Swanirman Sunirmit Publications of Research - 8th ICRTTEAS April 2026 | [2026-27]
ISSN [Online]: 2583-2654

Puretone: A Smart System for Voice-Based Automation and Verification

[1] Shyam Kumar G, [2]Sneha B, [3]Suri Arul U, [4]Tejashwini KR, [5]Shalini G, [6]Linda R
Department of CSE, Dr T Thimmaiah Institute of Technology

Abstract: Puretone is an AI-assisted voice recognition system that replicates the traditional roll-call attendance process by automating speaker verification when a student responds to their name. The system captures the voice input, reduces noise using deep learning, extracts key acoustic features, and matches them with stored voiceprints through a CNN-based model. It detects mimicked or altered voices, ensures accurate and secure attendance marking, and operates reliably even in noisy environments. A lightweight React– FastAPI web interface enables real-time processing and database updates, making Puretone suitable for education, workplaces, authentication, and communication applications.

Keywords: Voice Biometrics, Speaker Recognition, Deep Learning, Noise Reduction, AI Attendance System, Authentication, Voiceprint Matching.

I. INTRODUCTION

Attendance verification is a routine yet essential process in educational institutions and workplaces. Traditional roll-call methods rely heavily on manual identification, which is time- consuming, prone to human error, and vulnerable to proxy responses. With the growing need for secure and efficient authentication systems, voice biometrics has emerged as a reliable solution due to its uniqueness, convenience, and non-intrusive nature. Puretone is an AI-assisted voice recognition system designed to replicate the conventional roll-call procedure while automating the verification stage. Instead of removing the traditional method, the system enhances it by using artificial intelligence to authenticate the speaker when their name is called. Puretone captures the spoken response, processes it using advanced noise reduction techniques, extracts acoustic features, and performs speaker identification through a CNN-based model. This helps detect mimicked or altered voices and ensures accurate attendance marking even in noisy or unpredictable environments. By integrating a lightweight web-based interface with a Fast API backend, Puretone provides real-time processing, secure data management, and a user-friendly experience. The system is scalable and adaptable for various applications, including education, corporate environments, secure access control, and communication platforms. Puretone demonstrates how AI- driven voice biometrics can modernize traditional workflows without disrupting familiar practices.

II. LITERATURE SURVEY

Voice recognition, speech processing, and noise-robust audio analysis have been widely researched in recent years due to the increasing demand for intelligent, secure, and reliable human– machine interaction systems. Traditional speech models struggled with overlapping speech, noisy environments, variable recording conditions, and real-time processing constraints. To overcome these limitations, researchers have introduced advanced deep learning models such as RNNs, CRNNs, CNNs, and audio-visual learning systems. These studies demonstrate how modern neural networks improve accuracy, robustness, and adaptability in complex audio conditions.

Amberkar et al., in **[1]**, emphasized the role of Recurrent Neural Networks (RNNs) in improving continuous speech recognition tasks. Their study showed that RNN-based models trained on labeled datasets effectively capture temporal dependencies in

ISSN [Online]: 2583-2654

speech, resulting in improved recognition accuracy across sequential audio inputs.

Adavanne et al., in **[2]**, presented a Convolutional Recurrent Neural Network (CRNN) framework designed for Sound Event Localization and Detection (SELD). By combining spectrogram features with recurrent layers, the model demonstrated superior performance in detecting overlapping sound events in three- dimensional acoustic spaces.

Yousefi and Hansen, in **[3]**, proposed a CNN-based approach for recognizing overlapping speech segments. Their model utilizes MFCC features with deep convolutional layers to separate and identify multiple speakers within a single audio stream, significantly enhancing multi-speaker recognition and speaker count estimation.

Alharbi et al., in **[4]**, performed a systematic review of Automatic Speech Recognition (ASR) research from 2015–2020 using PRISMA methods. Their findings highlighted the primary challenges faced in ASR, including variability in accents, environmental noise, and limitations in real-time deployment. They identified gaps where further advancements are needed, particularly in robust speech modeling.

Afouras et al., in **[5]**, explored Deep Audio-Visual ASR systems that combine audio signals with lip-movement information. Their multimodal neural network outperformed traditional audio-only models, demonstrating particularly strong results in noisy environments where speech signals are distorted or partially masked.

Sankavi et al., in **[6]**, developed a deep-learning-based method for noisy speech classification. Their system, trained on noise-rich speech datasets, achieved improved accuracy in identifying degraded speech samples, proving effective for speech processing applications in challenging acoustic conditions.

Shetu et al., in **[7]**, compared multiple deep learning architectures for noise reduction under low Signal-to-Noise Ratio (SNR) conditions. Their evaluation identified several effective neural models capable of suppressing noise and enhancing speech clarity, particularly in extremely noisy environments.

Thakur et al., in **[8]**, reviewed modern advancements in NLP and AI for speech recognition. Their work summarized key algorithms, neural architectures, and language modeling frameworks while also highlighting remaining research gaps related to robustness, scalability, and dataset diversity.

Koç et al., in **[9]**, examined voice recognition techniques optimized for Edge-AI devices. By comparing phoneme-based and word-based models on microcontroller platforms, they found that phoneme approaches yield higher accuracy, lower memory consumption, and better real-time performance for lightweight embedded systems.

Cho and Wee, in **[10]**, introduced a multi-noise learning framework for speaker recognition. Their method leverages noise-invariant representations to maintain stable performance in diverse acoustic environments, resulting in significantly improved speaker identification accuracy under varying noise conditions.

III. PROBLEM DEFINITION

The challenge lies in designing a voice-based attendance system capable of:

- Accurately identifying speakers from high-dimensional audio inputs in real-time classroom environments.
- Handling background noise, overlapping speech, and variations in voice quality without compromising recognition accuracy.
- Detecting mimicked, altered, or fake voices to prevent proxy attendance and ensure security.
- Replicating the traditional roll-call process while automating verification without human involvement.

ISSN [Online]: 2583-2654

- Reducing dependency on controlled environments, unlike fingerprint or facial biometrics.
- Updating attendance records instantly with low latency for seamless classroom operation.

IV. METHODOLOGY

The Puretone system follows a structured pipeline to authenticate students' voices and automate attendance marking. The methodology consists of four main stages: audio capture, noise reduction, feature extraction, and speaker identification. Each stage ensures that the system performs accurately in real-time classroom environments.

1. ***Voice Capture :*** When a student's name is called, the system records their spoken response using a microphone. The audio is sampled at a fixed rate and prepared for analysis by removing silence and stabilizing volume levels.

2. ***Noise Reduction :*** To handle classroom noise, the recorded audio is passed through a deep learning–based noise suppression model (DeepFilterNet). This step removes background disturbances while preserving the speaker's unique voice characteristics.

3. ***Feature Extraction*** : The cleaned audio is converted into log-mel spectrogram features, which effectively represent the frequency patterns and temporal characteristics of the speaker's voice. These features serve as input for the recognition model.

4. ***Speaker Identification*** : A trained Convolutional Neural Network (CNN) compares the extracted features with stored voiceprints. The model generates a similarity score, which is used to determine whether the response matches the registered student.

V SYSTEM ARCHITECTURE

The architecture of the Puretone voice-based attendance system consists of multiple interconnected modules that work together to capture, process, analyze, and authenticate voice signals in real time. The system is designed to operate reliably in noisy classroom environments while ensuring secure and accurate speaker identification. The overall architecture is illustrated in Fig. 4 and described below.

A. Audio Input Layer : The system begins with the acquisition of raw audio signals. A microphone receives the primary voice signal along with background noise and other voice interferences. These combined inputs represent real-world classroom conditions where multiple sound sources coexist.

B. Signal Preprocessing Unit : The mixed audio captured by the microphone is sent to the Signal Preprocessing Unit, where filtering, normalization, and noise removal are performed. This stage enhances the quality of the audio by isolating the relevant speech segment and reducing unwanted disturbances.

C. Deep Feature Extraction Module : The cleaned audio is then processed by a Deep Feature Extraction Module, which converts the signal into high-level features such as log-mel spectrograms or learned embeddings. These features capture unique speaker characteristics needed for accurate recognition. The extracted features are stored in and retrieved from the Database for comparison during authentication.

D. AI Model Training and Anomaly Detection : The extracted features are used to train the AI model, typically a CNN-based or hybrid architecture for speaker identification. The model also incorporates an Anomaly Detection component that identifies mismatched or suspicious voice samples, helping to prevent proxy attendance and voice spoofing.

E. Decision Layer : Based on similarity scores and anomaly detection results, the Decision Layer determines whether the input voice matches the registered user. This decision is then forwarded to the backend for updating attendance records.

F. Backend Layer (FastAPI) : The backend, implemented using FastAPI, acts as the communication bridge between the AI model, database, and user interface. It:

- receives model predictions,

ISSN [Online]: 2583-2654

- updates and retrieves attendance data,
- handles API requests from the frontend.

G. Feedback Loop : A Feedback Loop continuously updates the system by sending new data to the database and retraining or fine- tuning the model when necessary. This improves accuracy over time and allows the system to adapt to voice changes and environmental variations.

H. User Interface : The User Interface (UI) allows educators or administrators to interact with the system. Through the UI, users can monitor attendance status, initiate recording, view logs, and manage data. The frontend communicates with the backend to provide real- time updates.

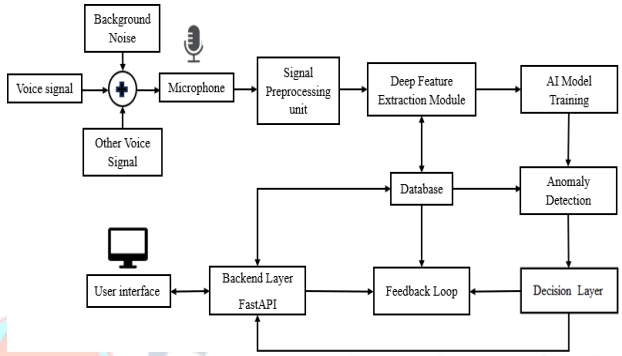

Fig 1. System Architecture of Puretone

VI DATA FLOW DIAGRAM

The data flow of the Puretone system outlines how raw audio is transformed into meaningful authentication decisions. The process begins with the collection of voice input and proceeds through several AI-driven stages that ensure noise-free analysis, secure verification, and continuous system improvement.

The flow starts with Audio Input, where the system captures the speaker's voice through a microphone. Since raw audio often contains background noise and environmental disturbances, it is immediately processed by a Noise Reduction module powered by DeepFilterNet. This step enhances clarity and ensures that only clean and relevant speech information is forwarded.

Once the audio is filtered, the system performs Speech-to-Text conversion using DeepSpeech. This converts the spoken response into text form, enabling further analysis and comparison. In parallel, the cleaned audio features are passed to the AI-driven Voice Recognition module, which identifies the speaker by analyzing unique vocal characteristics and comparing them with stored voiceprints.

Outputs from the recognition module are then evaluated across three critical analysis blocks. The Fake Voice Detection unit identifies spoofed or mimicked voice samples that could indicate proxy attendance. The Anomaly Analysis unit checks for irregularities such as abrupt changes in pitch, unfamiliar patterns, or mismatched acoustic features. The Stress/Emotion Detection module analyzes vocal cues to detect emotional variations or stress patterns, supporting advanced monitoring applications.

All these results merge into the Output and Continuous Learning stage, where the final decision—such as attendance marking or authentication approval—is generated. Simultaneously, the system stores new patterns and feedback to improve the model over time, enabling adaptive learning and increased accuracy with continued use.

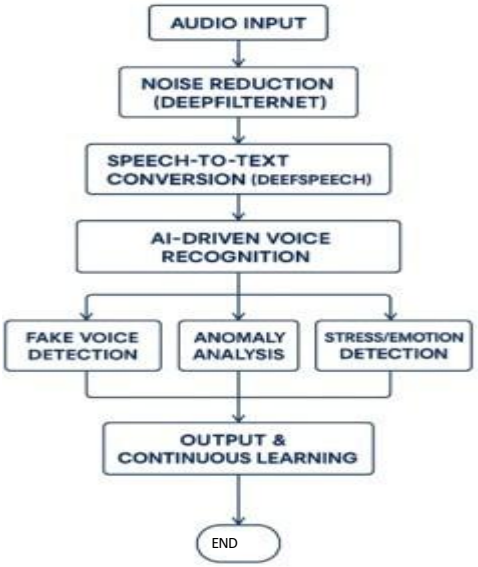

Fig 2 Flow Diagram

ISSN [Online]: 2583-2654

VII SOFTWARE AND HARDWARE REQUIREMENTS

The implementation of the Puretone AI-based voice recognition system requires a moderate computational setup capable of supporting real-time audio processing, deep learning inference, and web-based interaction. The hardware configuration includes an Intel Core i5 12th Generation processor with a minimum of 8 GB RAM, which ensures seamless execution of noise reduction models and speaker recognition algorithms. A storage capacity of at least 256 GB SSD is necessary to maintain datasets, logs, and trained model files. A high-quality USB microphone is essential for capturing clear voice input, and optional components such as CUDA-enabled GPUs can significantly accelerate model training and inference when required.

On the software side, the system operates on Windows 11 (64-bit), providing compatibility with essential development frameworks. Python serves as the primary backend programming language, supporting the integration of DeepSpeech for speech-to-text processing, DeepFilterNet for noise suppression, and TensorFlow or PyTorch for deep learning-based speaker recognition. Libraries such as Librosa, NumPy, Pandas, and SoundDevice enable audio analysis, preprocessing, and real-time recording. The web interface is developed using React.js, ensuring a responsive and lightweight user experience, while FastAPI handles backend communication and manages voice authentication requests. MongoDB functions as the main database for storing user information, attendance logs, and voice features, ensuring efficient retrieval and scalability. Visual Studio Code is used as the primary development environment due to its extensibility and support for full-stack development.

VIII BACKEND AND ALGORITHM

The backend of the PureTone system is implemented using Python and FastAPI, enabling efficient processing of audio inputs, extraction of voice features, and comparison with stored voiceprints. Librosa is used for audio handling and MFCC feature extraction, while

```python
def extract_features(audio_path):
    y, sr = librosa.load(audio_path, sr=16000)
    mfcc = librosa.feature.mfcc(y=y, sr=sr, n_mfcc=40)
    return np.mean(mfcc.T, axis=0)

def compare_voice_features(input_features, stored_features):
    return cosine_similarity([input_features], [stored_features])[0][0]

def identify_speaker(audio_path):
    input_features = extract_features(audio_path)
    best_match = None
    highest_score = 0.0

    for student in db.students.find():
        stored_features = np.array(student["voice_features"])
        score = compare_voice_features(input_features, stored_features)

        if score > highest_score:
            highest_score = score
            best_match = student["name"]

    if highest_score >= 0.80:
        return {"speaker": best_match, "confidence": highest_score}
    else:
        return {"speaker": "Unknown", "confidence": highest_score}
```

Fig 3 Backend Algorithm

a cosine similarity metric determines the match score between the input signal and registered student voice profiles. The backend interacts with the MongoDB database to retrieve stored features and finalize speaker identification based on confidence scores. This lightweight architecture ensures real-time processing with minimal latency.

The backend workflow consists of three major functions. The first function, extract_features(), loads the audio file, extracts MFCC features, and computes their mean representation. The second function, compare_voice_features(), calculates the cosine similarity between the input features and the stored features. The third function, identify_speaker(), iterates through all registered students in the database, computes similarity scores, and selects the highest-scoring match. If the score exceeds a predefined threshold (0.80), the system assigns the corresponding student as the identified speaker; otherwise, the result is marked as "Unknown."

Algorithm: Voice Identification Process

Step 1: Load the input audio file and extract MFCC features using a defined sampling rate.

Step 2: Convert extracted features into a numerical representation suitable for comparison.

ISSN [Online]: 2583-2654

Step 3: Retrieve stored voice features for each registered student from the database.
Step 4: Compute cosine similarity between the input features and stored voiceprints.
Step 5: Track the highest similarity score and the corresponding student identity.
Step 6: Compare the highest score with the threshold value (0.80). **Step 7:** If the score is equal to or greater than the threshold, return the student's identity with confidence.
Step 8: If the score is below the threshold, return "Unknown" with the computed confidence value.

IX RESULTS AND ANALYSIS

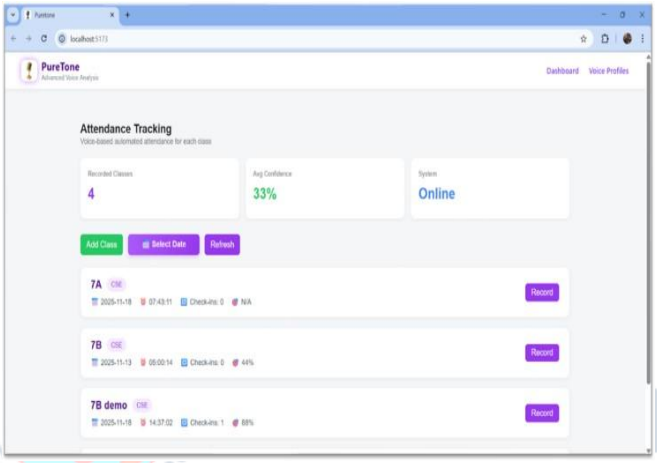

Fig 4 Puretone Dashboard interface

The Puretone system was tested in real-time classroom scenarios to evaluate its ability to record attendance, analyze voice responses, and generate meaningful feedback for instructors. The graphical interface, shown in Fig. 6 and Fig. 7, displays the automated attendance tracking results for multiple classes and demonstrates how the system processes and presents recognition outcomes.

The dashboard summarises key performance indicators such as the number of recorded classes, average confidence score, and system status, enabling users to quickly assess system performance. In the example captured, the system recorded four classes, achieving an average confidence level of 33%, which represents the model's certainty in distinguishing speakers based on their voiceprints. The interface also confirms that the system is online, indicating successful communication between the frontend, backend, and recognition modules.

For each class, the system displays essential attendance parameters including the recording time, number of check-ins, and the confidence score for identified students. The "Record" button allows instructors to initiate new recordings seamlessly. As shown in the results, one of the classes (7B demo) recorded one successful check- in with a confidence score of 88%, demonstrating the system's capability to authenticate students with high reliability under suitable conditions. Conversely, entries with lower confidence scores indicate either unclear audio input or insufficient voice training data.

The attendance table lists individual students along with their USN, time of response, confidence percentage, and attendance status. Students who did not respond or whose voice did not match the stored profile are marked as Not Marked, ensuring transparency in identification outcomes. The feedback option enables instructors to evaluate and refine system performance, contributing to continuous model improvement.

Overall, the results indicate that Puretone can successfully capture classroom attendance, process student voice responses, and provide an intuitive interface for reviewing recognition accuracy. The system demonstrates promising performance, particularly in controlled acoustic conditions, and highlights areas where further training or noise calibration may enhance recognition capability.

ISSN [Online]: 2583-2654

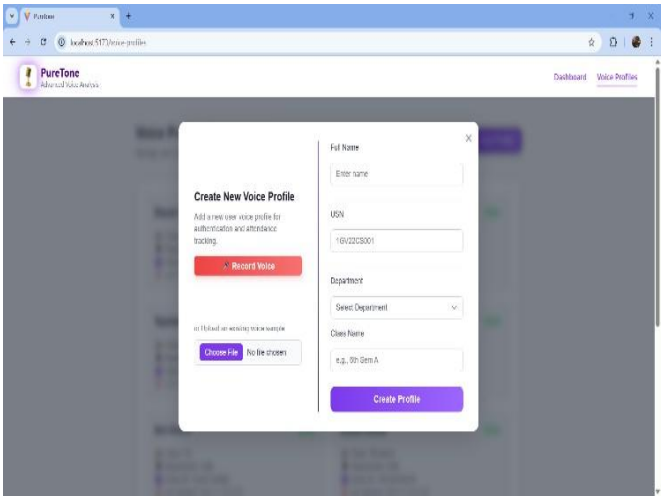

Fig 5 Voice profile creation interface

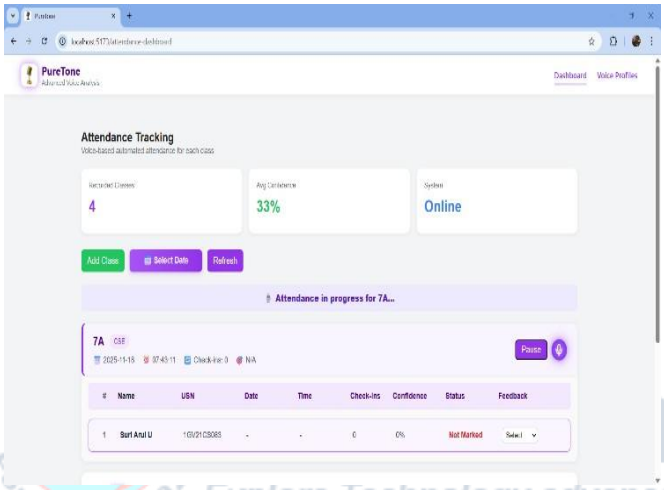

Fig 6 Demo Attendance progress interface

X FUTURE ENHANCEMENTS

The Puretone system offers a strong foundation for automated voice- based attendance, but several enhancements can further improve its accuracy, usability, and scalability. Future developments may include integrating multi-language voice recognition, enabling the system to support classrooms with diverse linguistic backgrounds. Adding mobile application support would allow teachers and administrators to manage attendance remotely and enable students to interact with the system from handheld devices.

To improve robustness in real-world environments, advanced anti- spoofing mechanisms using generative adversarial networks (GANs) can be introduced to better detect synthetic or replayed voices. The system can also incorporate emotion and stress analytics to support mental-health monitoring or identify unusual vocal behavior. Expanding the database infrastructure to cloud-based storage would allow large-scale deployment across institutions and enhance data accessibility.

Another important enhancement involves implementing edge AI processing, enabling the model to run on low-power devices and reducing dependency on high-end hardware. Continuous learning modules can also be strengthened so the system can adapt automatically to voice changes over time. These improvements would extend Puretone's applicability beyond attendance tracking into fields such as access control, virtual learning environments, and smart communication systems.

XI CONCLUSION

The Puretone system demonstrates an effective approach to automating classroom attendance through AI-driven voice recognition while retaining the familiar structure of traditional roll- call methods. By integrating noise reduction, speech-to-text processing, and deep learning–based speaker identification, the system provides a secure, contactless, and efficient solution for authenticating student responses. The inclusion of anomaly detection and fake voice identification further strengthens the system's reliability, reducing the risks of proxy attendance and misclassification.

The results show that Puretone can operate successfully in real-time classroom environments, offering accurate recognition and intuitive visual feedback through a responsive web interface. Although performance may vary with acoustic conditions and voice clarity, the architecture supports continuous learning and incremental improvement. Overall, Puretone presents a scalable and intelligent alternative to manual attendance processes, with the potential to extend into broader applications such as access control, communication enhancement, and educational analytics.

ISSN [Online]: 2583-2654

REFERENCES

[1] A. Amberkar, P. Awasarmol, G. Deshmukh and P. Dave, "RNN- Based Architecture for Improved Speech Recognition," Proc. Int. Conf. Signal Processing, pp. 112–118, 2018.

[2] S. Adavanne, A. Politis, J. Nikunen and T. Virtanen, "CRNN Framework for 3D Sound Event Localization and Detection," IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pp. 21–25, 2019.

[3] M. Yousefi and J. Hansen, "CNN-Based Overlapping Speech Recognition and Speaker Separation," IEEE Trans. Audio, Speech and Language Processing, vol. 28, pp. 326–338, 2020.

[4] H. Alharbi, M. Alsubhi, R. Almotairi and S. Choudhury, "Automatic Speech Recognition: A Systematic Review (2015– 2020)," Journal of Intelligent Systems, vol. 36, no. 4, pp. 455–468,2021.

[5] T. Afouras, J. S. Chung, A. Senior, O. Vinyals and A. Zisserman, "Deep Audio-Visual Speech Recognition in Noisy Environments," IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP),pp. 6568–6572, 2022.

[6] Sankavi K., Jyothish Lal G. and Premjith B., "Deep Learning– Based Noisy Speech Classification," International Journal of Speech Technology, vol. 25, pp. 215–224, 2023.

[7] S. S. Shetu, E. A. P. Habets and A. Brendel, "Deep Learning Approaches for Noise Reduction in Low SNR Conditions," IEEE Access, vol. 11, pp. 15423–15432, 2023.

[8] R. Thakur, L. Ahuja, S. Vashisth and R. Simon, "A Review on NLP and AI Advancements in Speech Recognition," ACM Computing Surveys, vol. 55, no. 7, pp. 1–29, 2023.

[9] Y. Koç, A. A. Tarçın and D. Köse, "Voice Recognition Models for Edge-AI Devices: A Comparative Study," IEEE Internet of Things Journal, vol. 11, no. 2, pp. 976–986, 2024.

[10] S. Cho and K. Wee, "Multi-Noise Learning for Robust Speaker Recognition," IEEE Trans. Neural Networks and Learning Systems, vol. 36, no. 1, pp. 88–97, 2025.