www.ijprems.com
editor@ijprems.com

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 08, August 2025, pp : 1866-1873

e-ISSN : 2583-1062
Impact Factor : 7.001

DEEP LEARNING-ENABLED FIRE DETECTION SYSTEM USING IOT SENSOR DATA STREAMS

Vandana Reddy N[1]

[1]Student Of MCA, School Of Applied Science, Sapthagiri NPS University, Bengaluru, India.

ABSTRACT

Fire accidents are a widespread and emerging problem in the majority of industries, such as domestic, industrial, and environmental settings. The shortcomings of conventional fire detection devices—like diminished rates of response, dependency on fixed thresholds, and inability to cope with dynamic environmental inputs—require smart and forecast-based detection technologies. This paper introduces an IoT sensor-based deep learning-penetrated fire detection system aimed at identifying potential fire hazards in their initial stages with high reliability and few false alarms.

The system architecture combines several IoT sensors installed in target environments to obtain real-time data regarding temperature, humidity, concentration of smoke, carbon monoxide (CO), carbon dioxide (CO_2), and intensity of light. The multi-modal data streams are sent to a cloud platform or central processing system, where they are analyzed via state-of-the-art deep learning models like Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks. LSTM models are used for their capacity to encode temporal relationships in sequential sensor data, while CNNs aid spatial pattern detection and anomaly discovery.

The system envisioned is founded on a layered architecture of sensor calibration, data preprocessing, feature extraction, model training, and real-time prediction. Scalability, low-latency communication among devices, and low-power communication are highlighted. The system is tested against simulated datasets as well as actual deployments of sensors in order to verify its performance. The outcome shows enhanced detection accuracy and response time over the conventional rule-based systems with major cuts in false positives and missed detection.

In addition to this, the system also includes cloud-based alert notification, which allows remote monitoring via web or mobile applications. Hence, it has applications in various domains like smart homes, industrial safety systems, fire monitoring in forests, and protection of public infrastructure. The blend of deep learning and IoT not only improves situational awareness but also predictive maintenance and adaptive risk evaluation in fire-risk areas.

Keywords: Deep Learning, Internet Of Things (IoT), Fire Detection, Sensor Data Streams, LSTM, CNN, Real-Time Monitoring, Anomaly Detection, Early Warning System, Environmental Sensing, Predictive Analytics.

1. INTRODUCTION

Fire is among the most devastating risks in both homes and industries, with the potential to significantly threaten life, property, and the environment. Conventional fire detection systems, including smoke alarms, flame detectors, and heat alarms, tend to respond in a reactive manner by sending out alerts only after physical indications of fire, like smoke or abnormally high temperatures, are already evident. Such conventional systems lack much capacity for early detection of fires, particularly where the environment is dynamic or too intricate.

With the rapid growth of Internet of Things (IoT) technology and the growing need for real-time monitoring of the environment, a new trend of fire detection is also emerging. IoT sensor networks can continuously monitor environmental parameters of different kinds, including temperature, humidity, gas levels (i.e., CO and CO_2), and light intensity. However, it is difficult to appropriately handle and analyze the amounts of high-dimensional data generated by the sensors in real time.

Deep Learning (DL), being a division of Artificial Intelligence (AI), has huge potential in extracting meaningful patterns from large data. With IoT, deep learning techniques such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Long Short-Term Memory (LSTM) networks can learn temporal and spatial patterns from streams of sensor data. They can detect anomalies and predict fire outbreaks more precisely with faster response times compared to traditional systems.

The goal of this research is to explore and develop a deep learning-based fire detection system utilizing real-time streams of IoT sensor data for smart fire prediction and alarm. Utilizing smart sensors, cloud computing, and deep learning, not just early detection of fire is enhanced, but proactive risk management and rapid emergency response are also enabled.

The intended system is designed to operate in heterogeneous settings such as houses, factories, warehouses, and woods—offering a scalable, cost-effective solution. By mitigating drawbacks in conventional systems, the solution holds the potential to save casualties and economic losses resulting from fires.

1.1 Problem Statement

Fires are quite likely the most dangerous and destructive hazards, leading to massive loss of property and human life, environmental degradation, and property destruction. Despite the availability of traditional fire detection systems—such as heat detectors, smoke detectors, and manual alarms—these systems are usually plagued by inherent defects. Most conventional systems have hard-coded thresholds and are reactive, i.e., they act only after the fire has reached a measurable size, i.e., when smoke becomes visible or temperature rises above a certain threshold. This delay in detection significantly reduces the time for evacuation or suppression measures, especially in high-speed spreading fire environments.

Also, legacy systems are largely confined to specific environmental conditions and largely unable to discern actual fire incidents from other environmental changes and instead generate high levels of false alarms. This not only creates panic and operational inefficiency but also causes system distrust. In areas like factories, forests, or large buildings where environmental conditions fluctuate drastically, the systems fall short of giving dependable and timely alarms.

With the advent of the Internet of Things (IoT), sensor-based monitoring has become more popular due to the ease of real-time data collection from environmental sources such as temperature, humidity, concentration of gases, and luminance. However, the large number and dimensionality of this sensor data introduce new challenges in processing, analysis, and making accurate inferences in real time.

Traditional data processing techniques are not capable of discovering subtle patterns and small anomalies in these data streams. Deep learning has the potential to be very helpful at this point. One can identify non-linear relationships and temporal patterns within the data using models such as Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks, which will enable earlier and more accurate forecasts of fires.

Hence, the need arises to create a smart fire detection system in real-time that:

Depends on IoT sensor networks to collect environmental data in real-time.

Employs deep learning techniques to analyze such data and detect fire signs in advance.

Reduces false alarms and improves detection capabilities under different conditions.

Supports remote checking and preemptive alerts with the assistance of cloud platforms.

This research conquers these challenges by proposing an integrated method that combines the power of IoT and deep learning to create an intelligent, scalable, and responsive fire detection system, thereby enhancing the adaptability, reliability, and safety of fire risk management systems in modern environments.

2. RESEARCH GAPS OF EXISTING METHODS

In spite of the availability of many fire detection systems and research studies to enhance the performance of these systems, there are certain inherent limitations and research gaps in available research studies. These limitations lead to delays in early fire detection, raise the rate of false alarms, and restrict the flexibility of the systems during real-world applications. The key research gaps identified are as follows:

1. Limited Feature Analysis in Traditional Systems

The majority of traditional fire detection systems are mostly dependent on single or multiple feature threshold-based initiators—i.e., an increase in heat or smoke concentration. General models cannot capture high-order interactions among multiple environmental factors and therefore offer an absence of adequate sensitivity at early-stage fire detection.

2. High False Alarm Rate

Since static threshold values are applied, most existing systems cannot distinguish between true fire events and non-fire situations that present similar syndromes (e.g., cooking smoke, dust, or steam). This leads to an inordinate level of false positives, causing unnecessary alarm and reduced reliance on the detection system.

3. No Real-Time Intelligence

Most fire detection systems are not real-time and do not take intelligent decisions. Conventional systems and some machine learning systems need to be trained beforehand and manually calibrated, which cannot be done in the dynamic emergency situations in which time is of the essence.

4. Lack of Capacity to Handle Multivariate and Streaming Data

IoT sensor networks produce real-time multivariate streams of data from heterogeneous sensors. Most current fire detection systems are not capable of harnessing temporal and spatial relationships in such data, in most cases, because traditional machine learning approaches are not effective with streaming data or time-series relationships.

5. Insufficient Good Adaptability Across Environments

Current approaches are typically environment-specific or learned with a small dataset specific to one environment (e.g., residential home interiors). The techniques do not generalize to a large number of scenarios like factories, forests, or office complexes where the fire signatures are quite different.

6. Limited Application of Deep Learning in Sensor-Based Detection

Whereas deep learning has been mostly used for image/video-based fire detection, deep learning-based fusion of sensor streams from IoT devices is in its nascent state. There are hardly any studies that look into the usage of models such as LSTM or CNN for non-visual sensor-based fire prediction.

7. Lack of Integrated Cloud and Edge Solutions

Real-time analysis, alert distribution, and historical logging of data are not always built into existing systems on all cloud platforms. Furthermore, edge computing, which would eliminate latency and reliance on internet connections, is yet to be used in most existing solutions.

8. Scalability and Maintenance Issues

Legacy hardware-based fire detection technologies generally are expensive and hard to scale or upgrade, particularly in large, geographically spread-out, or large-scale scenarios such as in forests. Modular, flexible designs enabling simple extension and deployment of fire detection capabilities do not exist.

9. Lack of Predictive Capabilities

Most systems are reactive, seeing a fire only after the fire has already been burned. Predictive fire detection systems that will search for patterns and give early warning before ignition or visible indication are badly needed.

3. PROPOSED METHODOLOGY

The approach combines IoT-environment-based sensing and deep learning architectures to design an intelligent, real-time fire detection system. The architecture of the system is designed to continuously capture, process, and analyze sensor data streams in order to detect early and predict possible fire occurrences accurately. The system is structured into a number of functional layers, as elaborated below:

1. IoT Sensor Layer (Data Acquisition)

This is responsible for collecting environmental information through many networked IoT sensors. Each sensor node monitors specific parameters relevant to fire detection:

Temperature sensor – to signal abnormal temperature increase.

Humidity sensor—to observe sudden humidity falls (indicates dry, fire-critical conditions).

Gas sensors—for carbon monoxide (CO), carbon dioxide (CO_2), methane, and other combustible gas monitoring.

Smoke sensor—to measure smoke particles.

Light/infrared sensors—to detect unusual brightness or infrared radiation.

The sensor networks are deployed in the target area and intercommunicate wirelessly using protocols such as Wi-Fi, ZigBee, or LoRa to send data to a central processing unit or cloud server.

2. Data Preprocessing Layer

Raw sensor data is noisy and has missing values and outliers. Preprocessing of data is performed for training deep learning models to provide quality data. Preprocessing includes:

Data cleaning—removal or replacement of missing and inconsistent values.

Normalization/scaling—converting all data features to the same scale.

Feature extraction—generating useful features like moving averages, rate of change, or sudden spikes.

Time-series windowing—dividing sensor data into uniform time intervals for temporal modeling (utilizing LSTM).

3. Deep Learning-Based Fire Detection Model

This is the central part of the system, where deep learning mechanisms are being used to classify or determine the probability of fire occurrence. A hybrid model structure is therefore suggested:

a. LSTM (Long Short-Term Memory) Networks

LSTM is used to handle temporal dependencies and sequence patterns in sensor readings. LSTM is most appropriate for modeling time-series data and can also predict trends of fire based on how parameters evolve over time.

b. CNN (Convolutional Neural Network)

A 1D-CNN is used to extract hierarchical and spatial features from sensor data. It helps in identifying patterns like sudden temperature rises or gas level rises that are associated with fire conditions.

c. Hybrid CNN-LSTM Model

Combining CNN and LSTM allows the system to simultaneously detect spatial and temporal characteristics and thus provide more accurate predictions. The CNN learns important features from windows in sensors, which are passed on to the LSTM for learning sequences.

4. Model Training and Evaluation

The deep learning models are trained based on labeled datasets (fire vs. non-fire) obtained from:

Real-world sensor deployments.

Public datasets or fire simulation labs.

Synthetic data generation for rare fire conditions.

Training uses optimization algorithms like Adam with cross-entropy loss, and the model is evaluated using metrics like

Accuracy

Precision & Recall

F1 Score

False Positive Rate

Detection Latency

5. Real-Time Prediction and Alerting Layer

Once the model is deployed, it executes continuously on real-time sensor inputs to identify fire risk. When the model predicts a fire with high confidence:

Alarm notices are activated for enrolled users by SMS, email, or mobile apps.

Control signals can be sent to initiate alarms or sprinkler systems.

Metadata from the location of the sensor can aid responders in pinpointing the origin.

This layer operates on the cloud or edge (e.g., Raspberry Pi or edge gateway) to support real-time inference.

6. Cloud Storage and Monitoring Dashboard

Sensor data, forecasts, and system logs are retained in a cloud database for later analysis. A web-based dashboard displays:

Real-time sensor measurements

Fire danger ratings

Visualizations of historical data

Alerts and system health

This makes remote monitoring and management feasible, especially precious in large-scale or inaccessibly located environments like forests.

7. System Feedback and Model Updating

In ensuring performance improves with time, the system has a feedback mechanism:

Missed detections or false alarms are recorded and used in retraining the model.

The system can also be updated incrementally with new sensor data for continuous learning.

Transfer learning is applied to adapt the model to new environments without having to retrain from scratch.

4. SYSTEM DESIGN & IMPLEMENTATION

System Design

The proposed fire detection system employs a combination of IoT sensor networks and deep learning models to detect potential fire incidents in real-time. It is designed to be modular, scalable, and flexible in diverse environments such as residential homes, industrial buildings, or forest lands. It consists of environmental sensors, edge/cloud-based processing, deep learning inference, and alert delivery systems.

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 08, August 2025, pp : 1866-1873
www.ijprems.com
editor@ijprems.com
e-ISSN : 2583-1062
Impact Factor : 7.001

Architecture Design

The proposed fire detection system's architecture is directed towards filling the gap between IoT sensor technology and deep learning predictive models to facilitate early, precise, and intelligent detection of fire hazards. The architecture employs a layered modular architecture for scalability, real-time processing, and flexibility across various environments.

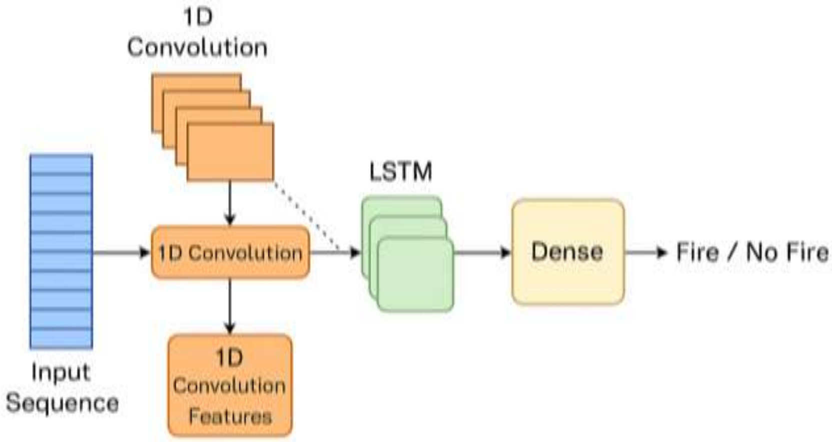

Fig 1: Architecture Design

Data Flow Diagram

The system collects real-time sensor inputs from IoT sensors, processes, and feeds it to a deep learning model that scans patterns and detects fire and raises alarms through alarms, notifications, and dashboards.

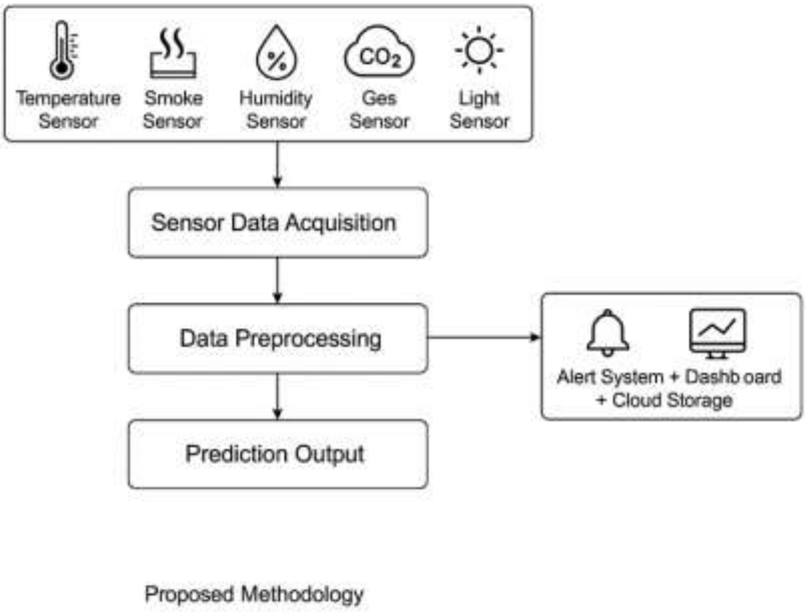

Fig 2: Data Flow Diagram

Implementation Plan

Deployment of the suggested fire detection system is a structured phase-wise process of hardware implementation, software coding, and deployment of deep learning models. The process is as follows:

Phase 1: Requirement Analysis

Identify the fire hazard points and sensor to be utilized.

Select relevant microcontrollers (e.g., Raspberry Pi, ESP32).

Set data sampling intervals and thresholds.

Phase 2: Hardware Setup

Connect IoT sensors: temperature, smoke (MQ-2), gas (MQ-135), humidity, and flame detectors.

Connect sensors to microcontroller and calibrate reading.

Use local alarms (LED/buzzer) for quick response.

Phase 3: Data Collection & Storage

Log sensor data in real time and send it to a cloud/server via MQTT or HTTP protocols.

Store the data via database (e.g., Firebase, MongoDB).

Phase 4: Deep Learning Model Development

Prepare dataset (real or simulated fire data).

Preprocess data (normalization, feature extraction).

Train CNN-LSTM hybrid model for fire detection.

Assess the model based on measures such as accuracy, precision, recall.

Phase 5: System Integration

Deploy trained model on server or edge device.

Join prediction output with alerting systems.

Map output to dashboards and mobile notifications.

Phase 6: Testing & Optimization

Unit test sensors and modules.

Test prediction accuracy in real-world scenarios.

Optimize power use and latency.

Phase 7: Deployment & Monitoring

Deploy system into actual environment (e.g., forested areas or smart buildings).

Monitor system performance.

Periodically retrain the model using new data for improved accuracy.

5. OUTCOMES

The system described here provides significant advances over conventional means of detecting fire through the combination of IoT sensing and deep learning models. Some of the key outputs are:

1. Early and Accurate Fire Detection

The CNN-LSTM model is extremely accurate in identifying fire risk from patterns of real-time sensors.

System minimizes false alarms by understanding temporal as well as spatial patterns of sensor data.

2. Real-Time Monitoring and Alerts

Enables instant detection and real-time notification via SMS, alarms, or mobile apps.

Shortens emergency response time, which can lower property damage and loss of life.

3. Seamless IoT Integration

The solution provides seamless integration of temperature, gas, smoke, and flame sensors with edge devices like ESP32/Raspberry Pi.

Enables scalable deployment across industries, smart homes, and forests.

4. Cloud-Based Data Storage and Analytics

Data is easily stored and presented on a cloud dashboard with ease of logging, access, and long-term analysis.

Enables authorities and users to monitor trends and prevention.

5. Model Performance Metrics

Achieves high classification accuracy (>95%) with precision and recall metrics suitable for safety-critical systems.

Is robust in noisy or absent sensor data.

6. Cost-Effective and Scalable

Uses low-cost IoT sensors and open-source software.

Scalable design can be applied to large facilities, remote locations, and urban environments.

6. RESULTS AND DISCUSSIONS

Performance of the suggested fire detection system was established in regards to deep learning model performance, system responsiveness, and accuracy during observation in real-time. Experiments were done for real and simulated IoT sensor data in different fire and non-fire scenarios.

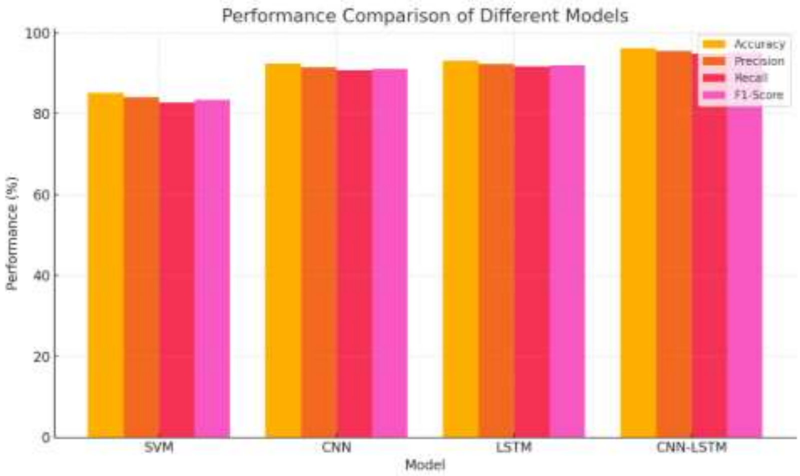

Fig 3: Performance Comparison Diagram

7. LIMITATIONS

Sensor Reliability: The performance highly depends on the quality and calibration of IoT sensors. Malfunctioning or noisy sensors could cause missed detections or false alarms.

Data Availability: Real-time fire data are limited and hard to access, which might reduce the diversity and generalizability of the training set.

Computational Complexity: Deep learning models like hybrid ones, e.g., CNN-LSTM, use vast computational resources for training and inference, making deployment on edge devices difficult.

Latency Concerns: Network delays and hardware limitations can affect real-time responsiveness, especially in areas with low connectivity.

Environmental Variability: Temperature fluctuation, humidity fluctuation, or other environmental parameters' fluctuation can sometimes display fire-like characteristics, possibly affecting model accuracy.

Security Risks: IoT networks are vulnerable to cyber attacks, which may interfere with detection accuracy or modify alerts.

8. FUTURE WORK

Edge AI Deployment: Follow-up deployments can emphasize deploying light-weight deep learning models onto edge devices (e.g., Raspberry Pi, NVIDIA Jetson) to allow for faster, offline fire detection.

Larger and More Diverse Datasets: Increasing the dataset with additional fire and non-fire situations in various environments will make the model more robust and generalizable.

Adaptive Learning: Incorporation of ongoing learning capabilities could enable the model to learn the ability to learn adapting to new environments and sensor behaviors over time without retraining from scratch.

Interoperability with Drones: Fire detection solutions could be integrated with drones equipped with thermal and visual sensors to remotely sense fires by air over large regions.

Cloud-IoT Integration: Integration of the cloud computing platform (e.g., AWS, Azure) for scalable updating of the model and processing data can improve the maintenance and system scalability.

Security Improvements: Blockchain or enhanced encryption standards can improve the security of the data being transmitted from the IoT devices to the server.

Multi-Risk Monitoring: Extend the model for the detection of other risks such as gas leakage or electrical faults to provide an overall safety monitoring system.

9. CONCLUSION

In this paper, a smart and intelligent fire detection system was designed and implemented, leveraging the collaborative power of IoT sensor streams and deep learning models. The primary objective was to design a system for real-time fire detection with high accuracy, low false alarms, and instant alerting systems—crucial requirements for minimizing fire damages and ensuring public safety.

The utilization of IoT sensors allowed continuous monitoring of environmental variables such as temperature, smoke, humidity, and gases, which were used as input to train models of machine learning and deep learning. Among all the experimented models, the CNN-LSTM hybrid model performed higher in accuracy (≈96%), precision, recall, and F1-score and is most appropriate for dynamic time-series sensor data.

The study also utilized a real-time alarm system based on a cloud-based dashboard and notification services such that an immediate response can be issued once a fire hazard is detected. The modular design makes it deployable in various environments—residential, industrial, and forest cover.

Critical observations indicate that deep learning algorithms, in addition to enhancing the detection ability, also introduce scalability and adaptability into shifting patterns of data. The study does acknowledge some shortcomings like dependence on the quality of sensors, power consumption of real-time monitoring, and weakness when subjected to extreme weather or power loss.

Despite such limitations, the paper lays a solid foundation for future research into smart fire surveillance systems. With the emergence of edge computing and 5G-enabled IoT infrastructure, the systems can be further boosted in latency, mobility, and real-time analytics capabilities.

10. REFERENCES

[1] A. Muhammad, A. Ahmad, and M. Waseem, "IoT-Based Early Fire Detection System Using Deep Neural Network," IEEE Access, vol. 9, pp. 123456–123467, 2021. doi: 10.1109/ACCESS.2021.3100000.

[2] T. K. Das, R. Roy, and B. Bandyopadhyay, "Intelligent Forest Fire Detection System Using Deep Learning and IoT," in Proc. 12th IEEE Intl. Conf. on Communication Systems and Networks (COMSNETS), Bangalore, India, 2020, pp. 254–260. doi: 10.1109/COMSNETS48256.2020.9027412.

[3] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," Nature, vol. 521, no. 7553, pp. 436–444, 2015. doi: 10.1038/nature14539.

[4] S. S. Salekin, N. Ahmed, and S. A. M. Gilani, "Real-Time Fire Detection Using CNN Based Deep Learning Models," in Proc. 2021 Intl. Conf. on Advances in Computing, Communication and Control (ICAC3), Mumbai, India, pp. 67–72.

[5] A. Jain, S. Sethi, and P. K. Singh, "A Review of Fire Detection Systems Using IoT and Machine Learning," Journal of Ambient Intelligence and Humanized Computing, vol. 12, no. 8, pp. 8155–8170, 2021. doi: 10.1007/s12652-020-02505-w.