Research Article
https://doi.org/10.33904/ejfe.1383234

Eur J Forest Eng 2023, 9(2):80-87

EJFE

An Internet of Things Platform for Forest Monitoring

Mustafa Sanlı

Aselsan Inc., 06200 Yenimahalle, Ankara, Turkiye

Abstract

Forests have a very important function in sustaining the natural life on our planet. However, wildfires, landslides, uncontrolled tree cutting, poaching, arson, and many other dangers threaten the forests and natural resources. Therefore, effective monitoring and observation of forests is crucial. This study explains the development and implementation of an Internet of Things (IoT)-enabled forest monitoring system as an innovative solution that will contribute to the protection of forests. The presented system provides real-time climate data in forestlands by using microcontrollers, low-cost sensors, data communication, and cloud platforms. It collects important information such as temperature, humidity, air quality, and ecological activities. Fire detection is achieved by associating the increase in CO gas concentration with the increase in temperature. Landslides are detected by measuring the acceleration of soil movement in 3 axes. Additionally, the system includes advanced machine learning-based acoustic tracking techniques to detect chainsaws, motor vehicles, screams, shouts, and gunshots. The IoT platform provides a web-based user interface and other tools to system users such as forest managers and researchers. These tools detect early signs of threats such as wildfires, landslides and illegal activities in forests. Our tests demonstrate the system's effectiveness in providing information for protecting and managing forests.

Keywords:, IoT, Sensor network, Information systems, Forest monitoring, Embedded systems, Artificial Intelligence.

1. Introduction

Forests play an important role in our planet's ecosystem. They provide a wide range of economic and social benefits. In addition, forests provide valuable resources that are used in industry, agriculture, and tourism. However, challenges such as illegal tree cutting, climate change, and natural disasters endanger these ecosystems. Advanced monitoring systems are required to protect and manage forests.

The rapid development of Internet of Things (IoT) technology resulted in the development of new approaches in the field of data collection. IoT technology enables to collect environmental data by using sensors, microcontrollers and various communication protocols. Our point is, this capability has high potential to change the way we monitor and protect forests. Designing IoT-based monitoring systems that provide real-time data collection and analysis enables more effective monitoring of forests.

This paper explores the design and implementation of an IoT-based forest monitoring system. By using the power of IoT, this system aims to ensure better management of forest ecosystems. It also helps to identify early signs of threats and reduce potential risks.

As the climate change continues to impact our planet, the need for robust forest monitoring systems becomes increasingly urgent. This paper presents an IoT-based forest monitoring system that can contribute to efficient forest management and conservation.

Various technology solutions have been presented in the past to monitor the forest environment. Several of these studies collected environmental data using different types of sensors (Yan et al., 2016; Gulci et al., 2018; Pokhrel and Soliman, 2018; Zope et al., 2020; Patel et al., 2020; Wiame et al., 2021; Imamoglu and Tas, 2022; Selle et al., 2022; Dampage et al., 2022; Krishnamoorthy et al., 2023). Another part of the studies was on detecting fires in the forest using video processing techniques (Nai-meng and Wan-jun, 2020; Al-Masri, 2021; Hu, 2021; Madkar et al., 2022; Nihei et al., 2022; Seric et al., 2022; Yang et al., 2022). However, none of these studies offer a complete system that protects forests from internal and external threats. Also, video processing-based solutions have additional challenges, such as placing cameras in hard-to-reach areas of forests and delivering electricity to these areas. In this paper, we present the design of an IoT-enabled, scalable, end-to-end forest monitoring system

*Corresponding Author: Tel: +90 312 5926000 E-mail: msanli@aselsan.com.tr
Received: 30 October 2023; Accepted: 13 December 2023

that leverages artificial intelligence techniques to monitor forests. The contributions of our proposed system can be summarized as follows:

- We offer an end-to-end system that includes IoT node hardware design, web server and database creation, web page development for the user interface, and node management infrastructure.
- Our proposed system can detect environmental conditions such as temperature, humidity, light intensity, and forest fires in hard-to-reach areas.
- Our solution uses cutting-edge artificial intelligence methods to detect tree cutting, motor vehicles, human shouts, and screams from ambient sounds.
- Land movement is detected by the accelerometer in 3 axes and compared with a model to detect landslides.
- Our custom web server design enables the scalability of the system by allowing the connection of many IoT nodes simultaneously. It also allows the use of load-balancing techniques to increase scalability further when necessary.
- The system management infrastructure allows users to define new IoT nodes in the system, prohibit compromised nodes from sending data, and set data collection intervals.

2. Materials and Methods

2.1. System Architecture

Our proposed system consists of monitoring units (IoT nodes), a web server and a web page developed for user interface. Figure 1 shows the main building blocks of our proposed architecture. In a typical use case, a number of monitoring units are spread throughout the forest, while system users access the collected data via the web page. Our proposed architecture can support the connection of hundreds of monitoring units thanks to its scalable software design.

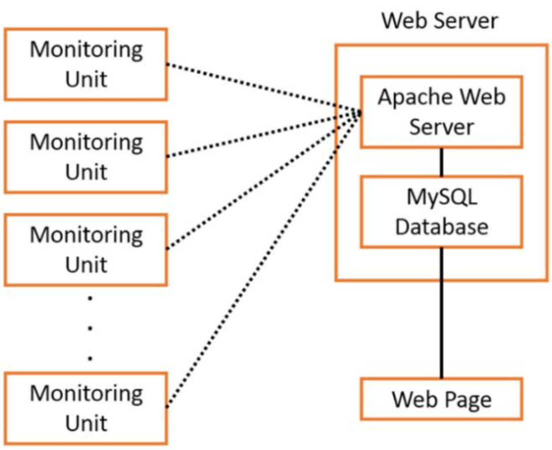

Figure 1. Block diagram of the system

2.1.1. Monitoring unit

The monitoring unit includes a Raspberry Pi microcomputer, an Arduino Uno microcontroller board, a GSM modem, and various sensors to collect environmental data. The sensors used in the monitoring unit are light, air quality, weather sensors, and accelerometer. Additionally, a microphone is utilized to listen to sounds in the forest. A photo resistor is used to quantify light intensity as a light sensor. Air quality sensor MQ7 is utilized to determine the CO concentration in the environment. The accelerometer is used to detect movement in the *x, y,* and *z* axes. Collected information is used to detect landslide occurrences. The light sensor, air quality sensor, and accelerometer provide analog outputs. The internal 10-bit analog-to-digital converter of the Arduino Uno microcontroller is used to transfer these signals to the digital domain. Arduino Uno sends sensor data to Raspberry Pi using the UART serial interface. Weather sensor DHT11 is used to measure temperature and humidity. DHT11 produces digital output and connects directly to Raspberry Pi. A USB microphone is connected to the Raspberry Pi to capture ambient sounds in the forest. These readings are sent to the web server via GSM modem at the time intervals specified in the user interface.

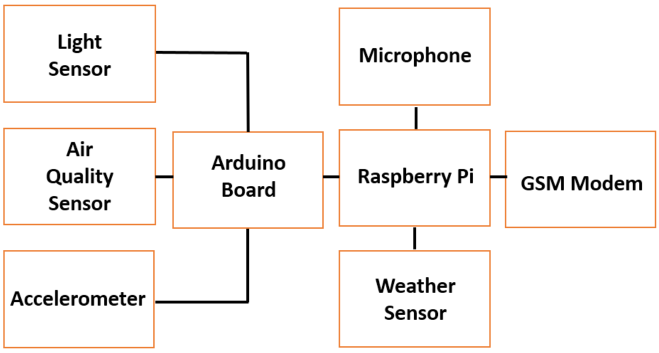

Figure 2. Block diagram of the monitoring unit

Raspberry Pi Microcomputer: Raspberry Pi4 Model B is a small sized minicomputer. It has a 1.5 GHz Quad-Core ARM Cortex-A72 64-bit processor (URL-1). It has 40 input and output pins and 2 GB RAM. It has many connection interfaces, such as Ethernet, USB, WiFi and Bluetooth, readily available on the board. Figure 3 shows a Raspberry Pi4 Model B microcomputer.

Figure 3. Raspberry Pi4 Model B

Arduino Uno Board: It is based on an 8-bit ATMega328P microcontroller. It provides 14 digital I/O pins and 6 analog inputs for accessing external elements. The board is equipped with an SRAM of 2KB and an EPROM of 1KB. Additionally, it provides a USB connection, and operates at a CPU speed of 16 MHz (URL-2). The board is shown in Figure 4.

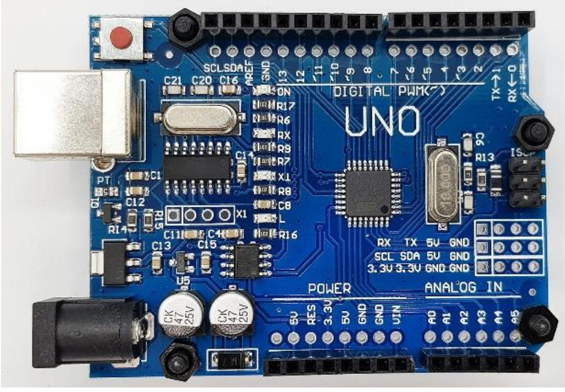

Figure 4. Arduino Uno microcontroller board

Photo Resistor: It is a light sensitive component that measures the light intensity. The resistance of a photo resistor decreases with increasing light intensity. A simple readout circuit can measure the voltage decrease on the photo resistor. Figure 5 shows the readout circuit that we use to measure light intensity. In the readout circuit, Vout will be proportional to the light intensity.

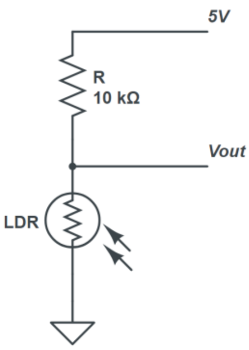

Figure 5. The readout circuit used to measure the voltage drop on the photo resistor

Air Quality Sensor: We use MQ7 in Figure 6 as the air quality sensor. It measures the CO concentration in the environment ranging from 20 to2000ppm (URL-3). It works at an extensive range of temperatures and responds rapidly to CO concentration changes.

Figure 6. MQ7 air quality sensor

Weather sensor: The DHT11 sensor is utilized as a weather sensor to measure the temperature and humidity (URL-4). The sensor in Figure 7 has a built-in microcontroller. As a result of this, it gives digital output.

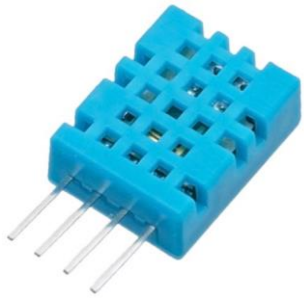
Figure 7. DHT11 weather sensor

Accelerometer: Accelerometer ADXL345 in Figure 8 is used to sense the acceleration in the *x, y,* and *z* axes (URL-5). In our case, we use it to detect landslides by sensing the movement of the earth. It provides an analog output proportional to the magnitude of acceleration.

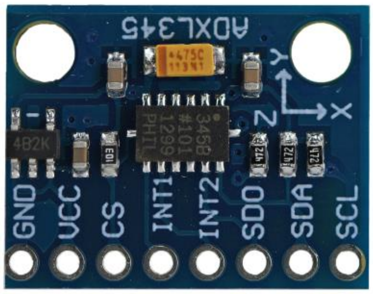

Figure 8. Accelerometer

GSM Modem: SIM808 module in Figure 9 consists of a GSM and GPS module (URL-6). We use this module to collect the GPS coordinates of the IoT node and send the data from the Raspberry Pi microcomputer to the web server. Microphone: An USB microphone is used to get the environmental sound in the forest.

Figure 9. SIM808 GSM modem

2.1.2. Web server

The web server is used to collect data from monitoring units. In our proposed architecture, we used an Apache web server running on the CentOS Linux operating system. The web server collects data from monitoring units using HTTP GET requests. In addition, the MySQL database is used on the same server to store the data collected from the monitoring units. The server-side code used to receive requests and interact with the database is written in PHP.

2.1.3. Web page

The web page offers a user-friendly interface where users can track current and historical sensor readings from monitoring units and manage the IoT network. There are four main sections on the web page: Monitoring, Archive, Alarms, and Management. In the monitoring section, current sensor readings from monitoring units are displayed. The archive section provided access to past readings by querying the data by specifying the monitoring unit, sensor type, and time interval. The Alarms section allows users to define triggered alarms when sensor readings meet predefined conditions. The management section is used to manage the IoT network. In this section, users can specify which monitoring units are allowed to transmit data, and what is the frequency of sensor readings for each monitoring unit.

2.2. Fire Detection Methodology

The Raspberry Pi continuously receives CO gas concentration and temperature readings from sensors in the monitoring unit. Associating the rise in CO gas concentration with the increase in temperature allows for the identification of fires. A detection alarm is produced when specific thresholds are exceeded. These threshold values can be adjusted by authorized users in the system management section of the web page. It enables early detection of fires in hard-to-reach areas of forests.

2.3. Landslide Detection Methodology

We use measurements from the accelerometer to detect the landslide. The accelerometer detects movement in 3 axes and sends the readings to the Raspberry Pi microcomputer via the Arduino board for evaluation. In Raspberry Pi, movement data is compared with a model to decide whether there is a landslide. In this model, parameters such as the duration of the movement, the number of axes participating in the movement, and the amount of acceleration are used to classify the movement as a landslide. All parameters used in the model can be configured by authenticated users via the web page.

2.4. Custom Sound Detection Methodology

In the monitoring unit, we try to detect sounds such as tree cutting, motor vehicles, screams, shouts and gunshots. The audio signal captured by the microphone is parameterized before being analyzed by machine learning methods to detect target sounds. Detection is achieved using acoustic models, which we specifically train for each target sound. The classification is binary; the target sound is present or absent. An alarm is activated to notify the user when the target sound is detected.

We use a machine learning classifier code, which we wrote in Python 3 running on Raspberry Pi to detect tree cutting, motor vehicle, screaming, shouting, and gunshot sounds coming from the environment. There are many classification methods to detect specific types of sounds from audio files. We use SVM (Support Vector Machines) as the classification method. SVM is commonly used in sound detection due to its improved performance compared to other classifiers (Ansari et al., 2021; Guo and Li, 2003). In this classifier, the averages and variances of periodically calculated feature vectors are used as input instead of frame-level feature vectors.

It is not possible to use the audio signal received from the microphone as input to classifiers. There is a need to find some parameters that can successfully represent the properties of the audio signal. These parameters are then extracted and fed to the classifier. In most conventional approaches, properties such as ZCR (zero crossing rate), pitch, and correlation are used for audio detection (Amado and Filho, 2008; Kathirvel et al., 2011). Another property used in sound detection is MFCC features. It is shown that MFCC features have high performance in sound detection with noise robustness (Oh et al., 2012). In this work, we utilized MFCC features due to their success in dealing with background noise.

Figure 10 summarizes the block diagram of the MFCC feature extraction workflow (Chung and Chung, 2017). The audio signal received from the microphone is sampled at 16 KHz. After sampling, it is divided into frames. In our implementation, the duration of each frame is 25 ms and the inter-frame interval is 10 ms. Before FFT (Fast Fourier Transform) is applied to the signal received from the microphone, it is pre-emphasized, to highlight the high-frequency components of the signal, as in Equation 1 (Chung and Chung, 2017). Finally, the hamming window is applied to the signal.

$$s(n) = s(n) - 0.9\, s(n-1) \qquad (1)$$

Mel-Filtering processes the results obtained from FFT, and the logarithm of the output is converted to 12-dimensional MFCC using Discrete Cosine Transform (DCT). From these 12-dimensional MFCCs, we used 36-dimensional MFCCs containing delta and acceleration coefficients as inputs to the SVM. We used the 36-dimensional MFCC feature vectors averaged across 20 frames to construct the SVM input of the SVM classifier. Figure 11 shows the SVM input vector used in the classifier. Figure 12 illustrates the SVM training and detection procedure. In the training phase, audio samples containing the sounds we want to detect are given to the system to calculate the SVM parametersw_k, x_k, and b. In the detection phase, the output of the classifier is utilized to decide whether the input data is one of the sounds we want to detect. The output $d(x)$ is calculated according to Equation 2 (Chung and Chung, 2017).

$$d(x) = \sum_{k=1}^{K} w_k\, G(x_k, x) + b \qquad (2)$$

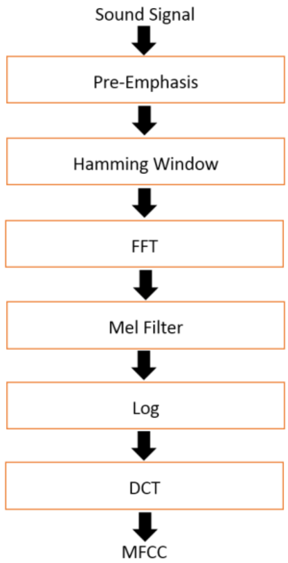

Figure 10. MFCC feature extraction process

K denotes the total number of support vectors, w_k denotes the weights, x_k are the support vectors and b is the bias received in the training process. After the calculation, a positive $d(x)$ value indicates that the input is the sound we want to detect, and a negative value indicates that it is not.

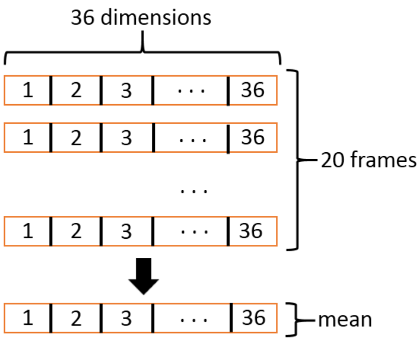

Figure 11. The SVM input vector used in the classifier

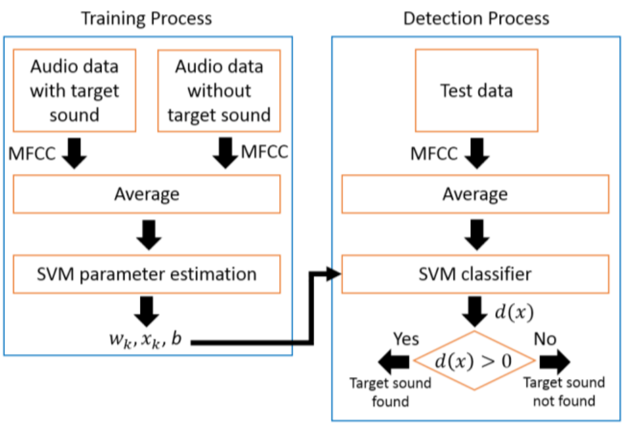

Figure 12. Training and detection procedure used in the classifier

3. Results and Discussion

In order to test the operation of the system end-to-end, we placed a monitoring device in our laboratory. We observed the sensor readings from the monitoring device via the user web page. Figures 13-16 shows the various environmental parameter variations with respect to the time. Figure 13 shows that the temperature takes values between 17°C and 24°C. Humidity in Figure 14 takes the minimum value when the temperature in Figure 13 is maximum.

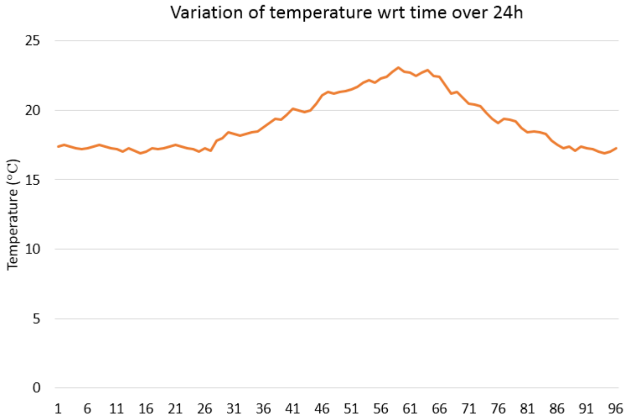

Figure 13. Variation of temperature in 24 hours

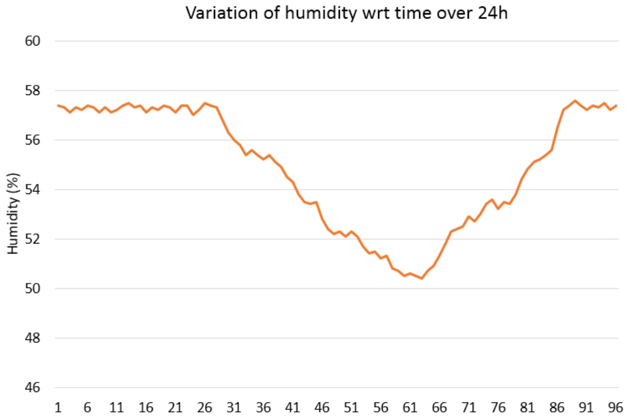

Figure 14. Variation of humidity in 24 hours

Figure 15 tells that the light intensity reaches its maximum during the day and is around zero at night. The CO concentration in Figure 16 is between 0.15 and 0.3 throughout the day. These results are in line with the measurements we made with our external measuring devices. Another important result of our tests is that there are no missing measurements, that is, the monitoring device was able to make measurements and send environmental data every 15 minutes.

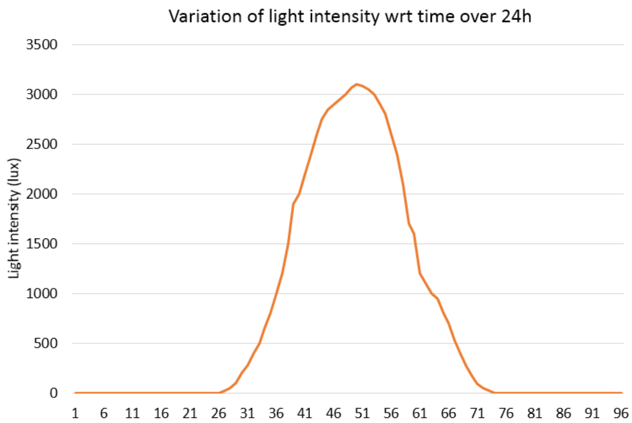

Figure 15. Variation of light intensity in 24 hours

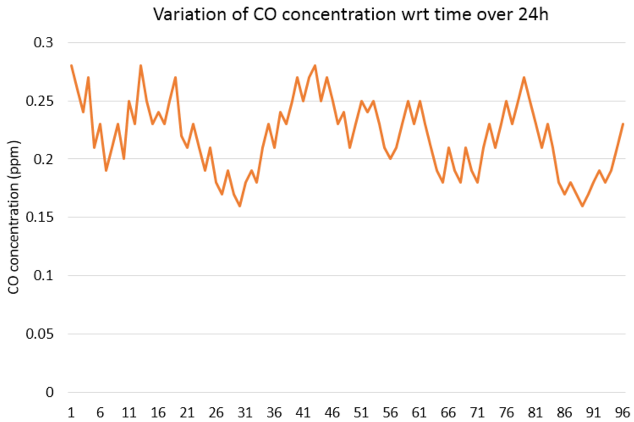

Figure 16. Variation of CO concentration in 24 hours

To measure the performance and accuracy of the sound detection in our system, we collected an audio dataset from the internet. The collected dataset consisted of 50 audio recordings for each target sound, such as chainsaws, motor vehicles, screams, shouts, and gunshots. Besides this data set, we had another non-target dataset consisting of 805 audio recordings of random sounds, including ambient forest sounds, animal sounds and human talking. The metrics we used in our performance analysis are precision, recall and F-measure. To calculate these metrics, the predictions of the sound detection system are evaluated with 4 parameters:

True positives (TP): Accurate prediction of target sounds
True negatives (TN): Accurate prediction of non-target sounds
False positives (FP): False prediction of target sounds
False negatives (FN): False prediction of non-target sounds

Using these parameters, precision, recall and F-measure are calculated as given in Equations 3-5 (Rijsbergen, 1979).

$$Precision\ (P) = \frac{TP}{TP + FP} \tag{3}$$

$$Recall(R) = \frac{TP}{TP + FN} \tag{4}$$

$$F - Measure(F1) = \frac{2PR}{P + R} \tag{5}$$

The results of sound detection test are presented in Table 1 with metrics precision, recall and F-measure. The results show that the precision rates for all sound types are above 80%. It is a solid indication that the system can detect specified sounds with quite high accuracy.

Table 1. Test results of the sound detection test.

	Precision	Recall	F-Measure
Chainsaw	0.92	0.94	0.93
Motor vehicle	0.94	0.96	0.95
Scream	0.88	0.90	0.89
Shout	0.82	0.92	0.87
Gunshot	0.98	0.96	0.97

The performance of the classifier can be better evaluated using the F-measure value as it includes both precision and recall in its calculation. It can be seen that all F-measure values in Table 1 are above 0.87. When these values are compared with the values obtained from the classifiers used in previous studies such as (Niessen et al., 2009; Dash and Solanki, 2019), it can be concluded that the classifier successfully detects sounds. The relatively lower precision value for shouting is due to the misclassifying some animal sounds as shouts. It can be further improved by training our model with more animal sounds labeled as non-target data.

4. Conclusion and Suggestions

Today, forest protection and management are very important for our planet. As a result, the use of Internet of Things (IoT) technology for forest monitoring is an innovative and transformative solution. This paper introduces "An Internet of Things Platform for Forest Monitoring", that is developed to manage and protect forests with the abilities provided by IoT technology.

The IoT-based forest monitoring system discussed in this paper makes use of sensors, data communications, and cloud-based platforms to collect real-time data from forests. This data contains important information about temperature, humidity, air quality, and wildlife activities. The information collected is a valuable resource for forest managers and researchers. Our point is, the proposed platform can help detect fires by associating the increase in CO gas concentration with the increase in temperature. It detects landslides by measuring acceleration in 3 axes. Additionally, the system includes advanced machine learning-based acoustic tracking techniques to detect chainsaws, motor vehicles, screams, shouts, and gunshots. The presence of these sounds creates an alarm in the system and informs the authorities about the abnormality. Real-time data collected by the IoT-based forest monitoring system provide significant convenience in protecting forests and the environment.

The application of IoT technology in the field of forest monitoring offers important advantages for a safe and sustainable future for the forests. Future work in this regard includes continued research for efficient sensor and data processing techniques, improvement of existing systems, and exploration of new applications. Adapting the Internet of Things to forest monitoring is not a solo effort. Protecting these important ecosystems takes collaboration, innovation, and shared commitment.

Ultimately, our proposed IoT-enabled platform provides an efficient tool for monitoring forests efficiently. In a world where rapid technological developments and transformations are frequently seen, the use of such new technologies in protecting and managing of the forests is important for the future of our planet.

Ethics Committee Approval: N/A.

Peer-review: Externally peer-reviewed.

Author Contributions: Concept: M.S.; Design: M.S.; Supervision: M.S.; Resources: M.S.; Data Collection: M.S.; Analysis: M.S.; Literature Search: M.S.; Writing Manuscript: M.S.; Critical Review: M.S.

Conflict of Interest: The authors have no conflicts of interest to declare.

Financial Disclosure: The authors declared that this study has received no financial support.

Cite this paper as: Sanlı, M. 2023. An Internet of Things Platform for Forest Monitoring, *European Journal of Forest Engineering*, 9(2):80-87.

References

Al-Masri, A.N. 2021. Automated deep learning based video summarization approach for forest fire detection. *Journal of Intelligent Systems and Internet of Things*, 5(2):54-61.

Amado, R.G., Filho, J.V. 2008. Pitch detection algorithms based on zero-cross rate and autocorrelation function for musical notes. *International Conference on Audio, Language and Image Processing*, Shanghai, China, pp. 449-454.

Ansari, M.R., Tumpa, S.A., Raya, J.A., Murshed, M.N. 2021. Comparison between support vector machine and random forest for audio classification. *International Conference on Electronics, Communications and Information Technology (ICECIT)*, 14-16 September, online, pp. 1-4.

Chung, S., Chung, Y. 2017. Scream sound detection based on SVM and GMM. *International Conference on Recent Trends in Engineering and Technology,* 3-4 May, Bangkok, Thailand, pp. 215-218.

Dampage, U., Bandaranayake, L., Wanasinghe, R., Kottahachchi, K. 2022. Forest fire detection system using wireless sensor networks and machine learning. *Scientific Reports*, 12:1-11.

Dash, T.K. and Solanki, S.S. 2019. Investigation on the Effect of the Input Features in the Noise Level Classification of Noisy Speech. *Journal of Scientific and Industrial Research*, 78:868-872.

Gulci, S., Akay, A.E., Spinelli, R., Magagnotti, N., 2018. Assessing the exposure of chipper operators to wood dust in a roadside landing area. *Fresenius Environmental Bulletin*, 27(6): 4132-4138.

Guo, G., Li, S. 2003. Content-based audio classification and retrieval by support vector machines. *IEEE transactions on neural networks*, 14(1):209-223.

Hu, M., Chai, H., Ren, Y. 2021. Forest fire video detection based on multi-scale feature fusion with data enhancement. *4th International Conference on Artificial Intelligence and Pattern Recognition,* 24-26 September, Xiamen, China, pp. 218-224.

Imamoglu, A.F., Tas, I. 2022. Preliminary application of a low-cost smart collar developed for wild animal tracking. *European Journal of Forest Engineering*, 8(1): 35-39.

Kathirvel, P., Manikandan, M.S., Senthilkumar, S., Soman, K.P. 2011. Noise robust zerocrossing rate computation for audio signal classification. *3rd International Conference on Trendz in Information Sciences & Computing (TISC2011)*, 8-9 December, Chennai, India, pp. 65-69.

Krishnamoorthy, M., Asif, M., Kumar, P.P., Nuvvula, R.S.S., Khan, B., Colak, I. 2023. A design and development of the smart forest alert monitoring system using IoT. *Journal of Sensors* 2023:1–12.

Madkar, S., Sakhare, D.Y., Phutane, K., Haral, A., Nikam, K., Tharunyha, S. 2022. Video based forest fire and smoke detection using YoLo and CNN. *International Conference on Power, Energy, Control and Transmission Systems (ICPECTS)*, 8-9 December, Chennai, India, pp. 1-5.

Nai-meng, C., Wan-jun, Y. 2020. Early forest fire smoke detection based on aerial video. *Journal of Physics: Conference Series*, 1684(1):12095.

Niessen, M.E., Krijnders, D., Andringa, T.C. 2009. Understanding a soundscape through its components, *Euronoise*, 26 - 28 October, Scotland, UK.

Nihei, K., Kai, N., Maruyama, Y., Yamashita, T., Kanetomo, D., Kitahara, T., Maruyama, M., Ohki, T., Kusin, K., Segah, H. 2022. Forest fire surveillance using live video streaming from UAV via multiple LTE networks. *IEEE 19th Annual Consumer Communications & Networking Conference (CCNC)*, 8-11 January, online, pp. 465-468.

Oh, S.G., Lee, J.U., Lee, H.S., Chung, Y.W., Park, D.H. 2012. Abnormal sound detection and identification in surveillance system. *Journal of KIISE*, 39(2):144-152.

Patel, Y.S., Banerjee, S., Misra, R., Das, S.K. 2020. Low-latency energy-efficient cyber-physical disaster system using edge deep learning. *21st International Conference on Distributed Computing and Networking*, 4-7 January, Kolkata, India, pp. 1-6.

Pokhrel, P., Soliman, H. 2018. Advancing early forest fire detection utilizing smart wireless sensor networks. In: Kameas, A., Stathis, K. (Eds.) Ambient Intelligence. AmI 2018. Lecture Notes in Computer Science, vol 11249. Springer, Berlin, pp. 63-73.

Rijsbergen, C.J.V. 1979. Information Retrieval, 2nd ed. Butterworth-Heinemann, Newton, MA, USA.

Selle, J., Krishna, A.V., Harish, M., Reddy, K.B. 2022. An IoT based Alert System with Gas Sensors in a WSN Framework for Evasion of Forest Fire. *2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, 28-29 April, India, pp. 1516-1521.

Seric, L., Ivanda, A., Bugaric, M., Braovic, M. 2022. Semantic conceptual framework for environmental monitoring and surveillance-a case study on forest fire video monitoring and surveillance. *Electronics,* 11(2):275.

URL-1, https://www.raspberrypi.org, Last Access: 01.10.2023.

URL-2, https://www.arduino.cc, Last Access: 01.10.2023.

URL-3, https://www. hwsensor.com, Last Access: 01.10.2023.

URL-4, https://www.microbot.it, Last Access: 01.10.2023.

URL-5, https://www.analog.com/en/products/adxl345 .html, Last Access: 01.10.2023.

URL-6, https://www.simcom.ee/modules/gsm-gprs-nss/ sim808, Last Access: 01.10.2023.

Wiame, B., Moussati, E., Omar, M., Mohammed, B. 2021. Early forest fire detection system using wireless sensor network and deep learning. *International Journal of Advanced Computer Science Applications*, 7(2):469–505.

Yan, X., Cheng, H., Zhao, Y., Yu, W., Huang, H., Zheng, X. 2016. Real-time identification of smoldering and flaming combustion phases in forest using a wireless sensor network-based multi-sensor system and artificial neural network. *Sensors,* 16(8):1228.

Yang, X., Wang, Y., Liu, X., Liu, Y. 2022. High-precision real-time forest fire video detection using one-class model. *Forests,* 13(11):1826.

Zope, V., Dadlani, T., Matai, A., Tembhurnikar, P., Kalani R. 2020. IoT sensor and deep neural network based wildfire prediction system. *International Conference on Intelligent Computing and Control Systems*, 13-15 May, Secunderabad, India, pp. 205–208.