Journal of Ubiquitous Computing and Communication Technologies (ISSN: 2582-337X)
www.irojournals.com/jucct/

Ubiquitous Monitoring and Alert Fire Detection System in Data Center using Logistic Regression

Belal K. ELFarra[1], Mamoun A. A. Salha[2]

Islamic University of Gaza, Gaza Strip, Palestine.

E-mail: [1]fbelal@iugaza.edu.ps, [2]masalha@iugaza.edu.ps

Abstract

Data centers are the core of any organization. They house its data, host its services, ensure business continuity, and serve as the bedrock of decision-making. Therefore, round-the-clock operation is essential to maintaining the continuity of services provided by an organization. This requires robust, standardized security systems to restrict exposure to threats to infrastructure and data. Yet, existing data center alarm systems do not have robust monitoring and reporting capabilities that generate premature action on high-priority alerts. In this research paper, we propose a data-driven automated method to optimize fire detection in data centers through the use of machine learning algorithms and sensor networks to inspect large amounts of data and detect patterns of fires. The installation of the proposed system includes the employment of an ESP32 development board to handle real-time data and wireless communication with different sensors, such as smoke detectors, temperature, gas, and motion sensors, in order to facilitate end-to-end monitoring. We have achieved an intelligent monitoring system through the employment of machine learning for handling sensor data, adaptively setting thresholds, and initiating corresponding actions ranging from sending alerts to alarm sounds. The system is highly accurate (≈97–98%) with strong confidence-based filtering and minimal computational cost, which renders it ideally suited for real-time indoor use. The method provides consistent detection through diversified materials (carton, cloth, electrical), filling the gap in situations where visual information is not necessarily accurate or reliable. Experimental results demonstrate the efficiency of the system

DOI: https://doi.org/10.36548/jucct.2025.3.003
Received: 08.08.2025, received in revised form: 10.09.2025, accepted: 27.09.2025, published: 07.10.2025

in the timely notification of fire cases, bearing witness to its practical application over vision-based or simulation-based approaches.

Keywords: Ubiquitous, Early Fire Detection, Data Center, Logistic Regression, Machine Learning, Fire Detection Systems.

1. Introduction

Unlike open flames, smoldering fires progress slowly and release less heat, making them harder for conventional alarms to detect in time. For this reason, it has become crucial to develop fire extinguishing systems that can provide reliable early detection of fires [1]. Because of their critical role, data centers demand specialized protection strategies that go beyond traditional fire alarms. Therefore, these centers are designed with the highest levels of protection and ensure continuity of service, in accordance with agreed-upon security systems and standards, including continuous monitoring and strong encryption to protect sensitive data from falling into unauthorized hands [1][2].

Due to the vital importance of data centers, it is crucial to emphasize safeguarding against hazards like fires, floods, and extreme temperatures. Consequently, data centers employ sophisticated systems to identify these hazards, such as smoke detectors, heat sensors, and energy monitors. Protection systems are created to provide early alerts and implement the required actions to control and extinguish fires. This entails establishing efficient fire prevention protocols and adequately prepared emergency response strategies [3][4].

Data centers use fire alarm systems that utilize a network of various sensors designed to monitor different aspects of a fire and enable continuous monitoring and data collection related to environmental conditions or specific phenomena in the area. This system is typically used to monitor temperature levels and trigger an immediate alarm in the event of a sudden or abnormal rise, indicating a potential fire.

Recent advances in IoT and ubiquitous computing have made it possible to embed sensors directly into infrastructure, enabling continuous monitoring and smarter fire detection. When context awareness is built into a data centers fire alarm system, operators and responders can better understand the exact conditions inside the facility. This leads to improvement in overall safety.

Fire alarm systems in data centers are typically integrated with server and equipment monitoring systems, enabling the performance and health of servers, power distribution units, and other critical equipment to be monitored. This integration enables appropriate safety measures to be taken immediately upon detection of abnormalities, such as overheating or electrical faults, that could lead to a fire.

This research focuses on developing effective fire detection systems that can provide early and accurate detection while minimizing false alarms. Different approaches, including chemical gas sensors, machine learning, fuzzy logic, and multi-sensor systems, are being explored to improve the reliability and performance of fire detection technologies.

The subsequent sections of this paper are structured as follows: The following section focuses on the literature review, starting with a discussion of the problem statement, then proceeding to an examination of related studies. Next, we explore the methodology, beginning with the suggested concept and subsequently introducing the proposed algorithm. The section on experiments and results displays the findings of our study. In conclusion, we summarize the main findings and contributions of our research, while also emphasizing possible avenues for future investigation.

2. Background

Applying machine learning to sensor grids leads to smart systems that enhance fire and smoke detection. Algorithms analyze data, recognize patterns, and differentiate incidents from normal variations by allowing dynamic threshold adjustments for an efficient and effective alarm system. This improves alarm triggering, detection accuracy, and reliability.

The ESP32 enhances fire alarm systems by overcoming connectivity challenges and enabling effective real-time detection. It supports wireless communication, integrates with various sensors, and performs on-board data processing. With its versatile GPIO pins and ADC, the ESP32 seamlessly interfaces with different sensor types. It can analyze sensor data using machine learning algorithms, promptly detecting fire or smoke incidents and triggering appropriate response actions such as activating alarms or notifying authorities. This context-aware system improves the overall reliability and responsiveness of fire alarm systems.

To overcome the communication management of the overall system, ESP-NOW is initialized as the main protocol. It offers a lightweight, low-power, and efficient communication solution for local area network applications involving ESP32 devices. It enables direct peer-to-peer communication between ESP32 devices without the need for a traditional Wi-Fi network infrastructure. This is particularly useful in scenarios where sensors need to communicate quickly and efficiently within a limited range.

To ensure that data center supervisors are fully aware of any unusual events, a redundant alerting mechanism has been activated. This can include a combination of audible alarms, visual indicators (such as flashing lights), and electronic notifications sent to designated personnel via email, SMS, or push notifications on their mobile devices. Redundant alerting mechanisms help ensure that fire incidents are communicated promptly to the necessary individuals.

As a first step, data preprocessing of the collected data performs necessary operations and transformations on the raw sensor data to prepare it for further analysis and interpretation. Data cleaning involves handling missing values, outliers, and erroneous data. Missing values can be imputed or removed depending on the context, outliers can be identified and dealt with appropriately, and any erroneous data can be corrected or discarded. The temperature sensor (LM35 & DHT11) operates based on the fundamental principle of voltage measurement across its terminals. As the temperature increases, the voltage across the sensor also increases accordingly. Specifically, for every one-degree change in temperature while the sensor is in operation, there is a corresponding voltage change of approximately 10 millivolts. This relationship between temperature and voltage variation forms the basis of the temperature sensor's functionality, formula (1) converts the raw analog value from the temperature sensor into a temperature value in Celsius degrees considering the reference voltage used.

$$Temreture\_Value = Sensor\_Value * 1100 / (1024 * 10) \quad (1)$$

Logistic regression is used to classify data points; the model learns the coefficients or parameters that best fit the training data using a maximum likelihood estimation method. These coefficients are then used to make predictions on new, unseen data by calculating the log-odds of the outcome and transforming them into probabilities using the logistic function. Each sensor reading is represented as a feature vector or input data point. Let's denote the feature vector as x, where $x = [x_1, x_2, x_3, ..., x_n]$ represents n sensor readings (e.g., smoke, temperature, flame intensity). Each xi corresponds to a specific sensor measurement. A labeled dataset is needed to train the machine learning model. The dataset consists of pairs (x, y), where x is the input feature vector, and y is the corresponding label indicating whether it represents a fire (y = 1) or a non-fire (y = 0) condition.

The logistic regression equation models the relationship between predictor variables (x_1, x_2, x_3, ..., x_n) and the log-odds (logit) of the binary outcome (y).

$$logit(p) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n \quad (2)$$

where logit(p) represents the log-odds of the probability (p) of the outcome, and β_0, β_1, β_2, ..., β_n are the coefficients or weights associated with each predictor variable, and x_1, x_2, ..., x_n are the corresponding values of the predictor variables.

To obtain the probability (p) of the binary outcome based on the logit, the logistic function (sigmoid function) is applied:

$$p = 1 / (1 + exp(-logit(p))) \quad (3)$$

In binary classification, a decision threshold is used to determine the boundary between the two classes. If the predicted probability (p) exceeds the threshold, the outcome is classified as the positive class (fire incident); otherwise, it is classified as the negative class (non-fire incident).

2.1 Problem Statement

Data centers face serious threats from fires, which can result in infrastructure damage and data loss. Traditional alarm systems in data centers lack comprehensive monitoring and suffer from manual calibration issues, leading to delayed responses and false alarms. We propose an automated, data-driven approach to calibrate alarm systems and monitor environmental conditions in data centers. Our goal is to enhance fire detection, provide early alerts, and minimize damage.

3. Literature Review

Lately, there has been increasing interest in offering fire systems capable of delivering dependable early fire detection [1]. Nevertheless, the majority of traditional fire science research relies on numerical experiments like CFD simulations [2][3][4] rather than on real experimentation.

Numerous studies regarding early fire detection have been published in the literature. Solorazano et al. suggested employing chemical gas sensors for fire detection rather than smoke-based systems to enable early fire identification [6][7]. This method primarily relies on the observation that certain fire types emit volatiles prior to smoke [8].

Moreover, there has been a growing interest in creating machine learning solutions utilizing actual fire datasets [9][10]. Nonetheless, the overwhelming majority of publicly accessible datasets concerning early fire detection research rely entirely on image data [11][12][13][14]. L. Wu et al. utilized temperature, smoke levels, and carbon dioxide data to create an early warning algorithm based on a back-propagation neural network that predicts the likelihood of fire occurrence, employing the National Institute of Standards and Technology (NIST) dataset for their research [10].

A fire detection algorithm utilizing fuzzy logic has been developed, applying temperature sensors to assess the direction of a fire emergency [15]. Although the utilized sensor nodes can regularly gather ambient temperature, interact with neighboring sensors, and retain information about neighbors such as Node ID and coordinates, the temperature sensor alone is insufficient for detecting fire outbreaks [16]. A smart Fire Monitoring and Warning System (FMWS) utilizing Fuzzy Logic is introduced to detect the actual presence of hazardous fire and notify the Fire Management System (FMS). The research explores the creation and implementation of a fuzzy logic fire monitoring system that transmits a notification message, a fascinating phenomenon that can be observed by WSN is fire occurring indoors or outdoors [16].

A study examines two fuzzy logic methods, incorporating temporal aspects, for overseeing and assessing fire confidence to enhance and decrease the quantity of rules necessary for making accurate decisions [17]. The research posits that this decrease might diminish sensor operations without significantly affecting operational quality and enhance battery longevity, thereby improving the efficiency, resilience, and cost-effectiveness of the sensing network. It utilizes fuzzy logic on the data gathered from sensors by transmitting it to the cluster head through an event detection mechanism. Several sensors are employed to identify the likelihood of fire and its direction. Every sensor node comprises several sensors that detect temperature, humidity, light, and CO levels to assess fire probability and the azimuth angle to determine the fire's direction [17][18][19].

An experiment is conducted utilizing the prototype fire detection equipment integrated with an IoT link to enhance the monitoring of fire hotspots. Employing fuzzy logic reduces the occurrence of false alarms from fire detection systems. The prototype serves as a learning tool for high school students specializing in computer engineering and networking [21]. It creates a WiFi ESP8266 board that is compatible with Arduino IDE [22].

Currently, the primary fire alarm systems depend on smoke detection, which entails recognizing airborne solid and liquid particles as well as gases emitted during pyrolysis or combustion [23]. Smoke detectors primarily utilize two methods: ionization sensors and photoelectric sensors. Ionization detectors utilize a radioactive material that emits alpha particles to ionize air molecules. When smoke is detected, the ions engage with smoke particles, diminishing the intensity of the electric circuit and activating the alarm. Photoelectric sensors, conversely, employ a light source and receiver to gauge light scattering triggered by smoke particles within the chamber. No matter the detection principle, the alarm triggers when the signals surpass set thresholds

Research analyzing photoelectric and ionization fire alarms under controlled settings has shown variations in sensitivity, response time, and dependability [24]. Ionization alarms usually react more quickly to open flame fires, whereas photoelectric alarms show quicker responses and greater sensitivity to smoldering fires. Different elements, like particle size, affect how well the detectors' function. For instance, photoelectric sensors quickly capture larger particles from smoldering fires, whereas ionization sensors detect smaller particles from flaming fires more rapidly. The tests further showed that smoldering fires produce more smoke particles compared to flame fires, and photoelectric alarms are capable of sensing certain smoldering fires that ionization alarms miss.

Nonetheless, smoke detectors encounter difficulties in differentiating between combustion particles and non-combustion particles, including water vapor, dust, and specific actions like cooking or cigarette smoke [25][26]. As a result, both photoelectric and ionization alarms can produce false alarms, causing occupants to disregard or deactivate the alarm alerts. To improve the accuracy of fire detection, extra sensors can be added. For example, detecting carbon monoxide (CO) can assist in eliminating false alarms triggered by common nuisance situations where CO is not produced [27]. Gas-based systems, in contrast to smoke-based alarms, necessitate several sensors or multi-criteria methods, as well as advanced data processing algorithms, to reduce false alarms and effectively identify fires.

4. Methodology

The goal of this paper is to identify signs of fire at an initial stage before they become critical or dangerous. This is done by simulating early fire detection using logistic regression. Logistic regression was selected for its simplicity, interpretability, and low computational overhead, enabling real-time processing on low-cost embedded platforms like the ESP32. Despite its linear nature, careful multi-sensor feature fusion allows it to achieve high accuracy across multiple materials, while its probabilistic outputs support robust confidence-based alerting. Compared to complex models like CNNs or LSTMs, logistic regression provides comparable performance on structured sensor data without requiring large datasets, high-end hardware, or extensive hyperparameter tuning, making it well-suited for practical deployment in indoor data center fire detection.

The study utilizes a real Indoor Laboratory dataset [28] which is composed of 8 CSV files, each corresponding to different experimental scenarios and materials such as carton, clothing, and electrical components. To build a unified dataset, all individual files were merged into a single structured DataFrame. This ensured that the model was trained and evaluated on the complete distribution of fire and non-fire conditions across different materials. The selected features for analysis included humidity, temperature, the MQ135 gas sensor, TVOC, and eCO2 level (see figure 1), while the target variable was the binary fire class (0 = no fire, 1 = fire). Prior to model training, the feature values were standardized using the StandardScaler technique to normalize variations in scale across different sensors. This preprocessing step ensured that no single feature dominated the learning process and improved the stability of the logistic regression model.

Our algorithm involves loading and preprocessing the data, splitting it into training and testing sets, creating a logistic regression model, fitting the model to the training data, predicting the target variable for the test set, and evaluating the model's performance using appropriate metrics. The logistic regression model is trained on the provided datasets to detect and predict fire incidents. The paper emphasizes the importance of early fire detection in preventing the spread of fire and minimizing damage, highlighting the role of advanced sensors, intelligent algorithms, and real-time monitoring in early fire detection systems. A practical system is developed after constructing the model, showcasing its effectiveness in real-world scenarios and validating its performance for accurate early fire detection. The methodology handled multicollinearity by normalizing all sensor features with

StandardScaler to prevent any single correlated sensor from dominating, by leveraging multiple sensor inputs rather than depending on one predictor, and by applying confidence-based filtering to minimize the impact of uncertain predictions caused by overlapping or correlated sensor data.

This system utilizes the same components as the dataset and leverages the model's insights to provide timely warnings and interventions, effectively preventing fire escalation and minimizing damage. This system works by setting up two levels of fire alerts. The first level is triggered when the model detects unusual conditions in the data center. At this stage, the system immediately sends a warning to the supervisor via mobile and email. If the situation worsens and crosses to the second level, a loud alarm is activated inside the data center, prompting quick action to reduce fire risks. By using this dual threshold approach, the system provides early warnings as well as critical alerts, making fire detection and response faster and more effective. The following pseudocode illustrates the process procedures. This pseudocode assumes that we have already imported the required libraries, loaded the training data into variables X and Y, and performed any necessary preprocessing steps. In this work, the dataset was divided into training and testing subsets using scikit-learn's built-in train_test_split function, ensuring proper evaluation of the model. It then creates an instance of the logistic regression model using LogisticRegression().

Pseudocode

---

1. Import the necessary libraries:

sklearn.linear_model, LogisticRegression, sklearn.model_selection, train_test_split, sklearn.metrics, accuracy_score, recision_score, confusion_matrix, and numpy.

2. Load the training data (features and target variables) into variables X and Y:

X = # load the feature data

Y = # load the target variable

3. Preprocess the data:

Perform feature scaling and handle missing values

4. Split the data into training and testing sets:

X_trn, X_ tst, Y_ trn, Y_tst = train_test_split(X, Y, test_size=0.2, random_state=42)

5. Create an instance of the logistic regression model:

model = LogisticRegression()

6. Fit the model to the training data:

model.fit(X_ trn, Y_ trn)

7. Predict the target variable for the test set:

Y_pred = model.predict(X_ tst)

8. Evaluate the model's performance using appropriate metrics:

accuracy = accuracy_score(Y_ tst, Y_pred)

precision = precision_score(Y_ tst, Y_pred)

confusion_mat = confusion_matrix(Y_ tst, Y_pred)

The model is trained on the training data using the fit method, and the target variable is predicted for the test set using the predict method. Finally, the model's performance is evaluated using a metric such as accuracy. A confidence threshold of 0.7 was applied, meaning that model predictions were only considered reliable when the highestclass probability exceeded 70%. Predictions below this threshold were treated as uncertain, ensuring that the system minimizes false alarms and prioritizes high-confidence fire detection. We can tune hyperparameters and save the trained model for future use.

An algorithm that outlines the steps involved in communication between a transmitter and a receiver using the ESP-NOW protocol:

1. Initialize the required libraries for the ESP-NOW protocol and Wi-Fi.
2. Specify the MAC address of the receiver device.
3. Define a structure that represents the data to be sent. Include fields for the necessary data types.
4. Implement a callback function that is triggered when data is sent. This function can handle the status of packet delivery.
5. Set up the ESP-NOW protocol by initializing it and registering the callback function for data sent events.
6. Add the receiver device as a peer using its MAC address.
7. Read the analog value from the temperature sensor or any other relevant sensor.
8. Prepare the data structure with the necessary data to be sent.
9. Send the data to the receiver device using the ESP-NOW protocol.

10. Handle the result of the data sending process. Print a success or failure message based on the result.
11. Introduce a delay before sending the next set of data.

5. Experimental and Results

To validate our concept, we developed a prototype of the system. Our efforts were divided into three phases: hardware implementation, software coding, and integration.

5.1 Indoor Laboratory Fire Dataset

The Real Indoor Laboratory Fire Dataset by [28] includes time-series data recorded in 8 controlled fire tests conducted under laboratory settings. All the experiments utilised various fire sources, i.e., 4 runs with electric fire, 2 runs with cardboard boxes, and 2 runs with clothes. The temperature, humidity, MQ139, TVOC, and eCO2 readings for every experiment were recorded from the start of the fires until the alarms started or rang for the fire. Hardware and software requirement

The experiments were conducted using a Windows 10 Education OS with an Intel Core i5-10500H processor speed of 2.50GHz and 16 GB of RAM, along with Python version 3.8. The datasets used for testing were synthetic, two-dimensional in nature and of varying sizes according to real clusters.

The following parameters were employed to generate the dataset:

- **Two ESP32 Development Boards:** These were the main microcontrollers of the system and established a communication channel with an ESP-NOW-based Wi-Fi module.
- **DHT11 Sensor:** This was used to detect temperature and humidity.
- **LM35 Sensor:** It provided the analog output proportional to the temperature.
- **MQ139 Gas Sensor:** This was utilized to detect specific gases in the air through changes in electrical conductivity. It is commonly used in air quality monitoring and gas detection in various applications.
- **TVOC:** This refers to the total concentration of volatile organic compounds in the air. Volatile organic compounds can originate from various sources and may affect indoor air quality.

- **eCO2:** This is an equivalent air concentration of carbon dioxide. It provides a general overview of the air quality and may be used to investigate indoor levels of contamination.

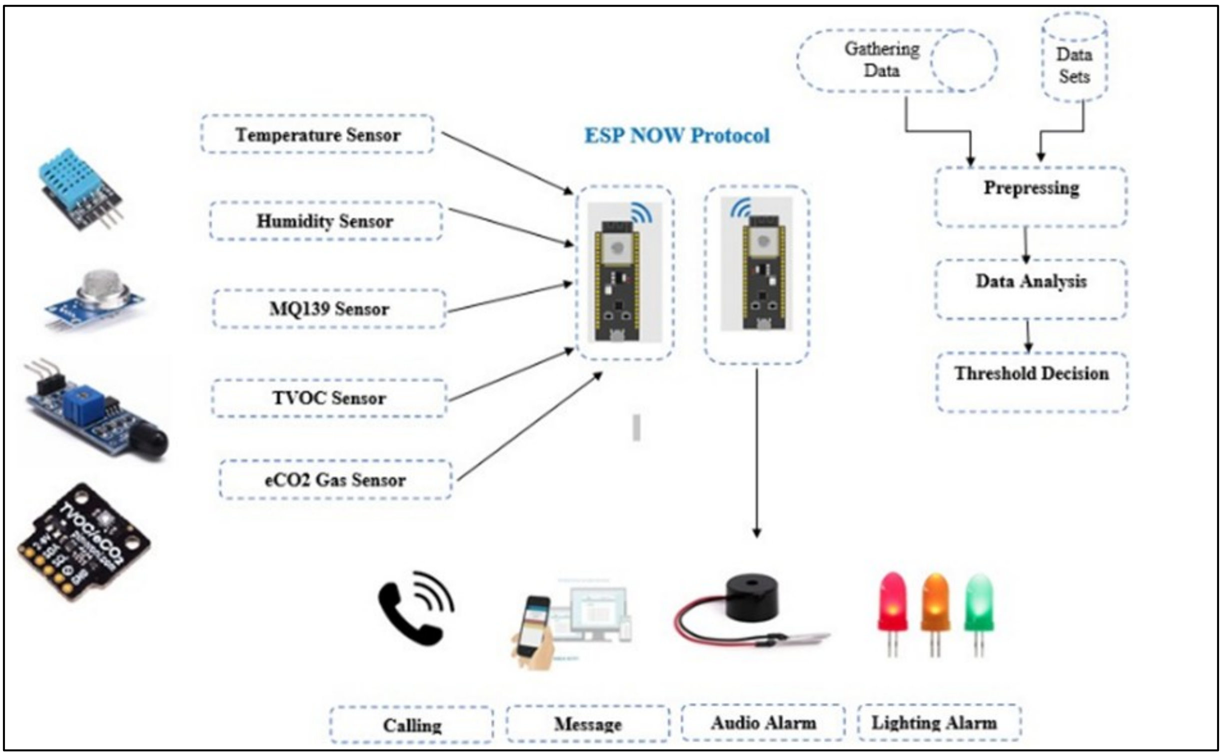


Figure 1. Implementation of System Prototype

- **Audible Alarm Buzzer:** The fire alarm controller generated audible alarm buzzers to create a loud signal, which notified the occupants of a fire and prompted them to undertake remedial actions, such as evacuation.
- **Visual Signals:** Strobe lights or flashing lights were used as visual signals to complement the sound alarms and achieve maximum visibility, particularly in cases with noisy conditions or for hearing-impaired persons.
- **Text Messages:** Text messages were utilized to alert those not on the premises but who were required to be notified of the activation of fire alarms. Off-site facility managers, remote monitoring systems, or emergency response teams may be among them.

5.2 Training System

We utilize supervised datasets as training data to construct a logistic regression model. The dataset is divided into a training set (90% of the data) and a test set (10% of the data). The training set is used to train the logistic regression model by adjusting its parameters through iterative optimization. This process aims to minimize the error between the predicted outputs and the actual target values. After training, the model is evaluated using the test set, which contains unseen data. Predictions are made using the trained model, and the accuracy metric is commonly used to assess its performance. The results may vary based on factors such as the quality and size of the specific experiment in the dataset, problem complexity, choice of hyperparameters, and evaluation metrics employed. The performance of the logistic regression model on the Indoor Laboratory Fire dataset is presented in Table 1.

Table 1. Logistic Regression Indoor Laboratory Fire Dataset

Data set (Experiment)	#of records	Accuracy
Carton_1	330	93.94%
Carton_2	456	98.91%
Clothing_1	918	97.28%
Clothing_2	2299	97.39%
Electrical_1	1140	96.77%
Electrical_2	991	97.99%
Electrical_3	2689	99.26%
Electrical_4	2974	99.22%

The accuracy of the logistic regression model is evaluated by comparing the predicted outputs to the actual target values in the test set. We evaluated performance using accuracy, which reflects the percentage of correct classifications made by the model on unseen test data. It provides an indication of how well the model performs in accurately classifying the data. The results indicate that the model achieved high accuracy rates for most categories, ranging from 93.94% to 99.26%, demonstrating its effectiveness in accurately classifying instances within each category.

The heatmap in Figure 2 shows a consolidated view of the model's performance using key evaluation metrics accuracy, precision, recall, and F1-score across different material types. The heatmap clearly shows that the model maintains consistently high performance across all material categories. Carton achieves the highest overall metrics (accuracy = 98.1%), followed closely by cloth (accuracy = 97.5%) and electrical fires (accuracy = 97.4%). The minimal variation between the categories indicates that the model generalizes well to different materials without significant bias.

While the heatmap provides a high-level overview, Figure 3 presents the confusion matrices for the carton, electrical, and cloth datasets to give a more detailed breakdown of predictions. For the carton dataset, the model achieved an accuracy of 98.1%, with most samples correctly classified and only a small number of false negatives in Class 1. The electrical dataset showed slightly lower performance with an accuracy of 97.4%, particularly strong in Classes 0 and 2, but with a relatively higher number of false negatives in Class 1, where some cases were misclassified as Class 0. This suggests the need for fine-tuning to improve sensitivity for Class 1. The cloth dataset also performed very well, reaching an accuracy of 97.5%, with large numbers of correct predictions across all classes and only a few misclassifications in Classes 0 and 1. Overall, these results highlight the strong effectiveness of the model, while pointing to Class 1 in the electrical dataset as the primary area requiring further improvement.

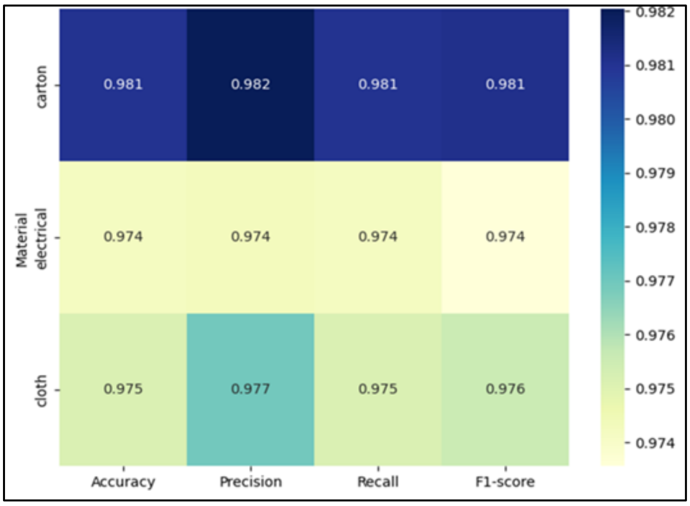


Figure 2. Performance Metrics Heatmap Across Materials

Regarding the eCO2 sensor, it did not detect significant amounts of particles in the air during the electrical-based fire incident. This can be attributed to the fact that the eCO2 sensor may not be specifically designed to detect particles or gases associated with electrical fires. It's important to consider that the lack of detection by the eCO2 sensor in this context does not necessarily indicate a fault or limitation of the sensor itself.

The dataset provided contains approximately 1.5% indication of early fire, suggesting that there are certain features or patterns within the data that can be associated with the early stages of a fire. The correlation analysis shown in Figure 6 confirms that temperature alone is not the dominant indicator of fire incidents. As shown in the correlation matrix, the strongest relationship with fire status was observed for the MQ139 gas sensor (r = 0.75), followed by eCO2 (r = 0.64) and TVOC (r = 0.59). While temperature showed a moderate correlation (r = 0.58), it was not the sole predictor of fire. Interestingly, humidity exhibited a negative correlation with fire status (r = -0.39), indicating a reduction in humidity during fire events. These findings demonstrate that the proposed model effectively leverages multiple sensor inputs rather than relying exclusively on temperature, which enhances its robustness and reliability in early fire detection.

The probability analysis in Figure 7 revealed that most predictions were made with high confidence. For electrical and cloth samples, over 95% of test cases exceeded the 0.7 confidence threshold, with average confidence values above 0.95 for correct predictions. Carton samples achieved lower stability (86.7% above the threshold), mainly due to the smaller dataset size. Importantly, misclassified samples consistently exhibited lower confidence levels (around 0.6), suggesting that confidence-based filtering can be applied in practice to reject uncertain predictions and further reduce errors. This demonstrates not only the accuracy but also the reliability of the logistic regression model in real-world fire detection scenarios. In the dataset, each sensor produces readings in very different ranges—for example, humidity is measured in percentages, temperature in degrees Celsius, gas concentrations in ppm, and eCO_2 in parts per million. If these raw values were fed directly into the logistic regression model, the features with larger numerical ranges (such as gas concentrations) could dominate the model coefficients, leading to biased weight assignments and reduced accuracy. By applying Standard Scalar, each feature was standardized to have a mean of zero and a standard deviation of one. This transformation placed all sensor readings on a comparable scale, ensuring that no single feature disproportionately influenced the learning process. In addition, it helped stabilize the training process, improved the numerical efficiency of logistic regression, and reduced issues related to multicollinearity among correlated sensor variables.

5.3 Performance of Model Training

Accuracy and loss during training and validation were measured to assess the performance of material-specific models, with each model trained and evaluated on distinct material types. The results demonstrate exceptional performance across all materials, with carton achieving 98.10% accuracy, electrical 97.42% accuracy, and cloth 97.52% accuracy mentioned in figure 6. This specialized approach yielded superior performance compared to the unified model, with an average improvement of 2-3% in classification accuracy across material types.

The loss curves shown in Figure 7 shows excellent generalization capabilities, with carton exhibiting the remarkable characteristic of validation loss (0.1275) being lower than training loss (0.1434), indicating perfect generalization. Similarly, electrical and cloth models maintained minimal gaps between training and validation losses (0.0618 vs 0.0717 and 0.0736 vs 0.0721 respectively), confirming effective learning without overfitting.

These comprehensive results validate the effectiveness of the material-specific approach, demonstrating that specialized models can capture material-specific patterns more effectively while maintaining robust generalization capabilities. The consistent high performance across diverse materials confirms the methodology's suitability for real-world deployment where material type can be predetermined.

5.4 Practical System Validation

To validate the effectiveness of our trained model, a practical experiment was conducted using the same sensors employed in the dataset collection. In this setup, the output value of Class = 0 indicates no fire detected, whereas Class = 1 represents fire detected.

As shown in Table 2, all temperature values remained within a normal operating range, confirming that the environment was stable during the test. Despite this stability, the system was still able to identify potential fire risks based on abnormal readings from the gas (MQ139), TVOC, and eCO2 sensors. This demonstrates the added value of a multi-sensor approach, where detection does not rely solely on temperature but instead integrates multiple indicators of fire conditions.

Specifically, rows 3–6, 8, and 14 of Table 2 were correctly classified as fire events (Class = 1), while all other rows were identified as no-fire conditions (Class = 0). Out of the 14 collected samples, the system accurately detected every case, resulting in 100% detection accuracy in this small-scale validation experiment.

Furthermore, the alarm system was successfully triggered in response to detected fire events, providing clear evidence of the system's reliability and real-world functionality. These findings reinforce both the validity of the trained logistic regression model and the robustness of the integrated sensing framework. Future work will focus on extending these tests to larger-scale and more diverse environments to further validate the system's performance under real operational conditions.

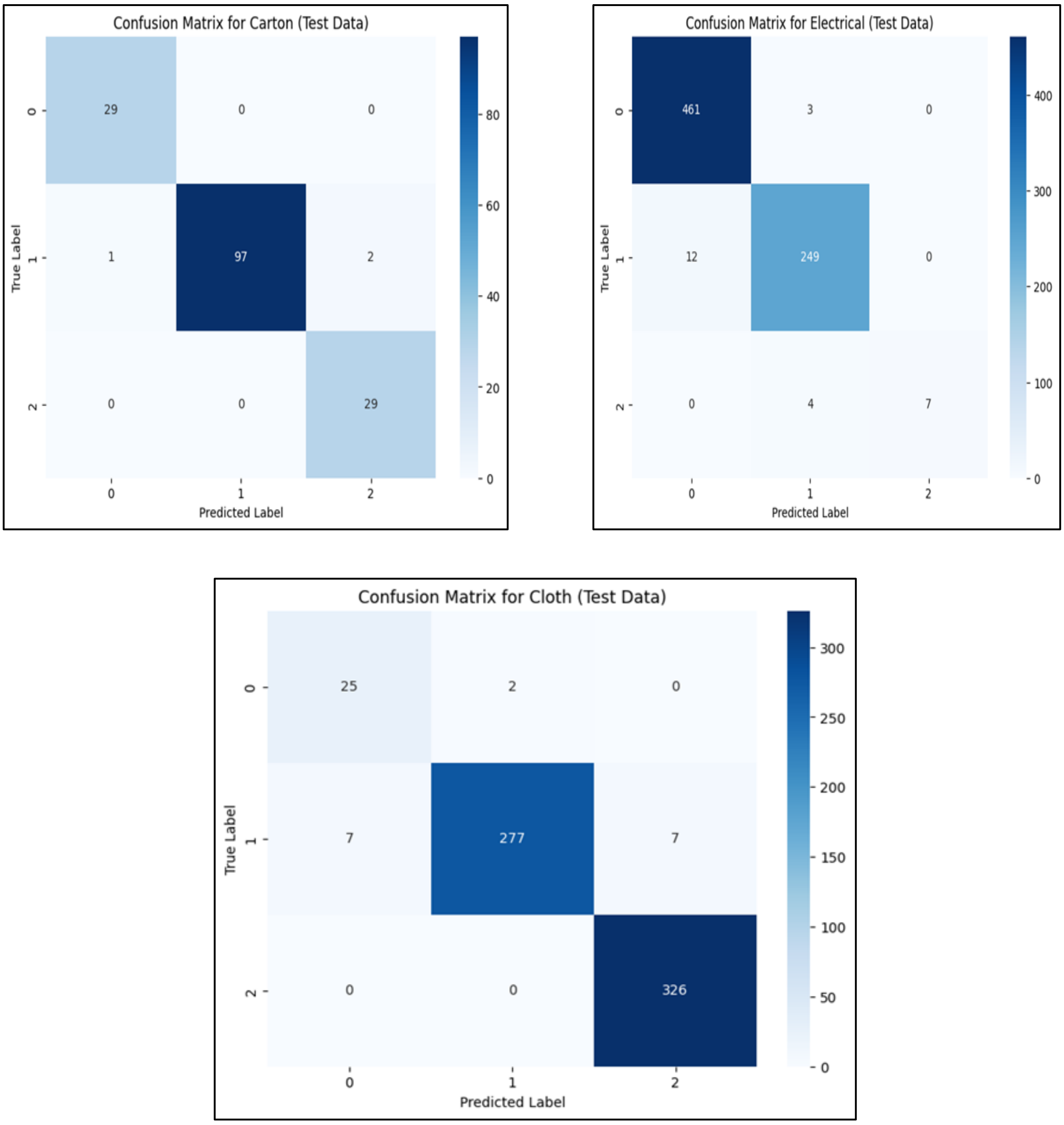


Figure 3. Confusion Matrix Heatmaps for the Three Material Categories (Carton, Clothing, Electrical)

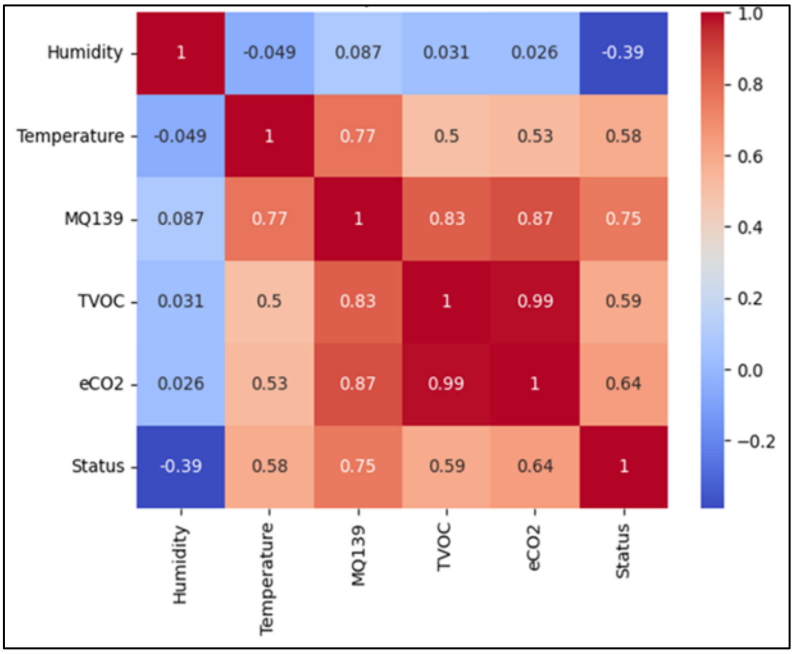
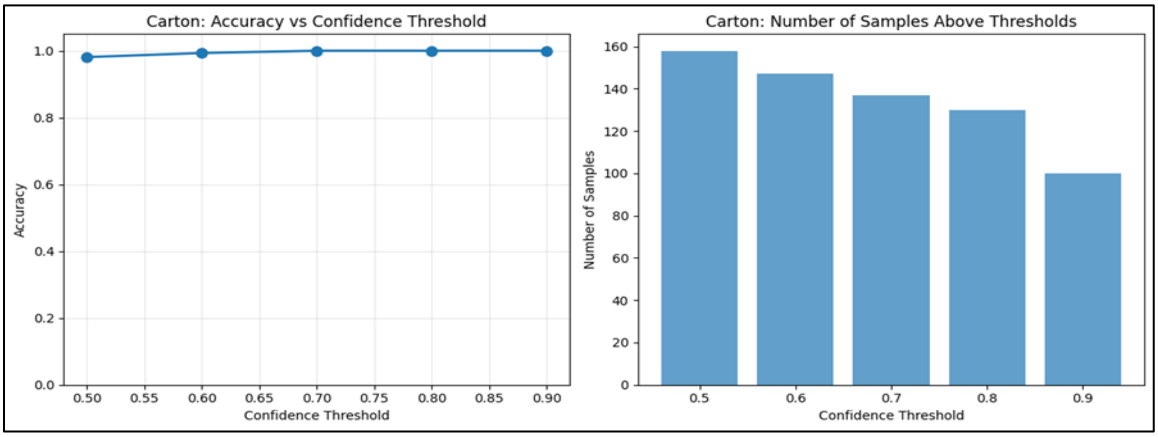


Figure 4. The Correlation Heatmap of Features vs Fire Status

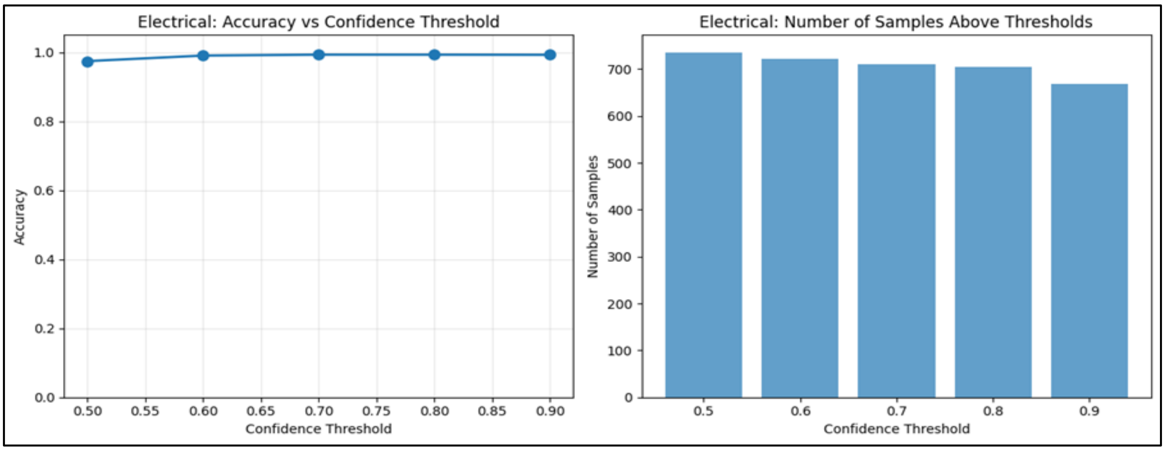
5.5 Comparative Overview

Table 3 presents a comparative overview of our multi-sensor fusion and logistic regression approach against selected state-of-the-art fire detection methods. The comparison highlights methodology, dataset type, key results, and practical relevance. Unlike most prior studies, which rely on image- or simulation-based data, our approach uses real sensor signals across multiple materials (carton, cloth, electrical), achieving high accuracy (≈97–98%) with robust confidence filtering and low computational complexity, making it suitable for real-time indoor fire monitoring.

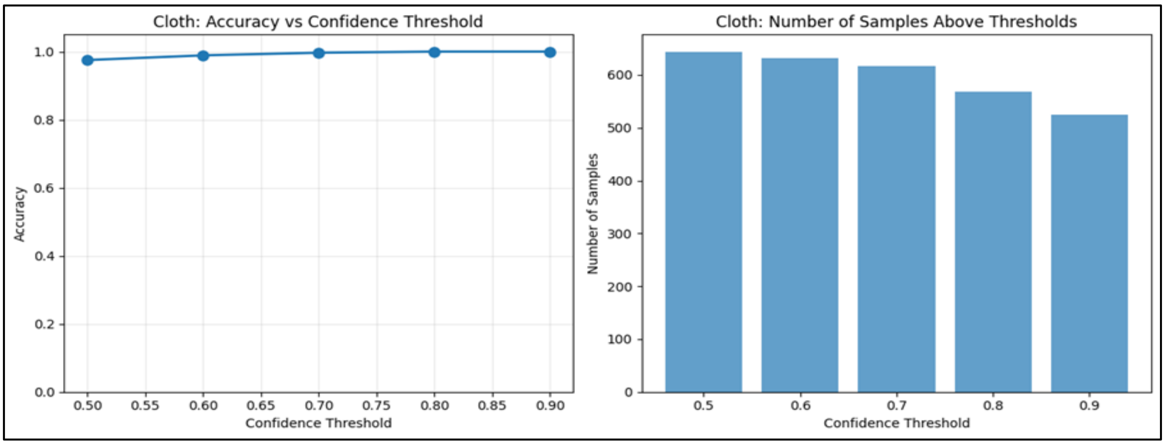
While the studies employ diverse datasets (from images and videos to simulations), the comparison remains meaningful, emphasizing methodological advances, accuracy benchmarks, and real-time applicability. Our work focuses on sensor-based data and logistic regression for early-stage indoor fire detection, addressing scenarios where visual data may be unavailable or impractical. This complements vision-based (e.g., CNN and segmentation) and simulation-based (e.g., LSTM-Kriging) approaches, illustrating that integrating multiple modalities can enhance fire safety and emergency response.



(a)



(b)



(c)

Figure 5. Accuracy with Different Thresholds (0.5–0.9) for the Evaluated Datasets

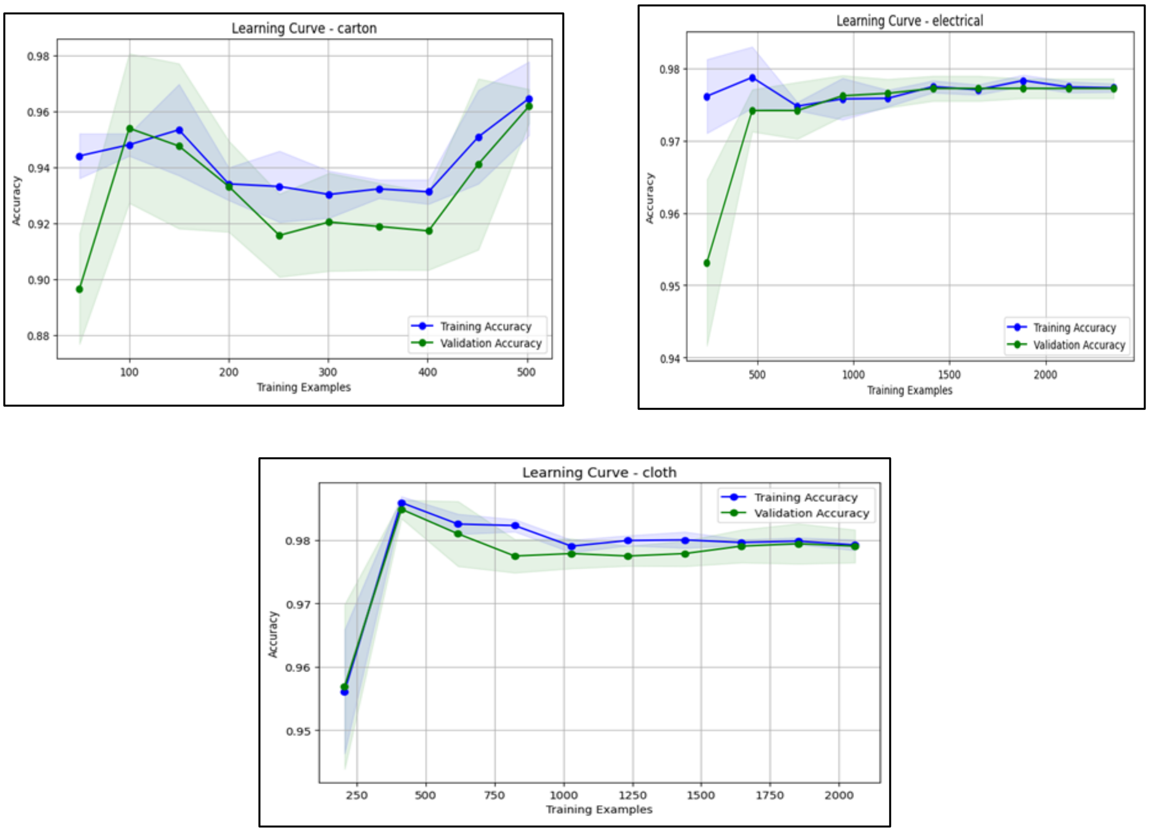


Figure 6. Training and Validation Accuracy for Carton, Electrical, and Cloth Materials

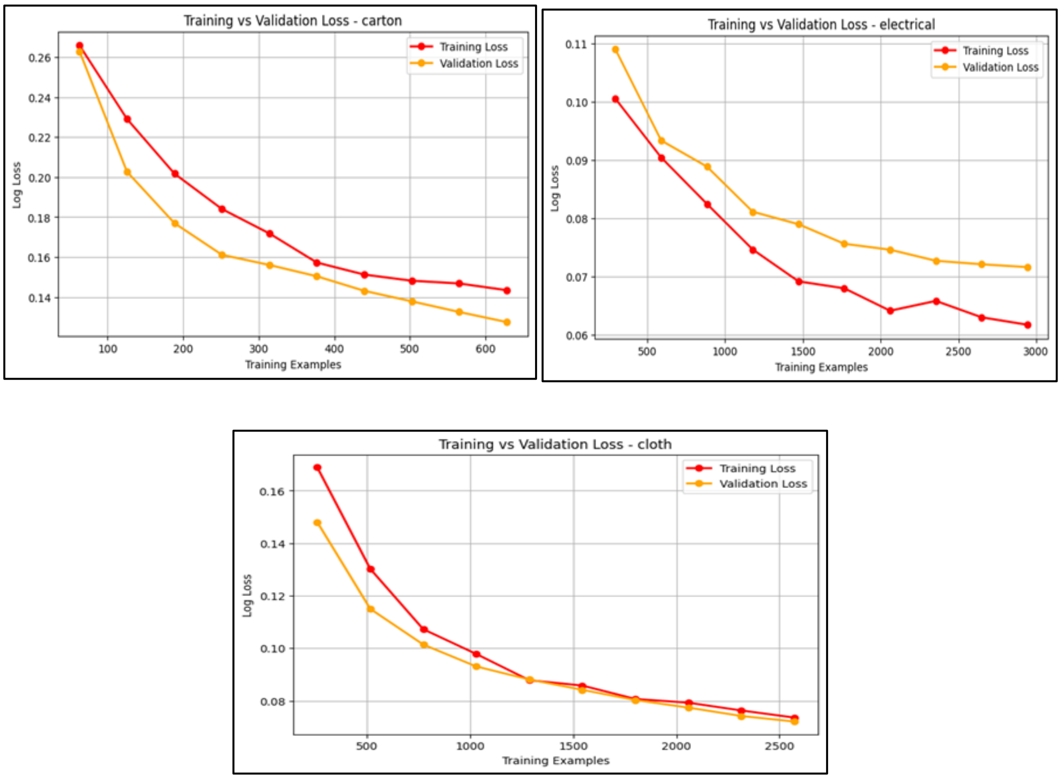


Figure 7. Training and Validation Loss for Carton, Electrical, and Cloth Materials

Table 2. Fire Detection Outputs

Humidity%	Temp.	MQ139	TVOC	eCO2	Class
69	24.9	342	1134	5260	0
71	24.3	159	14	401	0
68	24.9	362	1665	7566	1
74	21.6	184	1478	4956	1
73	21.6	200	2124	7956	1
76	24.7	274	2168	6356	1
68	22.7	78	244	489	0
70	22.3	128	7144	6100	1
66	22.1	86	959	841	0
74	22.9	94	974	983	0
66	22.9	121	3785	3468	0
70	21.8	68	11	423	0
66	22.2	78	286	524	0
75	22.3	131	5130	4572	1

Table 3. Comparison of our Multi-Sensor Fire Detection Approach with Existing Methods

Paper	Method	Dataset	Key Results	Comparison to Proposed Work
Proposed Work	Multi-sensor fusion + Logistic Regression	Real sensor data (carton, cloth, electrical)	Accuracy ≈ 97–98%; robust confidence filtering; low computational complexity; real-time indoor fire detection	Operates on low-cost sensors rather than images; high accuracy across multiple materials; suitable for indoor deployment
[29]	Fine-tuned CNN with adaptive prioritization and dynamic channel selection	Fire image datasets from surveillance cameras	High accuracy in fire detection; validated for real-world disaster management	Vision-based approach; sensor-free; highlights advantage of using multi-sensor data in indoor scenarios

[30]	LeNet5 CNN with L2 regularization	Fire/non-fire images	Train accuracy ≈ 87%, validation ≈ 71%, test ≈ 70%	Lower accuracy compared to our method; image-based CNN approach; demonstrates the benefit of sensor fusion for multi-material indoor fires
[31]	Hybrid Adaboost + MLP + CNN	Image/video fire datasets	≈ 99% accuracy with low false alarm rate	High-performing hybrid model; image/video data vs. real sensors; emphasizes computational efficiency advantage of logistic regression
[32]	Fire-Net (Landsat-8, optical + thermal modalities)	Satellite imagery of active fires	97.35% accuracy, including small fires	Remote sensing and large-scale detection; contrasts with indoor, multi-material sensor-based detection
[33]	Lightweight MobileNetV3, anchor-free structure	Self-built + public fire datasets	90.2% accuracy; 29.5 f/s real-time inference	Lightweight, real-time image-based model; allows direct comparison of efficiency vs. sensor-based multi-material approach

6. Conclusion

Data centers play an important role in data collection, processing, and decision-making. A variety of security measures have been implemented, including early warning systems, considering the need for their continuous functioning. This proposed work uses machine learning to interact with the sensor network and demonstrates the effectiveness of early fire detection using the logistic regression model and sensor data integration. In real-time monitoring, using innovative algorithms enables and improves fire detection by providing timely warnings that can reduce fire spread and damage. The research should be further developed to improve the parameter model by detecting modification techniques and optimizing the performance of the logistic regression model. These results lead to improved efficiency and dependability in fire detection. When logistic regression is suitable for current indoor fire detection using structured sensor data, future work could examine advanced models such as DNNs to improve detection in challenging conditions or with more complicated implementations. Future research should focus on improving these models, testing them using modern technologies, and performing major evaluations to verify their durability and dependability in real-world fire detection cases. This system is continuously improving to reduce fire accidents, safeguard people and properties, and contribute to overall safety and security.

References

[1] Gaur, Anshul, Abhishek Singh, Ashok Kumar, Kishor S. Kulkarni, Sayantani Lala, Kamal Kapoor, Vishal Srivastava, Anuj Kumar, and Subhas Chandra Mukhopadhyay. "Fire sensing technologies: A review." IEEE Sensors Journal 19, no. 9 (2019): 3191-3202.

[2] Fernández-Alaiz, Florencio, Ana Maria Castañón, Fernando Gómez-Fernández, Antonio Bernardo-Sánchez, and Marc Bascompta. "Determination and fire analysis of gob characteristics using CFD." Energies 13, no. 20 (2020): 5274.

[3] Maragkos, Georgios, and Tarek Beji. "Review of convective heat transfer modelling in cfd simulations of fire-driven flows." Applied sciences 11, no. 11 (2021): 5240.

[4] Kolaitis, Dionysios, Eleni Asimakopoulou, and Maria Founti. "CFD simulation of fire spreading in a residential building: the effect of implementing phase changing materials." In European Combustion Meeting. 2011.

[5] Sesseng, Christian. Mapping of gas concentrations, effect of deadair space and effect of alternative detection technology in smouldering fires. 2016.

[6] Fonollosa, Jordi, Ana Solórzano, and Santiago Marco. "Chemical sensor systems and associated algorithms for fire detection: A review." Sensors 18, no. 2 (2018): 553.

[7] Solórzano, Ana, Jordi Fonollosa, and Santiago Marco. "Improving calibration of chemical gas sensors for fire detection using small scale setups." In Proceedings, vol. 1, no. 4, p. 453. MDPI, 2017.

[8] Fleming, Joseph M., and Fire Marshal. "Smoke detector technology and the investigation of fatal fires." Fire and Arson Investigator 50, no. 3 (2000): 35-40.

[9] Andrew, A. M., A. Y. M. Shakaff, A. Zakaria, R. Gunasagaran, E. Kanagaraj, and S. M. Saad. "Early stage fire source classification in building using artificial intelligence." In 2018 IEEE Conference on Systems, Process and Control (ICSPC), IEEE, (2018): 165-169.

[10] Wu, Lesong, Lan Chen, and Xiaoran Hao. "Multi-sensor data fusion algorithm for indoor fire early warning based on BP neural network." Information 12, no. 2 (2021): 59.

[11] Maheen, Jubeena B., and R. P. Aneesh. "Machine learning algorithm for fire detection using color correlogram." In 2019 2nd international conference on intelligent computing, instrumentation and control technologies (ICICICT), vol. 1, IEEE, (2019): 1411-1418.

[12] Khan, Rubayat Ahmed, Jia Uddin, Sonia Corraya, and J. Kim. "Machine vision based indoor fire detection using static and dynamic features." International Journal of Control and Automation 11, no. 6 (2018): 87-98.

[13] Saponara, Sergio, Abdussalam Elhanashi, and Alessio Gagliardi. "Real-time video fire/smoke detection based on CNN in antifire surveillance systems." Journal of Real-Time Image Processing 18, no. 3 (2021): 889-900.

[14] Li, Pu, and Wangda Zhao. "Image fire detection algorithms based on convolutional neural networks." Case Studies in Thermal Engineering 19 (2020): 100625.

[15] Sarwar, Barera, Imran Sarwar Bajwa, Shabana Ramzan, Bushra Ramzan, and Mubeen Kausar. "Design and application of fuzzy logic based fire monitoring and warning systems for smart buildings." Symmetry 10, no. 11 (2018): 615.

[16] Maksimović, Mirjana, Vladimir Vujović, Branko Perišić, and Vladimir Milošević. "Developing a fuzzy logic based system for monitoring and early detection of residential fire based on thermistor sensors." Computer Science and Information Systems 12, no. 1 (2015): 63-89.

[17] Muralidharan, Aiswarya, and Fiji Joseph. "Fire detection system using fuzzy logic." Int. J. Eng. Sci. Res. Technol 3, no. 4 (2014): 6041-6044.

[18] Necsulescu, D., and Xuqing Le. "Type-2 Fuzzy Logic Sensor Fusion for Fire Detection Robots." In Proceedings of the 2 nd International Conference of Control, Dynamic Systems, and Robotics Ottawa, Ontario, Canada. 2015.

[19] Listyorini, Tri, and Robbi Rahim. "A prototype fire detection implemented using the Internet of Things and fuzzy logic." World Trans. Eng. Technol. Educ 16, no. 1 (2018): 42-46.

[20] Arduino. (2018, November 18). Arduino Basics.html. Retrieved April 20, 2018, from Arduino: http://www.arduino.cc

[21] Robot-r-us, WeMos D1 R2 WiFi ESP8266 Board Compatible with Arduino IDE (2018), 18 October 2017, https://www.robot-r-us.com/wireless_wifi/wemos-d1-r2-wifi-esp8266-development-boardcompatible-with-ardui.html

[22] ASTM Standard Terminology of Fire Standards; ASTM: West Conshohocken, PA, USA, 2004.

[23] Fleming, Joseph M. "Photoelectric and Ionization Detectors—A Review of The Literature Re–Visited." Retrieved Dec 31 (2004): 2010.

[24] Fabian, T.Z.; Gandhi, P.D. Smoke Characterization Project—Technical Report; Underwriters Laboratories: Northbrook, IL, USA, 2007. [Google Scholar]

[25] Keller, Alejandro, M. Rüegg, Martin Forster, Markus Loepfe, Rolf Pleisch, P. Nebiker, and Heinz Burtscher. "Open photoacoustic sensor as smoke detector." Sensors and Actuators B: Chemical 104, no. 1 (2005): 1-7.

[26] Meacham, Brian J. "The use of artificial intelligence techniques for signal discrimination in fire detection systems." Journal of Fire Protection Engineering 6, no. 3 (1994): 125-136.

[27] Liu, Zhigang, and Andrew K. Kim. "Review of recent developments in fire detection technologies." Journal of Fire Protection Engineering 13, no. 2 (2003): 129-151.

[28] Nazir, Amril, Husam Mosleh, Maen Takruri, Abdul-Halim Jallad, and Hamad Alhebsi. "Early fire detection: a new indoor laboratory dataset and data distribution analysis." Fire 5, no. 1 (2022): 11.

[29] Muhammad, Khan, Jamil Ahmad, and Sung Wook Baik. "Early fire detection using convolutional neural networks during surveillance for effective disaster management." Neurocomputing 288 (2018): 30-42.

[30] Roy, Sanjiban Sekhar, Vatsal Goti, Aditya Sood, Harsh Roy, Tania Gavrila, Dan Floroian, Nicolae Paraschiv, and Behnam Mohammadi-Ivatloo. "L2 regularized deep convolutional neural networks for fire detection." Journal of Intelligent & Fuzzy Systems 43, no. 2 (2022): 1799-1810.

[31] Saeed, Faisal, Anand Paul, P. Karthigaikumar, and Anand Nayyar. "Convolutional neural network based early fire detection." Multimedia Tools and Applications 79, no. 13 (2020): 9083-9099.

[32] Seydi, Seyd Teymoor, Vahideh Saeidi, Bahareh Kalantar, Naonori Ueda, and Alfian Abdul Halin. "Fire-Net: A Deep Learning Framework for Active Forest Fire Detection." Journal of Sensors 2022, no. 1 (2022): 8044390.

[33] Li, Yuming, Wei Zhang, Yanyan Liu, and Yao Jin. "A visualized fire detection method based on convolutional neural network beyond anchor." Applied Intelligence 52, no. 11 (2022): 13280-13295.