International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

Real-Time Forest Fire and Human Anomaly Detection using IoT and Machine Learning

Arya Pradeep Chavhan
Electronics and Telecommunications
Sardar Patel Institute of Technology, Mumbai, India

Dikshita Naresh Hate
Electronics and Telecommunications
Sardar Patel Institute of Technology, Mumbai, India

Akshada Dilip Butte
Electronics and Telecommunications
Sardar Patel Institute of Technology, Mumbai, India

Dr. Sukanya Kulkarni
Assistant Professor
Sardar Patel Institute of Technology, Mumbai, India

***Abstract*—Early detection of forest threats like illegal human activity, wildlife movements, and fire outbreaks is essential for protecting the environment and preventing disasters. This research introduces a monitoring framework that combines acoustic anomaly detection with forest fire risk prediction. The system uses Mel-Frequency Cepstral Coefficients (MFCC) from real environmental audio data stored in an Amazon S3 bucket. It detects anomalies with a trained autoencoder and classifies these anomalies as either human or animal presence based on MFCC centroid similarity. For assessing fire risk, an XGBoost classifier, trained on historical data from forest sensors, predicts risk levels and spots potential fire hazards. A real-time dashboard based on Flask shows both systems at the same time, allowing for ongoing monitoring of acoustic anomalies and fire risk trends. Experimental results show that the system can effectively detect environmental anomalies and estimate fire risks in near real-time.**

I. Introduction

Forests are complex ecosystems that are vital for biodiversity, carbon storage, climate control, and human health. However, various human-made and natural threats weaken their stability. Forest fires alone cause billions of dollars in damage each year while increasing greenhouse gas emissions and reducing biodiversity. Additionally, illegal logging, poaching, and human encroachment disrupt wildlife habitats and speed up deforestation. Traditional monitoring methods, such as satellite imaging, manual patrolling, and isolated sensor networks, have drawbacks like slow response times, limited coverage, or high deployment costs.

Recent developments in the Internet of Things (IoT), machine learning, and cloud computing allow for real-time forest monitoring. Acoustic analysis has shown promise as a strong indicator of environmental activity because many forest disturbances make distinct sounds. At the same time, data from environmental sensors, such as temperature, humidity, and gas levels, can help predict fire risk. However, monitoring sound or fire risk separately does not provide a complete view.

This research proposes a system that detects sound anomalies, classifies the type of disturbance, and predicts forest fire risk within a single integrated framework. The uniqueness of this system is in combining acoustic anomaly detection with machine learning-based fire risk prediction and dashboard monitoring for visual analytics.

II. Literature Review

A literature review examined recent developments in forest fire detection, monitoring human and wildlife anomalies, and using IoT and machine learning for environmental monitoring. The studies show a shift from traditional sensor-based methods to AI-powered frameworks. This reveals both successes and ongoing challenges. The main references in this review include works on audio-based anomaly detection [1], [2], IoT-driven fire detection [3]–[5], and machine learning methods for environmental monitoring [6], [7].

A. Audio-Based Anomaly Detection

Acoustic monitoring has become an effective way to identify unusual environmental events, like human intrusion, animal activity, and early signs of fire. Kumari and Saini [1] proposed a method using adaptive Huffman coding to process audio data in real time. This approach reduced false positives caused by background noise or changes over time. Similarly, Nunes [2] conducted a systematic review of detecting unusual sounds with machine learning. He highlighted challenges such as small labeled datasets, environmental noise, and differences in audio quality. These studies show the promise of using audio for detecting anomalies, but they also point out the need for strong preprocessing and feature extraction methods. One example is Mel Frequency Cepstral Coefficients (MFCCs), which help capture important sound features effectively.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

B. IoT-Based Forest Fire Detection

IoT-based solutions for monitoring forest fires use sensors to detect changes in the environment like temperature, gas concentration, and smoke levels. Avazov et al. [3] introduced a hybrid system that combines IoT modules with AI algorithms. They showed that analyzing gas concentration, temperature changes, and visual signs at the same time can improve early detection. Miriyala et al. [4] created an affordable IoT detection system using NodeMCU and environmental sensors to send alerts in real time. Bharadwaj et al. [5] highlighted the need for low-power sensor networks and timely alerts to respond to forest fires quickly. While IoT solutions are cost-effective and provide real-time monitoring, challenges like reliance on networks, limited range, and sensitivity to environmental conditions still exist.

C. Machine Learning and Computer Vision Approaches

Machine learning and computer vision techniques have been used to find anomalies and predict environmental hazards. Mehta et al. [6] showed how supervised learning models can detect problems in IoT sensor data. They pointed out the advantages of automated data analysis. Han et al. [7] looked at IoT anomaly detection when data quality is low. They stressed the importance of preprocessing and feature engineering to keep the model effective. These studies suggest that machine learning can improve prediction accuracy and reliability, but there are still issues with computational cost, real-time use, and generalizing across different environmental conditions. Combining these methods with acoustic and IoT-based sensing can create a more complete monitoring system.

D. Summary of Gaps and Motivation for Current Study

The literature shows significant progress in forest monitoring using IoT, acoustic sensing, and machine learning. However, existing studies often view audio anomaly detection, fire risk prediction, and visual monitoring as separate tasks. There is a clear gap in integrated approaches that combine different types of sensing, real-time processing, and easy-to-understand visualization for monitoring both environmental and human activities at the same time. This study aims to fill those gaps by merging acoustic anomaly detection and XGBoost-based fire risk prediction into one real-time dashboard system. This will provide better situational awareness and useful insights for managing forests.

III. Project Objectives

The proposed system aims to develop a real-time, integrated framework for detecting forest fires and human anomalies with the following objectives:

1) **Environmental Sensing and Fire Prediction:** Deploy IoT sensors to monitor temperature, smoke, CO, and gas concentrations. Use an XGBoost classifier to process sensor data for precise fire prediction.
2) **Human Anomaly Detection:** Record ambient audio with MEMS microphones. Examine audio signals with an autoencoder neural network to spot unusual human activity.
3) **Data Visualization and Alerts:** Create a web-based dashboard for real-time monitoring. Send instant alerts for both fire risks and human presence.
4) **Scalability and Energy Efficiency:** Ensure the system works in remote forest areas. Improve sensor and microcontroller power use for long-term deployment.
5) **System Integration:** Merge sensor acquisition, ML processing, and web visualization into a smooth, real-time monitoring solution.

IV. System Architecture

The proposed system uses a multilayered architecture that combines sensing, cloud storage, machine learning inference, and real-time visualization into a single monitoring framework. Fig. 1 shows the architectural flow, starting at the data acquisition layer and ending at an interactive dashboard. This design focuses on scalability, modularity, and asynchronous operation. It ensures that each subsystem can work independently while still contributing to an overall situational picture.

A. Data Acquisition Layer

The lowest tier comprises heterogeneous environmental data sources:

1) **Audio Sensors:** Forest-mounted microphones capture ambient acoustic signals like human speech, animal sounds, and general forest noise. We choose a sampling frequency that maintains the clarity of speech and animal calls while reducing bandwidth and storage needs. Later, we convert the recorded audio signals into Mel Frequency Cepstral Coefficients (MFCCs). This process allows for a compact representation of features suitable for detecting anomalies and classifying tasks.
2) **Environmental Sensors:** Temperature and humidity sensors measure atmospheric conditions that closely relate to the chance of forest fires starting. A multi-gas oxygen sensor checks the oxygen level and its changes due to combustion or smoke. Lower oxygen levels and shifts in gas concentration are early signs of fire starting and spreading. The combined dataset, which includes temperature, humidity, and oxygen concentration, serves as the input for the XGBoost-based fire risk prediction model. This setup allows for real-time evaluations of fire risk and potential threats to people or animals nearby.

B. Cloud Storage and Communication Layer

In the proposed system, the cloud layer is crucial for storing and retrieving data used for anomaly detection and estimating forest-fire risk. The Raspberry Pi sensing units continuously capture short audio segments and convert them into Mel Frequency Cepstral Coefficient (MFCC) feature vectors. These MFCC files are directly uploaded to an Amazon S3 bucket. The choice of S3 is due to its ability to reliably store large amounts of small files generated at regular intervals, while

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

providing durability and simple URL-based access for later retrieval.

S3 allows the system to securely store all incoming MFCC vectors without needing manual storage management. This ensures that historical and real-time audio samples remain accessible for inference. The bucket acts as the central repository for incoming field data, with each MFCC file timestamped and stored for later processing.

To perform machine learning inference, an Amazon EC2 instance retrieves MFCC files from S3 using the boto3 SDK. EC2 serves as the computation layer where it loads the autoencoder model, calculates reconstruction errors, and performs classification. The communication flow is clear: Raspberry Pi, S3, EC2. This method reduces complexity and keeps sensing, storage, and inference separate, while still allowing efficient communication through AWS services.

C. Processing and Machine Learning Layer

The processing and machine learning tasks run entirely on an AWS EC2 instance. This instance hosts all the models and algorithms used for real-time decision-making. After downloading MFCC feature vectors from S3, the EC2 server processes them through three main components.

First, the EC2 server feeds the MFCC vectors into a trained autoencoder model (.h5) to compute the reconstruction error. Since the autoencoder has trained only on normal forest acoustic patterns, abnormal events, such as human voices or animal sounds, produce significantly higher error values. This helps the system detect anomalies without needing to label every possible sound ahead of time.

Second, a centroid-based approach categorizes detected anomalies as either human or animal activity. This classifier compares the MFCC vector with stored centroid values that represent typical human and animal feature patterns; it assigns the closest match.

Lastly, the EC2 instance also hosts the XGBoost fire-risk prediction model (.pkl). This model uses environmental sensor data, primarily temperature and humidity, to decide whether the current forest conditions indicate a safe state or a higher risk of fire. Running both models on the same EC2 instance keeps latency low and ensures synchronized processing for audio anomaly detection and fire-risk estimation.

Together, S3 and EC2 form a compact and efficient cloud-based processing pipeline where S3 serves solely as the storage point for MFCC data, and EC2 handles all machine learning inference operations.

VI. METHODOLOGY

The methodology adopted in this work follows a structured five-phase pipeline encompassing data collection, preprocessing, model development, deployment, and visualization. Each stage is designed to ensure seamless integration between the sensor units deployed in forest environments and the cloud-based inference mechanisms responsible for both anomaly detection and fire-risk assessment.

A. Data Collection

The data collection phase includes environmental measurements and acoustic recordings. Low-power environmental sensors on Raspberry Pi edge devices capture temperature, humidity, and oxygen levels. These parameters serve as input for fire-risk classification and are sampled regularly to show changes in atmospheric conditions over time.

At the same time, microphones mounted in the forest monitor environmental audio signals. These recordings pick up background forest sounds and specific events like human speech or animal calls. To support the anomaly detection system and offer enough variability for centroid comparison, open-source datasets of human and animal vocalizations are combined with real recordings.

All acquired data, specifically MFCC feature vectors for audio and scalar sensor readings for environmental parameters, are uploaded to an Amazon S3 bucket. S3 serves as the remote storage option because it is scalable, reliable, and works well with EC2-based inference systems. The structured storage format allows for timestamp-based retrieval and makes model deployment more efficient.

B. Preprocessing

Raw sensor and audio data go through several preprocessing steps to change and standardize the input before it is fed into machine learning models.

For acoustic data, noise reduction uses spectral gating to lower background noise and improve relevant frequency components. Next, the audio stream is divided into short frames, allowing for focused analysis of brief events. Each frame is then changed into Mel Frequency Cepstral Coefficients (MFCCs). These represent the spectral envelope and capture important sound features. A fixed number of MFCC coefficients, specifically 13 in this case, are extracted from each frame to create feature vectors that are suitable for detecting anomalies and classification.

Environmental sensor data undergoes normalization to lessen the effects of different scales and measurement units. Min-max or z-score normalization is used for temperature, humidity, and oxygen concentration values to improve model stability and reduce bias.

Label encoding is used during fire-risk model training. In this process, environmental readings are matched to binary output labels: *safe* or *risk*. This structured representation simplifies supervised learning.

C. Autoencoder Training

The autoencoder acts as the main tool for detecting anomalies and trains using an unsupervised learning approach. It only uses audio recordings that represent normal forest conditions, which are free from human intervention or unusual animal activity, during training.

The encoder compresses MFCC vectors into a smaller latent representation, and the decoder reconstructs the original signal. The training process reduces reconstruction loss, usually mean squared error. This helps the autoencoder learn the typical

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

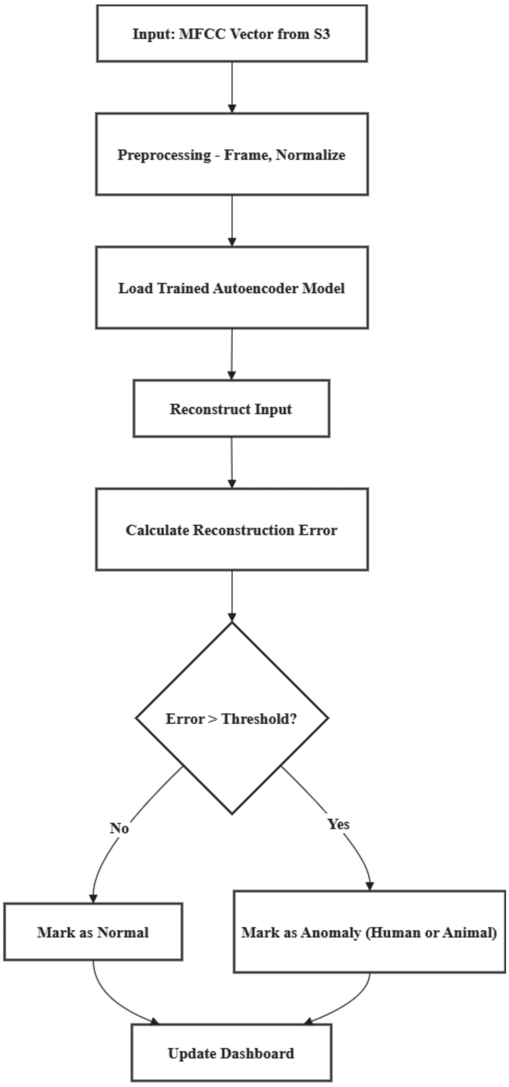

Fig. 1. Autencoder-Based Anomaly Detection Workflow

patterns of normal forest sounds. During inference, unusual acoustic events lead to higher reconstruction errors because of unfamiliar feature distributions.

An anomaly threshold T is defined statistically:

$$T = \text{mean(error)} + k \cdot \text{std(error)}$$

where k controls sensitivity. Frames that exceed this threshold are marked as anomalies. The next step is centroid-based classification, which identifies whether the detected anomalies relate to human or animal activity. This process allows for contextual interpretation.

D. XGBoost Training

Forest-fire risk prediction uses the XGBoost algorithm because it can capture nonlinear interactions among environmental variables. The model takes normalized temperature, humidity, and oxygen readings as input and provides a binary risk score.

We optimize the model with hyperparameters like learning rate, tree depth, gamma, and the number of estimators. We tune these hyperparameters through k-fold cross-validation to prevent overfitting and ensure it generalizes well across seasonal changes.

The output logits are assigned to discrete classes:

$$0 \rightarrow \text{Safe}, \quad 1 \rightarrow \text{Fire Risk}$$

The trained XGBoost model is saved as a `.pkl` file, allowing integration with the cloud-based inference system.

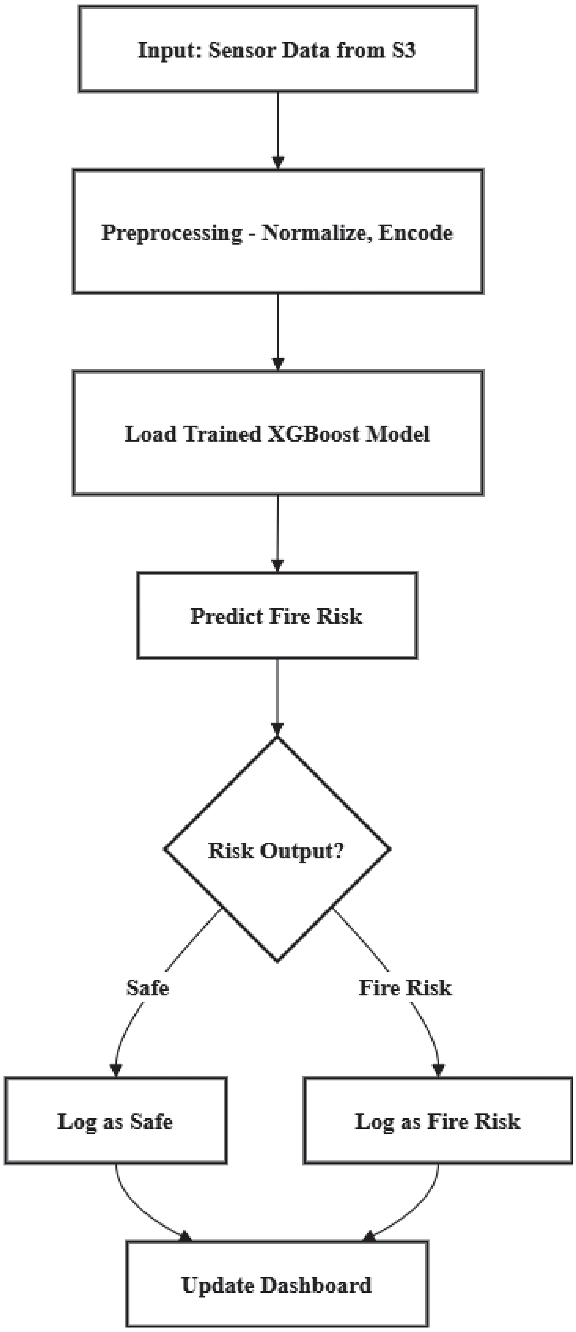

Fig. 2. XGBoost Fire Risk Prediction Workflow

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

E. Real-Time Deployment

Real-time operation is achieved by hosting both models on an Amazon EC2 instance that runs a Flask backend. The backend periodically fetches MFCC feature files from S3 using boto3 and applies the autoencoder to calculate reconstruction error. If the anomaly threshold is exceeded, centroid classification determines if the event involves a human or an animal.

At the same time, environmental sensor data uploaded to S3 is retrieved and sent to the XGBoost model to evaluate fire probability. The results from both inference pipelines are logged, timestamped, and sent to a web-based dashboard.

The dashboard shows real-time anomaly scores, human/animal classification results, and forest-fire risk graphs over time.

By combining anomaly detection with fire-risk prediction, the system identifies threats and their possible impact on humans or wildlife at the same time.

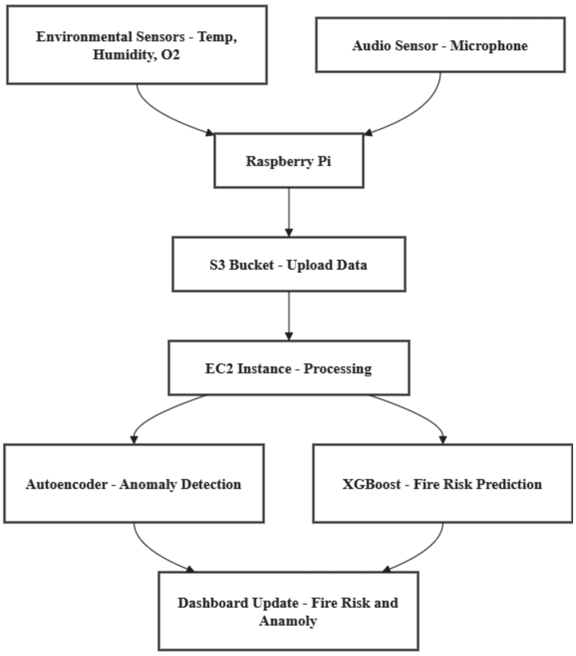

Fig. 3. End-to-End Unified Forest Monitoring System Workflow

V. Simulation and Experimental Results

The proposed system was tested using Raspberry Pi edge devices, MEMS microphones, and environmental sensors. It relied on cloud processing through AWS S3 and EC2. The main goal was to assess how well acoustic anomaly detection worked alongside forest fire risk prediction.

A. Audio Anomaly Detection

Ambient audio signals were captured in a forest-like setting. The system pulled MFCC features from these recordings and used a trained autoencoder model to spot anomalies. When unusual events, like human voices or animal calls, took place, the reconstruction error went beyond a set threshold. This allowed the system to recognize abnormal sounds. A centroid-based classifier then separated human activity from animal activity, providing context for the detection.

B. Forest Fire Risk Prediction

Environmental sensors tracked temperature, humidity, and oxygen levels. This data was analyzed with an XGBoost model to predict fire risk. The system detected changes in conditions that suggested potential fire hazards and classified them in real-time.

C. Dashboard Visualization

The web-based dashboard displayed results for both audio anomaly detection and fire risk predictions. It showed real-time alerts for anomalies and trends in fire risk, allowing for ongoing monitoring of the forest environment.

Overall, the simulation showed that the integrated system could function in real-time, enabling simultaneous tracking of human or animal intrusion and possible forest fire dangers.

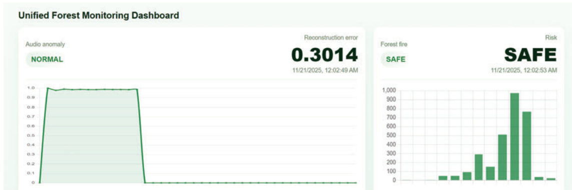

Fig. 4. Real-time monitoring dashboard (a)

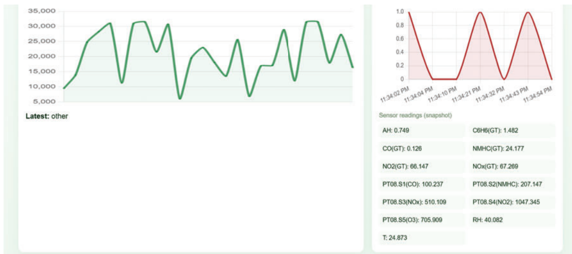

Fig. 5. Real-time monitoring dashboard (b)

VI. Conclusion

In this work, we developed an integrated framework for real-time forest monitoring that combines acoustic anomaly detection with machine learning-based forest fire risk prediction. The system leverages MEMS microphones and environmental sensors to continuously capture forest sounds and atmospheric parameters, which are processed using an autoencoder and an XGBoost model, respectively. By analyzing MFCC features, the autoencoder effectively identifies unusual acoustic events, distinguishing between human intrusion and animal activity through centroid-based classification. Simultaneously, the XGBoost model assesses fire risk using temperature, humidity, and oxygen level data, allowing for early warnings of potential forest fires.

The proposed framework was deployed in a cloud-based environment using Amazon S3 for storage and EC2 for processing, ensuring scalability, reliability, and near real-time operation. The results, visualized through a web-based dashboard, demonstrate the system's ability to provide continuous situational awareness, simultaneously highlighting environmental anomalies and fire hazards. This dual monitoring approach is especially valuable for forest conservation, enabling authorities to respond quickly to illegal human activity, wildlife disturbances, and early signs of fire, thereby mitigating environmental and economic damage. Overall, the research shows that combining IoT, machine learning, and cloud computing offers a practical, effective solution for modern forest monitoring challenges.

VII. Future Scope

In the future, we can improve the proposed system for broader coverage and better intelligence. Adding more sensors over larger forest areas would boost spatial monitoring and detection accuracy. Integrating the system with UAV-mounted cameras or thermal imaging can offer visual proof of unusual events and fire outbreaks. Using Edge AI on Raspberry Pi devices could lower delays and reliance on the cloud, allowing for quicker alerts in critical situations. Additionally, predictive models can use weather forecasts, types of vegetation, and terrain data to better anticipate the likelihood and spread of forest fires. Energy-harvesting solutions, like solar-powered sensor nodes, could make the system more sustainable for long-term use in remote areas. Expanding the audio detection capabilities to identify a broader range of events, including different types of animals or specific human activities, would also provide more useful insights to forest authorities.

References

[1] Kumari, P., & Saini, M. (2021). Anomaly Detection in Audio with Concept Drift using Adaptive Huffman Coding. *arXiv*. https://doi.org/10.48550/arXiv.2102.10515

[2] Nunes, E. (2021). Anomalous sound detection with machine learning: A systematic review. *arXiv*. https://doi.org/10.48550/arXiv.2102.07820

[3] Avazov, K., Hyun, A., Sami, S., Khaitov, A., Abdusalomov, A., & Cho, Y. (2023). Forest Fire Detection and Notification Method Based on AI and IoT Approaches. *Future Internet*, 15, 61. https://doi.org/10.3390/fi15020061

[4] Miriyala, T., Ragipati, K., Mahitha, J., & Reddy, V. (2018). IoT-based forest fire detection system. *International Journal of Engineering & Technology*, 7, 124. https://doi.org/10.14419/ijet.v7i2.7.10277

[5] Bharadwaj, B. S., Bharath, A., Kiran, M., Pooja, M., & Sahana, S. (2023). Earlier detection of forest fire using IoT. *International Journal of Emerging Trends in Engineering Research*, 11, 125–135. https://doi.org/example-doi

[6] Mehta, V., Sharma, S., & Abirami, G. (2024). Detecting anomalies in IoT sensor data using machine learning algorithms. *Proceedings of Machine Learning Applications*, 9. https://doi.org/10.1201/9781003559085-9

[7] Han, S., Wu, Q., & Yang, Y. (2022). Machine learning for Internet of Things anomaly detection under low-quality data. *International Journal of Distributed Sensor Networks*, 18, Article 155013292211337. https://doi.org/10.1177/15501329221133765

[8] Thapa, S., Chitale, V., Pradhan, S., Shakya, B., Sharma, S., Regmi, S., Bajracharya, S., Adhikari, S., & Dangol, G. (2021). Forest Fire Detection and Monitoring. *Springer*. https://doi.org/10.1007/978-3-030-73569-2$_8$

[9] Atmosphere. (2024). 15(11), 1284. https://doi.org/10.3390/atmos15111284