ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 12, Issue 03, March 2025

Fire Detection and Risk Prediction for Smart Safety: A Neural Network-Driven IoT Approach

[1] Dr R Jeya, [2] Rithish R, [3] Ganesh K, [4] Ashish Sukumar, [5] Ramuji Doniparthi

[1] [2] [3] [4] [5] Department of Computing Technologies, SRM Institute of Science and Technology, Kattankulathur, Chennai, India

Corresponding Author Email: [1] jeyar@srmist.edu.in, [2] rithishraghavan@gmail.com, [3] ganeshrajan2.00@gmail.com, [4] ashish11062003@gmail.com, [5] ramjid@gmail.com

***Abstract*— Fire accidents pose a significant threat to human life and property, making the development of predictive fire detection systems essential. Traditional systems rely primarily on gas sensors, detecting fires only after significant spread, which can result in severe property damage or even life-threatening situations. Our AI-driven fire detection system aims to address this by using a combination of sensors, including color, temperature, and gas sensors, to detect fires at an early stage. A neural network is implemented on the Arduino Nano 33 BLE Sense, enabling the microcontroller to predict fires based on proximity and color intensity patterns. Alerts are sent to users via a mobile app, while relevant authorities are notified through messaging protocols. The system also integrates a central server for data analysis and continuous improvement, enhancing prediction accuracy over time. This proactive approach ensures faster response times, minimizing damage and improving safety in various environments.**

***Index Terms*— ANN, Internet of Things, RGB, Arduino Nano 33 BLE, Fire Detection, ESP32, DHT11.**

I. INTRODUCTION

An approach to fire detection using video analysis of surveillance data is presented in this study. The two main advancements are a multiexpert system that incorporates supplementary data from color, shape variation, and motion analysis, and a system that analyzes motion. This method's key benefit is that it dramatically enhances the system's overall performance with very little work required during design. A novel motion description has been presented that uses a bag-of- words method. By reducing false positives while maintaining accuracy and running on embedded platforms, the suggested technique has been tested on a big collection of fire footage from both online and real-world sources [1].

The industrial sector suffers huge losses due to fire occurrences, which create irreversible damage and have an influence on the environment. There are a lot of safety regulations, but fires are still happening more often. The majority of businesses have installed smoke detectors with sensors, but their range isn't very good considering how big the building is. Computer vision-based fire detection systems that connect with cameras were developed because video surveillance systems are crucial for building automation. Identifying the edge of a flame helps to define the border between portions that are thermally reactive and those that are not. Although there are several methods for detecting fires, none of them take the stability of fire zones into consideration. A novel approach to distinguishing fire zones in video frames by means of color correlogram characteristics is presented in this work. Additionally, features based on saliency are extracted, and the color auto-correlogram feature is used to evaluate the color probability of nearby pixels. [2].

Recent advancements in sensing technologies, coupled with improvements in information and communication systems, have significantly enhanced fire detection and prevention capabilities. Contemporary fire detection systems now incorporate multiple parameters, including ambient heat, smoke, flames, and gas concentrations, to provide more reliable and efficient fire detection. However, despite these advancements, limitations persist, particularly in balancing early detection, minimizing false positives, and ensuring system scalability. Furthermore, many traditional systems rely on centralized cloud-based computing, which can lead to delays in critical decision-making during emergencies [3].

This project addresses these limitations by proposing an AI- driven fire detection system that operates on the Arduino Nano 33 BLE Sense platform, integrating TinyML for real-time, on- device processing. The system utilizes a range of sensors, including color, temperature, and gas sensors, to monitor environmental data and predict potential fire hazards at an early stage. A neural network is implemented to analyze sensor inputs such as temperature fluctuations, color intensity, and gas concentrations, allowing for accurate fire detection and rapid alerts. This approach significantly reduces the latency associated with cloud-based processing, enabling the system to function autonomously without sacrificing performance [4].

This system is ideal for real-time applications in many contexts since it can run on low-power embedded devices thanks to TinyML. To guarantee a prompt reaction, consumers receive alerts using a mobile app, and notifications are delivered to the appropriate authorities through messaging protocols. Furthermore, the system is designed with a central server that compiles data from sensors, allowing for continuing data analysis to continuously enhance the accuracy of predictions. Early fire detection may limit damage and save lives in both residential and industrial situations. This solution is scalable and versatile, making it a perfect fit for both settings [4][5].

By combining neural networks with advanced sensing technologies, this system overcomes the deficiencies of traditional fire detection approaches, providing a cost-effective, scalable, and real-time solution that offers greater reliability, fewer false positives, and enhanced safety for occupants [5].

This research suggests an AI-driven alternative to conventional fire detection systems that uses a combination of several sensors and machine learning to identify flames in their early stages. The system analyzes environmental data including temperature, gas levels, and color intensity in real-time using a neural network that runs on the Arduino Nano 33 BLE Sense. For training, validation, and testing purposes, a bespoke dataset is utilized, which is meant to mimic real-world fire incidents. The suggested approach outperforms current state-of-the-art fire detection systems in terms of accuracy, false alarm reduction, and reaction time.

The following are the main accomplishments of this study:

- Recognizing the significant deficiencies and shortcomings in existing fire detection systems,

especially those dependent on traditional sensor and video-based methodologies.

- Proposing an AI-driven, multi-sensor fire detection system that improves detection precision and velocity by amalgamating sensor data with sophisticated machine learning algorithms.
- Performing an exhaustive assessment of the proposed system with a bespoke dataset, accompanied by detailed performance metrics that illustrate its efficacy and dependability.

The subsequent sections of this work are structured as follows: Section II examines the current literature on fire detecting systems. Section III delineates the suggested solution, highlighting the innovative elements of the AI-driven methodology. Section IV delineates the study technique, encompassing the model architecture, dataset specifications, neural network configuration, and simulation framework. Section V examines the outcomes and offers a comprehensive review of the system's performance. Ultimately, Section VI finishes the article by summarizing the findings and outlining prospective future endeavors.

II. RELATED WORK

Recent advances in fire sensing technology have led to significant improvements in detection capabilities, leveraging developments in sensing, information, and communication technologies. Research in this domain has predominantly focused on enhancing fire detection accuracy, minimizing false alarms, and enabling faster response times through various methodologies. These efforts include advancements in video- based detection, sensor integration, and hybrid models combining different machine learning techniques.

A. Video-Mediated Fire Detection System:

Methods for fire detection based on video have been created to use frame-to-frame differences for the identification of possible fire areas. These approaches often evaluate attributes such as hue, form, and movement to identify fires, utilizing the erratic behavior of flames [6]. various methods employ classifiers such as the Bayes classifier to integrate the behavior of various characteristics, hence improving fire recognition. Nevertheless, these techniques frequently depend on immobile cameras and regulated surroundings, resulting in heightened false positives in dynamic real-world contexts [7]. Conversely, our study enhances vision-based fire detection by amalgamating it with additional sensor data (e.g., temperature and gas concentrations) to augment accuracy in both static and dynamic settings, providing a more resilient solution. Recent research have underscored the efficacy of You Only Look Once -based methodologies for fire and smoke detection in IoT monitoring systems[8], therefore augmenting the capabilities of conventional video-based techniques.

B. Wireless Sensor Networks (WSNs) and IoT-Based Systems:

Wireless Sensor Networks (WSNs) are extensively utilized for fire detection in residential and commercial environments owing to their capacity for continuous monitoring of environmental parameters, including temperature, smoke, and gas concentrations. These systems frequently depend on the detection of flames or smoke, leading to delayed notifications and the possibility of exacerbating fire events [9]. IoT-based systems have enhanced this domain by including devices such as NodeMCU with temperature and gas sensors, providing continuous monitoring and remote alerts [10]. Notwithstanding these developments, several systems rely on cloud-based processing, potentially causing delays in decision-making during emergencies. Our approach mitigates this constraint by employing on-device processing with the Arduino Nano 33 BLE Sense, allowing real-time analysis and expedited identification independent of external servers. Furthermore, recent studies indicate that utilizing multi-color space and backdrop modeling might markedly improve fire detection efficacy in wireless sensor networks (WSNs) [11].

C. Hybrid Machine Learning Models:

Recent studies have also explored hybrid machine learning approaches to improve fire detection accuracy. For instance, models that integrate logistic regression, support vector machines, and decision trees have been proposed to enhance detection performance and resilience across various fire scenarios [12]. These hybrid approaches achieve improved classification accuracy but often require extensive computational resources, making them more suitable for centralized systems rather than edge devices. Our project integrates a neural network directly on the Arduino Nano platform, allowing for efficient real-time processing with reduced computational overhead. The use of TinyML enables the system to adapt to new data, refining its predictive capabilities over time, and maintaining high accuracy in fire detection with minimal false positives. A recent paper discusses an improved fire detection approach based on YOLO-v8, which exemplifies advancements in hybrid machine learning techniques for smart cities [8][13]. Additionally, some studies have noted the importance of exploring recent advances and emerging directions in fire detection systems based on various machine learning algorithms [14][15].

D. Relation to Existing Work:

Our project builds upon the strengths of the aforementioned methodologies by combining multi-sensor data with AI-driven analysis for early-stage fire detection. Unlike video-based methods that are limited by background stability [7][8] or WSNs that require direct contact with fire or smoke [9][11], our approach incorporates temperature, gas concentration, and visual data. This integration allows for a more comprehensive assessment of fire risks, ensuring earlier detection and fewer false alarms. Additionally, by implementing a neural network on the Arduino Nano 33 BLE Sense, our solution reduces the latency associated with cloud-based systems, offering a cost- effective, scalable, and efficient approach for real-time fire detection.

III. PROPOSED SOLUTION

Fires pose a serious risk to people and their possessions, and the best way to limit damage and increase safety is to catch them early. This research presents a predictive fire detection system that integrates modern sensor technology with machine learning to provide real-time, early alerts of probable fire incidents. This technology, unlike conventional systems that depend on gas sensors and identify fires only after considerable damage has transpired, notifies users at the earliest stage, hence enhancing reaction times and mitigating risks.

The system employs multiple sensors—such as RGB color, temperature, and gas sensors—to detect environmental changes that signal early fire conditions. An artificial neural network (ANN) processes the data, recognizing complex patterns like temperature fluctuations and gas concentrations. This enhances accuracy and reliability, outperforming conventional fire detection methods.

A. Key Features:

- Early Detection and Proactive Alerts: The system leverages advanced sensors to detect early-stage fire indicators like color temperature and gas emissions, offering much faster detection compared to traditional systems. This proactive approach allows users to take action quickly by either extinguishing the fire or evacuating the area, minimizing damage and danger. By continuously monitoring environmental conditions and providing real-time alerts via a mobile app, the system significantly reduces the risk of large-scale fire outbreaks, offering a highly efficient and proactive fire detection method suitable for residential, commercial, and industrial settings.
- Hybrid Approach: The system integrates a variety of sensors, such as RGB color, temperature, and gas sensors, to create a comprehensive detection network.

While traditional gas sensors detect fires after flames have spread, the hybrid approach uses color and proximity sensors to catch fires at their earliest stages. This layered defense mechanism ensures that false alarms are minimized, as gas sensors serve as a secondary confirmation, making the system highly reliable. Additionally, the sensors' ability to function in various lighting conditions makes the system adaptable to different environments, providing robust and precise fire detection.

- AI-Powered Predictions: Using a neural network (ANN), the system continuously learns and adapts to new data, identifying patterns in sensor inputs that improve its ability to predict fires. By learning from historical data, the system refines its predictions over time, reducing false positives and negatives, and adjusting to different types of fires and environmental conditions. This AI-driven approach also allows for customized detection setups, making the system versatile enough to handle both industrial settings with hazardous materials and residential areas with common household fire risks.
- Cost-Effective and Scalable: The system is both economical and scalable, employing inexpensive components such as the Arduino Nano 33 BLE Sense and the APDS9960 sensor, hence enhancing accessibility for a diverse user base. The modular architecture facilitates effortless expansion, permitting the incorporation of supplementary sensors without necessitating substantial alterations to the fundamental configuration. This renders the system flexible for bigger buildings or locations with fluctuating fire dangers, guaranteeing that customers may enhance or expand their fire detection capabilities over time without incurring significant expenses.
- Mobile and Cloud Integration: One of the system's key strengths is its mobile app, which provides users with real-time alerts about fire risks, enabling immediate

action even when they are offsite. The system also uses cloud-based data storage and analytics, allowing centralized control, long-term monitoring, and detailed analysis of fire incidents. This integration ensures that users have constant access to sensor data, enabling swift decision-making, and improving the system's performance through historical data analysis, making it an ideal solution for both personal and large- scale deployment.

This approach markedly improves the speed, reliability, and preciseness of fire detection systems by mixing modern machine learning algorithms with IoT technologies. The predictions generated by the neural network, in conjunction with real-time sensor data, guarantee that users are notified of potential fire threats at the earliest opportunity. This novel methodology can preserve lives, reduce property damage, and establish a new benchmark for fire detection systems in residential and industrial environments.

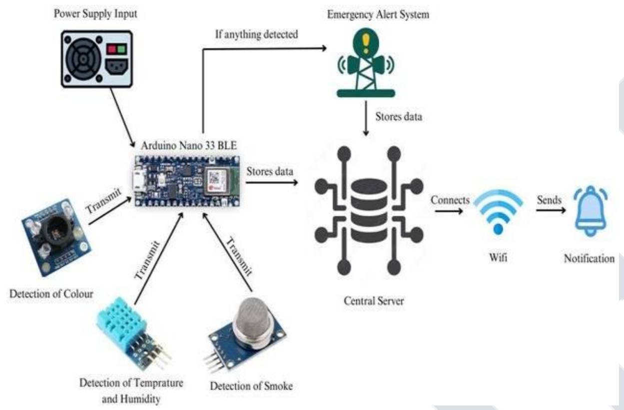

Fig. 1. System Architecture for the Fire Prediction Model

IV. RESEARCH METHODOLOGY

A. Model Architecture

This system design demonstrates a robust approach for live fire detection and environmental monitoring by integrating many sensors with an Arduino Nano 33 BLE. The power supply is the system's core, providing constant and dependable power to the whole design. A reliable power source is essential for the uninterrupted operation of the Arduino Nano 33 BLE, which functions as the system's main microcontroller. The Arduino collects and analyzes data from several connected sensors to monitor environmental conditions, making real-time decisions based on the sensor outputs.

The APDS9960 color sensor is mostly tasked with identifying variations in light or color, such the flickering of flames or the emergence of certain shades that may indicate irregular situations. For instance, firelight often produces a certain pattern of illumination within the red and yellow spectrum, which the APDS9960 is capable of detecting. This feature improves the system's capacity to identify possible fire dangers promptly.

A crucial element is the DHT11 or DHT22 sensor, which quantifies both temperature and humidity. These two environmental factors are critical indications of possible fire hazards. A fast rise in temperature, particularly alongside a decrease in humidity, may indicate an early warning of fire. The system perpetually monitors these quantities, transmitting them to the Arduino Nano 33 BLE for processing. Should temperature measurements surpass a specified safety level, the system identifies this as a potential fire hazard and initiates required measures.

The MQ-2 smoke sensor is essential for smoke detection, a very accurate sign of fire. The MQ-2 sensor identifies gas concentrations including methane, butane, propane, and smoke in the atmosphere. When smoke levels exceed a specific threshold, this information is transmitted to the Arduino for processing. The sensor's capacity to identify a diverse array of gases renders it especially efficient in settings where smoke or toxic gasses may indicate the occurrence of fire.

Upon collection of data from various sensors, the Arduino Nano 33 BLE analyzes the incoming information instantaneously. It locally retains essential data and continually assesses whether any sensor values exceed the established fire risk criteria. Should any parameter surpass safe thresholds, the Arduino promptly activates a sequence of actions, including relaying the gathered data to a central server using Wi-Fi. This connectivity functionality guarantees that sensor data is kept both locally and remotely for subsequent analysis or alert production. The data transmission to the server enables the system to utilize more robust computational capabilities, perhaps executing machine learning models to identify trends and evaluate the probability of a fire.

The central server retains all sensor data, serving as a long-term archive for environmental conditions throughout various durations. It also facilitates more intricate data analysis, encompassing pattern detection and risk forecasting. The server may be configured to initiate warnings upon the detection of hazardous patterns, such as a rapid increase in both smoke levels and temperature. This central hub facilitates the integration of data from numerous Arduino units, creating a networked system for extensive monitoring in environments such as buildings, industries, or campuses.

Simultaneously with data transfer, the Arduino activates the emergency warning system. This system may incorporate loud alarms, lights, or automatic notification systems that alert authorities or safety workers to the identified hazard. The emergency alert can be triggered using data from one or more sensors, guaranteeing prompt issuance of warnings in the case of fire or dangerous situations.

The architecture's integration of real-time local processing via the Arduino and distant data storage and analysis through the central server renders the system exceptionally efficient. The system employs a mix of sensors for smoke, temperature, humidity, and light to deliver a comprehensive fire detection mechanism. The incorporation of Wi-Fi connection guarantees constant data availability for remote access, facilitating real- time monitoring, decision-making, and prompt action. This system may be expanded to oversee bigger regions or employed in smaller, more focused applications where fire detection and environmental monitoring are essential.

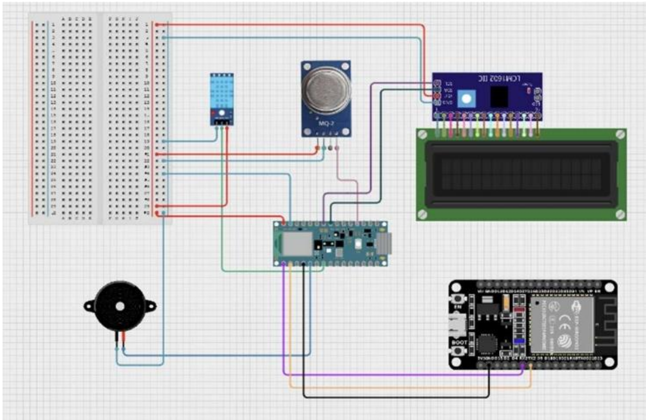

Fig. 2. Proposed Circuit Diagram for the Fire Prediction Model

B. Data Description

The dataset for this project comprises RGB color and proximity data collected under controlled conditions using the APDS9960 sensor. The data was gathered from candlelight flame experiments, simulating fire-like conditions, with variations in proximity and ambient lighting. The dataset includes RGB values representing the flame's color intensity, along with proximity readings, which serve as key features for the predictive fire detection model. This structured data was saved in CSV files, ensuring ease of access and use for preprocessing and model training.

Dataset Information: The dataset is crucial for training a predictive fire detection system as it accurately simulates real- world fire scenarios in a controlled setting. The dataset captures differences in flame behavior, including closeness and intensity, providing diverse and precise data for model training, enabling the neural network to learn and generalize fire predictions under various settings. The dataset's structure and organization guarantee good data quality, enhancing the fire detection system's precision and dependability.

C. Neural Network:

The neural network architecture for the fire detection system is constructed with TensorFlow and Keras, according to a sequential design. The input layer receives three floating-point values that reflect the normalized RGB data obtained from the APDS9960 sensor. It has 128 neurons that use the ReLU activation function in its first hidden layer. This lets the network find non-linear patterns in the data that are necessary to tell the difference between fire and non-fire situations. The second concealed layer has 64 neurons, utilizing the ReLU activation function to further enhance the characteristics collected from the sensor input. In order to minimize overfitting and enhance the network's generalization across various datasets, a Dropout layer is then included, which randomly sets 30% of the input neurons to zero during training.

The last concealed layer has 32 neurons utilizing the ReLU activation function to enhance the depth and intricacy of the network's feature extraction mechanism. It has one neuron with a sigmoid activation function in the output layer. This neuron sends out a chance between 0 and 1 that tells us if the input data fits a fire situation or not. This design enables the network to process RGB and proximity sensor data efficiently, facilitating real-time forecasts regarding fire situations with elevated accuracy and dependability.

```
Model: "sequential_12"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_44 (Dense)            (None, 128)               512

 dense_45 (Dense)            (None, 64)                8256

 dropout_5 (Dropout)         (None, 64)                0

 dense_46 (Dense)            (None, 32)                2080

 dense_47 (Dense)            (None, 2)                 66

=================================================================
Total params: 10914 (42.63 KB)
Trainable params: 10914 (42.63 KB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

Fig. 3. Layer-wise Breakdown of Neural Network Model Parameters

D. Simulation Framework

The backend is hosted on an ESP32, which serves as a central server. It receives the RGB and proximity data, processes it, and feeds it into the neural network model, which is converted to TensorFlow Lite format to run efficiently on the microcontroller. This enables the backend to make real-time predictions by analyzing sensor inputs and determining the likelihood of a fire.

In addition to making predictions, the backend is responsible for managing communication between the microcontroller, users, and authorities. Upon detecting a fire, the backend triggers alerts, sending notifications through messaging protocols or a mobile application. The ESP32's network capabilities, such as built-in Wi-Fi and Bluetooth, allow it to communicate seamlessly with external devices and cloud services. The backend also stores historical sensor data on a remote database for long-term analysis and model refinement. This combination of real-time prediction and data management ensures that users receive timely and reliable alerts, while the system continuously improves its accuracy.

ISSN (Online) 2394-2320
International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 12, Issue 03, March 2025

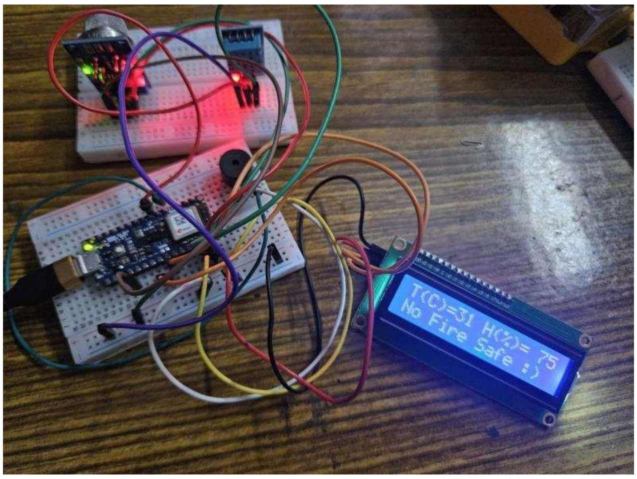

Fig. 4. Prototype of Arduino-based fire detection system displaying sensor readings and fire status

V. RESULTS AND DISCUSSIONS

The developed ANN-based fire detection system achieved a high training accuracy of 94.24% over 100 epochs, demonstrating consistent performance between training and validation with minimal overfitting. The system effectively captured early fire indicators like temperature, gas concentrations, and color changes from the input sensors, allowing for timely fire detection.

Fig. 5. Training and Validation of Accuracy Graph

Compared to traditional fire detection systems, which rely on single sensors and often suffer from delays or false alarms, this multi-sensor approach (gas, temperature, color) significantly enhanced detection speed and accuracy. The integration of proximity and color intensity data from the APDS9960 sensor allowed for early-stage fire detection, with minimal false positives. Real-time data transfer enabled continuous monitoring and system improvement.

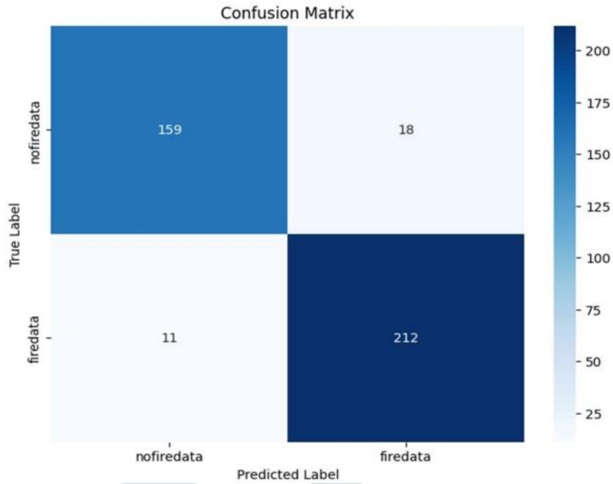

Fig. 6. Confusion Matrix of the Test Dataset

For future enhancements, expanding the training dataset to cover a wider range of environments and fire types could improve the system's adaptability and accuracy. Additionally, implementing advanced hyperparameter optimization and fine- tuning the neural network model may further reduce false positives and improve detection precision. Exploring the integration of more sophisticated sensors, such as thermal cameras or advanced smoke detectors, could also enhance performance in diverse scenarios, making the system more robust and scalable.

VI. CONCLUSION

The ANN-based fire detection system proved to be effective, cost-efficient, and reliable, with an accuracy of 94.24%. It provides timely alerts by detecting early fire indicators, making it suitable for residential and commercial applications. Future improvements in model architecture and dataset diversity could enhance its accuracy and adaptability across various environments, offering a scalable solution for fire safety.

REFERENCES

[1] P. Foggia, A. Saggese and M. Vento, "Real-Time Fire Detection for Video-Surveillance Applications Using a Combination of Experts Based on Color, Shape, and Motion," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 25, no. 9, pp. 1545-1556, Sept. 2015, doi: 10.1109/TCSVT.2015.2392531.

[2] J. B. Maheen and R. P. Aneesh, "Machine Learning Algorithm for Fire Detection using Color Correlogram," 2019 2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT), Kannur, India, 2019, pp. 1411-1418, doi: 10.1109/ICICICT46008.2019.8993246.

[3] A. Gaur Anshul and Singh, Abhishek and Kumar, Ashok and Kulkarni, Kishor S. and Lala, Sayantani and Kapoor, Kamal and Srivastava, Vishal and Kumar, Anuj and Mukhopadhyay, Subhas Chandra, "Fire Sensing Technologies: A Review," in IEEE Sensors Journal, vol. 19, no. 9, pp. 3191-3202, 1 May1, 2019, doi: 10.1109/JSEN.2019.2894665.

[4] A. Gkogkidis, V. Tsoukas, S. Papafotikas, E. Boumpa and A. Kakarountas, "A TinyML-based system for gas leakage detection," 2022 11th International Conference on Modern Circuits and Systems Technologies (MOCAST), Bremen, Germany, 2022, pp. 1-5, doi: 10.1109/MOCAST 54814.2022.9837510.

[5] D. P. Sharma, Dilli Prasad and Beigi-Mohammadi, Nasim and Geng, Hongxiang and Dixon, Dawn and Madro, Rob and Emmenegger, Phil and Tobar, Carlos and Li, Jeff and Leon-Garcia, Alberto., "Statistical and Machine Learning Models for Predicting Fire and Other Emergency Events in the City of Edmonton," in IEEE Access, vol. 12, pp. 56880- 56909, 2024, doi: 10.1109/ACCESS.2024. 3390089.

[6] A. Gaur Anshul and Singh, Abhishek and Kumar, Ashok and Kulkarni, Kishor S. and Lala, Sayantani and Kapoor, Kamal and Srivastava, Vishal and Kumar, Anuj and Mukhopadhyay, Subhas Chandra, "Fire Sensing Technologies: A Review," in IEEE Sensors Journal, vol. 19, no. 9, pp. 3191-3202, 1 May1, 2019, doi: 10.1109/JSEN.2019.2894665.

[7] V. Kumari and F. A. Ali, "Early Detection of Forest Fire Using Internet of Things," 2022 1st IEEE International Conference on Industrial Electronics: Developments & Applications (ICIDeA), Bhubaneswar, India, 2022, pp. 24-28, doi: 10.1109/ ICIDeA53933.2022.9970130.

[8] Talaat FM, ZainEldin H. An improved fire detection approach based on YOLO-v8 for smart cities. Neural Computing and Applications. 2023 Oct;35(28): 20939-54.

[9] Khalid W, Sattar A, Qureshi MA, Amin A, Malik MA, HUSSAIN K. A smart wireless sensor network node for fire detection. Turkish Journal of Electrical Engineering and Computer Sciences. 2019;27(4): 2541-56.

[10] N. Komalapati, V. C. Yarra, L. A. V. Kancharla and T. N. Shankar, "Smart Fire Detection and Surveillance System Using IOT," 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS), Coimbatore, India, 2021, pp. 1386-1390, doi: 10.1109/ICAIS50930.2021. 9395841.

[11] Diaconu BM. Recent advances and emerging directions in fire detection systems based on machine learning algorithms. Fire. 2023 Nov 17;6(11):441.

[12] Jana S, Shome SK. Hybrid ensemble based machine learning for smart building fire detection using multi modal sensor data. Fire Technology. 2023 Mar;59(2):473-96.

[13] Muhammad K, Ahmad J, Baik SW. Early fire detection using convolutional neural networks during surveillance for effective disaster management. Neurocomputing. 2018 May 2;288:30-42.

[14] Zhang D. A Yolo-based Approach for Fire and Smoke Detection in IoT Surveillance Systems. International Journal of Advanced Computer Science & Applications. 2024 Jan 1;15(1).

[15] Khalil A, Rahman SU, Alam F, Ahmad I, Khalil I. Fire detection using multi color space and background modeling. Fire technology. 2021 May;57:1221-39.