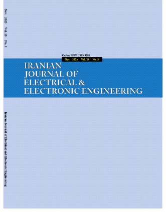
Iranian Journal of Electrical and Electronic Engineering 03. (2026) 3733

IUST
Iran University of
Science and Technology

Iranian Journal of Electrical and Electronic Engineering

Journal Homepage: ijeee.iust.ac.ir



Reducing False Alarms in Fire Detection Systems with YOLOv11 and Multi-Sensor Validation

Kausar Ahmed*, Bibhor Regan Gomes*, S.M. Jobair Hossain*, Emran Khan Musa*, Fuyad Hasan Bhoyan*, Md Humaion Kabir Mehedi**, Jia Uddin***(C.A.)

Abstract: Fires in indoor spaces such as residential and office buildings pose significant threats to human lives and property, causing substantial damage each year. Early and accurate fire detection plays a critical role in mitigating these risks and ensuring timely responses. However, conventional methods such as smoke sensors, temperature indicators, and standalone computer vision models suffer from limitations like false alarms, delayed detection, and high hardware demands. To address these challenges, we propose a novel three-layer verification framework for indoor fire detection to reduce false alarms, integrating smoke sensors, computer vision, and temperature monitoring into a multi-modal validation framework. The process begins with smoke sensors detecting potential fire incidents. The custom-trained YOLOv11n computer vision model verifies the detection using predefined thresholds, allowing immediate response without waiting for temperature escalation. If the computer vision model does not confirm the fire, the system initiates a temperature check as a final validation layer. Experimental evaluation of our model demonstrates a significantly high precision of 0.979 and a recall of 0.971. This layered approach ensures comprehensive detection, balancing reliability and resource efficiency. Our proposed hybrid AI-physical systematic framework demonstrates significant potential in reducing false alarms, improving detection accuracy, and prioritizing methodological scalability over industrial hardware. It lays the foundation for more reliable and energy-efficient fire safety solutions in smart buildings and industrial safety applications.

Keywords: False Alarm, Internet of Things (IoT), Computer Vision, Indoor Fire.

1. Introduction

FIRE is one of nature's most destructive forces, causing catastrophic consequences such as loss of life, property damage, and equipment destruction, especially in indoor settings. According to the U.S. Fire Administration, in 2022, there were 374,300 fires in residential buildings, which caused 2,720 deaths, 10,250 injuries, and over \$10.8 billion in damages. Additionally, 129,500 non-residential fires caused 140 deaths, 1,300 injuries, and over \$3.7 billion in losses [1]. These alarming statistics underscore the need for early and accurate fire detection systems to mitigate harm and ensure timely responses.

There are some limitations in existing fire detection solutions. Hardware-based detectors—heat, smoke, and flame sensors—suffer from delayed responses (e.g., heat detectors react 8 minutes slower than smoke detectors [2]). While faster smoke detectors are sensitive to strong airflow, it can cause a delay in smoke detection by 9 seconds [2]. Though flame sensors are efficient and quick, they are prone to false alarms because of their sensitivity to various IR sources like sunlight. Vision-based systems using surveillance cameras struggle in indoor environments, such as with obstructed views, reflections, and adversarial false alarms [3]. Furthermore,

---

Iranian Journal of Electrical & Electronic Engineering, 2026.
Paper first received 08 Feb 2025 and accepted 27 Dec 2025.

* The authors are with the Department of Computer Science and Engineering (CSE), University of Liberal Arts Bangladesh (ULAB), Dhaka, Bangladesh.
** The author is with the Department of Department of Computer Science and Engineering (CSE), BRAC University, Dhaka, Bangladesh.
*** The author is with the AI and Big Data Department, Endicott College, Woosong University, Daejeon, 34606, Republic of Korea.
Corresponding Author: Jia Uddin.
Email: jia.uddin@wsu.ac.kr

standalone sensor systems risk frequent false alarms [4], while continuous computer vision models like YOLO demand excessive computational resources, making them impractical for standard surveillance setups.

Motivated by the limitations of conventional fire detection systems, we propose a novel three-layer framework that integrates smoke sensors, computer vision, and temperature-based validation for early and accurate fire detection. We focused on developing a validation methodology, rather than hardware superiority. The system operates hierarchically: smoke sensors serve as the first layer to identify potential fire incidents, followed by a YOLOv11n-based computer vision model that verifies the presence of fire. If no visual fire is confirmed, a temperature-based validation layer acts as the final checkpoint. Our components of this framework are illustrative; thus, these can be replaced by industrial-grade alternatives. By combining sensors with computer vision, our method ensures rapid, energy-efficient detection without compromising reliability, aligning with the four-step fire protection framework (prevention, evacuation, rescue) recommended in prior research. [5].

The key contributions of this research are outlined below:

- By incorporating a three-layer verification method, our approach significantly reduces false alarms caused by standalone smoke or flame sensors. The hierarchical process makes sure that random smoke, non-fire IR sources, and airflow disturbances do not trigger unnecessary alerts.
- Unlike conventional deep-learning-based fire detection models that require high-end computational resources, our YOLOv11n-based vision model is optimized to run on low-power surveillance computers. This makes the system practical for deployment in standard indoor setups without the need to invest in additional hardware.
- The combination of computer vision and energy-efficient sensors ensures that our system operates with minimal power consumption. Additionally, the system's modular nature allows for cost-effective implementation without requiring expensive infrastructure modifications.
- The system can function wirelessly, eliminating the need for extensive wiring and reducing the complexity of the installation. Remote-controlled configuration improves ease of deployment and management of the system, making it suitable for various indoor environments, including industrial and residential settings.
- This system can be integrated with pre-existing CCTV surveillance networks, eliminating the need for an entirely new fire detection setup. This integration reduces installation costs and enhances the efficiency by utilizing already-installed cameras for visual verification.
- The modular design of our fire detection system allows for easy expansion and adaptation to different indoor environments. Whether deployed in factories, office spaces, or residential buildings, the system can be customized to meet specific fire safety requirements without requiring extensive modifications.

The remainder of this paper is organized as follows: Section 2 reviews related work on IoT and ML-based fire detection. Section 3 details our methodology, including hardware setup and YOLOv11n architecture. Section 4 presents experimental results and comparative analysis. Finally, Section 5 summarizes the findings and outlines directions for future work.

2. Literature Review

In recent years, the increasing prevalence of fire hazards has driven the need for smarter and more efficient detection systems. The development of IoT-based fire detection and prevention systems has gained significant attention in recent years, with various researchers proposing innovative solutions tailored to different environments and use cases. For instance, some researchers have implemented IoT architectures to monitor agricultural resources, with their proposed system integrating smoke and flame sensors with a Raspberry Pi for local data processing and the ThingSpeak platform for data storage and visualization, enabling early fire detection and efficient monitoring [6]. These IoT-based systems have proven to be highly effective in detecting fires in remote areas where traditional monitoring methods may not be feasible. Even in industrial contexts, IoT systems have also proven valuable. Authors [7] have designed a wireless sensor network-based fire detection system for garment factories, capable of locating fire indoors. Their evacuation process can further be improved by incorporating a more precise location tracking system. This paper proposes that type of system using a Global Positioning System (GPS) and a floor plan to track the exact fire location [8]. Similarly, in another study [9], researchers proposed a fire alarm navigation system using IoT, integrating GPS, flame, temperature, and smoke sensors, buzzers, LEDs, and a GSM module for early fire detection and emergency notifications, in order to reduce loss of lives and properties. The combination of these systems can lead to more sophisticated procedures to ensure a timely response and minimize potential damage.

Indoors, IoT systems have been leveraged for enhancing safety in residential and commercial settings. A smart kitchen safety and automation solution was developed to address gas leaks, fire, and high-temperature hazards by automatically activating safety measures and notifying users through the Blynk app, thereby providing real-time response capabilities [10]. In a similar way, this research presented an IoT-based fire detection system utilizing temperature, smoke, and gas sensors to identify early risks [11] . The study demonstrated the effectiveness of real-time monitoring and remote accessibility in enhancing fire safety measures. Additionally, it emphasized the role of automated alert systems in reducing property damage and ensuring occupant safety. Likewise, a low-cost smart fire monitoring and detection system was proposed to collect real-time data on temperature, humidity, and smoke levels, ensuring timely alerts and evacuation for homeowners through a dashboard and mobile notifications [12] . Meanwhile, this study [13] introduced a home fire alarm system that integrates a microcontroller with multiple sensors to enhance fire safety. By combining combustible gas, temperature, and flame detection modules, the system provides early warnings, sound and light alarms, and exhaust functions. Its design highlights advantages such as accurate detection and easy installation, offering a practical solution for home fire safety. Some IoT solutions focus on scalability and adaptability across different environments. A mesh network of ESP8266 nodes equipped with sensors for parameters such as methane, humidity, or temperature was proposed to monitor fire hazards. This system, connected to a Raspberry Pi central controller via the MQTT protocol, could alert users and emergency services through SMS, calls, and alarms. Its simplicity and flexibility made it suitable for both households and businesses [14] . Along the same lines, this research proposed a fire detection system that uses multiple sensors and IoT technology to enable early fire identification and real-time alerts [15] . The study underscored the importance of rapid detection in preventing asset loss and ensuring occupant safety.

By integrating advanced technologies such as IoT, GPS, and real-time monitoring systems, these approaches collectively demonstrate the growing potential of IoT-based solutions in addressing fire hazards across diverse environments, from agricultural sectors to urban homes and industrial complexes.

Furthermore, researchers have also explored integrating IoT with machine learning (ML) or using standalone ML models to achieve greater accuracy in fire detection. An IoT-based wireless sensor network combined with machine learning techniques demonstrated early detection and prevention capabilities by using multiple criteria-based sensing, real-time data analysis, and classification models to alert authorities promptly [16] . Vision-based systems have also been developed, such as a fire and smoke detection solution utilizing convolutional neural networks (CNNs) to provide real-time detection and crucial information to fire services, including the fire's location, size, and spread [17].

Deep learning-based object detection methods have also significantly advanced fire detection. Researchers proposed a smart fire detection system (SFDS) using the YOLOv8 algorithm to enable real-time fire detection in smart cities. By integrating IoT and Fog-Cloud computing, the system enhanced accuracy and response time, ultimately achieving 97.1% precision, reducing false alarms, and improving fire safety management [18],[36] . Similarly, other studies have explored transformer-based enhancements to fire detection models. TRA-YOLO, a novel fire detection model, combined a Transformer encoder with CNNs while introducing an improved ACK-Res2Net module to refine feature extraction for flames and smoke. Additionally, an adaptive spatial feature fusion (ASFF) structure was implemented to suppress background interference, leading to higher accuracy and real-time performance compared to YOLOv5 [19].

Further optimization of YOLO-based models has been explored to enhance fire detection in complex environments. The ESFD-YOLOv8n model incorporated Wise-IoU version 3 (WIoUv3) and GELAN blocks to improve smoke and fire detection accuracy while ensuring real-time performance. By replacing the C2f module with a residual block, researchers reduced computational complexity and minimized overfitting, resulting in a more efficient detection model [20] . Another refinement, FFS-YOLO, was introduced as a single-stage detector tailored for factory settings. By integrating the Parameter-Free Attention Module (SimAM) and ResNet-SimMix into YOLOv7, the model improved small-scale fire and smoke detection. The addition of an extra prediction head enabled better multi-scale feature fusion, thus making it a robust solution for early fire detection in industrial environments [21].

Additionally, fire detection in urban and indoor settings presents unique challenges due to the presence of small targets and high false alarm rates. To address these concerns, the CAGSA-YOLO algorithm was designed as a fire safety detection model, refining YOLOv5 by incorporating the CARAFE module for enhanced feature extraction. The model also introduced a new scale detection layer to improve performance on larger fire instances while leveraging the lightweight C3Ghost module to maintain efficiency. The proposed modifications significantly improved fire detection accuracy while reducing model size and computational demands, making it an effective solution for fire monitoring in complex urban environments [22].

Advancements in computer vision algorithms have further refined detection capabilities, with studies leveraging state-of-the-art object detection models such as Faster-RCNN, R-FCN, SSD, and YOLOv3 for image-based fire detection [23] . Beyond detection, ML regression models have been proposed for improving the accuracy of fire identification [24] . These innovative approaches collectively contribute to the development of more robust, efficient, and real-time fire detection systems that can be deployed across various environments, ensuring faster response times and enhanced safety.

3. Methodology

We have discussed our proposed system's structure and execution process in the following section. This proposed system consists of both an IoT sensor-driven hardware implementation and a YOLOv11n-driven software implementation. In this system, the IoT sensors will be used for environmental monitoring, for example, measurement of environmental substances like smoke level and temperature level, to determine potential fire situations in the sensor's operation area. And the YOLOv11n will serve as a software-based layer of detection for faster and more accurate responses. These integrations will allow our system to determine a fire situation and mitigate any false positive situation.

The specific steps to identify fire situations and mitigation strategies using this multimodal approach are presented in the following subsections. This methodology covers the hardware setup, dataset description, model training, YOLOv11 architecture, proposed algorithm, system workflow, and evaluation framework.

3.1 Hardware Setup

The implementation begins with the integration of carefully selected components. However, components were selected for accessibility and proof-of-concept purposes. The framework is hardware-agnostic; industrial sensors can be used for deployment. Components used include:

- **MQ2 Gas Sensor:** This sensor will monitor smoke and gas levels to detect potential fire-related anomalies. MQ2 is a versatile MOS (Metal Oxide Semiconductor) sensor that can detect a wide range of gases including LPG, Smoke, Alcohol, Propane, Hydrogen, Methane, and Carbon Monoxide (CO) concentrations in the air [25] , making it an excellent choice for building an indoor air quality monitor, or an early fire detection system. This sensor also has an excellent detection range from 200 to 10,000 ppm, which operates on 5V DC and consumes approximately 800 mW, making it a compelling solution for smoke detection caused by several types of fires.
- **LM35 Temperature Sensor:** LM35 is a precision integrated-circuit temperature sensor. This will be used for tracking environmental temperatures to cross-validate smoke detection. This sensor has a relatively high accuracy of ±0.5°C, a wide temperature detection range from -55°C to +150°C, and a fast response time [26], which are crucial for early fire detection. With its analog output, less than 60 mA current drain, and very low self-heating (0.08°C in still air), this sensor promises more accurate and continuous temperature monitoring compared to digital alternatives.
- **ESP32 Microcontroller:** ESP32-WROOM-32D is a powerful, generic Wi-Fi + Bluetooth® + Bluetooth LE MCU module. This microcontroller has an 80 MHz to 240 MHz clock frequency CPU, only 80 mA average operating current, and is operable up to 85°C [27]. With its Wi-Fi support to directly connect to the Internet, it is perfect for wireless system setup. Additionally, its high-performance integration, wireless transmission distance, power consumption, and network connectivity make it the perfect hub for our system.

3.2 Dataset Description

Our proposed system implemented a computer vision-based visual detection mechanism. For this, we have used a pretrained model. To get more enhanced detection, we have custom trained our model with a custom fire dataset, including images mostly from indoor fire situations. The dataset we used for training is an open-source dataset, which we have sourced from Roboflow, published in the Roboflow Universe [28] . The author of the dataset is the University of the Philippines, Cebu. The dataset contains a total of 9,128 images. The images are annotated with bounding boxes around the fire. The dataset consists of six types of fire flames: Cooking Oil, Electrical, Gas, Liquid, Metal, and Solid fires. These categories were used for classifying and detecting various types of fire during the model's evaluation. Fig 1 demonstrates the distribution of images in each class.

The images of the dataset are 320×240-pixel size and 0.08 MP. This perfectly reflects real-world CCTV limitations, aligning with our priority to be compatible with standard existing surveillance systems. The authors mentioned some preprocessing on the dataset. As of them, images were resized or stretched to 320x240 pixel size. Some images are auto-oriented to the size; however, no augmentations were applied. There are no null or missing annotations; each image in the dataset is annotated with bounding boxes around the fire regions and labeled based on the flame type. The sample input images from the dataset are shown in Fig 2.

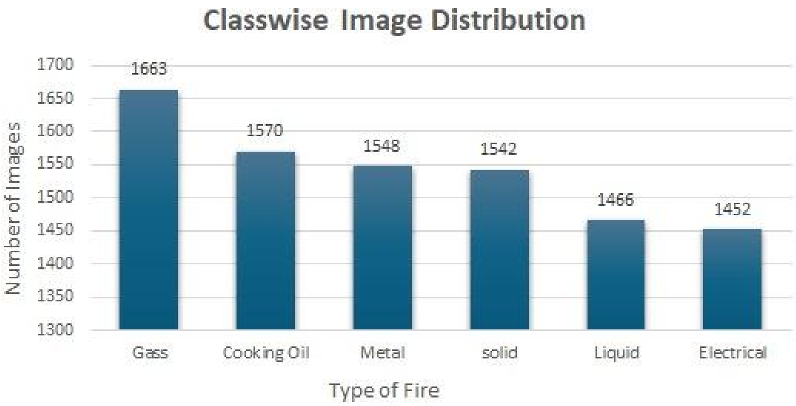


Fig 1. Dataset image distribution

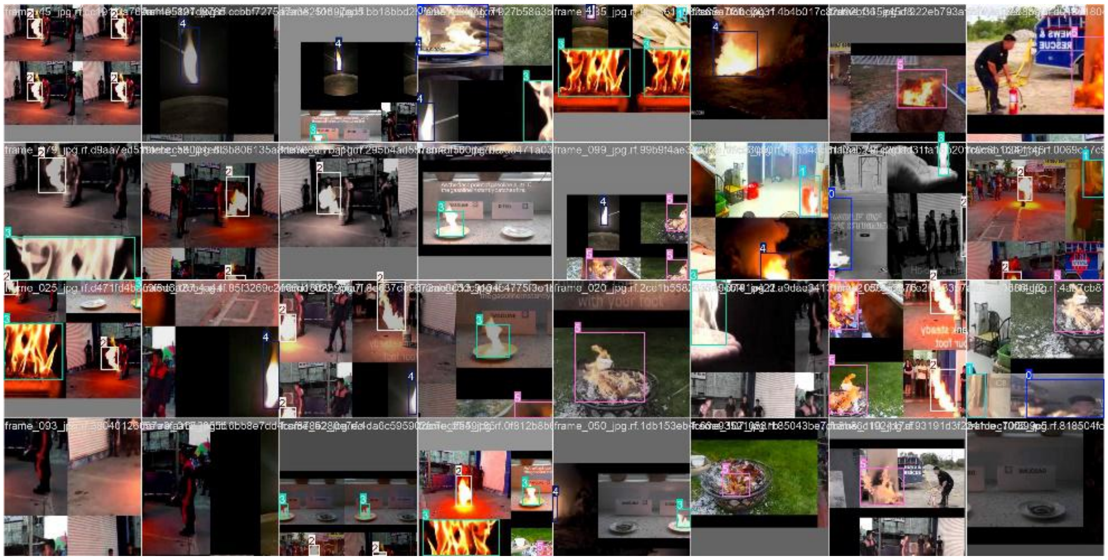


Fig 2. Dataset sample

3.3 Model Training

For our proposed system's computer vision-based validation layer, we have used the YOLOv11n model. The YOLO framework revolutionized object detection with its unified neural network architecture, simultaneously handling both bounding box regression and object classification tasks [29].

Data Split: Among the 9128 total images, the dataset was split into 70% (6388) for training the model, 15% (1374) for testing, and 15% (1366) was set aside for model validation. This split ensures a proper balance between the test, train, and validation sets with enough data in each set, allowing the model to be trained effectively.

Hyperparameters: Different hyperparameters, including batch size, optimizer, learning rate, image resolution, training epochs, and number of epochs, were first defined, and through an iterative validation process, these metrics were optimized.

Training Procedure: To train the computer vision model, we used Google Colab utilizing a Tesla T4 GPU (15095 MiB). After training the model, we used the model to predict fire occurrences on unseen data. For an extra layer of validation, we have tested the model on both static images and videos apart from the dataset. Which ensures the model's robustness in a wide range of real-world scenarios.

3.4 YOLOv11 Architecture

The YOLOv11 is a state-of-the-art, real-time object detection architecture algorithm developed by Ultralytics [30]. For this study, we used the YOLOv11n (YOLOv11 nano). This is the smallest among the YOLOv11 series, which is selected to balance the computation overhead. This is an advanced CNN-based model of the YOLO series, with key architectural enhancements that improve feature extraction efficiency, detection accuracy, and computational speed. Its design focuses on balancing power and practicality [31] . The model integrates the C3k2 block (Cross Stage Partial with kernel size 2), which replaces previous C2f blocks, leading to faster and more efficient operations. The model architecture consists of three main components: the backbone, which extracts multi-scale feature representations; the neck, which refines and aggregates features; and the head, which generates object detection predictions. It offers multiple variants, from nano to extra-large. The architecture achieves a balance between speed and accuracy, making it suitable for deployment in both edge devices and high-performance computing environments. The key architectural modules of the YOLOv11 model are illustrated in Fig 3.

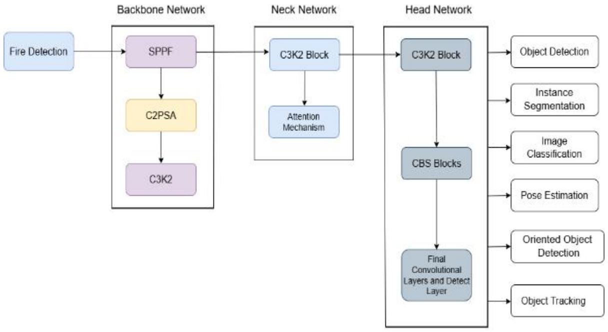


Fig 3. Key architectural modules in YOLOv11 [31]

3.5 Proposed Algorithm

Based on our core principle operational concepts, the surveillance system will follow a sophisticated algorithm. The algorithm is designed to minimize computation overhead; the YOLO/temperature checks activate only after any smoke detection. The proposed algorithm is outlined in the following Algorithm 1.

3.6 System Workflow

In this proposed system, we have designed a sophisticated fire detection system to reduce the number of false alarms through a multi-layer fire breakout validation process. The system architecture encompasses three key steps: smoke detection, computer vision-based fire detection and validation, and temperature-based detection cross-validation. The system confirms a fire event only when at least two of these validation layers agree, ensuring a balanced trade-off between detection accuracy and false alarm prevention. The operation of the system is as follows:

Algorithm 1: Continuous Fire Detection System

```
Initialize system parameters
0.1 Set thresholds: SMOKE_THRESHOLD, TEMP_THRESHOLD
0.2 Define constant: FIRE_CHECK_DURATION, TEMP_CHECK_INTERVAL, TEMP_CHECK_ATTEMPTS.
0.3 Load YOLOv11 model for fire detection
0.4 Establish Connection to ESP device with ESP_IP.
Start Monitoring Loop
While system is running do
Retrieve Smoke sensor data using get_smoke_level().
    If smoke level > SMOKE_THRESHOLD
    then
        Perform fire validation steps:
        Camera-based validation:
        activate YOLOv11n model with a live camera feed using detect_fire().
        if YOLO detects fire, then
            Save the detected fire frame with timestamp.
            trigger_alarm()
        else
            Temperature-based validation step:
            for TEMP_CHECK_ATTEMPTS iterations
            do
                Retrieve temperature sensor data using monitor_monitor().
                if temperature > TEMP_THRESHOLD
                then
                    trigger_alarm()
                end if
                Wait for TEMP_CHECK_INTERVAL
                seconds before the next check.
            end for
        end if
    else
        continue monitoring
    end if
    Wait for 5 seconds before the next iteration.
end while
```

1) Primary Detection Layer (MQ2 Sensor): The MQ2 gas sensor monitors smoke concentration continuously all the time and acts as the system's primary trigger. An empirically determined smoke threshold will be used to ensure sensitivity and trigger activation. Upon crossing this threshold, the system will consider this a primary detection.

2) Visual Validation Layer (YOLOv11 Model): If the MQ2 sensor detects smoke, the YOLOv11 model will start analyzing the camera feed for visual confirmation of fire. An empirically determined confidence threshold is used to ensure reliable detection across diverse fire types and environment setups. If the YOLO model also detects fire, then the alarm is triggered based on this two-step verification.

3) Environmental Cross-Validation (LM35 Sensor): In cases where the YOLOv11 model does not confirm fire, the LM35 temperature sensor provides a secondary check. Because previously smoke was detected, and the model is not 100% accurate, this check will ensure robust fire detection. An empirically determined temperature threshold validates the presence of fire during ambiguous smoke scenarios. If the temperature level crosses the threshold, then the alarm will be triggered. Otherwise, the system will consider the smoke increase as a false positive and will again monitor normally.

The ESP32 microcontroller was programmed using the Arduino IDE, and communication protocols are established over Wi-Fi for wireless communication. Also, it implements error-handling mechanisms to manage sensor data integrity. The system is deployed using the Python programming language. This program runs on a surveillance computer on an IDE, processing sensor data and camera feed in real time. Python libraries such as OpenCV, PySerial, and Ultralytics are utilized for video feed processing and hardware communication.

Each layer contributes not only to the overall goal of reducing false alarms caused by non-fire smoke sources, such as cigarettes or incense, while ensuring reliable detection of actual fires but also to a very fast and efficient 2-step detection using computer vision. Fig 4 presents a flowchart of the system architecture and work procedure. Fig 5 demonstrate interactions between hardware and software components, relationships and data pathways between the parts (controller, camera, and sensors).

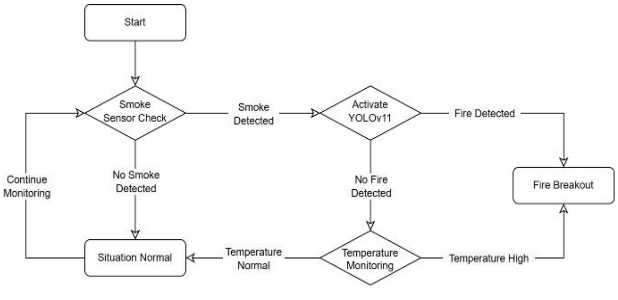


Fig 4. System architecture diagram

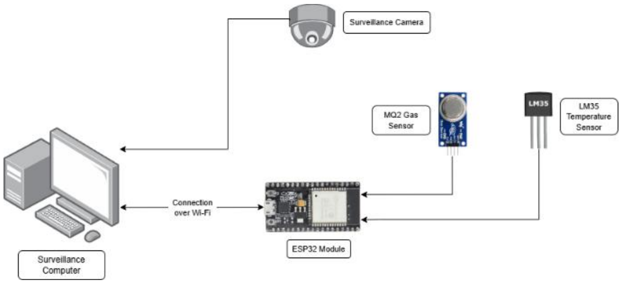


Fig 5. Assembly of the proposed framework

3.7 Evaluation Framework

To ensure the reliability of the system's first line of validation, the MQ2 smoke sensor was tested using controlled smoke sources like aerosol sprays and personal-use gas lighters. These tests were done to verify how the sensor responds to different smoke intensities and confirm that it was sensitive enough to detect potential fire hazards. For evaluating the YOLOv11-based fire detection model, we tested its performance using a live camera feed connected to a laptop. The model was initially tested with fire images and video clips of fire to assess its capability to differentiate moving flames. To further validate its performance, we conducted a controlled test with various lighting conditions (presented in later sections). This step ensured that the model could detect actual flames in real-time scenarios.

Since replicating real fire temperatures in a controlled indoor environment was not feasible for us and we were not able to expose the LM35 sensor to extreme fire temperatures, we verified its functionality within the controlled test environment by decreasing the threshold, ensuring that all three validation layers worked effectively in conjunction to provide a reliable and optimized fire detection system.

4. Result

This section discusses the experimental results of our proposed system's computer vision model for rapid second-layer validation. To implement this, we have trained the YOLOv5, YOLOv8, YOLOv9, and YOLOv11. These models are trained with various training settings and hyperparameters to ensure accurate fire detection. However, other efficient vision models can also be used in this vision validation layer.

Table 1 presents the utilized unified parameters and their respective values. These models were trained with a learning rate of 0.01 and a weight decay of 0.001. While training, we have considered six fire classes (types of fire). All the models are trained on all 9128 images categorized into six classes. The training process involves learning to predict the bounding box around the test fire images.

Table 1. Parameter used for model training

Parameter	Value
Batch Size	16
Number of epochs	25
Optimizer	AdamW
Pretrained	COCO model
Learning rate	0.01
Weight decay	0.005
Patience	100

Models are trained with 25 epochs to avoid overfitting, and following the approach suggested in [32], we have trained the model with a large dataset with fewer epochs. For all training sessions, the batch size was set to 16. Standardizing 640 x 640-pixel images are used during training.

4.1 Model Performance Analysis

The performance comparison of our tested YOLO models (YOLOv5, YOLOv8, YOLOv9, and YOLOv11) unveils various key points, including their strengths, weaknesses, and capabilities in object detection.

YOLOv5 displays an effective learning process, including an almost straight box loss decrease in the training process with a stable decreasing curve for metrics. The box_loss and cls_loss values are consistently decreased, and the mAP, precision, and recall values are improved significantly, suggesting YOLOv5's generalization capability over unseen data. However, analyzing the precision and recall graph, it can be seen that some confusion may have occurred in the middle of the learning process. Even considering these minor ups and downs, the smooth line of the curves demonstrates a very high score. But it still indicates some room for enhancement.

YOLOv8 exhibited higher performance with accuracy, precision, and recall. The model's box loss decreases consistently, and the other losses also show an almost steady curve during the training and validation stage, showing that the model learning was effective and generalized to the new data. However, some curves present some minor fluctuations, indicating some mistakes in the training process. The mAP50 and the recall curve show a steep inclination and fluctuations before steady learning, suggesting a potential improvement area.

In Fig 6, we can see that YOLOv9 showed an overall balanced performance. The smoothed line shows a steady loss decrement and increasing precision, recall, and mAP values. Although before the training process ended, the curves became stable, the curves demonstrate some steep inclined angles, which indicate some troubles in the process. Some in-depth layers might reveal the exact cause of this behavior, or this can be an anomaly that might not be seen in the retraining process. Even with these minor issues, YOLOv9 presents a promising result in fire detection.

YOLOv11 demonstrates a promising result, presented in Fig 7. Most of the performance metrics show a very good stable loss decrement, such as box loss, class loss, and other losses. The mAP50, mAP50-95, precision, and recall curves present an impressive improvement in the training process. Some minor fluctuations in the precision and recall curve can be potentially caused by the steep incline in the learning graph. Though these can be examined further, their smoothed curve already exhibits an impressive result.

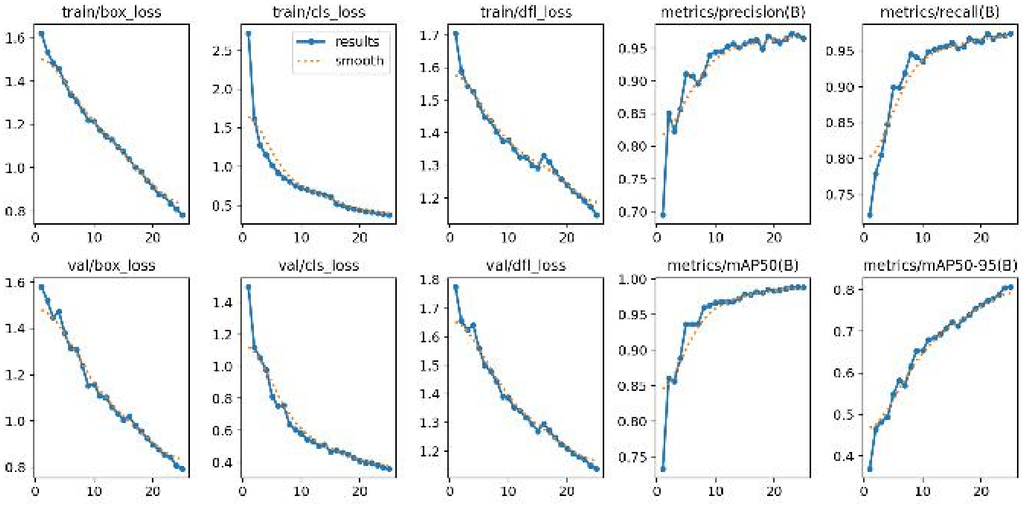


Fig 6. Training curve based on YOLOv9

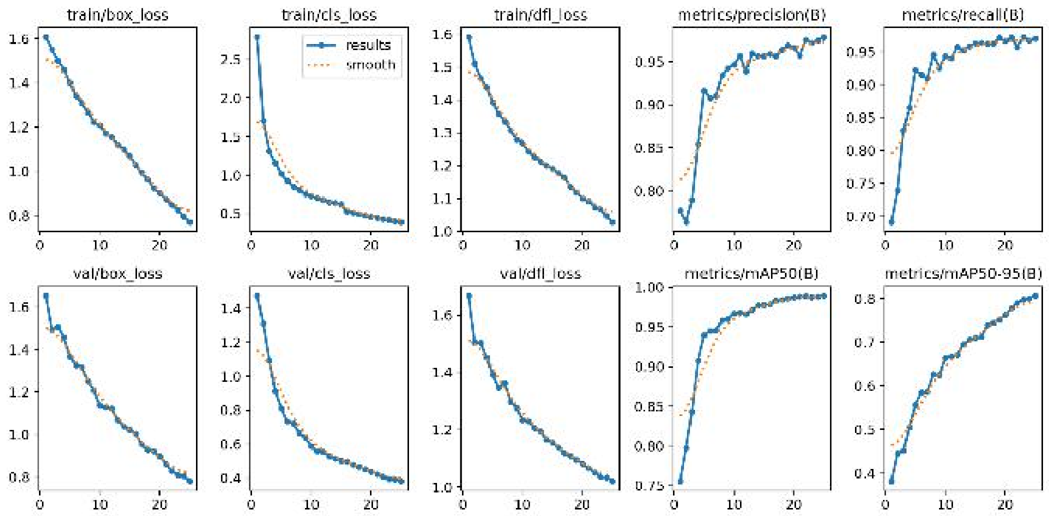


Fig 7. Training Curve Based on YOLOv11

The performance testing of various YOLO models (v5, v8, v9, and v11) for proposed fire detection was compared to assess these models' effectiveness. Our analysis was conducted on all fire classes of the dataset, shown in Table 2. During the training, metrics like Mean Average Precision (mAP) at 0.5 intersection, or mAP@50, and mAP50-95 were monitored to evaluate the model's learning capabilities. Additionally, for getting more closure understanding, we have calculated the F1-score of the model using the formula,

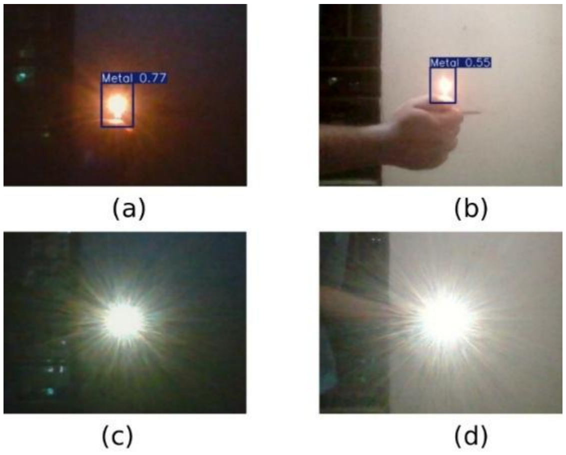
$$\text{F1 Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \qquad (1)$$

Table 2. Testing performance of YOLOv11n, v9n, v8n, v5n

Model	Epoch	Class	Precision	F1-score	MAP@0.5	MAP@0.95
YOLOv11n	25	All	97.90%	97.90%	98.90%	80.60%
YOLOv9t	25	All	96.52%	96.96%	98.81%	80.80%
YOLOv8n	25	All	97.01%	97.39%	98.99%	81.62%
YOLOv5n	25	All	97.50%	97.44%	99.05%	80.70%

From our experimental results, we can conclude that the YOLOv11n-based model demonstrated superior performance, with the highest precision, recall, and F1 score among the tested models, reflecting the highest capabilities for detecting all types of fire classes in the dataset. Its mAP@50 and mAP50-95 scores also indicate the robustness of the model over other tested models. Moreover, the YOLOv11n-based model presented a consistent improvement in performance across evaluation metrics, validating its reliability in fire detection tasks. As shown in Fig 8, the YOLOv11n-based system successfully identifies fire occurrences across various scenarios. Fig 9 presents evidence of fire detection. Fig. 9(a) and 9(b) show successful fire detection in different lighting conditions. Additionally, 9(c) and 9(d) show the model's robustness against false positives; here the model successfully ignores bright glare from a light source in dark and bright lighting conditions. including different lighting conditions, flame intensities, and background clutter highlighting the model's ability of generalization.

Fig 8. Detection sample



(a) (b)

(c) (d)

Fig 9. Visual detection results from the YOLOv11n model. (a) and (b) show detection on controlled flame sources with corresponding bounding boxes and confidence scores. (c) and (d) demonstrate the model's correct identification and ignorance of intense artificial light glare.

In Fig 10, we presented the confusion matrix of the YOLOv11n model, and in Fig 11, the confusion matrix of other tested models. From the plots, we can notice the effectiveness and the detection capability of the model across all fire classes of the dataset. With near-perfectly accurate detection in each class, this indicates a high recall rate of the model. Although it is capable of detecting all these fire class classifications, for our proposed system, we are only considering a fire/non-fire situation. So, its classification capability serves as an added strength of the model, indicating the minimizing of the false positive rate and increasing true positive anomalies.

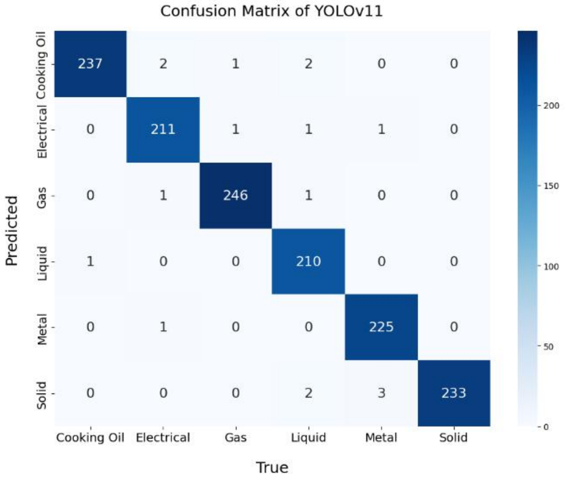
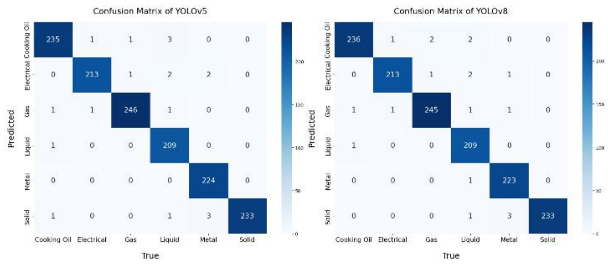
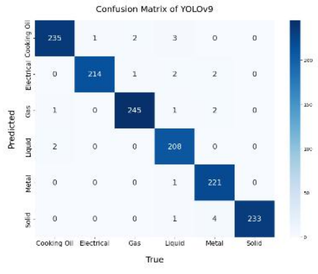


Fig 10. YOLOv11 confusion matrix



(a) Confusion Metrix for YOLOv5

(b) Confusion Metrix for YOLOv8



(c) Confusion Metrix for YOLOv9

Fig 11. Confusion matrix of YOLO v5, v8, v9

4.2 System Performance Comparison

Our proposed framework integrates both hardware and software components. As we prioritized efficiency and detection speed, our framework design was also guided by these requirements and analyzing existing systems. Within this context, temperature sensor capabilities of various systems were observed. Many studies [8] [9] [10] [11] [13] [15] were utilizing DHT-11 sensors, which operate within limited environmental ranges (0°C - 50°C) and exhibit moderate accuracy (±2°C). Even to demonstrate our proposed framework, we incorporated a more sophisticated temperature sensor, the LM35, with extended range and precision. This sensor offers a wider range (-55°C to 150°C) and higher accuracy (± 0.5°C) compared to the DHT-11. Additionally, LM35 consumes significantly less power, functioning efficiently at 60 µA, compared to DHT-11's 0.5 mA to 2.5 mA range.

In the case of ML, we have compared our model with the existing fire detection models. The comparison from Table 3 demonstrates that the proposed model surpasses earlier studies with a precision of 97.9%, the highest among the results from the chosen related works. For example, it surpasses the YOLOv3 model [23] by a significant margin of 10.8 and exhibits a slight but greater improvement of 0.8 over YOLOv8 [18], among other comparisons.

Table 3. Accuracy comparison with existing works

Study	Model	Precision
Talaat et al. [18]	YOLOv8n	97.1%
Xiang et al. [19]	TRA-YOLO	78.7%
Mamadaliev et al. [20]	ESFD-YOLOv8n	80.1%
Phan et al. [21]	FFS-YOLO	91.0%
Wang et al. [22]	CAGSA-YOLO	89.7%
Li et al. [23]	YOLOv3n	87.8%
Our Study	YOLOv11n	97.9%

4.3 False Alarm Mitigation Strategy

The system follows a structured three-layer validation approach, in which each layer is designed to support the others and ensure the misleading triggers from environmental factors, such as dust, steam, or non-fire infrared sources, do not result in unnecessary alerts. To minimize false alarms, this system implements a strategic step-by-step validation process:

1) *Smoke Detection Trigger:* The MQ2 sensor, serving as the system's first line of detection, acts as an energy-efficient trigger, being the only component of the system activated and monitoring all the time. When smoke concentration exceeds a set threshold, the system activates the next validation layer instead of immediately raising an alarm, reducing false positives caused by temporary or harmless smoke sources.

2) *Rapid Visual Analysis:* Upon smoke detection, the YOLOv11 model immediately processes the camera feed for detecting fire. This step ensures that alarms are only triggered if both smoke and visual fire evidence are present, significantly reducing the chances of false alarms caused by non-fire smoke sources. If the model detects flames and validates their presence with high confidence, the system promptly confirms a fire event and enables rapid response without waiting for further verification.
3) *Fail-Safe Temperature Validation:* If visual confirmation does not detect any fire due to obstructions or other factors, the system activates the LM35 sensor, acting as the secondary validation mechanism, to monitor temperature changes. This ensures detection in cases where the YOLOv11 model may be inconclusive due to obstructions or low visibility. A rise in temperature beyond a critical threshold serves as the final confirmation, allowing the system to find the difference between harmless visual anomalies and real fire outbreaks.

The system's architecture optimizes resource utilization by keeping the YOLOv11 model and LM35 sensor idle until needed and relying on the LM35 sensor as a backup validation mechanism. This helps in conserving energy and reducing unnecessary computational load.

Altogether, in this strategy, Fire detects a fast response time and at least 2-layer validation. When smoke is detected and YOLOv11 detects it, then the fire is triggered. By following this principle, the system can easily filter out most of the false alarm calls. And as the YOLOv11 model is activated after the smoke detection, intentional fire or fire images to confuse the system can also be prevented.

5. Conclusion

This paper presents a novel three-layer fire detection and validation method designed to overcome the limitations of traditional fire detection methods. By integrating smoke sensors, computer vision-based fire detection, and temperature-based cross-validation, the system demonstrates exceptional potential for reducing false alarms while maintaining rapid and accurate capabilities of detection. Our proposed framework's core innovation lies in the layered hierarchical validation logic. The implementation results highlight the efficiency of this multimodal approach, balancing reliability and optimization of resources effectively. The YOLOv11 model achieved impressive performance metrics, including a precision of 0.979, showing off robust fire detection across various types of fire. The hierarchical validation process makes sure of timely detection of actual fire events, offering a comprehensive and scalable solution. This conceptual evaluation underlines the system's potential to enhance fire safety in indoor environments. Thus, this system can contribute to minimizing false alarms through a three-layer verification process while ensuring cost-effective, wireless, and low-power fire detection. It can seamlessly integrate with existing CCTV networks and remain scalable for various indoor environments. With further advancements and IoT integration, the proposed system can play a crucial role in enabling safer and more reliable fire detection mechanisms for diverse real-world applications.

In the future, some areas can be explored further. This framework can be further validated with high-end industry-grade sensors (e.g., photoelectric detectors). Insights from studies suggest that deploying the system on edge devices like NVIDIA Jetson could improve robustness and real-time performance [33] . Additionally, a photoelectric smoke sensor-based system can be faster and more accurate [34] . An IoT cloud-based communication system would enable remote monitoring and an advanced alert system using a Messenger cloud [35] . Finally, low-power hardware could enhance energy efficiency, making the system suitable for continuous use.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Conceptualization: Kausar Ahmed, S.M. Jobair Hossain; methodology: Kausar Ahmed, Bibhor Gomes. S.M. Jobair Hossain, Emran Khan Musa; validation, Kausar Ahmed, Emran Khan Musa; Formal Analysis: Kausar Ahmed, Bibhor Gomes; Investigation: Emran Khan Musa, S.M. Jobair Hossain; Resources: Kausar Ahmed; Data Curation: Emran Khan Musa, S.M. Jobair Hossain; Writing—Original Draft Preparation: Kausar Ahmed, Bibhor Gomes and Fuyad Hasan Bhoyan; writing—review and Editing: Md Humaion Kabir Mehedi, Kausar Ahmed, Bibhor Gomes; Visualization: Kausar Ahmed; Supervision: Jia Uddin; Project Administration: Kausar Ahmed. All authors have read and agreed to the published version of the manuscript.

Funding

This research is funded by Woosong University Academic Research 2026.

References

[1] "Residential fire estimate summaries (2014-2023)." Accessed: Sep. 04, 2025. [Online]. Available: https://www.usfa.fema.gov/statistics/residential-fires/

[2] Y. Ahn, H. Choi, and B. S. Kim, “Development of early fire detection model for buildings using computer vision-based CCTV,” Journal of Building Engineering, vol. 65, p. 105647, 2023, doi: https://doi.org/10.1016/j.jobe.2022.105647.

[3] J. Pincott, P. W. Tien, S. Wei, and J. K. Calautit, “Indoor fire detection utilizing computer vision-based strategies,” Journal of Building Engineering, vol. 61, p. 105154, Dec. 2022, doi: 10.1016/J.JOBE.2022.105154.

[4] H. Fang, M. Xu, B. Zhang, and S. M. Lo, “Enabling fire source localization in building fire emergencies with a machine learning-based inverse modeling approach,” Journal of Building Engineering, vol. 78, p. 107605, Nov. 2023, doi: 10.1016/J.JOBE.2023.107605.

[5] C. J. Hsiao and S. H. Hsieh, “Real-time fire protection system architecture for building safety,” Journal of Building Engineering, vol. 67, 2023, doi: 10.1016/j.jobe.2023.105913.

[6] A. Morchid, Z. Oughannou, R. El Alami, H. Qjidaa, M. O. Jamil, and H. M. Khalid, “Integrated internet of things (IoT) solutions for early fire detection in smart agriculture,” Results in Engineering, vol. 24, p. 103392, Dec. 2024, doi: 10.1016/J.RINENG.2024.103392.

[7] T. Islam, H. A. Rahman, and M. A. Syrus, “Fire detection system with indoor localization using ZigBee based wireless sensor network,” 2015 4th International Conference on Informatics, Electronics and Vision, ICIEV 2015, Nov. 2015, doi: 10.1109/ICIEV.2015.7334000.

[8] Y. C. See and E. X. Ho, “IoT-Based Fire Safety System Using MQTT Communication Protocol,” International Journal of Integrated Engineering, vol. 12, no. 6, pp. 207–215, 2020, doi: 10.30880/IJIE.2020.12.06.024.

[9] I. Ehsan et al., “Internet of Things-Based Fire Alarm Navigation System: A Fire-Rescue Department Perspective,” Mobile Information Systems, vol. 2022, no. 1, p. 3830372, Jan. 2022, doi: 10.1155/2022/3830372.

[10] M. Logeshwaran and J. J. Jeya Sheela, “Designing an IoT based Kitchen Monitoring and Automation System for Gas and Fire Detection,” Proceedings - 6th International Conference on Computing Methodologies and Communication, ICCMC 2022, pp. 346–353, 2022, doi: 10.1109/ICCMC53470.2022.9754118.

[11] S. Debnath, J. Gautam, and S. K. Rai, “IoT Based Smart Home and Office Fire Notification Alert System,” 2024 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation, IATMSI 2024, 2024, doi: 10.1109/IATMSI60426.2024.10503336.

[12] S. K. Mekni, “Design and Implementation of a Smart fire detection and monitoring system based on IoT,” Proceedings - 2022 4th International Conference on Applied Automation and Industrial Diagnostics, ICAAID 2022, 2022, doi: 10.1109/ICAAID51067.2022.9799505.

[13] X. Feng, C. Wang, F. Anwar, R. I. Boby, and M. M. Rashid, “Design and Application of Home Fire Alarm System,” J Phys Conf Ser, vol. 1871, no. 1, p. 012125, Apr. 2021, doi: 10.1088/1742-6596/1871/1/012125.

[14] A. Mahgoub, N. Tarrad, R. Elsherif, A. Al-Ali, and L. Ismail, “IoT-based fire alarm system,” Proceedings of the 3rd World Conference on Smart Trends in Systems, Security and Sustainability, WorldS4 2019, pp. 162–166, Jul. 2019, doi: 10.1109/WORLDS4.2019.8904001.

[15] M. Hariprasath, U. Prakash, M. V. Raja, and S. R. V. Bharathi, “Fire Alert System for Home using IoT,” Proceedings of 2024 International Conference on Science, Technology, Engineering and Management, ICSTEM 2024, 2024, doi: 10.1109/ICSTEM61137.2024.10560639.

[16] N. Gupta, P. Deshpande, J. Diaz, S. Jangam, and A. Shirke, “F-ALERT: EARLY FIRE DETECTION USING MACHINE LEARNING TECHNIQUES,” Int. J. of Electronics Engineering and Applications, vol. 9, no. 3, pp. 34–43, doi: 10.30696/IJEEA.IX.III.2021.34-43.

[17] J. Pincott, P. W. Tien, S. Wei, and J. Kaiser Calautit, “Development and evaluation of a vision-based transfer learning approach for indoor fire and smoke detection,” Building Services Engineering Research and Technology, vol. 43, no. 3, pp. 319–332, May 2022, doi: 10.1177/01436244221089445.

[18] F. M. Talaat and H. ZainEldin, “An improved fire detection approach based on YOLO-v8 for smart cities,” Neural Comput Appl, vol. 35, no. 28, pp. 20939–20954, Oct. 2023, doi: 10.1007/S00521-023-08809-1/FIGURES/9.

[19] S. Xiang, S. Yin, G. Yu, X. Xu, and L. Yu, “Factory Fire Detection using TRA-YOLO Network,” Proceedings of the 36th Chinese Control and Decision Conference, CCDC 2024, pp. 3777–3782, 2024, doi: 10.1109/CCDC62350.2024.10588005.

[20] S. Smith et al., “ESFD-YOLOv8n: Early Smoke and Fire Detection Method Based on an Improved YOLOv8n Model,” Fire 2024, Vol. 7, Page 303, vol. 7, no. 9, p. 303, Aug. 2024, doi: 10.3390/FIRE7090303.

[21] D. T. Phan, K. H. Yap, K. Garg, and B. S. Han, “Vision-Based Early Fire and Smoke Detection for Smart Factory Applications Using FFS-YOLO,” 2023 IEEE 25th International Workshop on Multimedia Signal Processing, MMSP 2023, 2023, doi: 10.1109/MMSP59012.2023.10337640.

[22] X. Wang, L. Cai, S. Zhou, Y. Jin, L. Tang, and Y. Zhao, “Fire Safety Detection Based on CAGSA-YOLO Network,” Fire 2023, Vol. 6, Page 297, vol. 6, no. 8, p. 297, Aug. 2023, doi: 10.3390/FIRE6080297.

[23] P. Li and W. Zhao, “Image fire detection algorithms based on convolutional neural networks,” Case Studies in Thermal Engineering, vol. 19, p. 100625, Jun. 2020, doi: 10.1016/J.CSITE.2020.100625.

[24] U. Dampage, L. Bandaranayake, R. Wanasinghe, K. Kottahachchi, and B. Jayasanka, “Forest fire detection system using wireless sensor networks and machine learning,” Sci Rep, vol. 12, no. 1, pp. 1–11, Dec. 2022, doi: 10.1038/s41598-021-03882-9

[25] B. K. Mohammed, M. B. Mortatha, A. S. Abdalrada, and H. T. H. Salim ALRikabi, “A comprehensive system for detection of flammable and toxic gases using IoT,” Periodicals of Engineering and Natural Sciences, vol. 9, no. 2, pp. 702–711, 2021, doi: 10.21533/PEN.V9I2.1894.

[26] “LM35 LM35 Precision Centigrade Temperature Sensors,” 1999, Accessed: Sep. 04, 2025. [Online]. Available: www.ti.com

[27] “ESP32-WROOM-32D & ESP32-WROOM-32UDatasheet Version 2.6,” Sep. 2025. [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32d_esp32-wroom-32u_datasheet_en.pdf

[28] “Fire Source Detection Model > Overview.” Accessed: Sep. 04, 2025. [Online]. Available: https://universe.roboflow.com/university-of-the-philippines-cebu/fire-source-detection

[29] R. Khanam, M. Hussain, R. Hill, and P. Allen, “A Comprehensive Review of Convolutional Neural Networks for Defect Detection in Industrial Applications,” IEEE Access, vol. 12, pp. 94250–94295, 2024, doi: 10.1109/ACCESS.2024.3425166.

[30] “Ultralytics YOLO11 - Ultralytics YOLO Docs.” Accessed: Sep. 04, 2025. [Online]. Available: https://docs.ultralytics.com/models/yolo11/

[31] R. Khanam and M. Hussain, “YOLOv11: An Overview of the Key Architectural Enhancements,” Oct. 2024, Accessed: Sep. 04, 2025. [Online]. Available: https://arxiv.org/pdf/2410.17725

[32] A. Komatsuzaki, “One Epoch Is All You Need,” Jun. 2019, Accessed: Sep. 04, 2025. [Online]. Available: https://arxiv.org/pdf/1906.06669

[33] S. Cho, C. Seo, D.-J. Shin, and J.-J. Kim, “A Deep Learning Framework Performance Evaluation to Use YOLO in Nvidia Jetson Platform,” Applied Sciences 2022, Vol. 12, Page 3734, vol. 12, no. 8, p. 3734, Apr. 2022, doi: 10.3390/APP12083734.

[34] W. Choi and I. Y. Jung, “Multi-Sensor Photoelectric Fire Alarm Device Implementation for Early Fire Detection in Campsites,” Applied Sciences 2024, Vol. 14, Page 9965, vol. 14, no. 21, p. 9965, Oct. 2024, doi: 10.3390/APP14219965.

[35] R. Muhendra and A. Amin, “Real-Time Monitoring: Development of Low Power Fire Detection System for Dense Residential Housing Based on Internet of Things (IoT) and Cloud Messenger,” Scientific Journal of Informatics, vol. 8, no. 2, pp. 202–212, Nov. 2021, doi: 10.15294/SJI.V8I2.30811.

[36] S. Rahman, M. Z. Uddin, M. A. Hossain, and M. S. Islam, “Real-time smoke and fire detection using you only look once v8-based advanced computer vision and deep learning,” Indonesian Journal of Applied Artificial Intelligence Systems, 2024. [Online]. Available: https://ijaas.iaescore.com/index.php/IJAAS/article/view/21220

Biographies

Kausar Ahmed is currently a third-year Computer Science and Engineering (BSc in CSE) student at the University of Liberal Arts Bangladesh (ULAB). His current works include Machine Learning, Artificial Intelligence and its impact. His research interests include RAG, NLP, Generative Al, and Machine Learning. He can be contacted by email: kausar.ahmed.cse@ulab.edu.bd

Bibhor Regan Gomes is currently pursuing a B.Sc. in Computer Science and Engineering (CSE) at the University of Liberal Arts Bangladesh (ULAB). He demonstrates a strong interest in computer vision-based projects and is eager to explore the world of NLP. His contact email is: regan.gomes.cse@ulab.edu.bd

S.M. Jobair Hossain is currently pursuing a B.Sc. in Computer Science and Engineering (CSE) at the University of Liberal Arts Bangladesh (ULAB). He specializes in machine learning. Alongside ML, he keeps a strong interest in the field of mathematics. He also shows key interests in network security and cybersecurity. His email is: jobair.hossain.cse@ulab.edu.bd

Emran Khan Musa is currently pursuing a B.Sc. in Computer Science and Engineering (CSE) at the University of Liberal Arts Bangladesh (ULAB). He specializes in machine learning. His primary research interest lies in deep learning and advanced computer networks. He is also actively involved in social work. He can be contacted by email: emran.khan.cse@ulab.edu.bd

Fuyad Hasan Bhoyan is currently pursuing a B.Sc. in Computer Science and Engineering (CSE) at the University of Liberal Arts Bangladesh (ULAB). He demonstrates a strong interest in emerging technologies within the field of computer vision and has contributed to several research projects. He can be contacted by email: fuyadhasanbhoyan@gmail.com.

Md Humaion Kabir Mehedi (Member, IEEE) received a B. Sc. in Computer Science from BRAC University, Dhaka, in 2022. He is currently studying M.Sc. in Computer Science and Engineering at BRAC University. He has more than 3 years of research experience in Advanced Artificial Intelligence, Machine Learning and Deep Learning. Particularly in the areas of applied machine learning, medical image processing, computer vision, and natural language processing. He has more than 20 publications in prestigious book chapters and conference proceedings.

Jia Uddin is an Associate Professor in Artificial Intelligence and BigData Department at Woosong University, South Korea. Before that, he was an associate professor at BRAC University. He completed his Ph.D. from the University of Ulsan and MS from Blekinge Institute of Technology, Sweden. His research interests are Signal Processing, Industrial Fault Diagnosis, and Computer Vision. He has served as the Chair and a reviewer of numerous renowned conferences.