# Early Forest Fire Detection Using Mesh Network

Nishitha Ravella
Computer Science
Cleveland State University
Cleveland Ohio United States
n.ravella31@vikes.csuohio.edu

Samarendra Pitta
Computer Science
Cleveland State University
Cleveland Ohio United States
s.pitta53@vikes.csuohio.edu

Vinay Kumar Buddi
Computer Science
Cleveland State University
Cleveland Ohio United States
v.buddi@vikes.csuohio.edu

Harichandana Jalagam
Computer Science
Cleveland State University
Cleveland Ohio United States
h.jalagam@vikes.csuohio.edu

## ABSTRACT

The Internet of Things will play a significant part in today's world. These devices are specifically intended for monitoring, and the data collected is transferred and stored on the AWS cloud. This paper talks about the design and implementation of the Early Forest Fire Detection using Mesh Network, which was created by utilizing IOT components such as the Raspberry Pi 4, DS18B20, Raspberry Pi camera, and MQ135.

The deployed program would use sensors to collect data depending on temperature, smoke. The collected data would be transported to the AWS cloud, which helps in monitoring of data from anywhere in the world. This project helps in finding fire in forest at early stage to prevent damage to wildlife and vegetation in forest.

## CCS CONCEPTS

• IOT • Wireless Communication • Hardware • Programming • AWS Cloud • Software

## KEYWORDS

Internet of things, AWS Cloud, Forest Fire Detection, sensors, Raspberry Pi, DS18B20, MQ 135, Putty, SSH, BATMAN routing protocol and Mesh Network.

**ACM Reference format:**

Nishitha Ravella, Samarendra Pitta, Vinay Kumar Buddi and Harichandana Jalagam. 2023. Early Forest Fire Detection using Mesh network. In *Proceedings of ACM Woodstock conference (WOODSTOCK'18). ACM, New York, NY, USA, 2 pages.* https://doi.org/10.1145/1234567890

## 1 INTRODUCTION

In the current world, Internet of Things (IOT) is emerging technology which have multiple applications in the real world. The devices used are capable of detecting, processing, and transferring data wirelessly to the cloud. The cloud would store and analyze the data before presenting it to humans into readable format.

Forest plays a major role in ecosystem on earth. But the problem is sometimes fire ignites in the forest which leads to air pollution and damaging of vegetation and wildlife in forest. Another issue that arises is that in the case of a forest fire, the authorities are slow to respond, one of the causes is lack of forest fire information. As a result, we conduct research with the goal of detecting fires and informing authorities through an email in the case of fire.

Another issue arises if there is no network service provider for transmitting fire information throughout the forest. The idea is to create a public network that allows fire detectors to communicate data out of the forest in the case of a fire. As a result, we require a gadget capable of transmitting data across the forest for which we can use Wi-Fi for shorter distances.

The system that we build involves planting multiple nodes in the forest, each of which is equipped with a DS18B20 sensor as a temperature sensor reader, as well as a MQ-135 sensor as a smoke detector. We used Pi camera to capture images and store them in AWS cloud. We created a wireless mesh network system.

In AWS cloud, we use many services like AWS S3 bucket, AWS IOT core, AWS Rekognition, AWS Dynamo DB, AWS Lambda, AWS SNS and IAM. We integrate all the services to store and analyze data, and to trigger an email upon detection of fire.

In this forest fire detection system, we designed and implemented such that a user can review the data in real-time from across the world. The most important advantage of AWS cloud is we can scale according to our needs. Suppose if we need to add more devices or increase the area of detection in forest. We can achieve it by using AWS cloud.

### 1.1 Problem Statement

Forest Fires had become a major threat worldwide, leading to many negative impacts on human habitat and forest ecosystems. Climate changes and the greenhouse effect are some of the consequences of this devastation. A higher proportion of forest fires are caused by human activity. Therefore, to minimize damage caused by forest fires, it is necessary to detect forest fires at the initial stages. This project can be used to detect forest fires at the initial stage using wireless sensor networks.

# 2 Evaluation

## 2.1 Existing System

Earlier versions of the Forest Fire Detection System used connected sensors to communicate data to the standalone application workstation, and it had limited connectivity choices and data storage capacity, with data kept in a local server based on the intended architecture. Some of the drawbacks of utilizing the present Forest Fire detection system include that it is expensive to buy and install, that it requires frequent maintenance to work effectively, that it is hard to deploy, and that it is vulnerable to bad factors.

# 3 Design and Analysis

## 3.1 Raspberry Pi

Raspberry pi is small and powerful single-board computer. It has a Broadcom system-on-chip (SoC), which includes a CPU, GPU, and RAM. To run raspberry Pi, we need install OS called ‘Raspbian’. We connected data pins, Ground pins (GND) and Power (VCC) to appropriate sensors. Due to its size, it consumes less power but powerful. For Pi camera we separate slot in board.

Fig 1. Raspberry Pi module connected to power.

## 3.2 Sensors

3.2.1 DS18B20: The DS18B20 sensor is a digital temperature sensor which uses 1-Wire protocol which means multiple DS18B20 sensors can be connected to single raspberry Pi using a single data line. Since the temperature in forest fire can reach higher temperatures. We need to use sensors which can withstand higher temperatures and give accurate data.DS18B20 has temperature measurement range from -55°C to +125°C, which is suitable for forest fire detection. In DS18B20 sensor we have 3 pin interfaces, consisting of power (VDD), ground (GND), and data (DQ) lines. The Power (VDD) is connected to pin number 2, ground (GND) is connected to pin number 6 and data pin connected to pin 7 or GPIO 4 in raspberry Pi.

Fig 2. DS18B20 sensor

3.2.2 MQ135: To measure the air quality The MQ-135 gas sensor is used, which detects a wide variety of gases including carbon monoxide, sulfur dioxide, ammonia, smoke, and nitrogen oxides. MQ135 has great sensitivity and rapid reaction, as well as a long life and ease of implementation. It has an operating voltage of up to +5V and an analog output voltage of 0V to 5V. The MQ-135 may function as an analog or digital sensor. In our system we connected VCC to pin number 1, ground to pin number 9 and digital pin to pin number 13 or GPIO 27. Figure 3 shows an image of the MQ-135 gas sensor.

Fig 3. MQ-135

3.2.3 Pi camera: The Raspberry Pi Camera is a small camera module designed to work with Raspberry Pi single-board computers. It connects to the Raspberry Pi's CSI (Camera Serial Interface) connector, allowing for simple board integration. Raspberry Pi Camera provides both still image capture and video recording, making it a versatile option for a variety of visual tasks. There are visible light and infrared camera modules available, allowing it to be used in a variety of lighting settings.

Fig 4. Pi Camera

## 3.3 AWS Cloud

AWS cloud collects the data from raspberry pi and stores the temperature and smoke data in AWS DynamoDB which comes from DS18B20 and MQ-135. The data from Pi camera is sent to

S3 bucket. We used lambda function to trigger an email when an image with fire is inserted into S3 bucket.

To classify the image, we trained AWS rekognition by giving fire images and non-fire images dataset and train the model which takes about 60-120 mins to create a new AI model. After the model is trained click on start model to use trained model in AWS. We used SNS topic to send an email when the lambda function is triggered.

## 3.4 Block Diagram

Figure gives the block diagram of Forest fire detection using mesh network in which DS18B20, MQ135 and pi camera sends the data to AWS Cloud. Then in cloud it triggers the events to send an email if fire is detected.

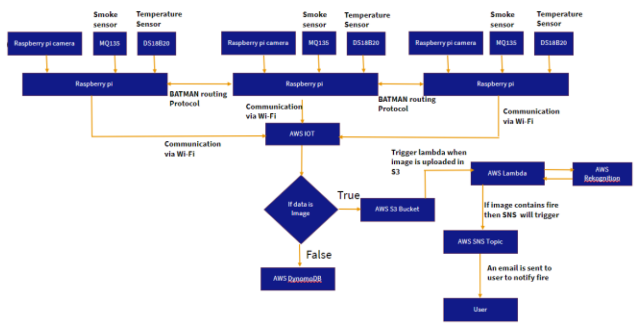


Fig 5. Block Diagram for Forest Fire Detection System

## 4 Proposed System

The proposed system is a real-time IOT Forest Fire detection system that uses raspberry Pi and other sensors to monitor the fire with different parameters like temperature and smoke. We are using DS18B20 to detect elevated temperatures, MQ-135 to detect air quality and Pi camera to capture images every 5 seconds while the raspberry Pi will send the data over Wi-Fi. This data is then uploaded to AWS cloud to detect whether the uploaded image contains fire or not. The data from DS18B20 and MQ-135 will be sent to AWS DynamoDB which stores data in tables with timestamps. Email alerts are sent when an image with fire is uploaded.

## 5 Implementation:

Implementation is critical to the success of any advanced or cutting-edge technology; creating the project plan and concrete architecture assisted us in completing the entire environmental monitoring system. The following points will outline how the project was carried out.

1. Firstly, all Raspberry Pi’s are connected to the same Wi-Fi network.
2. We installed BATMAN, a routing protocol which establishes nodes to create mesh network.
3. To manage mesh network, a utility called batctl is installed in raspberry Pi. By using the below command. sudo apt-get install -y batctl
4. Bread board acts as a medium to make connection with raspberry Pi and sensors.
5. Create a thing in AWS IOT core which makes MQTT connection to raspberry Pi and download the certificate, private key and root key which are created during IOT core.
6. Upload all files into all raspberry Pi’s and store them in a particular folder.
7. All the required hardware connections are made with DS18B20, MQ-135, Pi camera and raspberry Pi.
8. We trained an image classification model using AWS Rekognition which contains two folders fire_images and nofire_images.
9. By using the above two folders we trained model and got F1 score of 1.0, it is score of confidence and accuracy.
10. The data collected from sensors will be sent to AWS cloud from raspberry Pi via Wi-Fi.
11. Data collected from DS18B20 and MQ-135 will be sent to AWS IOT Core.
12. To transfer data from AWS IOT Core to AWS DynamoDB we used rules in IOT core page which transfers data to table by giving DynamoDB table name, partition key and value.
13. To capture image, we used Pi camera, which captures a photo every 5 seconds and sends data to S3 bucket.
14. S3 bucket can be accessed by giving access key and secret key of the bucket in python script file.
15. We created a lambda function which triggers when an image is inserted into S3 bucket.
16. In lambda function script we fetch the S3 Bucket details and use pretrained model from AWS Rekognition to find if uploaded image file contains fire or not.
17. If the image contains fire, then an AWS SNS topic will trigger, and email will be sent.

## 6. Experimental Results

- Figure 6, Figure 7, and Figure 8 will show the integration and assembly of hardware components for forest fire detection system. In the system we have raspberry Pi, DS18B20, MQ-135 and Pi camera are connected to each other via connecting wires.

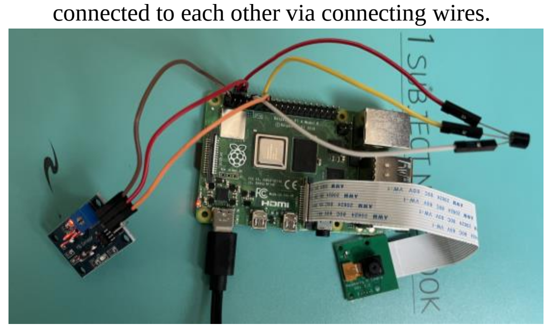

Fig 6. Raspberry Pi 1 hardware connections

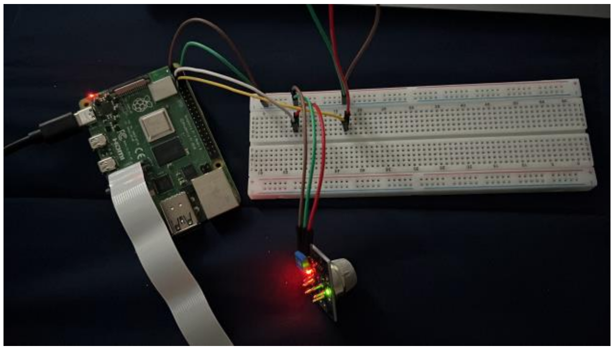

Fig 7. Raspberry Pi 2 hardware connections

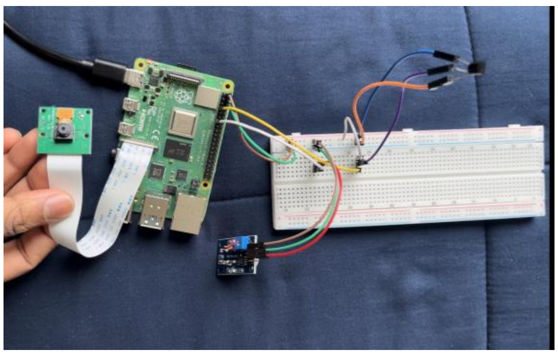

Fig 8. Raspberry Pi 3 hardware connections

Table 1. will provide the details of the system hardware, display, and cloud service.

| System Component | Details |
|---|---|
| Sensors | DS18B20, MQ-135, Pi camera |
| Network Connectivity | Wi-Fi |
| Microprocessor | Raspberry Pi 4 |
| Cloud | AWS Cloud |
| Routing protocol | BATMAN |

Table 1. System Component Details This section majorly concerns about the experimental outcomes of our Forest Fire detection system.

- Figure 9 has command prompt which shows we upload script.py, root, private key, and certificate file to all raspberry pi one by one by executing the commands given in figure 9.

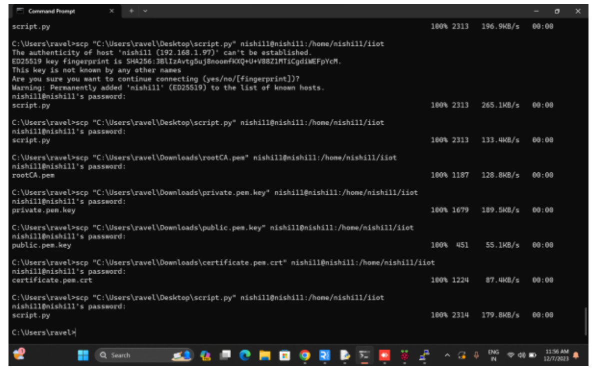

Fig 9. Uploading of files to raspberry pi from windows system.

- Figure 10 shows two terminals which are connected to two raspberry Pi by using Putty. We ping a raspberry Pi from other raspberry Pi terminal which gives packets in return which shows mesh network.

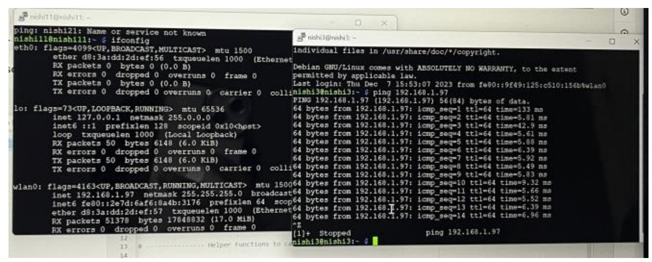

Fig 10. Two Terminals of raspberry Pi shows Mesh Network

- Figure 11, Figure 12, and Figure 13 show that python file is executed in all raspberry pi's.

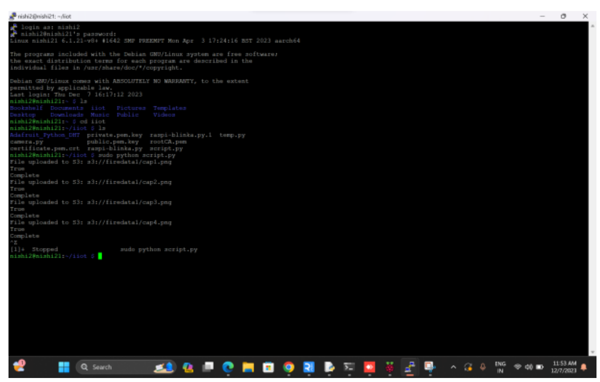

Fig 11. Execution of python file on Raspberry 1

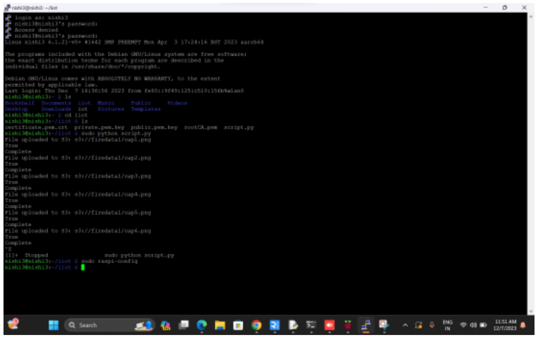

Fig 12. Execution of python file on Raspberry 2

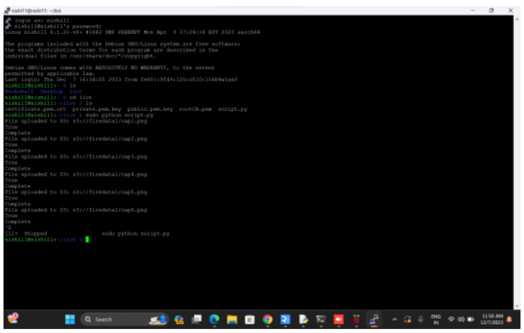

Fig 13. Execution of python file on Raspberry 3

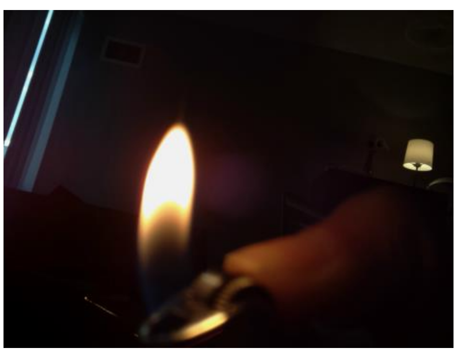

Fig 14. Fire image captured by Pi camera.

- Figure 15 shows that data captured from Pi camera is sent to S3 bucket.

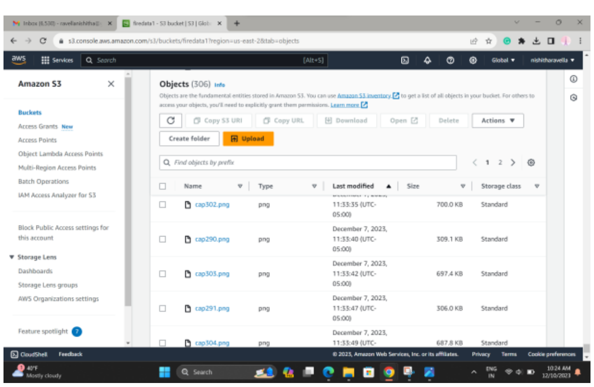

Fig 15. S3 bucket with image objects from Pi camera

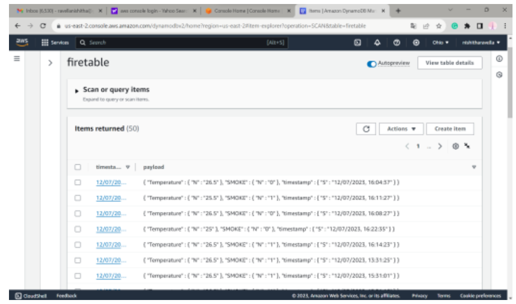

Fig 16. Temperature values are sent to AWS Dynamo DB

- Figure 17 shows that we created a trained model using dataset with fire and non-fire images.

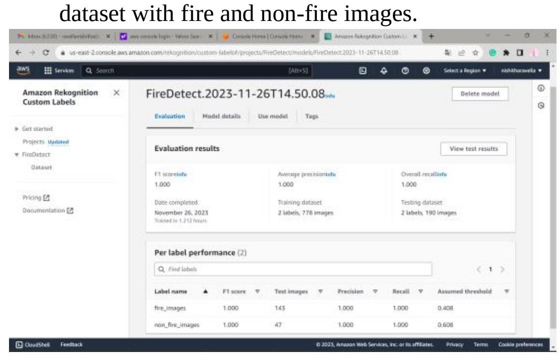

Fig 17. Created a model with F1 score of 1.00.

- Figure 18 shows that we started the model to use in forest fire system.

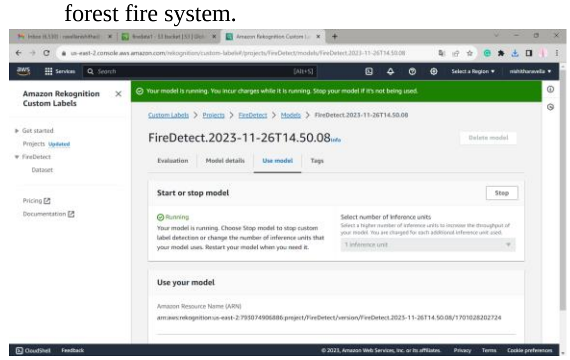

Fig 18. AWS Rekognition model is started.

- Figure 19 show flow of Lambda function

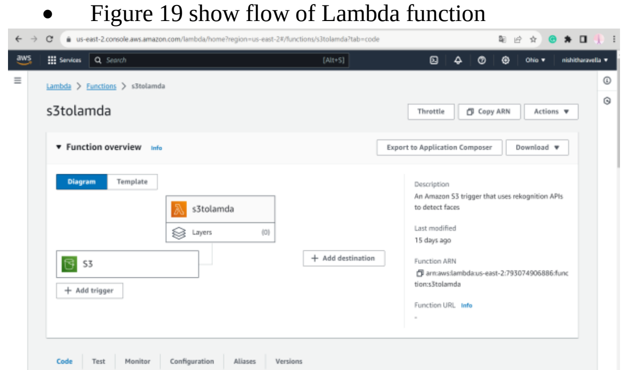

Fig 19. Lambda functions triggers on uploading a file in S3.

- Figure 20 shows that we created a SNS topic with endpoint as email.

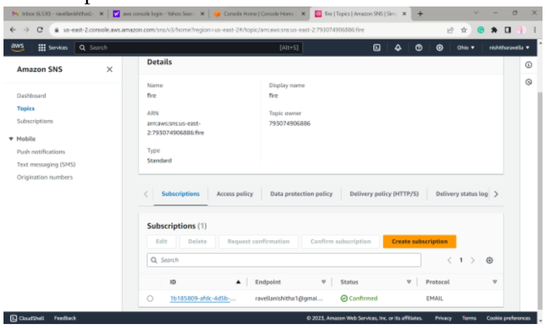


Fig 20: SNS named fire is created with email protocol.

- Figure 21 shows that mail is successfully received when fire is placed in front of Pi camera

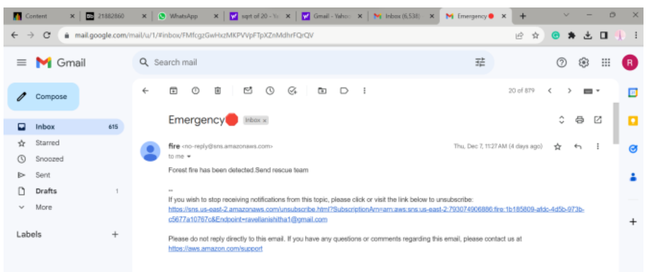


Fig 21. An email is received through AWS SNS when fire image is detected.

## TEAM MEMBERS ROLE:

- Nishitha: Developing the python code to integrate raspberry pi and AWS Cloud. Integration AWS services that are required for the classification of images and alerting the users with notification.
- Samarendra: Integration of mesh network data for analysis and reporting.

## CONCLUSION:

We also successfully connected AWS cloud to raspberry Pi system which gives use real-time forest fire detection. We made sure that proper data is sent to cloud and analysis of data is done correctly for which we used AI model with F1 score of 1.00.

## ACKNOWLEDGMENT

The major goal is to monitor and gather data on various environmental conditions in real time, as well as to be able to download and analyze data. It is a low-cost, effective, and dependable solution for monitoring and analyzing data from IoT sensors. You may acquire superior safety precautions and wellbeing by employing this technique.

## REFERENCES

[1] Gokay Saldamli, Sumedh Deshpande, Kaustubh Jawalekar, Pritam Gholap June 2019. *Wildfire Detection using Wireless Mesh Network*. Computer Engineering Department, San Jose University, CA, 95192, USA.

[2] Adnan, A. Ejah Umraneni Salam, Arham Arifin, Muhammad Rizal. October 2019. *Forest Fire Detection using LoRa Wireless Mesh Topology*. Computer Engineering Department, Hasanuddin University, Indonesia.

[3] Wardi, Dewiani, M Baharuddin, S Panggalo, MFB Gufran. Sepetember 2019. *The Performance of Routing Protocol OLSR and BATMAN in Multi-hop and Mesh Ad Hoc Network on Raspberry Pi.*