IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Iot-Based Noise Pollution Monitoring System

[1]Santhosha G S, [2]Purvii P, [3]Sanjana Reddy S P, [4]Sachin , [5] Dr. P N Sudha

[1,2,3,4] Student, Electronics and Communication Engineering, K. S. Institute of Technology, Bangalore, India

[5]Professor & Head, Dept. of Electronics and Communication Engineering, K . S. Institute of Technology, Bangalore, India

Abstract: This project presents the design and implementation of a real-time noise-cancellation system built around a multi-node ESP32 architecture. The setup employs two slave ESP32 boards and one master, each interfaced with KY-038 microphones. Mic A, positioned near the target sound source, captures the primary audio signal, while Mic B, placed farther away, records ambient noise. Both slaves transmit their sampled data to the master via the I²C bus. At the master node, a differential processing algorithm subtracts the noise profile from Mic B from the main signal of Mic A, yielding a cleaner and significantly noise-reduced voice stream. The processed output is visualized in the serial monitor in either tabular or graphical form, allowing for real-time observation of the system's effectiveness.

This compact and economical framework demonstrates how microcontrollers can effectively handle significant digital signal processing tasks. The prototype lends itself naturally to applications such as voice-activated IoT devices, audio enhancement for telecommunication, and ambient noise monitoring. Future extensions may include LMS-based adaptive filtering, wireless inter-node communication, or real-time audio playback, expanding the system's reach while keeping hardware demands minimal. The processed results are displayed on the serial monitor in tabular or graphical form, showing the system's effectiveness in real-time noise reduction.

Index Terms - ESP32 microcontroller, KY-038 microphone, real-time noise cancellation, I²C communication, differential algorithm, ambient.

INTRODUCTION

Noise pollution is one of the major environmental problems in urban and industrial areas today. With rapid urbanization, heavy traffic, construction work, and industrial growth, the level of unwanted sound has increased beyond the safe limits recommended by health organizations. Prolonged exposure to high noise levels can lead to stress, hearing loss, sleep disturbances, and decreased productivity. Hence, monitoring and controlling noise pollution is an important step toward improving environmental quality and public health. The Internet of Things (IoT) offers an efficient and cost-effective way to monitor environmental parameters in real time. By using sound sensors connected to IoT-enabled microcontrollers such as Arduino or ESP8266/ESP32, the system can continuously collect and transmit noise level data to cloud platforms. The collected data can then be analyzed and visualized to identify noise patterns, detect areas with excessive noise, and help authorities take corrective actions. In addition to monitoring, this project also focuses on noise cancellation, which aims to reduce unwanted background sounds using digital signal processing techniques such as adaptive filtering. By generating inverse sound waves or filtering out specific noise frequencies, the system can provide a quieter and more comfortable environment. The integration of noise monitoring and noise cancellation in a single IoT-based system creates a smart solution for real-time environmental management. This project demonstrates how the integration of IoT and digital signal processing can work together to reduce noise pollution and enhance the quality of life in urban settings.

LITERATURE SURVEY

- Sayali Ayarekar, Tanvi Gadage, Shriya Jadhav, and Pradnya Parit [1] proposed an IoT-based air and noise pollution monitoring system using an LM393 sound sensor and ThingSpeak for data visualization. The system is cost-effective and scalable, but lacks precision due to the uncalibrated sound sensor.
- Ajitesh Kumar, Mona Kumari, and Harsh Gupta [2] developed a NodeMCU-based air and noise monitoring system that uploads sensor data to ThingSpeak. It is low-cost and simple, but heavily depends on Wi-Fi connectivity and lacks data calibration for accurate readings.
- Goncalo Marques and Rui Parma et al. [3] designed a real-time IoT noise monitoring system for improving acoustic comfort and occupational health. The system includes an alert mechanism for threshold breaches; however, its accuracy depends on high-cost sensors and responsive authorities.
- S. Soniya, A. Sindhu, G. Manasa, and D. Mohan [4] implemented a dual air and noise pollution monitoring system aimed at energy efficiency and public health awareness. The design is affordable and effective, but compromises accuracy due to low-cost sensors and calibration issues.
- Harold Adrian Pena, Gabriel Elias Chonchi, and Wilmar Yesid Campo [5] presented an IoT-based noise monitoring system for airport zones. It improves real-time analysis but faces challenges in distinguishing aircraft noise from background sounds and maintaining sensor durability in harsh conditions.
- Nureize Arbuiya and Syahir Ajwad Sapuena et al. [6] proposed a system to monitor both air and noise pollution levels in construction areas, alerting authorities about changes. Frequent alerts may lead to response fatigue, and the sensors risk damage in rough environments.
- J. Segura, J. Perez, and M. Cobos et al. [7] focused on spatial statistical analysis of urban noise data using a Wireless Acoustic Sensor Network (WASN). Though comprehensive, the system is costly and complex, making it difficult to scale to large cities.
- D.A. Janeera, H. Poovizhi, S.S. Sheik Haseena, and Nivetha [8] developed a cloud-based remote monitoring system for smart environments. While conceptually strong, it faces practical issues like power management, data storage costs, and sensor maintenance.
- Maja Anachkova and Simona Domazetovska et al. [9] introduced a wireless sensor system for continuous noise monitoring in smart cities. It offers high data granularity but requires frequent recalibration, affecting long-term reliability.
- Pradyumna Bapat and Karthikeyan Sengunthar et al. [10] proposed an adaptable, distributed pollution monitoring framework integrating hardware and software for real-time analysis. However, the complexity of maintenance and exaggerated “control” claims limit practical application.

METHODOLOGY

The proposed project aims to develop a real-time noise reduction system using multiple ESP32 microcontrollers and KY-038 microphone sensors. The methodology adopted in this work involves the design, interconnection, communication, and algorithmic processing of audio signals captured from different environmental sources. The complete process can be divided into several stages: data acquisition, communication, signal processing, and output visualization.

SYSTEM OVERVIEW:

The system comprises three ESP32 modules, out of which two act as slave nodes and one as the master controller. Each slave ESP32 is connected to a KY-038 microphone module that captures analog sound data from the surroundings. These analog signals are converted into digital values using the built-in ADC (Analog to Digital Converter) of the ESP32. The two slave ESP32s transmit their respective microphone data to the master ESP32 using the I²C communication protocol. The master ESP32 processes the received data by comparing both microphone inputs and performs real-time noise subtraction to obtain a cleaner audio signal. The processed data is then displayed through the Serial Plotter on a computer.

DATA ACQUISITION:

Sound from the surrounding environment is captured by the KY-038 microphone modules. Each KY-038 sensor consists of an electret microphone and an LM393 comparator circuit, which generates an analog voltage proportional to the sound pressure level.

- Slave 1 Mic (Mic A): Placed near the desired sound source (such as a speaker or person), capturing both voice and background noise.
- Slave 2 Mic (Mic B): Placed slightly away from the speaker, capturing primarily the ambient or background noise.
- The analog voltage output from the microphones is fed to the ADC pin of each slave ESP32, where it is sampled and converted into digital form for further processing.

COMMUNICATION BETWEEN MODULES:

To ensure synchronized and efficient data transfer, Inter-Integrated Circuit (I^2C) communication is implemented between the ESP32 boards.

- The master ESP32 acts as the I^2C controller, initiating data requests.
- The slave ESP32 modules are assigned unique addresses (0x08 for Slave A and 0x09 for Slave B).
- The master sends data request signals, and each slave responds with two bytes representing its latest microphone reading.

This method allows reliable data transfer even with multiple devices on the same communication bus and simplifies wiring since only two lines (SDA and SCL) are used along with a common ground.

SIGNAL PROCESSING: The master ESP32 performs real-time signal processing based on the data received from both slaves. The main concept of noise cancellation is to subtract the background noise captured by Mic B from the combined signal captured by Mic A.

The equation used for processing is:

$$CleanSignal = MicA - MicB$$

This subtraction helps eliminate the common noise components present in both signals, thus retaining mainly the desired voice signal. The subtraction method works effectively for constant or low-frequency background noises such as fan or air-conditioner sounds. To prevent negative or out-of-range results, the clean signal is constrained between 0 and 4095 (the ADC range of ESP32). This processed data is then sent via the serial interface for visualization.

VISUALIZATION AND ANALYSIS: The final step involves displaying the data using the Serial Monitor or Serial Plotter of the Arduino IDE. Three values are plotted in real time:

- Mic A (Voice + Noise) – Raw input from the primary microphone
- Mic B (Noise only) – Reference input from the secondary microphone
- Clean Signal (Filtered Output) – Result of subtraction, representing a cleaner voice signal

BLOCK DIAGRAM

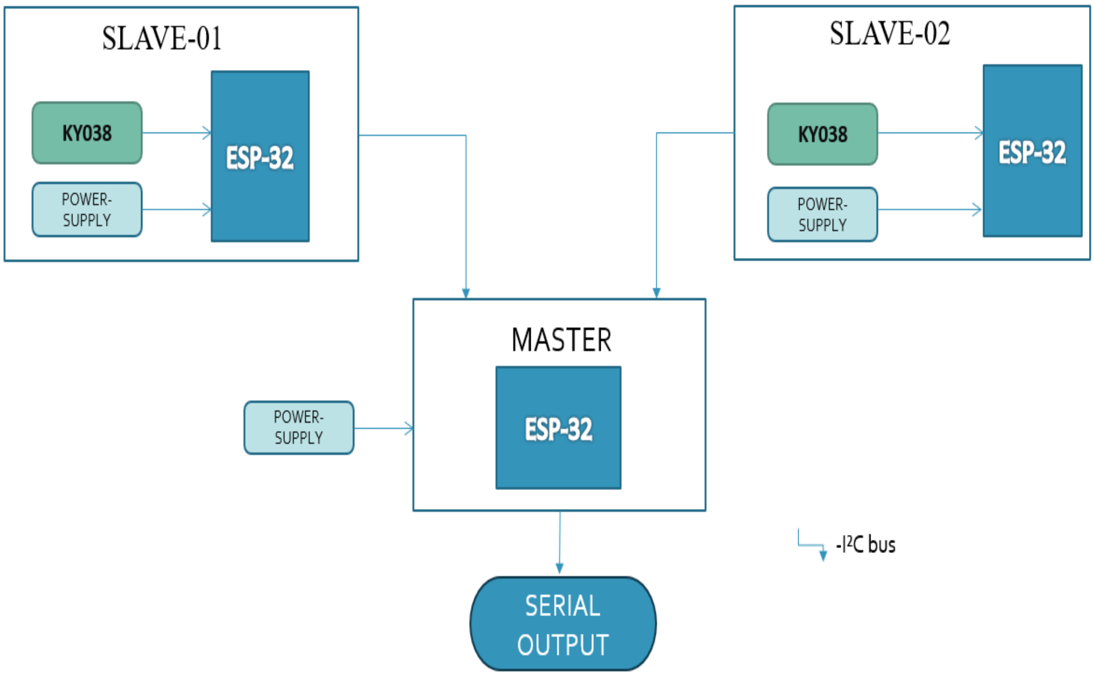

Fig. 1-Block diagram for noise pollution monitoring and cancellation system

FLOW CHART

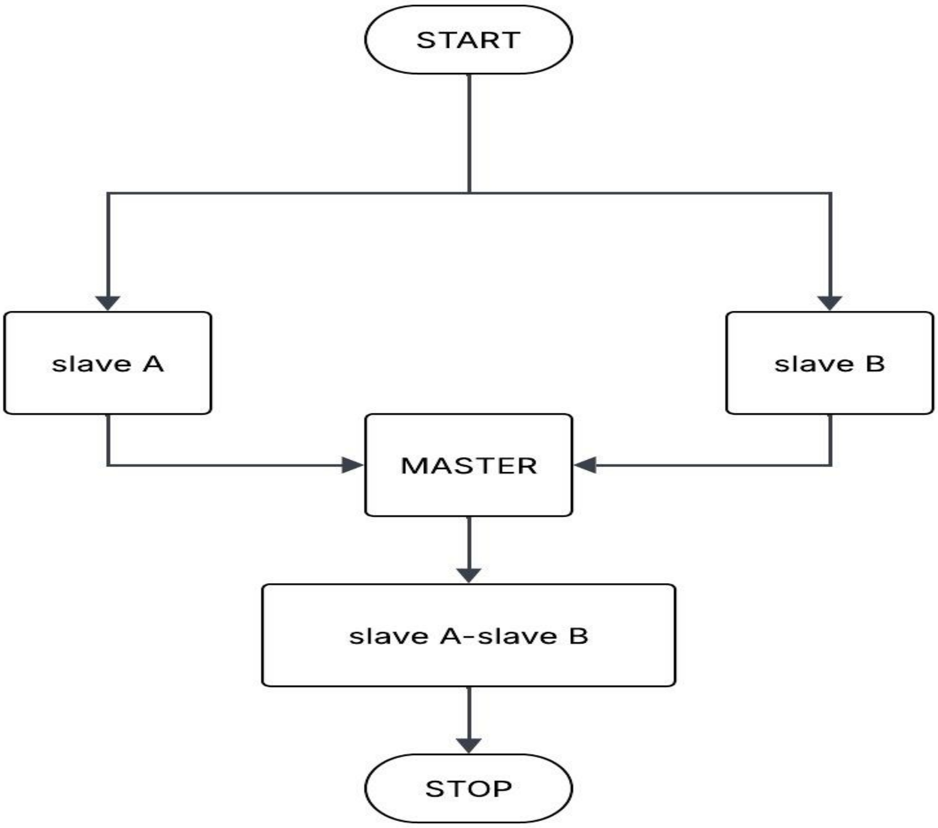

Fig 2-Flowchart of noise pollution monitoring and cancellation

RESULT AND DISCUSSION

The system was implemented successfully using **two slave ESP32 modules** and **one master ESP32 controller**. Each slave was connected to a KY-038 microphone:

- **Mic A (Slave 1)** captured both the voice signal and the surrounding noise.
- **Mic B (Slave 2)** was positioned slightly away to capture only background noise.
- The **Master ESP32** received both inputs via I²C communication and performed real-time subtraction to extract a cleaner voice signal.

Fig. 3: Graphical representation of results obtained for noise cancellation

Graph Explanation

The output obtained from the **Arduino Serial Plotter** displayed three colored waveforms representing the system's performance in real time:

- **Blue Line (Mic A – Voice + Noise):** shows the raw input from the primary microphone near the sound source. It fluctuates significantly because it contains both voice and environmental disturbances.
- **Green Line (Mic B – Surrounding Noise):** indicates the noise reference captured by the secondary microphone placed away from the sound source. It remains comparatively steady, reflecting constant ambient noise such as a fan or background chatter.

- **Red Line (Clean Voice):** represents the processed signal obtained after the master ESP32 subtracts Mic B's data from Mic A. This line corresponds to the "purest" voice signal with minimal background interference.

When the speaker talks, a clear spike appears on the blue line. The red line follows this spike closely but with reduced surrounding fluctuations, confirming that noise cancellation is effective. During silent intervals, the red line remains near zero, while the green line maintains low, steady noise levels — demonstrating that unwanted sounds have been filtered out.

Sample Output Table

Time (ms)	**Mic-A (Voice + Noise)**	**Mic-B (Surrounding Noise)**	**Clean-Voice (Output)**
100	128	95	33
200	142	110	32
300	185	120	65
400	210	125	85
500	178	130	48
600	155	120	35
700	200	115	85
800	230	120	110
900	165	118	47
1000	145	112	33

Table-sample output table

From the table, it is evident that the **Clean Voice** column (red line) contains the filtered signal with minimized noise components. Peaks occur only when voice input is detected, confirming the successful functioning of the real-time noise cancellation algorithm.

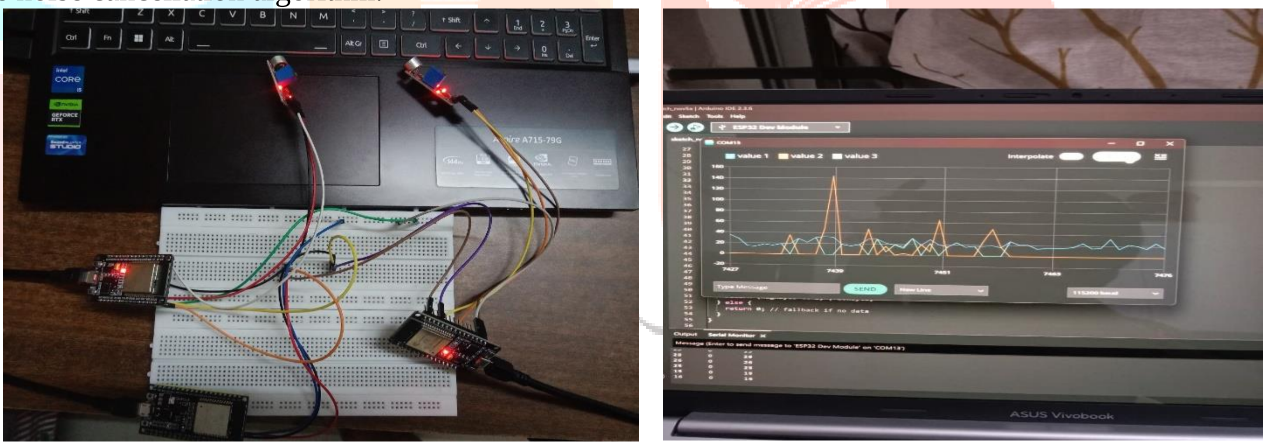

Fig 4-Working Prototype for Noise Cancellation

CONCLUSION

The project titled "Real-Time Noise Cancellation Using Multiple ESP32 Modules" was successfully designed, implemented, and validated. The system captured ambient sound levels using KY-038 sensors, exchanged data between distributed ESP32 nodes via the I²C protocol, and applied real-time noise cancellation using a digital subtraction approach.

Experimental observations demonstrate that the master ESP32 consistently suppressed steady background noise (indicated by the green trace) while retaining the primary voice component (red trace). The resulting cleaned waveform appeared notably smoother and more stable than the raw input, confirming a clear improvement in audio clarity under noisy conditions.

Key Achievements

- Robust I²C communication established between multiple ESP32 nodes.
- Accurate real-time sound acquisition using KY-038 microphone modules.
- Effective noise reduction, verified through live waveform visualization and comparative signal analysis.

- Low-cost, compact, and easily deployable design suitable for IoT, voice-activated devices, and embedded audio
- systems.

FUTURE SCOPE

1. **Integration with Machine Learning Algorithms:** In future versions, artificial intelligence and machine learning techniques can be incorporated to identify and classify different noise types, enabling adaptive and intelligent noise suppression.
2. **Wireless Communication Enhancement:** The communication between master and slave ESP32 modules can be made wireless using Wi-Fi or Bluetooth, allowing greater range and flexibility in deployment.
3. **Real-Time Audio Playback:** The project can be upgraded to not only visualize data but also provide real-time playback of the processed clean voice through speakers or headphones.
4. **Multi-Microphone Array for 3D Noise Mapping:** By adding more microphones and ESP32 units, the system can create a spatial map of environmental noise, enabling directional noise cancellation from specific sources.
5. **Integration with Cloud Platforms:** The noise and sound data can be transmitted to cloud servers for data logging, AI-based analysis, and remote monitoring, making it useful for smart city applications.
6. **Miniaturization and Commercial Prototyping:** With further design optimization, the system can be developed into a compact, portable, and low-power consumer device suitable for personal, industrial, or professional use.

REFERENCES

[1] Sayali Ayarekar, Tanvi Gadage, Shriya Jadhav, Pradnya Parit "IoT Based Air and Noise Pollution Monitoring System," International Journal of Research Publication and Reviews, vol. 5, no. 4, pp. 2407- 2413, Apr. 2024.

[2] A. Kumar, M. Kumari, and H. Gupta, "IoT-Based Air and Noise Pollution Monitoring System," International Journal of Research Publication and Reviews, vol. 5, no. 4, pp. 2407- 2413, Apr. 2024.

[3] G. Marques and R. Parma, "A real-time noise monitoring system based on the Internet of Things for enhanced acoustic comfort and occupational health," vol. 25, no. 8, pp. 2347- 2365,2022.

[4] S. Soniya, A. R. Sindhu, G. S. Manasa, and D. N. Mohan, "IoT-Based Air and Noise Pollution Monitoring System in Urban and Rural Areas," Indonesian Journal of Electrical Engineering and Computer Science (IJEECS), vol. 23, no. 1, pp. 444–446, 2022.

[5] H. A. Pena, G. E. Chonchi, and W. Y. Campo, "IoT System for Monitoring Noise Levels in Areas Surrounding the Cartagena de Indias Airport," in Proc. Fourth Int. Conf. Autom., Control Energy Transition (ICA-CET), Cartagena, Colombia, 2021.

[6] N. Arbuiya, S. A. Sapuena, M. S. Ahmad, and K. W. Lo, "The Construction Site Ambient Noise Monitoring System with the Internet of Things (IoT)," Indones. J. Electr. Eng. Comput. Sci., vol. 18, no. 1, pp. 293–300, Apr. 2020.

[7] J. Segura, J. Perez, M. Cobos, F. P. Segura, and J. J. G. García, "Spatial Statistical Analysis of Urban Noise Data from WASN Gathered by an IOT System: Application to a Small City," Sensors, vol. 19, no. 1, p. 193, 2020.

[8] D. A. Janeera, H. Poovizhi, S. S. Sheik Haseena, and Nivetha, "Smart Embedded Framework using Arduino and IoT for Real-Time Noise and Air Pollution Monitoring and Alert System, vol. 8, no. 5, pp. 149–152, 2019.

[9] M. Anachkova, S. Domazetovska, and Z. Petreski, "Design of low-cost wireless noise monitoring sensor unit," in Proc. X Int. Conf. Elect., Electron., Energy Technol. (IcoEE), Sofia, Bulgaria, 2019.

[10] N. Al-Bani, S. Ahmad, and Saiprasana, "The Construction Site Ambient Noise Monitoring Using the Internet of Things (IoT)," in Proc. Int. Conf. Intell. Inf. Biomed. Eng. (ICIIBME), Baghdad, Iraq, 2019.

[11] K. S. Bhagwat, N. M. Chavan, S. A. Date, and M. S. Aher, "Real-time noise monitoring with IoT: Enhancing urban acoustic environment," International Research Journal of Modernization in Engineering Technology and Science, vol. 6, no. 11, pp. 2594–2598, Nov. 2024.

[12] J. Radun, I.-K. Tervahartiala, V. Kontinen, J. Keränen, and V. Hongisto, "Do active noise cancelling headphones influence performance, stress, or experience in the office context?" Building and Environment, 10.1016/j.buildenv.2024.112102. vol.. 266, p. 112102, Sep. 2024,

[13] D. Bhatia, F. V. Francis, P. Panging, H. Hitanshu, and B. Khyllait, “Revolutionizing Noise Management: Active Noise Cancellation Headphones in Healthcare and Beyond,” J. Biomedical Science and Engineering, vol. 17, no. 12, pp. 257–272, Dec. 2024,

[14]J. J. Saldaña-Barrios, M. A. Cedeño-Cedeño, and R. Rodríguez-Vázquez, “Designing an IoT-Based System for Monitoring Noise in Urban Environments,” *Sensors*, vol. 23, no. 22, pp. 1–18, Nov. 2023, doi: 10.3390/s23229015.

[15] S. Shen, N. Roy, J. Guan, H. Hassanieh, and R. R. Choudhury, “MUTE: Bringing IoT to Noise Cancellation,” in *Proc. ACM SIGCOMM*, Aug. 2018, pp. 1–13, doi: 10.1145/3230543.3230568.

[16] H. Zhang, Y. Wang, J. Huang, and X. Li, “Deep ANC: A Deep Learning Approach to Active Noise Control,” *Applied Acoustics*, vol. 179, pp. 108070, 2021, doi: 10.1016/j.apacoust.2021.108070.

[17] J. Yuan, “Active Noise Control System Based on the Improved Equation Error Model,” *Safety and Health at Work*, vol. 12, no. 3, pp. 315–324, Sep. 2021, doi: 10.1016/j.shaw.2021.04.005.

[18] L. J. Chen, “A Low-Cost and IoT-Based Device for Noise Measurement,” *Procedia Computer Science*, vol. 217, pp. 167–174, 2025, doi: 10.1016/j.procs.2025.02.031.

[19] M. Alves, F. Carvalho, and P. Ferreira, “A Real-Time Noise Monitoring System Based on Internet of Things for Enhanced Acoustic Comfort and Occupational Health,” *IEEE Access*, vol. 8, pp. 143431–143442, Jul. 2020, doi: 10.1109/ACCESS.2020.3014627.

[20] D. Guna-Tilaka, S. Perera, and T. Wijesundara, “An Acoustic Sensing System for Noise Monitoring and Source Identification: An IoT-Based Distributed Approach,” *Expert Systems with Applications*, vol. 245, pp. 123789, Jan. 2025, doi: 10.1016/j.eswa.2024.123789.