \lambda_d \mathcal{L}_d \tag{9}$$

The localization loss ($\mathcal{L}_b$) is calculated as a Complete Intersection-over-Union (CIoU) loss as shown in Eq. (10):

$$\mathcal{L}_b = 1 - CIoU(\hat{B}, B^*) \tag{10}$$

Classification loss ($\mathcal{L}_c$) is the binary cross-entropy loss over class logits as shown in Eq. (11):

$$\mathcal{L}_c = \mathrm{BCE}(\mathbf{z}, \mathbf{y}) \tag{11}$$

Distribution focal loss ($\mathcal{L}_d$) predicts bounding-box localization as a discrete probability distribution as shown in Eq. (12):

$$\mathcal{L}_d = \mathrm{DFL}(\mathbf{r}, \mathbf{t}) \tag{12}$$

YOLOv12 is selected as the detection model for our proposed severity-aware detector. Unlike previous versions of YOLO, YOLOv12 uses attention-based feature extractors to enhance the model's contextual awareness. When dealing with object detection for fire monitoring applications, there are three main criteria for a satisfactory detector: (i) inference speed, (ii) the ability to extract discriminative features, and (iii) stable recognition performance in adverse conditions. Fire and smoke can often appear chaotic in shape, movement, and pixel contrast. YOLOv12's network architecture can improve severity-level discrimination through enhanced feature extraction and attention-based feature representation. Compared with raw pixel features, contextual information and texture features can be extracted as potential cues of the severity level.

YOLOv12 shares similar network architectures with most one-stage detectors. There are three main components in the YOLO series of detectors: backbone, neck, and head. The backbone extracts feature maps that embed spatial features such as color and texture, as well as the semantic features of the input image. The neck module further merges multi-scale feature maps generated by the backbone. The head on top performs classification and localization tasks based on features provided by the neck. For fire and smoke detection, this model achieves high performance across both classification and localization.

The training strategy is slightly modified to improve feature stability and benefit the subsequent severity-level classification task. Specifically, we initially train our model on three classes (i.e., Person, Fire, and Smoke), allowing our backbone to learn more stable features for each object category and to recognize spatial patterns between humans and dangers. After the basic detector is well-trained, we further add four categories to fire and smoke, making it a seven-class severity-aware detector. In this second stage of training, we freeze the initial layers of the backbone network to maintain low-level features. The intuition behind our choice of training procedure comes from the transferability of low-level features. In practice, many studies have shown that the first layers of CNNs encode generic low-level features, such as simple spatial patterns. On the contrary, higher layers learn dataset-dependent semantic concepts. Furthermore, freezing the initial layers prevents large modifications to the loss landscape, thus ensuring convergence of our training process [44]. We must emphasize, however, that the two-stage methodology presented here deviates from conventional fine-tuning. While the model is first trained on base data to obtain generalized feature representations, we further finetune it on domain-specific data with different class distributions and severity-aware labels to help it adapt and better learn feature representations for complex, fine-grained detection.

5 Implementation

Python was used to implement the training process for the detection model. The YOLOv12 and RT-DETR models were trained using Ultralytics, which uses PyTorch. The SSDLite320-MobileNetV3 model was trained using the PyTorch and Torchvision Libraries. Other libraries used include Python Ain't Markup Language (PyYAML) for loading the dataset configurations and tqdm for showing training progress. All models were trained on Google Colab's High-RAM environment with an NVIDIA A100 GPU and CUDA, using Python 3. This allowed large models to be trained quickly and provided consistency across all detection model benchmarks.

5.1 Dataset Construction and Experimental Design

We collected and annotated our own image dataset for this research to address the fire/smoke detection problem with coarse labels and severity-aware consideration. The images we collected came from public, open-source repositories and varied in illumination, object size, background, and environment. After preprocessing the data and prior to dataset splitting, duplicate and duplicate-like images were filtered out to reduce ambiguity and prevent future data leakage between the training and validation sets. It's worth noting that the data consists of annotated images, not video frames, and that detection is handled by YOLOv12. The severity labels (i.e., Small, Medium, and High for fire and Light, Medium, and Dense for smoke) were determined according to annotation guidelines that use visual factors (e.g., size of bounding box, intensity/brightness, and area of distribution) to ensure label consistency across images, though this approach retains a degree of heuristic judgment. All bounding boxes were manually annotated in the Roboflow software and then exported as YOLO annotations. The overall dataset contains around 6500 images. To ensure system robustness and avoid false alarms, 15% of the overall dataset has been intentionally curated as negative samples. Class distribution has been carefully prepared to ensure near balance in the overall instance count across all classes. The dataset has been divided using the 80–20 approach for training and validation. The same dataset has been used throughout the training of both stages of the proposed model. The dataset used to train and test the first stage of our proposed model was formatted for coarse three-class detection. The dataset used to train and test the second stage of our proposed model is the same as above, but formatted for seven-class detection.

5.1.1 Stage I: Three-Class Dataset Configuration

The first experiment setup provides a coarse-grained detection setting with three major classes, which include Person, Fire, and Smoke[1]. The number of instances per class is set to 3000 annotated samples to ensure sufficient samples for major detection intents. This detection setup defines the first stage of object-level detection. It learns generic representations of fire and smoke detection prior to incorporating severity awareness in stage two. Fig. 2 shows an example of representative samples with manually annotated bounding boxes for three classes.

5.1.2 Stage II: Seven-Class Severity-Aware Dataset Configuration

The setting of the second experiment stage is fine-grained severity-aware detection. For this stage's dataset configuration, we classify fire and smoke detection into several levels based on the presence of the Person class. Fire detection is divided into three sub-classes named SmallFire, MediumFire, and HighFire; Smoke detection is divided into three sub-classes named LightSmoke, MediumSmoke, and DenseSmoke.

[1]The Fire–Smoke–Person Detection Dataset (3-Class) used in this study is publicly available at (https://kaggle.com/datasets/80cd1c92a408c9e03332b182b0a7e4faf167f22cf50363bd77587380a5f1ced6)

Each class comprises 1000 annotated instances to ensure near-balanced distribution across severity levels[2]. The severity-aware categorization was initially estimated using automated measurements of visual characteristics within each annotated bounding box. In particular, the relative size of the detected region, along with intensity and brightness cues, was used to estimate the severity level. Based on these measurements, fire and smoke instances were automatically divided into three groups corresponding to low, medium, and high intensity levels. The resulting labels were manually reviewed and verified to ensure consistency and accuracy across the dataset. Fig. 3 presents the same representative samples from the dataset with annotated bounding boxes for different severity levels.

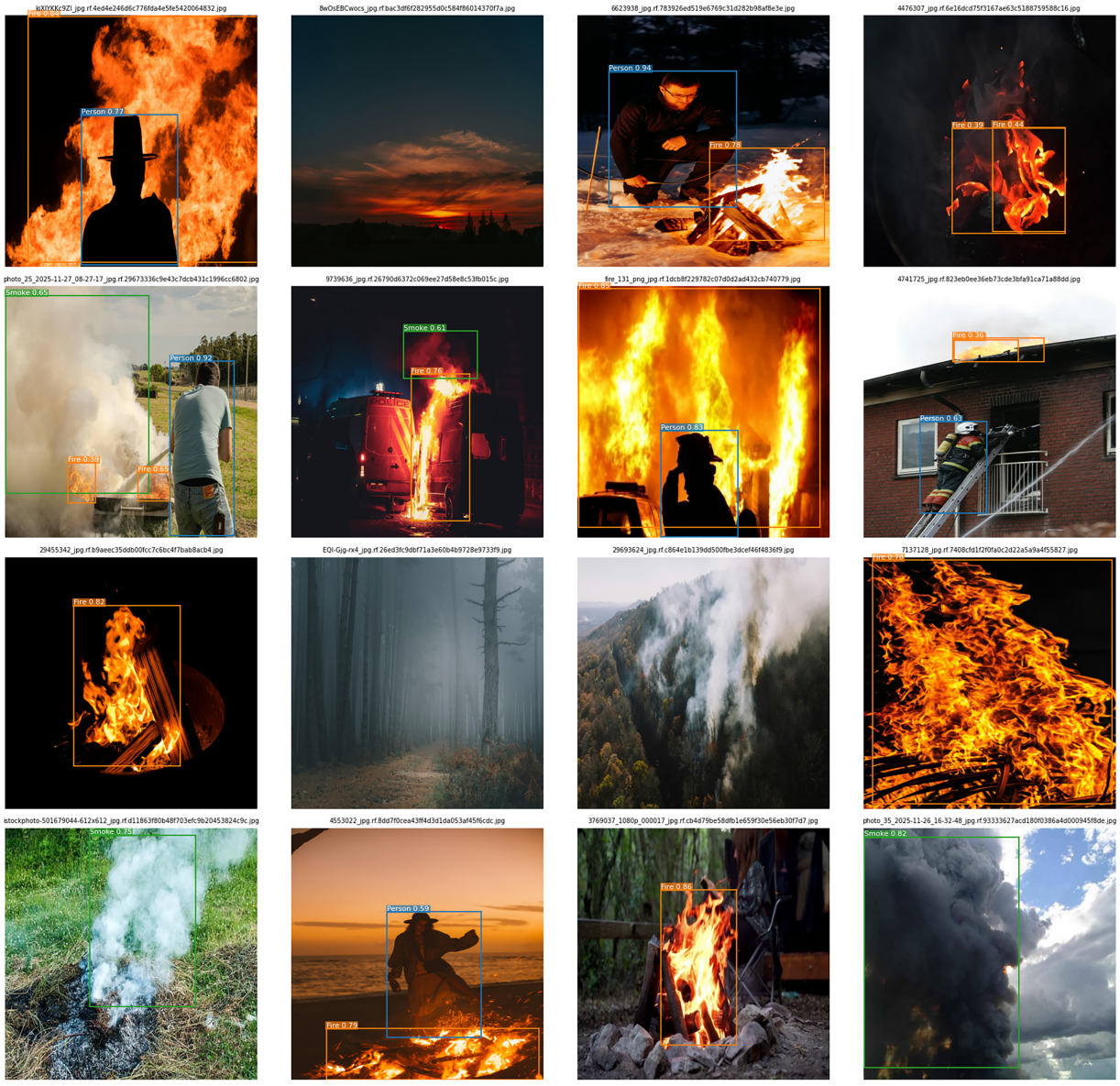

Figure 2: Representative annotated samples from the three-class dataset.

[2]The Fire–Smoke–Person Severity Dataset (7-Class) used in this study is publicly available at (https://kaggle.com/datasets/753f066efcddbcad72d47ee42178638f39e9df259d8501f5f6615f3c91ebafa2).

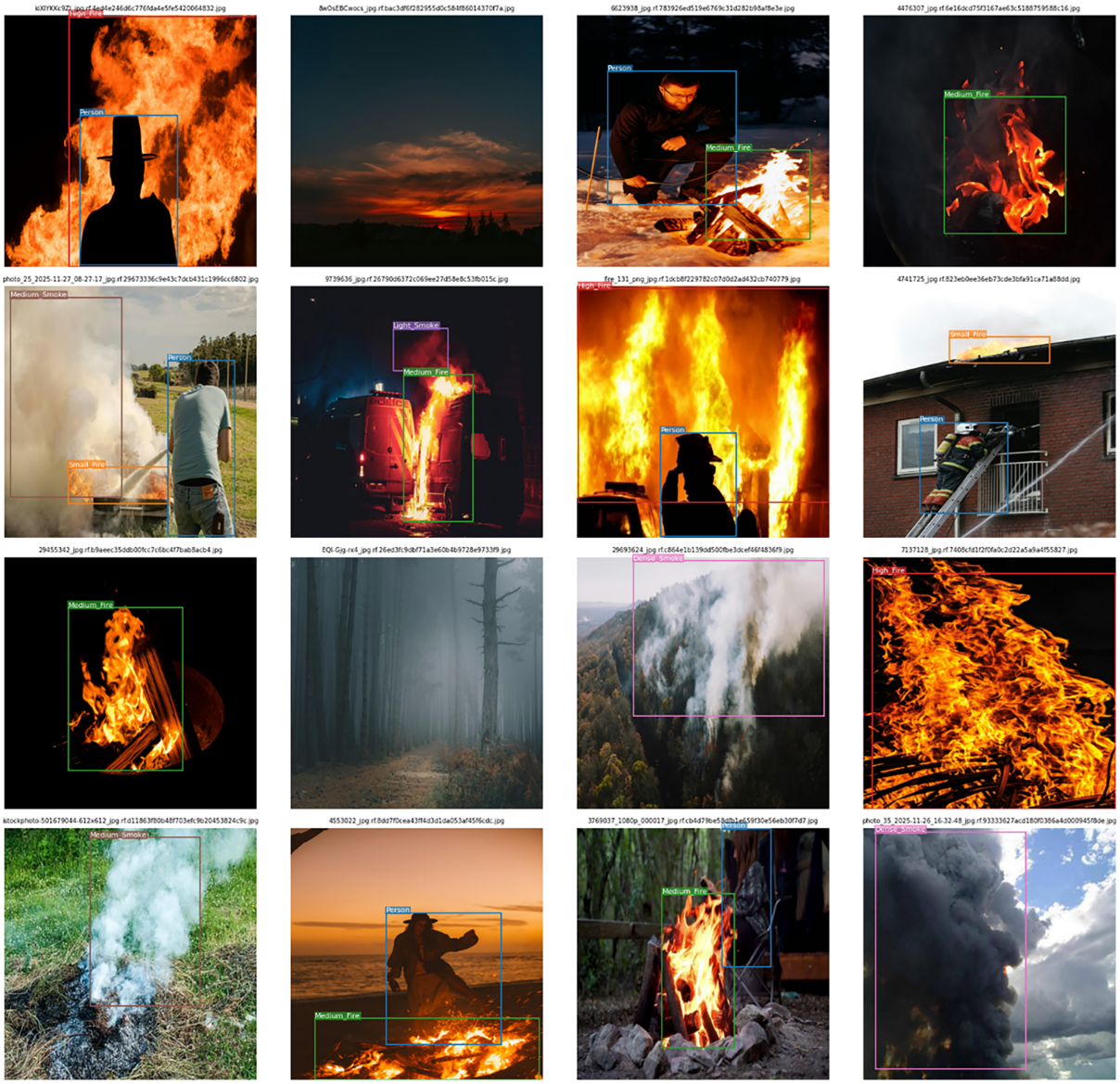

Figure 3: Representative annotated samples from the seven-class dataset.

5.2 Severity-Aware Labeling Protocol

Inspired by the consistency-aware annotation and pseudo-labeling approaches in the object detection literature, we introduce a novel severity-aware labeling protocol. This severity-aware annotation protocol aims to reduce subjective labeling bias and improve annotation consistency, following conventionally accepted practices in quality assessment of object detection annotations and in consistency-aware labeling methodologies [45,46]. The severity-aware labeling process consists of four main stages, including (i) feature extraction and preprocessing, (ii) feature normalization, (iii) severity score computation, and (iv) threshold-based severity assignment.

5.2.1 Feature Extraction and Preprocessing

All features were extracted per object, where the object could be either fire or smoke, using the annotated bounding box. Given the YOLO-format bounding box annotation, coordinates were converted to pixel coordinates, and the object bounding box was cropped from the original image. Features were computed from the cropped object region to extract visual indicators of severity. For fire regions, we extract features, including (i) the normalized bounding-box size (i.e., Area denoted as α), computed as $\alpha = w \times h$, where w and h represent the width and height of the bounding box, (ii) the mean value of the V channel in the Hue Saturation Value (HSV) color space (i.e., brightness denoted as β), and (iii) the mean value of the S channel in the HSV color space (i.e., Saturation denoted as S). For smoke regions, we extract features, including (i) the normalized bounding-box size (i.e., Area denoted as α), (ii) the standard deviation of grayscale intensity values (i.e., Contrast denoted as χ), and (iii) the inverse saturation (i.e., Grayness denoted as γ), computed as $\gamma = 255 - \text{mean}(S)$ where higher values indicate denser smoke regions.

5.2.2 Feature Normalization

In order to normalize the difference between scales of extracted features, min–max normalization was applied, scaling all feature values to fall within the range of $[0, 1]$ using Eq. (13):

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{13}$$

where x is the original value of the feature. $x_{\min}$ and $x_{\max}$ are the minimum and maximum values of the feature across the entire dataset. In which the normalization of fire-related features was performed separately from that of smoke-related features.

5.2.3 Severity Score Computation

A combined severity score was calculated for each object through a weighted average of the normalized features. The fire severity score is computed based on the normalized area denoted α', the normalized brightness denoted β', and the normalized saturation denoted S' using the Eq. (14):

$$S_{\text{fire}} = 0.5\alpha' + 0.3\beta' + 0.2S' \tag{14}$$

Similarly, the smoke severity score is computed based on the normalized area denoted α', the normalized grayscale contrast denoted χ', and the normalized grayness measure denoted γ' using the Eq. (15):

$$S_{\text{smoke}} = 0.2A' + 0.5(1 - C') + 0.3G' \tag{15}$$

Weighting factors were chosen to reflect the relative importance of an object's spatial extent vs. its visual properties in determining the severity of fire and smoke. It is important to note that the weighting coefficients used to compute the fire/smoke severity score were empirically selected for annotation purposes. These weights were not optimized or tuned based on a physical model. For example, we assigned a larger weight to the normalized flame area when computing fire severity, since the extent of the area covered is often the easiest visual cue for determining severity (i.e., brightness and saturation being next). Contrast-opacity-like terms were weighted most heavily for smoke since it has the largest visual effect on perceived smoke severity (i.e., grayness and extent next). These weights were determined through visual inspection, domain knowledge, and iterative refinement of annotations during dataset annotation to ensure consistent severity labeling.

5.2.4 Threshold-Based Severity Classification

Severity levels were determined using quantile thresholds based on the distribution of estimated severity scores. Two thresholds were used, including q_1 = 33% (i.e., 33rd percentile) and q_2 = 66% (i.e., 66th percentile). The threshold values used in this work are shown in Table 2.

Table 2: Severity threshold values.

Type	q_1	q_2
Fire	0.3250	0.4311
Smoke	0.5394	0.6342

5.2.5 Severity Assignment Rules

Based on the computed severity scores and threshold values, the final severity labels were assigned for fire and smoke based on the severity assignment rules detailed in Table 3.

Table 3: Severity assignment rules.

Severity Class	Assignment Rules
Small Fire	$S_{\text{fire}} < q_1$
Medium Fire	$q_1 \leq S_{\text{fire}} < q_2$
High Fire	$S_{\text{fire}} \geq q_2$
Light Smoke	$S_{\text{smoke}} < q_1$
Medium Smoke	$q_1 \leq S_{\text{smoke}} < q_2$
Dense Smoke	$S_{\text{smoke}} \geq q_2$

5.2.6 Annotation Consistency and Bias Mitigation

To alleviate annotation bias and variability, we first obtained severity labels automatically using the quantitative severity-aware labeling protocol described above, rather than subjective manual estimation. We then manually verified the labels against unified annotation guidelines. Unclear samples were further manually adjusted to ensure annotation reliability and reduce inter-rater inconsistency.

5.3 Training Configuration and Hyperparameters

All models were trained using a standardized experimental protocol comprising two consecutive stages. For each stage, the models were trained for 100 epochs with a batch size of 32, using a cosine-based learning rate scheduler. The training sets were split 80–20, and the model was trained on a centralized GPU environment. The training settings were deterministic, and the normalization approach was the same for all models in both stages of experimentation. For all evaluated models, moderate geometric and color-based augmentations were applied to improve generalization and maintain experimental consistency across architectures. The augmentation pipeline included HSV jittering, scaling, translation, flipping, resizing, and normalization. In addition, the input image resolution was standardized to 640 × 640 to ensure fair architectural comparison under identical spatial input conditions. The image was resized to the specified resolution, and the pixel values were normalized to 0–1.

Stage I provided a stable baseline for a three-class detection model (Person, Fire, Smoke). The goal was to ensure convergence before introducing fine-grained severity discrimination in Stage II. Stage II represents the main severity-aware model configuration. In this stage, the model was trained to discriminate between seven classes. While the overall training protocol was similar to that in Stage I, the hyperparameters were fine-tuned to support fine-grained multi-class discrimination. Table 4 summarizes the hyperparameters adopted in Stage II.

Table 4: Training hyperparameters used for the evaluated detection models.

Hyperparameter	YOLOv12s	RT-DETR-L	SSDLite-MobileNet
Input resolution	640 × 640	640 × 640	640 × 640
Epochs	100	100	100
Batch size	32	32	32
Optimizer	AdamW	AdamW	SGD (momentum = 0.9)
Initial learning rate	0.0013 (lrf = 0.02)	0.0008 (lrf = 0.01)	0.0010 (backbone), 0.0025 (head)
LR scheduling	Cosine decay	Cosine decay	Cosine annealing (Tmax = 100)
Loss Function	Composite (Box + Cls + DFL)	Hungarian-based (L1 + GIoU + CE)	MultiBox (Smooth L1 + CE)
Backbone freezing	First 8 blocks	First 8 blocks	First 8 blocks
Normalization	[0, 1] scaling	[0, 1] scaling	[0, 1] scaling

As indicated in Table 4, the YOLOv12s and RT-DETR-L models were optimized using the AdamW optimizer with cosine schedule decay. In contrast, the SSDLite320-MobileNetV3 model was optimized using the SGD optimizer with differential learning rates for the backbone and detection head networks. In Stage II, the backbone network was frozen to support fine-grained multi-class learning. This was intended to avoid overfitting in the early layers of the network for the severity-aware model configuration. While the overall training parameters were similar to those adopted in Stage I, the optimization parameters were fine-tuned to support the severity-aware model configuration. Furthermore, all models were evaluated on the same device and with the same inference settings. We ran experiments on an NVIDIA A100 GPU with a batch size of 32 and an input resolution of 640 × 640. Latency measurements were taken without gradient calculation in inference-only mode. Reported numbers are mean per-image inference times under the same evaluation settings.

6 Evaluation and Experimental Results

In this section, we comprehensively evaluate our proposed detection model. In each experiment, YOLOv12s, SSDLite320-MobileNetV3, and RT-DETR-L are trained and evaluated with the same training/validation split, for fair comparison. To evaluate our models quantitatively, we use some commonly used object detection metrics, including precision, recall, F1-score, and mean Average Precision (mAP) at standard IoU thresholds. We also report per-class performance and confusion matrices to better understand our model stability and error cases. It's worth noting that the results presented were obtained using a test set drawn from the same distribution as the training data. This work did not use external, unseen datasets. In addition, the proposed two-stage training strategy leads the models to stable convergence and strong severity-aware discrimination performance. However, we did not perform ablation studies on stage-wise

training, backbone freezing, or the addition of negative samples separately. As such, the individual effect of each component on the final detector performance cannot be directly reported.

6.1 Stage I: Three-Class Detection (Baseline Evaluation)

Table 5 presents the baseline performance of the three models examined for the three-class setting (i.e., Person, Fire, Smoke). YOLOv12s had the highest overall detection performance among the models. It had the highest mAP@0.5 precision, recall, and F1-score among all the models, with scores of 0.823, 0.823, 0.791, 0.759, and 0.770, respectively. This could be attributed to the fact that it is a convolutional model that was optimized for object detection. It also had a robust anchor-box detection mechanism. RT-DETR-L had slightly lower mAP@0.5 precision, recall, and F1-score than YOLOv12, with scores of 0.774, 0.785, 0.736, and 0.760, respectively. This could be attributed to the query-based model occasionally failing to detect small objects due to its limited number of queries. SSDLite320-MobileNetV3 had stable but lower overall detection performance than the other models. This could be attributed to its lower representational capability, due to a lightweight backbone network optimized for computational efficiency. Despite these differences, all models learned the primary detection task and converged consistently. Establishing a reliable baseline before progressing to the more challenging severity-aware configuration in Stage II.

Table 5: Performance comparison of Stage I three-class detection evaluated models.

Model	mAP@50	Precision	Recall	F1-Score
YOLOv12	0.823	0.791	0.759	0.770
RT-DETR	0.774	0.785	0.736	0.760
SSD-MobileNet	0.758	0.716	0.744	0.730

6.2 Stage II: Severity-Aware Detection (Core Evaluation)

Stage II evaluates the proposed models using the fine-grained, seven-class, severity-aware configuration. Unlike Stage I, this setup requires the models to discriminate between multiple fire and smoke intensity levels while maintaining stable Person detection performance. Stage II is the main experiment conducted in this paper. Evaluation includes detailed analysis of model stability in fine-grained settings, misclassification behavior, and the ability to distinguish severity levels.

6.2.1 YOLOv12s Severity-Aware evaluation

Table 6 shows the overall quantitative results for YOLOv12s in a seven-class severity-aware setting. From the table, we can see that we have obtained an mAP@50 of 0.929 along with 0.884 precision, 0.876 recall, and an F1-score of 0.88. It is important to note that we introduced about 15% negative samples, aiming to better train the model on fire/smoke features and the distinguishing patterns that set them apart from often similar false alarm instances. We did not run a specific ablation experiment to quantify the precise impact of including these negatives. However, our proposed framework demonstrates high precision, indicating robustness to false positives. Additionally, the average latency for each image was measured at 2.48 ms, indicating good computational efficiency during centralized evaluation.

Table 6: YOLOv12 performance metrics.

Performance Metric	Score
mAP@50	0.929
Precision	0.884
Recall	0.876
F1-score	0.880
Latency (per image)	2.48 ms

As seen from the class-wise YOLOv12 evaluation in Fig. 4, it maintains good, stable performance across most severity levels, even for Person. With a precision of 0.932 and a recall of 0.867, we achieved an F1-score of 0.898 for the Person class. This indicates that our introduction of fine-grained fire and smoke severity levels has not affected person detection performance. For fire categories, High-Fire achieved the best performance with an F1-score of 0.973, backed by decent precision and recall of 0.984 and 0.963, respectively. This indicates good consistency in detecting fire in our dataset. The medium fire category also performed well, with an F1-score of 0.926. Small Fire, on the other hand, shows low precision, scoring 0.81, but high recall, scoring 0.867, indicating possible over-detection in low-prevalence fire cases. Regarding smoke severity levels, we noticed a larger drop in performance. Dense Smoke again demonstrated high detection performance, achieving an F1-score of 0.916. Medium Smoke showed balanced precision and recall, resulting in an F1-score of 0.854. However, due to the difficulty of light smoke objectively standing out from background clutter, Light Smoke proved to be the hardest case for YOLOv12, achieving a low recall of 0.711 and an F1-score of 0.750. Overall, YOLOv12 shows promisingly stable performance when discriminating across multiple classes of smoke severity, without hurting performance on Person detection.

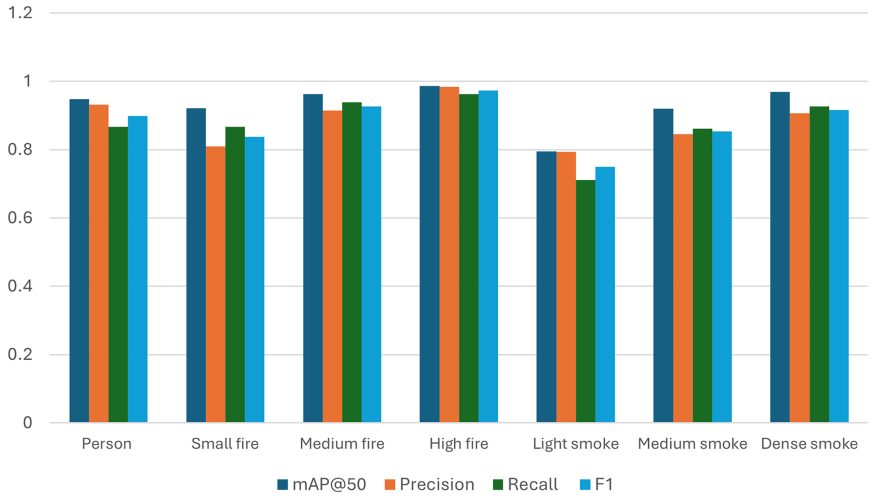

Figure 4: Class-wise performance metrics on the validation set for the YOLOv12s model.

Fig. 5 shows the confusion matrix for the YOLOv12 model. From the figure, we can see strong diagonal dominance across all severity levels, indicating high true-positive rates. Classification was most stable for High-Fire, followed closely by Medium Fire and Dense Smoke, with little to no misclassification across other categories. Most misclassifications occurred between adjacent severity levels (i.e., Small Fire, Medium Fire, Light Smoke, and Medium Smoke). This makes sense as neighboring levels of intensity

would have similar visual features. Light Smoke also had considerably lower stability, with more predictions being confused towards higher smoke density categories and also towards the background. This may be due to low-opacity smoke patterns being harder to distinguish from background scenes, especially when illumination varies. Background false positives were also very low, further demonstrating that introducing negative samples made our model more robust against false alarms. The Person class also shows strong diagonal dominance and limited confusion between fire and smoke categories, indicating that it can reliably differentiate between persons and fires/smoke. In terms of potential impacts on risk-assessment-oriented operations, the severity-level confusion errors introduced by the proposed model affect fire monitoring missions differently depending on severity. Errors due to confusion over severity levels between close ranks (e.g., High Fire vs. Medium Fire) may result in residual situation awareness, enabling on-the-ground emergency personnel to take precautionary measures. However, severe errors that would ultimately lead to the prediction of dangerous fire patterns as background activity could stall emergency mobilization efforts. The fewer number of background false negatives, along with reduced non-adjacent confusion overall in our proposed formulation, demonstrates increased resilience of our approach towards risk-aware fire monitoring operations.

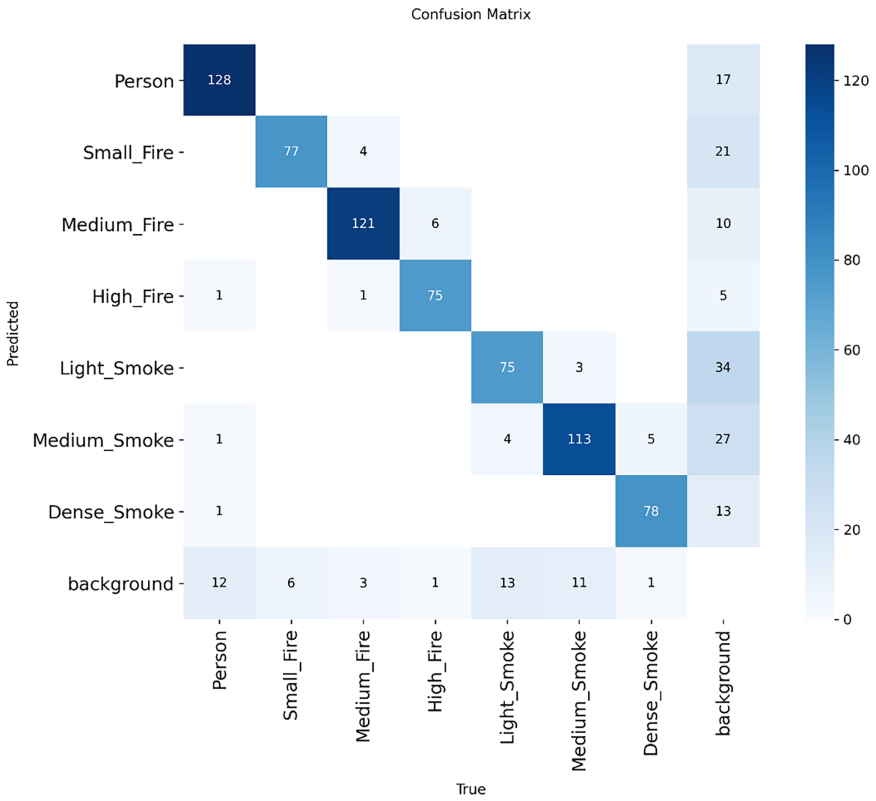

Figure 5: Confusion matrix for YOLOv12 model.

Fig. 6 consists of Precision-Recall curves for each class for the YOLOv12 model. As we can see, High Fire and Dense Smoke have near-perfect Precision-Recall curve values. Area under the Precision-Recall curve for Light Smoke is considerably lower because detecting light smoke is difficult. But since our overall mAP@50 was high at 0.929, we can safely say that our model can make fine distinctions in fire severity while maintaining strong localization.

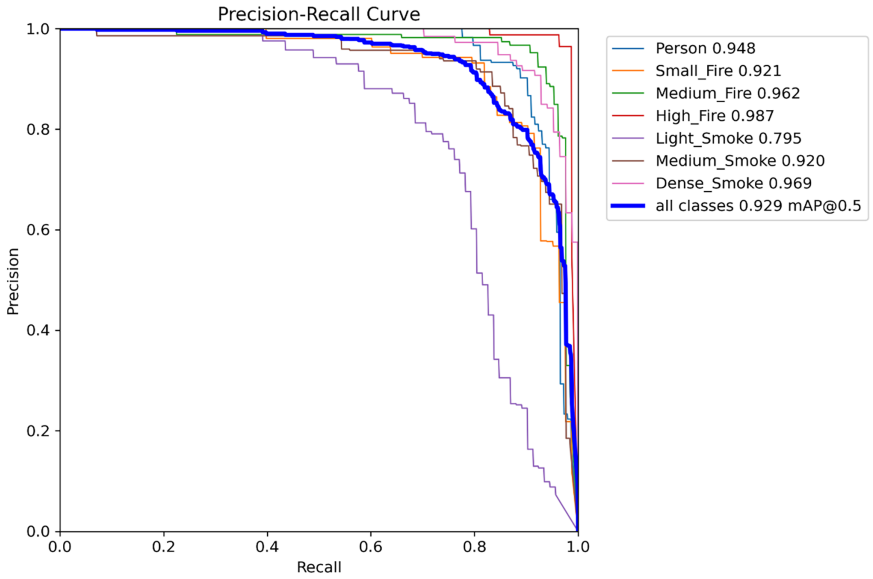

Figure 6: YOLOv12 Precision–Recall curves.

Figs. 7–9 show the performance of YOLOv12s across various confidence thresholds for precision, recall, and F1-score. As seen in the precision vs. confidence plot, precision increases as the confidence threshold rises and approaches 1 at higher thresholds. This is expected, as we are filtering out false-positive predictions by increasing our confidence threshold. On the other hand, recall changes little for low to medium confidence thresholds, but it degrades significantly as we set higher thresholds. This is because setting a high confidence threshold will cause us to filter out correct positive predictions. As shown in the F1–Confidence curve, our best model performance occurs around a threshold of 0.417, achieving an F1-score of 0.88 across all classes.

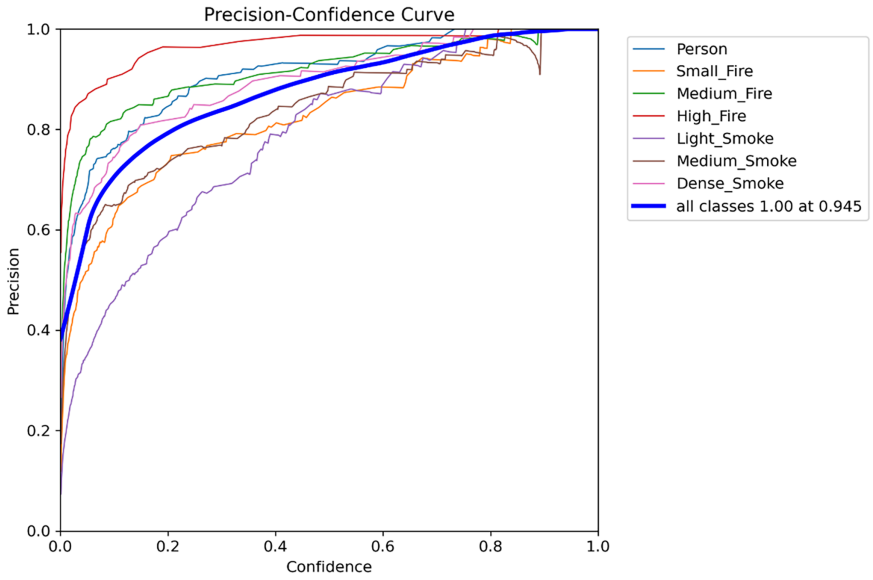

Figure 7: Precision for YOLOv12 model.

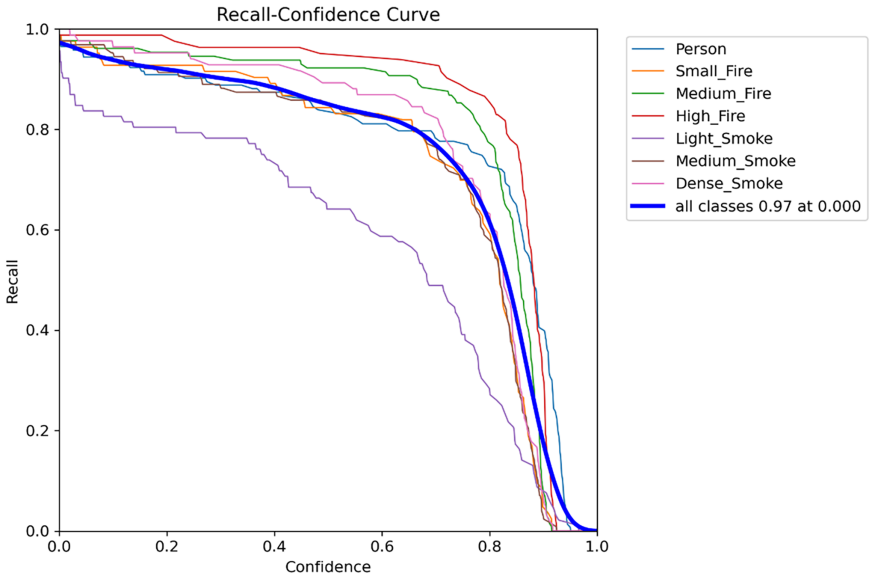

Figure 8: Recall for YOLOv12 model.

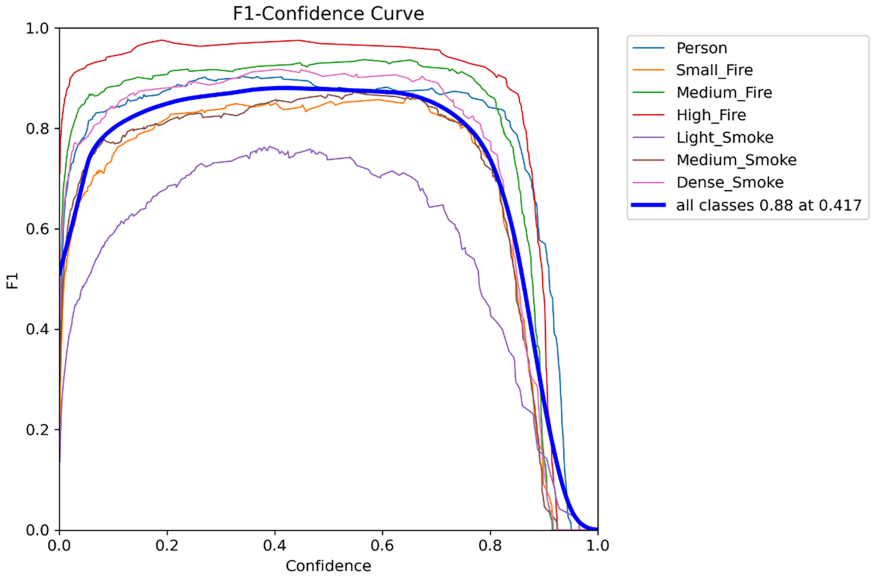

Figure 9: F1-score for YOLOv12 model.

As shown in Fig. 10, the curves represent the training and validation processes. The curves converge stably throughout the entire 100 epochs. The box loss, classification loss, and DFL loss curves decline steadily, with no signs of divergence between the training and validation curves. Meanwhile, the precision, recall, and mAP metrics are increasing steadily in the early stages and then remain at high levels in the latter stages.

Given the strong balance between detection accuracy and computational efficiency demonstrated by YOLOv12s, a detailed analysis was conducted to examine its severity-level discrimination capability.

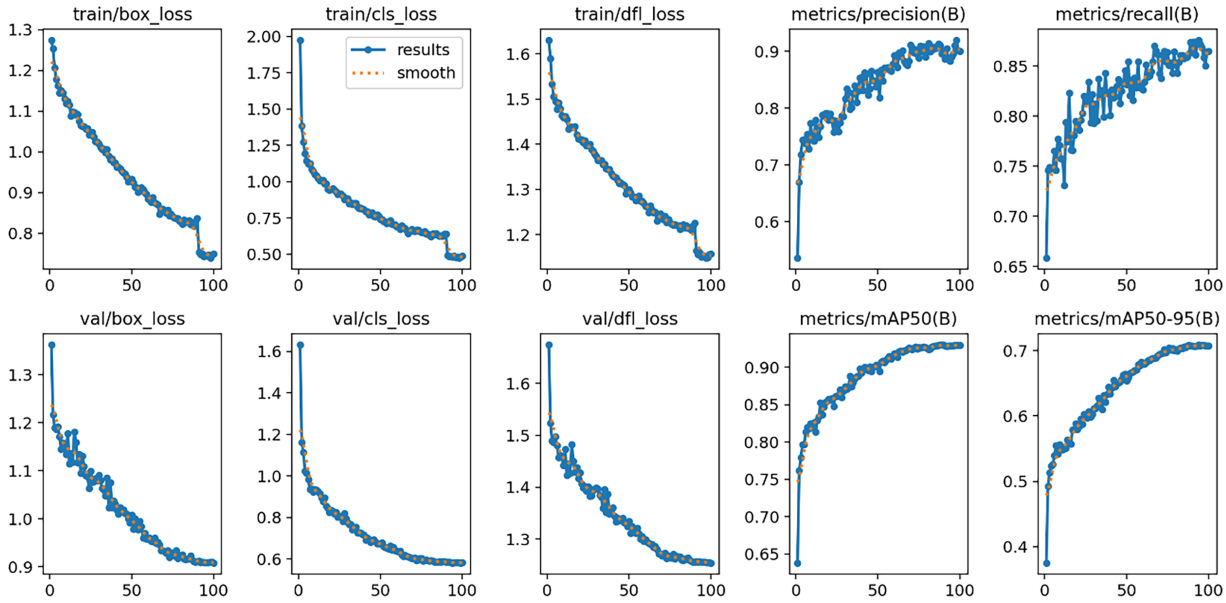

Figure 10: YOLOv12 training and validation performance across 100 epochs.

6.2.2 RT-DETR Severity-Aware evaluation

As shown in Table 7, RT-DETR-L achieved the highest overall detection accuracy under the severity-aware configuration, reaching an mAP@50 of 0.931, with a precision of 0.925, a recall of 0.915, and an F1-score of 0.92. While the model demonstrates superior accuracy compared to YOLOv12s, its latency averages 5.21 ms per image, reflecting increased computational complexity.

Table 7: RT-DETR detection performance metrics.

Performance Metric	Score
mAP@50	0.931
Precision	0.925
Recall	0.915
F1-score	0.920
Latency (per image)	5.21 ms

As shown in Fig. 11, the results for each category are stable as well. Fire-related categories perform very well again, especially Small Fire scoring in F1-score 0.957 and Medium Fire scoring in F1-score 0.954, which means RT-DETR can effectively distinguish between the severities of fire. Person is still performing well, with an F1-score of 0.923 and only slightly lower precision, 0.906, than recall, 0.94. Increasing the label granularity from Stage I to Stage II did not appear to affect the model's performance on Person detections, which remain well above baseline. Smoke-related categories, however, perform worse than other categories, and again, Medium Smoke scoring in the F1-score of 0.862 shows the largest discrepancy between precision and recall. Overall, results across classes are stable, indicating that RT-DETR can perform fine-grained severity discrimination.

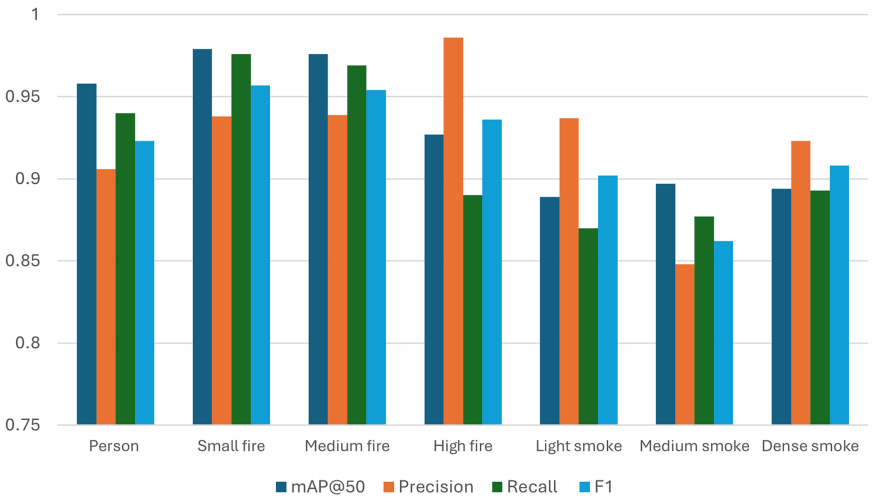

Figure 11: Class-wise performance metrics on the validation set for the RT-DETR model.

Fig. 12 shows the confusion matrix for the RT-DETR model. There are many true positives for Medium Fire 126 and Small Fire 82 as well. Compared to YOLOv12s, confusion between different severities is lower. False positives towards the background are still quite high for RT-DETR, especially for Person predictions 74. Overall, RT-DETR predicts more person as interacting with the background. Nonetheless, the diagonal for person is still quite high, indicating that most predicted humans were actually humans and not misclassified as fire or smoke.

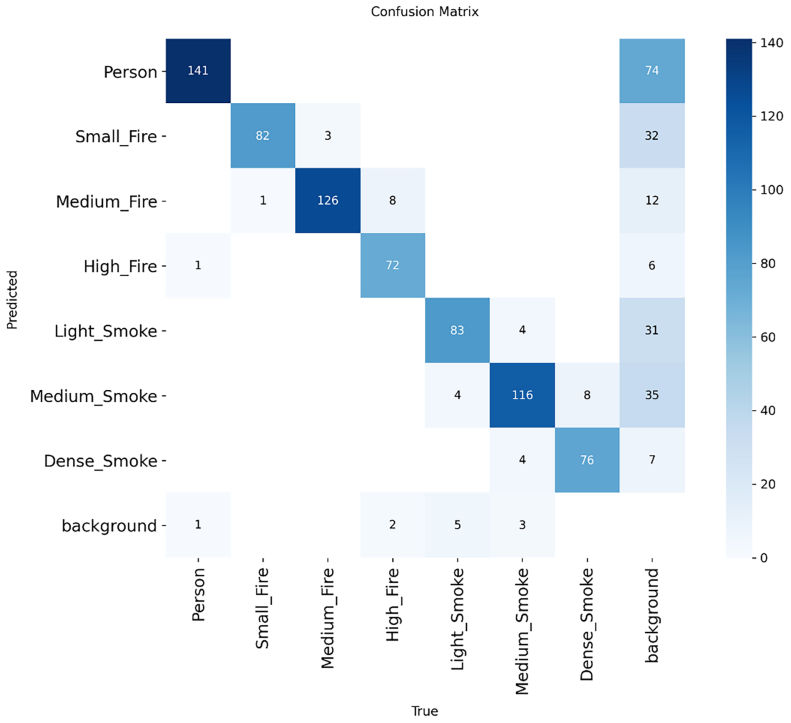

Figure 12: Confusion matrix for RT-DETR model.

RT-DETR achieves better overall detection results when running in severity-aware mode. Again, it struggles less to distinguish among different fire severities than it does among other categories. Even though RT-DETR requires more computational power, it outperforms YOLOv12s in centralized class-wise evaluation.

6.2.3 SSDLite-MobileNet Severity-Aware evaluation

As shown in Table 8, the SSDLite320-MobileNetV3 model had an overall mAP@50 of 0.812 with precision at 0.749, recall at 0.778, and an F1-score of 0.763. The SSDLite320-MobileNetV3 achieved competitive inference efficiency, with an average time of 3.90 ms per image, placing it between YOLOv12s and RT-DETR-L in latency. Though the SSDLite320-MobileNetV3 has lower detection accuracy than RT-DETR and YOLOv12s, it maintains a reasonable precision-recall trade-off and good computational efficiency for its architecture.

Table 8: SSDLite-MobileNet detection performance metrics.

Performance Metric	Score
mAP@50	0.82
Precision	0.75
Recall	0.78
F1-score	0.76
Latency (per image)	3.901 ms

As shown in Fig. 13, the class-wise evaluation chart shows that the F1-score for Person detection was 0.714, with a precision of 0.762 and a recall of 0.671. The best F1-score of 0.926 was recorded for the Medium Fire class. The F1-score of 0.832 was recorded for the Medium Smoke class, and the Dense Smoke class recorded an F1-score of 0.732. The worst F1-score of 0.487 was recorded for the Small Fire class, with a low recall of 0.349. The High-Fire class recorded a very high recall of 0.951 but a lower precision of 0.582. The recall of 0.522 was recorded for the Light Smoke class. The SSD-MobileNet model achieved higher F1-scores for the medium severity classes but lower F1-scores for the smaller or lighter classes. The low F1-score for the Small Fire and Light Smoke classes indicates that the model has lower sensitivity for small-scale fire and low-visibility smoke.

The confusion matrix shown in Fig. 14 further explains the model's predictive tendencies. Person classification exhibits moderate diagonal dominance, with several background classifications, which aligns with the decrease in recall observed during class-wise evaluation. Large true-positive values were observed for Medium Fire 125 and Medium Smoke 114 predictions, demonstrating the model's consistent detection performance on mid-level severity samples. Misclassifications were observed more often at lower levels of fire and smoke. Small Fire samples were frequently misclassified as background, as evidenced by the noticeable reduction in recall. Additional overlap was also observed between lighter and denser smoke levels, particularly between Light Smoke and Medium Smoke samples. This is expected due to the relatively subtle visual differences between smoke density categories. Additional background classifications were also observed in several fire and smoke samples, indicating that the model missed some detections.

Overall, SSD-MobileNet achieves strong latency performance and provides reliable detection accuracy. However, it struggled to detect both Small Fire and Light Smoke samples, which would allow it to be used in finer-grained, severity-aware applications.

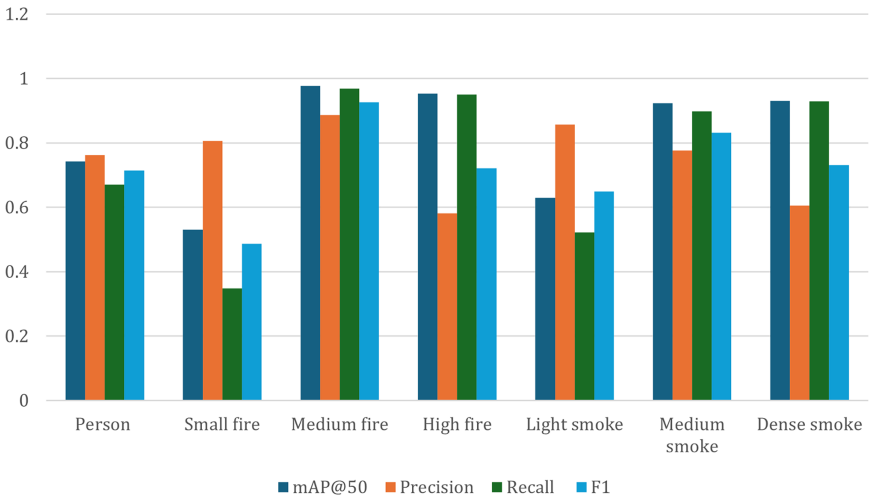

Figure 13: Class-wise performance metrics on the validation set for the SSDLite-MobileNet model.

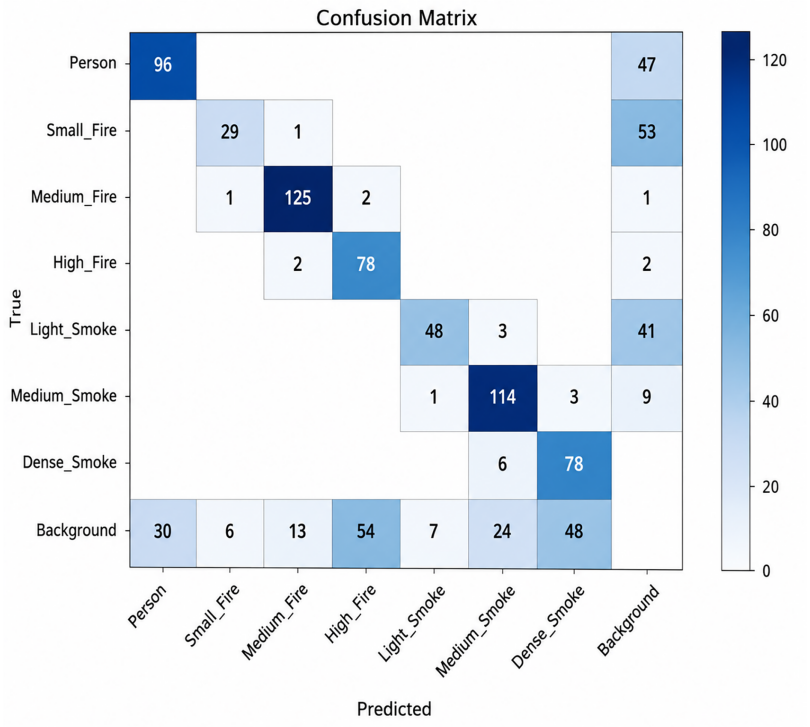

Figure 14: Confusion matrix for SSDLite-MobileNet model.

6.2.4 Cross-Model Comparative Performance Analysis

Table 9 offers a summarized comparison of all architectures. RT-DETR-L achieved the highest overall mAP@50 of 0.931, closely followed by YOLOv12s at 0.929, while SSDLite-MobileNet had a stable but lower mAP@50 of 0.812. For computational efficiency, YOLOv12s was the fastest, with a detection speed of 2.48 ms, whereas RT-DETR-L had the highest processing time at 5.21 ms. For individual classes, transformer-based detection performed better, especially for Small Fire and Medium Fire, with RT-DETR-L achieving the highest F1-score. YOLOv12s was found to be exceptionally stable at detecting High-Fire, with balanced

performance across all classes. However, SSD-MobileNet was found to be unstable at detecting subtle fires, especially Small Fire, with a low recall.

Table 9: Comparative validation performance and average latency per image across all evaluated models.

Model	mAP@50	Precision	Recall	F1-Score	Latency (per image)
YOLOv12	0.929	0.884	0.876	0.880	2.48 ms
RT-DETR	0.931	0.925	0.915	0.920	5.21 ms
SSD-MobileNet	0.822	0.75	0.78	0.76	3.901 ms

For the smoke-related classes, YOLOv12s exhibited consistent performance, whereas RT-DETR showed slight degradation in smoke-class accuracy relative to fire detection. The performance of SSD-MobileNet was moderate for the Medium Smoke and Dense Smoke classes, but the model performed poorly on the Light Smoke class. Considering the models' performance on the Person class, the RT-DETR model achieved the highest accuracy, followed closely by YOLOv12s, whereas the SSD-MobileNet performed comparatively lower, showing reduced sensitivity to multi-class severity. Overall, the performance of the three models clearly indicates that the trade-off for the three models is: for the highest accuracy, the RT-DETR model can be selected; for the best trade-off between accuracy and latency, the YOLOv12s model can be selected; and for the computationally efficient baseline model, the SSD-MobileNet can be selected. However, it is important to note that although RT-DETR had the highest overall mAP@50, the overall performance gap between RT-DETR scoring 0.931 and YOLOv12s scoring 0.929 is only 0.002. Nevertheless, given YOLOv12's much lower inference latency, achieving 2.48 ms/image and its multi-class severity discrimination, it was chosen as the main architecture for the proposed severity-aware detection model. This is because, in real-world applications of fire monitoring systems, one of the key requirements is response time, as even a small delay in detection can impact early hazard recognition. The much lower latency of YOLOv12s makes it suitable for real-time fire detection applications, as detection time is considered a key contributing factor in reducing potential damage. On one hand, the reported inference latency of 2.48 ms/image was achieved with an NVIDIA A100 GPU and a centralized experiment setting. If we consider a more realistic inference deployment setting with embedded edge devices such as Raspberry Pi or NVIDIA Jetson boards for IoT-based continuous surveillance, inference latency would likely increase due to limited processing power, memory bandwidth, communication, and networking delays. On the other hand, YOLOv12s's relatively low latency and efficiency indicate promise for model optimizations targeting edge AI hardware.

To understand the predictive performance of our proposed severity-aware multi-class detection model, we conducted an error analysis using confusion matrices. As can be observed, errors were separated into two categories: (i) adjacent severity misclassification error that the prediction only misses with exactly one severity level under the same fire/smoke category, and (ii) multi-level severe misclassification error that missed predictions include severity level gaps greater than 1 or incorrect category/background prediction. From the data presented in Table 10, we observe that RT-DETR shows the highest proportion of adjacent-level mistakes, 84.2%, suggesting that most of its errors stem from confusion between visually adjacent severity levels. YOLOv12s also showed decent severity consistency, whereas SSDLite320-MobileNetV3 suffered severely from misclassification, suggesting poor fine-grained robustness in discriminating between fire/smoke severity. While RT-DETR had the best average detection accuracy measured by mAP@50 and F1-score, YOLOv12s performed significantly faster and had lower latency, making it better suited for real-time fire monitoring. Hence, we view choosing YOLOv12s as an engineering decision that trades off accuracy for computational efficiency, rather than claiming it has the best overall empirical performance.

Table 10: Severity-level misclassification analysis.

Model	Adjacent Severity Misclassification (%)	Multi-Level Severe Misclassification (%)
YOLOv12s	39.7	60.3
RT-DETR	84.2	15.8
SSD-MobileNetV3	15.2	84.8

While our model achieves competitive severity-aware detection results, we note several practical considerations before real-world deployment. These include resource constraints of edge/embedded devices, latency constraints for emergency-response use cases, changing environmental factors (e.g., illumination, smoke opacity, occlusions, backgrounds with similar appearance), and requirements for large-scale deployment (e.g., network bandwidth, cloud-edge coordination, alert cascade failures). The proposed model (i.e., YOLOv12) demonstrated high efficacy when tested on the newly curated severity-aware dataset. Nonetheless, future research should focus on assessing its performance on external datasets to enable cross-dataset comparisons, thereby analyzing robustness, generalization, and the broader applicability of severity-aware learning in fire monitoring contexts.

6.3 Ablation Study for the Two-Stage Training Strategy

To verify the effectiveness of the proposed two-stage training strategy, we also conducted an experiment with YOLOv12 without it for a fair comparison. Specifically, we removed the two-stage training strategy by training directly on the seven-class severity-aware dataset, without first training the detector on coarse-grained fire, smoke, and person classes. The confusion matrix illustrates the results of using YOLOv12 without employing the two-stage training strategy, as shown in Fig. 15.

From the confusion matrix in Fig. 15, we observe that the classification performance suffers considerably when bypassing the proposed two-stage training strategy. There is significant confusion among adjacent fire severities (i.e., Small Fire, Medium Fire, and High Fire). For instance, there are 41 samples in the Small Fire category that were incorrectly predicted as the Medium Fire category, and 44 samples in the Medium Fire category that were incorrectly predicted as the High Fire category. A similar pattern can be found in smoke severity estimation, where there is heavy confusion among Light Smoke, Medium Smoke, and Dense Smoke. Besides, the number of hazardous samples wrongly classified as background has also increased significantly. This implies that training YOLO from scratch on seven-class data yields poor robustness to classifying subtle fire/smoke patterns against backgrounds compared to the proposed training framework. Moreover, we note that training the YOLOv12 detector from scratch on seven-class data fails to preserve the severity ordering, as we observe clear confusion between non-adjacent severities, such as Small Fire and High Fire. We do recognize that some confusion between adjacent severities is to be expected, since fire/smoke severity may exhibit gradual transitions across different levels. However, the number of samples with such confusion grows significantly when the two-stage training strategy is discarded. We attribute this to the proposed two-stage training process, which helps build a stable representation in Stage I and fine-tunes the detector to discriminative fire/smoke severities in Stage II.

Comparing with results reported in Table 11, the proposed two-stage training strategy leads to +0.332 improvement in mAP@50, +0.356 improvement in Precision, +0.269 improvement in Recall, and +0.318 improvement in F1-score. These results confirm that our two-stage training strategy substantially improves severity-aware discrimination performance, thereby validating its effectiveness.

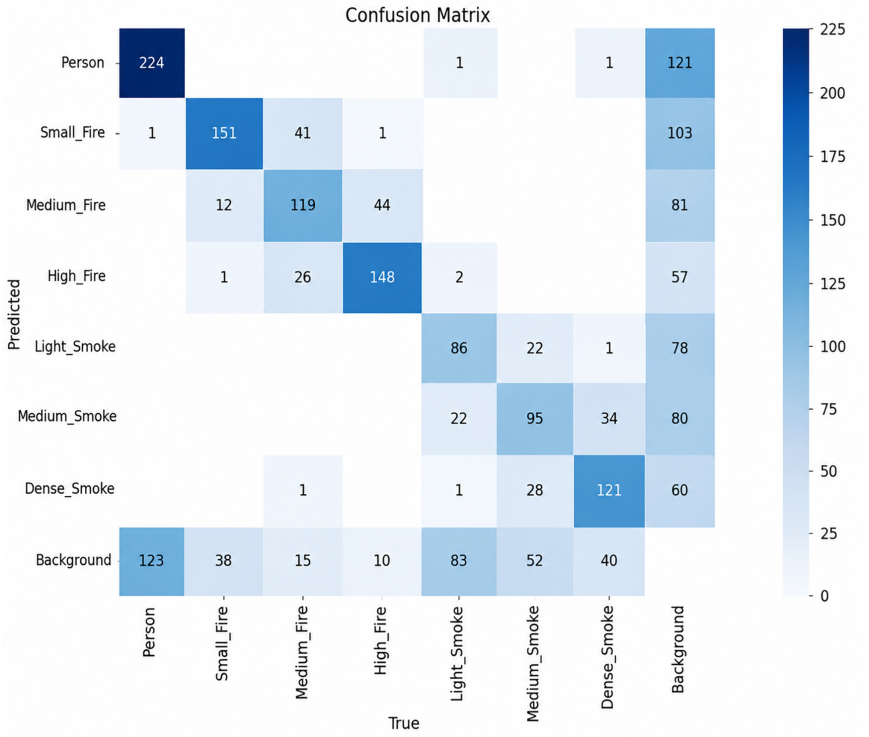

Figure 15: Confusion matrix for YOLOv12 model without two-stage training strategy.

Table 11: Ablation comparison performance for the two-stage training strategy across YOLOv12.

Model/Strategy	mAP@50	Precision	Recall	F1-Score
YOLOv12/with strategy	0.929	0.884	0.876	0.880
YOLOv12/without strategy	0.597	0.528	0.607	0.562

7 Conclusion and Future Work

In this paper, we introduce a severity-aware multi-class fire and smoke detection model to enhance situation awareness and aid vision-based hazard monitoring systems. Standard methods for automatic fire detection typically provide information on the presence or absence of fire/smoke regions in the frame. However, by introducing a categorical, severity-aware fire/smoke detection formulation into the standard hazard-monitoring system architecture, we modeled fire hazards as coarse-grained hierarchical categories comprising interdependent classes representing multiple levels of fire/smoke severity and human presence. The fundamental novelty of this system lies in modeling fires as categories rather than as a unit-class detection problem, enabling richer context-aware information that supports better-informed emergency responses. We designed the proposed system after a layered hazard-monitoring architecture involving visual sensing, deep learning–based detection, and response coordination modules. Towards enhancing feature stability and alleviating training burden, we propose a two-stage training strategy where coarse-grained 3-class labels Person, Fire, and Smoke were utilized in stage I training to allow convergence to reasonably stable object-level feature embedding. Stage II training associated detection targets with seven-class severity-aware labels, allowing fine-grained differentiation between low, medium, and high fire intensity, as well as light,

medium, and dense smoke levels. We evaluate our approach experimentally on our collected dataset of 6500 labeled images.

We compare the performance of three detection models trained under the same experimental setting YOLOv12s, RT-DETR-L, and SSDLite320-MobileNetV3. Results indicate robust detection performance across all models, with RT-DETR-L exhibiting the highest overall accuracy mAP@50, scoring 0.931, followed by YOLOv12s scoring 0.929, and SSDLite-MobileNetV3 scoring 0.882. While the transformer-based Real-Time Detection Transformer (RTDETR) model exhibited the highest accuracy, our selected YOLOv12s model provides the fastest inference time, at 2.48 ms/image, while maintaining competitive precision, recall, and F1-score. Given comparable accuracy and large speedup from YOLOv12s inference, we favor this model for real-time applications. Across all evaluated models, our experimental results demonstrate that the severity-aware detection network generalizes to fine-grained fire and smoke recognition without sacrificing accuracy in person detection. Extending the detection model beyond binary-level fire alarm systems provides additional context and granularity to risk-aware monitoring systems or early-stage disaster response. Our experiments yielded promising results, yet we acknowledge that challenging real-world environments, such as nighttime, obscured views, or atmospheric conditions (e.g., fog and dust), may lead the model to make false detections. Another limitation in our work is that we do not test on completely unseen external datasets to measure generalization performance. In addition, our study shows that still images cannot represent the temporal evolution of fire and smoke growth patterns.

Future work will consider a range of extensions to improve our proposed severity-aware detection model. Firstly, we will evaluate the effectiveness of our model on other independent fire/smoke datasets. We also plan to study domain adaptation and the generalization ability of our severity-aware labeling approach on unseen environments and imaging conditions. The data collection will be expanded to include more challenging scenes and to learn better features for handling nighttime, obscured views, and atmospheric conditions. Adding more diversity to the training and validation datasets can increase robustness to challenging fire-detection conditions, such as nighttime monitoring, low and high lighting, and congested backgrounds. Sampling more data from marginal classes, such as small flames or low-density smoke, could also enable more reliable detection of these early warning signs. Moreover, we plan to conduct cross-dataset validation on heterogeneous real-world datasets to assess robustness and scalability. Controlled ablation experiments are another focus of future work to quantify the effects of negative samples on false-positive reduction and overall model robustness. Secondly, we plan to expand our model to a spatiotemporal hybrid by incorporating video frames as inputs, enabling it to learn temporal features and provide a better estimate of severity. Thirdly, we aim to investigate the deployment of our detection model on embedded GPUs and edge computing devices (e.g., NVIDIA Jetson and Raspberry Pi) under realistic IoT monitoring conditions. Future work will include analyzing how accuracy degrades under limited computing resources. Finally, detection could be improved by using additional sensing modalities. One example is sensor fusion with a thermal camera or infrared imagery, which can provide useful cues when smoke is present or when flames are less bright. Video stream inputs could also be used to enable predictive analysis of fire hazards by training sequential models to forecast early fire growth.

Acknowledgement: Not applicable.

Funding Statement: The authors received no specific funding for this study.

Author Contributions: Conceptualization, Ayman Noor and Talal H. Noor; methodology, Aminah Almehmadi, Aziza I. Noor, Hanan Almukhalfi and Talal H. Noor; validation; Aminah Almehmadi, Ayman Noor, Aziza I. Noor and Hanan Almukhalfi; formal analysis, Aminah Almehmadi, Aziza I. Noor and Talal H. Noor; investigation, Aminah Almehmadi, Ayman Noor, Aziza I. Noor and Talal H. Noor; resources, Aminah Almehmadi, Ayman Noor, Hanan Almukhalfi and

Talal H. Noor; data curation, Aminah Almehmadi, Ayman Noor, Aziza I. Noor and Hanan Almukhalfi; writing—original draft preparation, Aminah Almehmadi, Ayman Noor and Aziza I. Noor; writing—review and editing, Ayman Noor and Talal H. Noor; visualization, Aminah Almehmadi, Aziza I. Noor and Hanan Almukhalfi; supervision, Ayman Noor and Talal H. Noor; project administration, Talal H. Noor; funding acquisition, Ayman Noor, Aziza I. Noor, Hanan Almukhalfi and Talal H. Noor. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The Fire–Smoke–Person Detection Dataset (3-Class) used in this study is publicly available at (https://kaggle.com/datasets/80cd1c92a408c9e03332b182b0a7e4faf167f22cf50363bd77587380a5f1ced6). Where the Fire–Smoke–Person Severity Dataset (7-Class) used in this study is publicly available at (https://kaggle.com/datasets/753f066efcddbcad72d47ee42178638f39e9df259d8501f5f6615f3c91ebafa2).

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Shi L, Wang J, Li G, Chew MYL, Zhang H, Zhang G, et al. Increasing fire risks in cities worldwide under warming climate. Nat Cities. 2025;2(3):254–64. doi:10.1038/s44284-025-00204-2.
2. Jin C, Wang T, Alhusaini N, Zhao S, Liu H, Xu K, et al. Video fire detection methods based on deep learning: datasets, methods, and future directions. Fire. 2023;6(8):315. doi:10.3390/fire6080315.
3. Ismail ND, Ramli R, Ab Rahman MN. A systematic literature review of vision-based fire detection, prediction and forecasting. J Kejuruter. 2025;37(1):191–218. doi:10.17576/jkukm-2025-37(1)-13.
4. Chaturvedi S, Khanna P, Ojha A. A survey on vision-based outdoor smoke detection techniques for environmental safety. ISPRS J Photogramm Remote Sens. 2022;185(14):158–87. doi:10.1016/j.isprsjprs.2022.01.013.
5. Khan F, Xu Z, Sun J, Khan FM, Ahmed A, Zhao Y. Recent advances in sensors for fire detection. Sensors. 2022;22(9):3310. doi:10.3390/s22093310.
6. Rehman A, Kim D, Anand P. Convolutional neural network model for fire detection in real-time environment. Comput Mater Contin. 2023;77(2):2289. doi:10.32604/cmc.2023.036435.
7. El-afifi MI, Team S, M Elkelany M. Development of fire detection technologies. Nile J Commun Comput Sci. 2024;7(1):58–66. doi:10.21608/njccs.2024.263103.1027.
8. Deng L, Wu S, Zou S, Liu Q. Large-space fire detection technology: a review of conventional detector limitations and image-based target detection techniques. Fire. 2025;8(9):358.
9. Wen C, Li K, Liao Y, Xiao Z. Design of an intelligent alarm system based on multi-sensor data fusion. J Phys Conf Ser. 2021;1961(1):012025. doi:10.1088/1742-6596/1961/1/012025.
10. Barmpoutis P, Papaioannou P, Dimitropoulos K, Grammalidis N. A review on early forest fire detection systems using optical remote sensing. Sensors. 2020;20(22):6442. doi:10.3390/s20226442.
11. Dilshad N, Khan T, Song J. Efficient deep learning framework for fire detection in complex surveillance environment. Comput Syst Sci Eng. 2023;46(1):749–64. doi:10.32604/csse.2023.034475.
12. Avazov K, Mukhiddinov M, Makhmudov F, Cho YI. Fire detection method in smart city environments using a deep-learning-based approach. Electronics. 2021;11(1):73. doi:10.3390/electronics11010073.
13. Saleh A, Zulkifley MA, Harun HH, Gaudreault F, Davison I, Spraggon M. Forest fire surveillance systems: a review of deep learning methods. Heliyon. 2024;10(1):e23127.
14. Wang Z, Wu L, Li T, Shi P. A smoke detection model based on improved YOLOv5. Mathematics. 2022;10(7):1190. doi:10.3390/math10071190.
15. Vasconcelos RN, Franca Rocha WJ, Costa DP, Duverger SG, MMd S, Cambui EC, et al. Fire detection with deep learning: a comprehensive review. Land. 2024;13(10):1696. doi:10.3390/land13101696.
16. Kang J, Tariq S, Oh H, Woo SS. A survey of deep learning-based object detection methods and datasets for overhead imagery. IEEE Access. 2022;10:20118–34. doi:10.1109/access.2022.3149052.
17. Amjoud AB, Amrouch M. Object detection using deep learning, CNNs and vision transformers: a review. IEEE Access. 2023;11(1):35479–516. doi:10.1109/access.2023.3266093.

18. Noor A, Almukhalfi H, Souza A, Noor TH. Towards a real-time indoor object detection for visually impaired users using raspberry Pi 4 and YOLOv11: a feasibility study. Comput Model Eng Sci. 2025;144(3):3085–111. doi:10.32604/cmes.2025.068393.
19. Saponara S, Elhanashi A, Gagliardi A. Real-time video fire/smoke detection based on CNN in antifire surveillance systems. J Real Time Image Process. 2021;18(3):889–900. doi:10.1007/s11554-020-01044-0.
20. Qu X, Dong H, Tan X, Li Z. Real-time fire detection and response system using machine vision for industrial safety. Int J Mod Phys C. 2026;37(06):2542004. doi:10.1142/s0129183125420045.
21. Vorwerk P, Kelleter J, Müller S, Krause U. Classification in early fire detection using multi-sensor nodes—a transfer learning approach. Sensors. 2024;24(5):1428. doi:10.3390/s24051428.
22. Alkhammash EH. Multi-classification using YOLOv11 and hybrid YOLO11n-MobileNet models: a fire classes case study. Fire. 2025;8(1):17.
23. Khan RA, Hussain A, Bajwa UI, Raza RH, Anwar MW. Fire and smoke detection using capsule network. Fire Technol. 2023;59(2):581–94. doi:10.1007/s10694-022-01352-w.
24. Sousa MJ, Moutinho A, Almeida M. Thermal infrared sensing for near real-time data-driven fire detection and monitoring systems. Sensors. 2020;20(23):6803. doi:10.3390/s20236803.
25. Kallianiotis A, Papakonstantinou D, Tolias IC, Benardos A. Evaluation of fire smoke control in underground space. Undergr Space. 2022;7(3):295–310. doi:10.1016/j.undsp.2021.07.010.
26. Wang L, Zhang X, Li L, Li B, Mei Z. Experimental study on early fire smoke characteristics in a high-volume space: a fire detection perspective. Fire. 2024;7(9):298. doi:10.3390/fire7090298.
27. Al-Smadi Y, Alauthman M, Al-Qerem A, Aldweesh A, Quaddoura R, Aburub F, et al. Early wildfire smoke detection using different YOLO models. Machines. 2023;11(2):246. doi:10.3390/machines11020246.
28. Ramos LT, Casas E, Romero C, Rivas-Echeverría F, Bendek E. A study of YOLO architectures for wildfire and smoke detection in ground and aerial imagery. Results Eng. 2025;26:104869. doi:10.1016/j.rineng.2025.104869.
29. Wang H, Fu X, Yu Z, Zeng Z. DSS-YOLO: an improved lightweight real-time fire detection model based on YOLOv8. Sci Rep. 2025;15(1):8963. doi:10.1038/s41598-025-93278-w.
30. Geng X, Han X, Cao X, Su Y, Shu D. YOLOV9-CBM: an improved fire detection algorithm based on YOLOV9. IEEE Access. 2025;13:19612–23.
31. Lin Z, Yun B, Zheng Y. LD-YOLO: a lightweight dynamic forest fire and smoke detection model with dysample and spatial context awareness module. Forests. 2024;15(9):1630.
32. Xue Z, Kong L, Wu H, Chen J. Fire and smoke detection based on improved YOLOV11. IEEE Access. 2025;13(11):73022–40. doi:10.1109/access.2025.3564434.
33. Zhang Z, Tan L, Robert TLK. An improved fire and smoke detection method based on YOLOv8n for smart factories. Sensors. 2024;24(15):4786. doi:10.3390/s24154786.
34. Zhou K, Jiang S. Forest fire detection algorithm based on improved YOLOv11n. Sensors. 2025;25(10):2989. doi:10.3390/s25102989.
35. Yang M, Qian S, Wu X. Real-time fire and smoke detection with transfer learning based on cloud-edge collaborative architecture. IET Image Process. 2024;18(12):3716–28. doi:10.1049/ipr2.13187.
36. Yang X, Li Y, Chen Q. Automated image-based fire detection and alarm system using edge computing and cloud-based platform. Internet Things. 2024;28(1):101402. doi:10.1016/j.iot.2024.101402.
37. Borges N, Fonseca L, Barreto PS, Alchieri E, Caetano MF, Resende P, et al. A fire management intelligent system for the Brazilian cerrado biome based on a deep learning two phase detection method. J Reliab Intell Environ. 2025;11(1):5. doi:10.21203/rs.3.rs-4865999/v1.
38. El-Madafri I, Peña M, Olmedo-Torre N. Dual-dataset deep learning for improved forest fire detection: a novel hierarchical domain-adaptive learning approach. Mathematics. 2024;12(4):534. doi:10.3390/math12040534.
39. Sultan T, Chowdhury MS, Safran M, Mridha M, Dey N. Deep learning-based multistage fire detection system and emerging direction. Fire. 2024;7(12):451. doi:10.3390/fire7120451.
40. Suh Y. Vision-based detection algorithm for monitoring dynamic change of fire progression. J Big Data. 2025;12(1):134. doi:10.1186/s40537-025-01211-9.

41. Sun B, Cheng X. Smoke detection transformer: an improved real-time detection transformer smoke detection model for early fire warning. Fire. 2024;7(12):488. doi:10.3390/fire7120488.
42. Ghaffarian S, Taghikhah FR, Maier HR. Explainable artificial intelligence in disaster risk management: achievements and prospective futures. Int J Disaster Risk Reduct. 2023;98(3):104123. doi:10.1016/j.ijdrr.2023.104123.
43. Mustafa AM, Agha R, Ghazalat L, Sha'ban T. Natural disasters detection using explainable deep learning. Intell Syst Appl. 2024;23(3):200430. doi:10.1016/j.iswa.2024.200430.
44. Dobrzycki AD, Bernardos AM, Casar JR. An analysis of layer-freezing strategies for enhanced transfer learning in YOLO architectures. Mathematics. 2025;13(15):2539. doi:10.3390/math13152539.
45. Hussain A, Ullah K, Afaq M, Munsif M, Hussain A, Baik SW. Quality over quantity: a data-centric survey of annotation errors in object detection datasets. Artif Intell Rev. 2026;59:107.
46. Li G, Li X, Wang Y, Wu Y, Liang D, Zhang S. Pseco: pseudo labeling and consistency training for semi-supervised object detection. In: European Conference on Computer Vision (ECCV). Berlin/Heidelberg, Germany: Springer; 2022. p. 457–72.