

Research Paper

Automated Wildlife Monitoring System Using Raspberry Pi

¹Bandari Samatha, ²Dr. Abdul Rahim

¹M. Tech Student, Dept of ECE, MALLA REDDY ENGINEERING COLLEGE FOR WOMEN, Maisammaguda, Dhulapally, Kompally, Secunderabad-500100, Telangana, INDIA.

²Assistant professor, Dept of ECE, MALLA REDDY ENGINEERING COLLEGE FOR WOMEN, Maisammaguda, Dhulapally, Kompally, Secunderabad-500100, Telangana, INDIA.

ABSTRACT

The Automated Wildlife Monitoring System Using Raspberry Pi is an intelligent solution developed for protecting wildlife and monitoring environmental conditions in forest areas. The system uses Raspberry Pi Pico as the main controller and integrates sensors such as DHT11, MQ2 gas sensor, and fire sensor. These sensors continuously monitor temperature, humidity, smoke, and flame conditions. The collected data is processed in real time to detect abnormal environmental situations. Whenever dangerous conditions are identified, a buzzer generates an alert to warn nearby authorities. An LCD display provides real-time information about environmental parameters. The system operates automatically without requiring constant human intervention. It reduces response time and helps prevent forest disasters. The proposed system is cost-

effective, reliable, and suitable for remote areas. It contributes significantly to wildlife conservation and environmental safety.

Keywords: *Raspberry Pi Pico, Wildlife Monitoring, Fire Sensor, DHT11 Sensor, MQ2 Gas Sensor, Environmental Monitoring, Forest Protection, IoT, LCD Display, Buzzer.*

INTRODUCTION

Wildlife conservation has become increasingly important due to deforestation, climate change, and human activities affecting natural ecosystems. Forest regions are vulnerable to hazards such as wildfires, harmful gases, and sudden environmental changes that threaten the survival of animals and plants. Conventional monitoring methods mainly depend on manual inspections and camera surveillance, which are time-consuming

and inefficient. Continuous monitoring of remote forest areas is difficult using traditional approaches. To overcome these limitations, automated monitoring systems have been developed using embedded technologies. The proposed system utilizes Raspberry Pi Pico as the central processing unit for collecting and analyzing environmental data. Sensors such as DHT11, MQ2, and fire sensors are employed to detect temperature, humidity, smoke, and flame conditions. Real-time monitoring enables early identification of hazardous situations. An LCD screen displays sensor readings, while a buzzer generates alerts during emergencies. The system minimizes the need for human intervention and improves response speed. Its compact design and low power consumption make it suitable for long-term deployment.

LITERATURE SURVEY

Recent developments in embedded systems and IoT technologies have greatly improved wildlife monitoring applications. Researchers have proposed various methods for monitoring environmental conditions using sensor-based systems. Temperature and humidity sensors such as DHT11 are widely used for detecting climatic changes that may affect wildlife habitats. Gas sensors like MQ2 have been

employed for identifying harmful gases and smoke, which are indicators of fire hazards. Flame sensors are utilized to provide early warning of forest fires by sensing infrared radiation. Several studies have integrated these sensors with microcontrollers such as Arduino and Raspberry Pi. Real-time monitoring and cloud-based platforms have enhanced accessibility and data analysis capabilities. Wireless sensor networks have been introduced to improve coverage over large forest regions. Solar-powered systems have also been proposed to address energy requirements in remote locations. Alert mechanisms using GSM modules and IoT notifications enable faster response from authorities. LCD displays and buzzers are commonly used for local warning purposes.

EXISTING SYSTEM

Existing wildlife monitoring systems mainly rely on manual patrolling and camera surveillance. Forest officials inspect large areas periodically to identify hazards and illegal activities. These methods require significant manpower and are time-consuming. Monitoring remote forest regions continuously is difficult using traditional techniques. Some systems use GPS tracking devices and camera traps for observing animal movements.

However, these approaches do not provide environmental monitoring capabilities. Basic fire detection systems are available but operate independently without integrating multiple sensors. Most existing systems lack real-time alert mechanisms, leading to delayed emergency responses. High installation costs and maintenance requirements are additional drawbacks. Therefore, traditional methods are inefficient for providing comprehensive wildlife protection.

PROPOSED SYSTEM

The proposed Automated Wildlife Monitoring System provides a smart and efficient solution for forest protection. Raspberry Pi Pico acts as the central controller of the system. DHT11 sensor monitors temperature and humidity levels continuously. MQ2 sensor detects smoke and harmful gases present in the environment. Fire sensor identifies flame occurrences at an early stage. All sensor readings are processed in real time by the controller. Abnormal conditions activate a buzzer to generate warning alerts. An LCD screen displays environmental information continuously. The system reduces human intervention and enables rapid response to disasters.

ARCHITECTURE

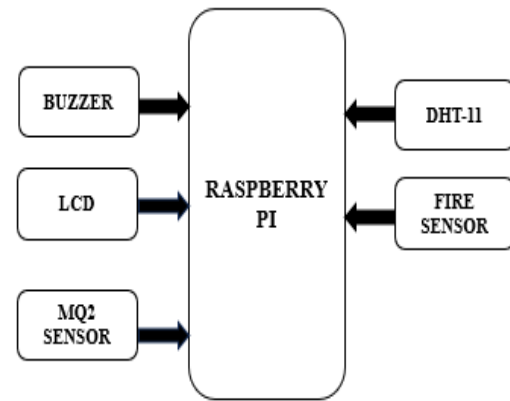


Fig: 1 Architecture of Wildlife Monitoring System Using Raspberry Pi

The architecture of the Automated Wildlife Monitoring System consists of Raspberry Pi Pico, DHT11 sensor, MQ2 gas sensor, fire sensor, LCD display, and buzzer. Raspberry Pi Pico serves as the processing unit of the system. The DHT11 sensor measures temperature and humidity values. MQ2 sensor detects smoke and hazardous gases. Fire sensor senses flames and infrared radiation. Sensor data is continuously transferred to the Raspberry Pi Pico. The controller analyzes the received information and identifies abnormal conditions. The LCD module displays real-time environmental parameters. A buzzer produces warning signals whenever dangerous situations are detected. This architecture provides efficient and continuous environmental monitoring.

MEDHOLOGY DISCRIPTION

The methodology begins with collecting environmental information using multiple sensors. DHT11 sensor continuously measures temperature and humidity levels. MQ2 sensor detects smoke and harmful gases such as carbon monoxide and methane. The fire sensor identifies the presence of flames in the surroundings. All sensor outputs are connected to the Raspberry Pi Pico microcontroller. The controller continuously receives and processes sensor values. Threshold values are predefined for identifying abnormal conditions. When temperature rises beyond the threshold or smoke is detected, the controller generates warning signals. The buzzer provides immediate alerts for nearby authorities.

Simultaneously, sensor readings are displayed on the LCD screen. The system operates automatically without human intervention. Continuous monitoring helps in early detection of environmental hazards. The collected data can be stored and analyzed for future applications. The system can also be integrated with IoT platforms for remote monitoring. Thus, the methodology ensures efficient wildlife protection and disaster prevention.

HARDWARE AND SOFTWARE REQUIREMENTS



Fig: 2 Raspberry Pi

Raspberry Pi acts as the central controller of the Automated Wildlife Monitoring System. It receives data from all connected sensors and processes the information continuously. Based on the sensor readings, it controls the output devices and generates alerts during abnormal conditions. Thus, Raspberry Pi ensures efficient and real-time environmental monitoring.



Fig:3 LCD display

The LCD display provides real-time information about sensor readings and system status. It displays temperature, humidity, gas levels, and warning messages. The display enables users to monitor environmental conditions easily. It offers a simple and effective interface for observing system operation.



Fig:4 Buzzer

The buzzer acts as an alerting device in the monitoring system. It produces an audible warning whenever abnormal conditions are detected. The buzzer helps nearby authorities or personnel respond quickly to emergencies. This feature enhances the safety and reliability of the wildlife monitoring system.



Fig:6 Fire Sensor

The fire sensor is responsible for detecting the presence of flames and fire hazards. It senses infrared radiation emitted by flames and sends signals to the Raspberry Pi. When fire is detected, immediate action can be initiated through alerts. This helps in preventing forest fires and protecting wildlife.



Fig: 5 DHT-11 Sensor

The DHT11 sensor is used to measure temperature and humidity levels in the surrounding environment. It continuously monitors climatic conditions that affect wildlife habitats. The sensor provides accurate readings to the Raspberry Pi for analysis. This helps in detecting abnormal environmental changes at an early stage.



Fig:7 MQ2 Sensor

The MQ2 sensor is used to detect harmful gases such as smoke, methane, and carbon monoxide. It continuously monitors air quality and identifies hazardous conditions. The sensor transmits gas concentration data to the Raspberry Pi for processing. Early detection of smoke and gases helps reduce environmental damage.

A. Arduino IDE

The Arduino Integrated Development Environment (IDE) is utilized for developing, compiling, and uploading the embedded program to the NodeMCU (ESP8266). It provides a user-friendly platform for writing code in Embedded C and supports a wide range of libraries for sensor interfacing and device control. The IDE enables efficient implementation of data acquisition, processing logic, and actuation control within the system.

RESULTS AND DISCUSSION

The developed system successfully monitored environmental conditions in real time. Temperature and humidity values were accurately measured using the DHT11 sensor. MQ2 sensor effectively detected smoke and harmful gases. The fire sensor provided immediate flame detection capabilities. Sensor readings were displayed continuously on the LCD screen. The buzzer generated alerts whenever abnormal conditions occurred. The system responded quickly to emergency situations and minimized delay. Experimental results demonstrated reliable and stable operation. The proposed system proved to be cost-effective and energy-efficient. Overall, the obtained results indicate that the system is suitable for wildlife monitoring applications.

CONCLUSION

The Automated Wildlife Monitoring System Using Raspberry Pi provides an effective solution for protecting forest ecosystems. It continuously monitors environmental parameters using multiple sensors. The system detects fire, smoke, and abnormal climatic conditions at an early stage. Real-time monitoring improves the efficiency of disaster prevention. The buzzer and LCD provide instant alerts and information. Automation reduces the need for manual supervision. The system is economical, reliable, and easy to deploy in remote regions. It enhances wildlife safety and supports environmental conservation. The proposed model contributes to maintaining ecological balance. Hence, it serves as a valuable tool for smart forest management.

FUTURE SCOPE

The system can be enhanced by integrating IoT technologies for remote monitoring. GSM modules can be used to send SMS alerts to forest officials. Cloud platforms can store sensor data for long-term analysis. Wireless sensor networks can improve coverage over large forest areas. Solar power systems can be incorporated for sustainable operation. GPS modules may be used for identifying the exact location of hazards. Artificial intelligence techniques can predict

environmental risks based on historical data. Camera modules can be added for visual monitoring of wildlife activities. Mobile applications can provide real-time access to environmental information.

REFERENCES

- [1] Raspberry Pi Foundation, *Raspberry Pi Pico Datasheet*, Raspberry Pi Foundation, 2021.
- [2] Aosong Electronics Co., Ltd., *DHT11 Temperature and Humidity Sensor Datasheet*, China, 2020.
- [3] Hanwei Electronics Group, *MQ2 Gas Sensor Datasheet*, Zhengzhou, China, 2019.
- [4] Flame Detection Sensor Module, *Fire Sensor Module Datasheet and Specifications*, 2020.
- [5] S. Monk, *Programming Raspberry Pi Pico in Python*. New York, NY, USA: McGraw-Hill Education, 2021.
- [6] M. Banzi, *Getting Started with Arduino*, 2nd ed. Sebastopol, CA, USA: O'Reilly Media, 2011.
- [7] R. Kamal, *Embedded Systems: Architecture, Programming and Design*, 3rd ed. New Delhi, India: McGraw-Hill Education, 2017.
- [8] N. Sriskanthan and F. Tan, "Wireless Sensor Networks for Forest Fire Detection," in *Proc. IEEE International Conference on Networking, Sensing and Control*, 2008, pp. 1–6.
- [9] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless Sensor Networks: A Survey," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, Aug. 2002.
- [10] C. Hartung, R. Han, C. Seielstad, and S. Holbrook, "FireWxNet: A Multi-Tiered Portable Wireless System for Monitoring Weather Conditions in Wildland Fire Environments," in *Proc. ACM International Conference on Mobile Systems, Applications and Services (MobiSys)*, 2006, pp. 28–41.
- [11] J. Lloret, M. Garcia, D. Bri, and S. Sendra, "A Wireless Sensor Network Deployment for Rural and Forest Fire Detection and Verification," *Sensors*, vol. 9, no. 11, pp. 8722–8747, 2009.
- [12] A. Kumar, "IoT Based Forest Fire Detection System," *International Journal of Engineering Research and Technology (IJERT)*, vol. 7, no. 5, pp. 1–5, 2018.
- [13] K. K. Patel and S. M. Patel, "Internet of Things-IoT: Definition, Characteristics, Architecture, Enabling Technologies, Application & Future Challenges,"

International Journal of Engineering Science and Computing, vol. 6, no. 5, pp. 6122–6131, 2016.

[14] Raspberry Pi Foundation, “Raspberry Pi Documentation.” [Online]. Available: <https://www.raspberrypi.org/documentation>. [Accessed: Jun. 20, 2026].

[15] ElectronicWings, “Embedded Systems and Sensor Tutorials.” [Online]. Available: <https://www.electronicwings.com>. [Accessed: Jun. 20, 2026].

[16] Electronics For You, “Embedded Systems and IoT Tutorials.” [Online]. Available: <https://www.electronicsforu.com>. [Accessed: Jun. 20, 2026].

[17] Circuit Digest, “IoT and Sensor Interfacing Projects.” [Online]. Available: <https://www.circuitdigest.com>. [Accessed: Jun. 20, 2026].

[18] IEEE Xplore Digital Library, “Forest Fire Detection Using IoT Systems.” [Online]. Available: <https://ieeexplore.ieee.org>. [Accessed: Jun. 20, 2026].

[19] Springer Publications, *Environmental Monitoring Systems Using Sensors*. Cham, Switzerland: Springer, 2020.

[20] Google Scholar Database, “Research Papers on Wildlife Monitoring Systems.” [Online]. Available: <https://scholar.google.com>. [Accessed: Jun. 20, 2026].

13. Babburi, S. Lightweight Distributed Provenance Framework for Edge and IoT Data Systems.

14. Gaddam, S. From Fixed Specifications to Self-Adapting Systems: A Machine Learning Perspective on Software Engineering.

15. Immadi, S. K. (2025). Optimizing ERP for Human Capital Management. *Applied Research for Growth, Innovation and Sustainable Impact*, 377–384. <https://doi.org/10.1201/9781003684657-63>

16. Reddy, S. K. R. Developing a Modular AI Framework to Enhance Scalability and Personalization in Next-Generation Reward Platforms.

17. Poojari, R. Frameworks for Data Management and Lineage in Large-Scale Healthcare Data Systems.

18. Mahimalur, R. K., Vargam, M., & Manoharan, D. Devops Lifecycle Management And Cloud Migration Assessments: A Security-Driven CI/CD Perspective.

19. Viswanathan, V. Generative AI for Smarter Workforce Planning and Enterprise Resource Decisions.
20. Gajula, S. (2026, March). Two Pillars of Banking Intelligence: A Comparative Analysis of AI Techniques for Fraud Prevention and Churn Mitigation. In 2026 14th International Symposium on Digital Forensics and Security (ISDFS) (pp. 1-6). IEEE.
21. Maturi, S. Y. (2023). Crowdsourced frontier: Unveiling autonomous adversarial cybercapabilities via open AI competition. *International Journal of Intelligent Systems and Applications in Engineering*, 11(1s), 275–284.
22. Venkata Ramana, P. (2024). AI-driven predictive analytics in ERP systems for proactive supply chain optimization. *International Journal of Research in Information Technology and Computing*, 8(4).
23. Srikanth Kavuri. (2025). AI-DRIVEN TEST AUTOMATION FRAMEWORKS: ENHANCING EFFICIENCY AND ACCURACY IN SOFTWARE QUALITY ASSURANCE. *International Journal of Applied Mathematics*, 38(10s), 699–710. <https://doi.org/10.12732/ijam.v38i10s.990>
24. Venkata Pavan Kumar Gummadi. (2026). Infrastructure Optimization Techniques for Enterprise Integration Platforms: A Comprehensive Analysis. *Computer Fraud and Security*, 37–44. <https://doi.org/10.52710/cfs.875>
25. Shashank, A. (2025). AI-Enhanced ETL Processes: Leveraging Artificial Intelligence for Optimized Data Integration Systems. *Journal Of Multidisciplinary*, 5(8), 219-225.
26. Kandula, S. T. R., Susarla, R. S., & Boyapati, P. K. (2025, July). Enhanced Cyber Security Using Global Local Artificial Neural Network Based Intrusion Detection in Big Data Environment. In 2025 IEEE 4th World Conference on Applied Intelligence and Computing (AIC) (pp. 426-431). IEEE.
27. Boyapati, P. K. Building a centralized data operations hub for healthcare enterprise integration. *IJSAT-Int. J. Sci. Technol.* 16 (2). <https://doi.org/10.71097/IJSAT.v16.i2.3219>