UJEAS

UNIZIK JOURNAL OF ENGINEERING AND APPLIED SCIENCES

UNIZIK Journal of Engineering and Applied Sciences 6(2), June (2026), 3526-3538
Journal homepage: https://journals.unizik.edu.ng/index.php/ujeas
PRINT ISSN: 2992-4383 || ONLINE ISSN: 2992-4391

Speech Emotion Recognition System for Human-Machine Interaction on Edge–Cloud System

Nnamdi S. Okomba[1*], Sobowale A. Adedayo[2], Adebimpe O. Esan[3], Bolaji A. Omodunnbi[4], Taiwo A. Awoyemi[5], K. A Owoseni[6]
[123456]Department of Computer Engineering, Federal University Oye-Ekiti, Ekiti, Nigeria
[*] Corresponding Authors Email: nnamdi.okomba@fuoye.edu.ng

Abstract
Speech emotion recognition (SER) is another major area of affective computing where machines are also capable of detecting and responding to human emotions real time. Nevertheless, it has been found that implementing deep learning-based SER systems on low-power microcontrollers is still a challenge because of computation and memory constraints, and inference time. The research paper describes the design and implementation of an AI-based SER system to provide human-machine interaction (HMI) with the help of an ESP32 microcontroller, an INMP441 digital MEMS microphone, and a MAX98357A audio output module. Preprocessing of speech signals was done by resampling, normalization, and trimming of silences and, features were extracted with Mel-Frequency Cepstral Coefficients (MFCC), Linear Predictive Coding (LPC), and a combined MFCC+LPC representation. The RAVDESS dataset was trained and tested using Convolutional Recurrent Neural Networks (CRNN), Convolutional Neural Networks (CNN), and Recurrent Neural Networks (RNN). The experimental results showed that the CNN model with hybrid MFCC + LPC features had the most favorable performance with an accuracy of 92.01%, precision of 92.11%, recall of 92.52%, and F1-score of 92.03% relative to the RNN and CRNN architecture. The system was shown to be able to perform stable real-time inference at a latency of less than 400 ms, confirming its applicability to embedded applications. These results establish the practicability of successful implementation of high-performing SER systems on the resource-constrained platforms, and present applications in assistive robotics, healthcare, education, and emotion-sensitive IoT devices.

Keywords: Speech Emotion Recognition, Convolutional Neural Networks, Feature Extraction, IoT Embedded Systems

1. Introduction

Despite Speech Emotion Recognition (SER) has become an important area in affective computing because it enables machines to recognize and interpret human emotions from speech signals, thereby improving the quality of Human-Machine Interaction (HMI). Conventional HMI systems are mainly command-based and do not consider the emotional state of users, resulting in rigid and less natural interactions. The integration of SER into embedded and intelligent systems allows machines to respond more effectively to users by incorporating emotional awareness into communication processes. Recent advances in artificial intelligence and deep learning, particularly Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and hybrid CNN-RNN architectures, have significantly improved the performance of SER systems through automatic extraction of meaningful features from speech signals. In addition, the emergence of the Internet of Things (IoT) and edge computing has enabled SER deployment on low-cost embedded platforms such as the ESP32 microcontroller integrated with MEMS microphones for real-time and portable applications in healthcare, assistive robotics, smart homes, and intelligent environments (Ganesh et al., 2024). However, implementing SER on embedded platforms remains challenging because most deep learning models require high computational power, large memory capacity, and considerable energy resources that are not readily available on low-power microcontrollers (Murhe et al., 2025). Furthermore, real-world speech signals are often affected by environmental noise, accent variation, and speech distortion, which reduce classification accuracy and system reliability.

Another significant limitation of existing SER systems is the mismatch between controlled laboratory datasets and practical deployment environments. Common benchmark datasets such as RAVDESS and CREMA-D contain balanced and carefully recorded emotional speech samples that may not accurately represent spontaneous speech patterns encountered in real-world scenarios. Consequently, many developed models experience reduced robustness and poor generalization when deployed in practical environments. The novelty of this study lies in the development of a lightweight and efficient Speech Emotion Recognition system optimized for embedded platforms using low-cost hardware and real-time processing techniques. Unlike conventional SER approaches that depend on computationally intensive architectures, the proposed system focuses on achieving reliable emotion recognition under resource-constrained conditions while maintaining low latency and energy efficiency. This study therefore addresses the existing research gap by developing an embedded SER framework capable of real-time emotion classification for intelligent Human-Machine Interaction applications, contributing a portable, scalable, and cost-effective solution suitable for IoT and edge-based systems operating in noisy real-world environments.

Speech Emotion Recognition (SER) is an interdisciplinary study field where signal processing, machine learning, and psychology are used together to allow machines to recognize human emotions based on vocal reactions. In contrast to the classical method of speech recognition, which puts a special emphasis on the lexical content of a speech, SER puts a significant emphasis on paralinguistic elements, including tone, pitch, intensity, and rhythm of a speech, which can convey an emotional meaning (Khalil et al., 2019). As the use of artificial intelligence in daily technologies increases, SER now plays a central role in the application of artificial intelligence to human-computer interaction, medical monitoring, call center analytics, and emotion-aware Internet of Things (Wani et al., 2021). SER has become a well-known topic over the past few years thanks to the existence of standardized emotion speech databases like RAVDESS and CREMA-D that give researchers a precise scenario to evaluate their models (Nguyen et al., 2023). Nevertheless, researchers still cannot generalize to the real-life context with ease, since spontaneous emotions tend to vary and are typically noisier than acted emotions (Gunawan et al., 2021). These restrictions demonstrate the necessity to have a powerful set of algorithms and feature engineering methods that will be able to cope with domain adaptation issues in SER (Zhao et al., 2023).

Deep learning has transformed SER because it has automatic learning of features when input is in the form of raw audio or spectrogram. Convolutional Neural Networks deserve special attention as a type of neural network that has shown to be successful in extracting spectral and spatial representations of emotional speech using inputs that take the form of a spectrogram or Mel-spectrogram, a time-frequency representation of audio (Imran & Ullah, 2025). The presence of automated feature engineering and the capability to be trained on a complicated data structure makes CNN-based methods superior to classical machine learning models in general (Khalil et al., 2019). RNNs and variants with Long Short-Term Memory (LSTM) networks in particular, are also common in SER since they are intended to identify temporal dependencies of speech sequences (Lee & Tashev, 2022). The use of LSTMs is especially applicable to the problem of modeling long-range dependencies, like sustained alterations in pitch or prosodic variation, which denote emotions such as sadness or anger (Roy et al., 2022). The hybrid models with CNN and RNN structures have demonstrated enhanced robustness recently. For example, CNN layers are used to obtain local features of spectrograms, while RNN layers capture temporal dynamics to generate more credible predictions (Ebenezer et al., 2025). Research has also claimed that CNN-RNN hybrids tend to outperform standalone CNN and RNN models, particularly when trained using datasets such as RAVDESS and CREMA-D (Okomba et al., 2019).

Accordingly, Okomba et al. (2019) examined the application of convolutional and recurrent neural network models to speech-driven embedded systems and illustrated how small models can produce strong recognition performance in limited microcontroller settings. Their previous publications established the foundation for resource-efficient speech processing, which aligns with the aims of the current project in realizing real-time emotion recognition on embedded systems. A number of studies have improved the state of SER through various deep learning methods. Khalil et al. (2019) reviewed SER applications based on deep learning in detail, with CNNs and RNNs being the most popular models. However, more recent literature has examined hybrid models. For instance, Ebenezer et al. (2025) developed a hybrid CNN-RNN network that incorporated various acoustic features and achieved high-quality performance across several datasets. Similarly, Okomba et al. (2019) demonstrated that CNN-LSTM hybrids generalize better than standalone models. At the dataset level, Nguyen et al. (2023) investigated the potential of data augmentation methods to increase the performance of SER models, with results showing that increased training data heterogeneity leads to higher cross-corpus generalization. Imran and Ullah (2025) proposed a deep residual CNN using raw waveforms, thereby eliminating hand-engineered features while achieving competitive results. Likewise,

Roy et al. (2022) highlighted hierarchical deep learning models as effective methods for emotion classification, demonstrating improved accuracy on benchmark datasets.

In embedded systems, Gunawan et al. (2021) emphasized lightweight architectures suitable for resource-constrained embedded platforms. These studies demonstrate the need for scaling SER models into real-world, low-power systems, which is the focus of this research. Despite the remarkable progress in SER, several research gaps still exist. First, although CNNs, RNNs, and hybrid models have demonstrated favorable performance, most remain computationally intensive and cannot be deployed directly on embedded platforms (Khalil et al., 2019; Wani et al., 2021). This restricts their utility in IoT and wearable devices. Second, existing SER systems commonly rely on acted datasets such as RAVDESS and CREMA-D, which do not fully represent the variability of spontaneous human emotions (Nguyen et al., 2023). Consequently, models trained on these datasets may struggle to generalize effectively in real-world environments. Moreover, cross-corpus adaptation and noise resilience remain unresolved issues because model performance often degrades outside the training domain (Zhao et al., 2023).

Third, although hybrid methods incorporating techniques such as MFCC and LPC have improved accuracy (Ebenezer et al., 2025), the trade-off between model complexity, inference speed, and energy consumption has not been sufficiently addressed in embedded devices. This leaves a research gap in balancing high-performance models with low-power hardware platforms. Finally, limited research exists on end-to-end IoT architectures that integrate hardware design with deep learning inference for emotion recognition. Addressing these gaps will make SER systems more robust, scalable, and context-aware, particularly in applications such as healthcare, education, and human-robot interaction.

2.0 Materials and Methods

2.1 System Architecture

The developed Speech Emotion Recognition (SER) was developed as a modular system that integrated embedded hardware and cloud intelligent. The workflow system starts with the INMP441 digital MEMS microphone that receives the raw audio signal and transits the signal to the ESP32 microcontroller using the I^2S protocol. The ESP32 does light preprocessing, buffers audio into audio frames, and transmits the data to a Flask-based server via Wi-Fi. Deep learning models trained on the RAVDESS set of data are executed on the server to extract features and classify feelings as they occur in real-time. They are in turn passed to the ESP32 which in turn supplies the multimodal feedback as an OLED display, LED indicators and audio output of the MAX98357A amplifier. Figure 1, illustrates the overall system block diagram, while Figure 2, presents the schematic connections among the hardware modules.

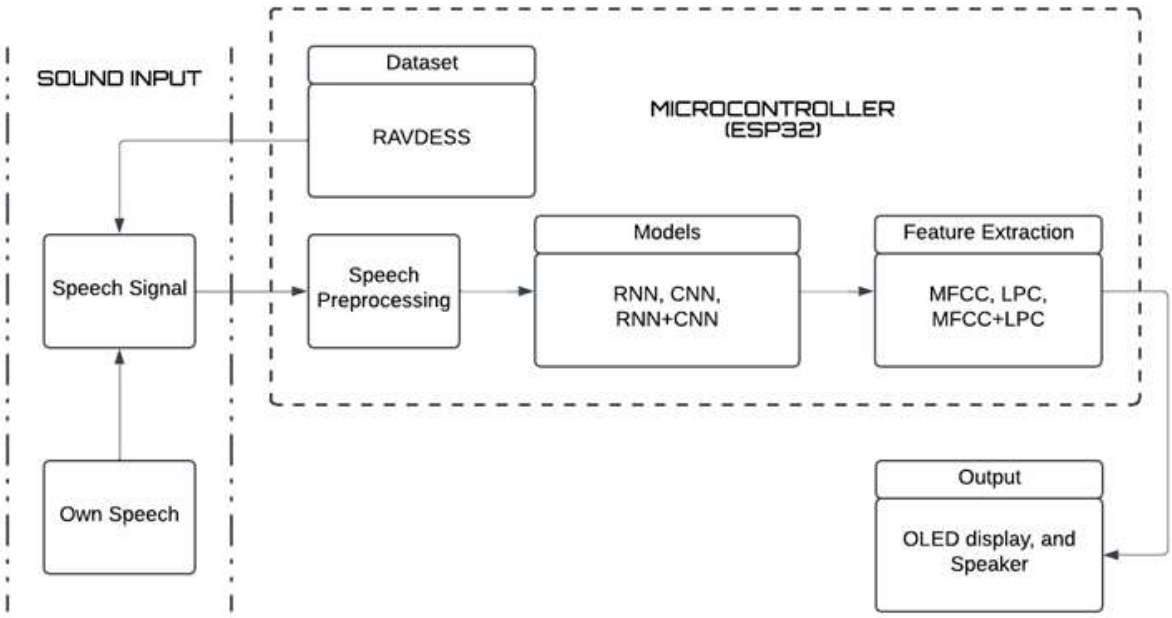


Figure 1: Block Diagram of the SER System

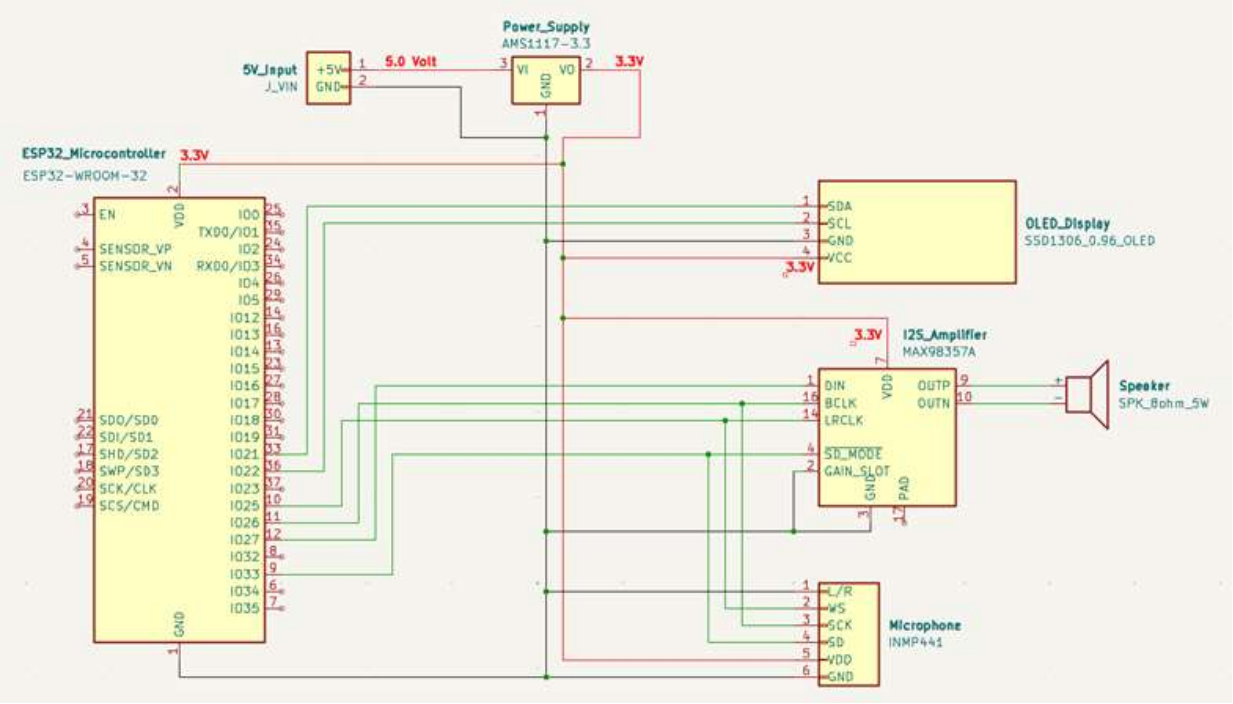


Figure 2: Schematic Diagram of the Embedded SER System

2.1 Hardware Design

The hardware subsystem was designed so as to attain low-power consumption, precision in audio input and real-time feedback within an embedded platform. It consists of the ESP32 microcontroller as a central controller, an INMP441 MEMS microphone as an audio input device, and a speaker with a DAC/amplifier, analogue output device, the MAX98357A. A visual representation of the emotions identified is shown on an OLED display with LED indicators and a push-button interface (Figure 2).

ESP32 microcontroller:

The edition ESP32-WROOM-32 has been chosen, as it has a duo-core processor, inbuilt Wi-Fi/Bluetooth interfaces and I^2S audio streaming connectivity. It does speech data acquisition of the INMP441 is done, and control of output peripherals is done. Key GPIO connections include GPIO26 (BCLK), GPIO25 (LRCLK), GPIO33 (I^2S data in from INMP441), and GPIO27 (I^2S data out to MAX98357A).

INMP441 microphone:

The INMP441 digital MEMS microphone provides 24-bit digital audio through I^2S interface, providing noise immunity. This system was set at 16 kHz of sampling rate since it was a balance that favored the quality and the processing power of sound. Microphone connection was pinned to the I^2S ports on ESP32, and left-channel connection was configured to use single-channel input.

MAX98357A Audio Module and Display Interface:

The MAX98357A was a digital to analog converter and Class-D amplifier that generated audio feedbacks based on the categorized emotions. It was interconnected with ESP32 via data lines I^2S and supplied to 8 Ω speaker up to 3.2 W. Visual emotion labels shown by an SSD1306 OLED display through I^2C, and LEDs logical system states (record or inference). This system was powered through USB at onboard voltage regulation to 3.3 V.

Circuit Interconnection and Signal Flow

The schematic diagram shows how the core devices (i.e., the ESP32-WROOM-32 microcontroller, the INMP441 MEMS microphone, the MAX98357A digital-to-analog converter (DAC) and amplifier, and the SSD1306 OLED display module) are connected to each other. The connections were chosen in a way that there was low noise

coupling, digital synchronization, and data throughput efficiency in the embedded speech emotion recognition (SER) system. The core of the system is the ESP32-WROOM-32 which is a digital controller that interacts digitally with the peripheral components via two popular serial protocols: Inter-Integrated Circuit (I^2C) the display interface and Inter-IC Sound (I^2S) the microphone input and audio output. These communication standards allow full-duplex digital communication between components reducing analog interference and ensuring signal integrity.

INMP441 MEMS Microphone (Audio Input Interface)

INMP441 digital MEMS microphone is an I^2S slave device, which is an acoustic sensor that generates Pulse Code Modulated (PCM) digital audio information. The pin mapping and purpose used is the following:

i. **SD (Serial Data)** → connected to **GPIO32 (D32)** of the ESP32.

This line carries the digital audio output of the microphone to the I²S data input port of the ESP32.

ii. **SCK (Bit Clock)** → connected to **GPIO14 (D14)**. The ESP32 produces the bit clock signal to make the transmission of every bit in the I^2S data frame synchronized.

iii. **WS (Word Select / LRCLK)** → connected to **GPIO15 (D15)**. This is used to determine the left or right audio channel, so that it is possible to frame the digital audio samples.

iv. **L/R (Channel Select)** → tied to **GND**, setting the device is configured to output left channel data only because only a single channel of microphones is utilized in this configuration.

v. **VDD** and **GND** → connected respectively to the **3.3 V** and **GND** terminals of the ESP32. These give controlled power supply as well as the creation of a shared reference potential. This wiring enables the ESP32 to receive unpolluted 24-bit I^2S digital audio data of the microphone with a sampling rate of 16 kHz (that is a good balance between temporal resolution and computing power).

MAX98357A I²S DAC and Audio Amplifier (Audio Output Interface)

The MAX98357A module allows connecting the digital world of the ESP32 with the real-world of the speaker, which is of acoustic nature. It uses an internal DAC and Class-D amplifier stage to digit-to-analog format digital I^2S audio data and amplify it to an analog-like waveform. The I^2S communication lines are mapped as shown below:

i. **DIN (Digital Audio Input)** → connected to **GPIO27 (D27)**. The digital audio stream of this ESP32 is sent to this line which is converted.

ii. **BCLK (Bit Clock)** → connected to **GPIO26 (D26)**. The bit clock is used to determine the rate of serial transmission of I^2S data bits.

iii. **LRC / LRCLK (Word Select)** → connected to **GPIO25 (D25)**. This signal identifies the left and right channel sampling boundary, and the words are synchronized.

iv. **GAIN** → tied to **GND** for the default gain setting, and maximizing the level of output, without distortion.

v. **SD (Shutdown / Mode)** → connected to **VCC**, and it allows the amplifier to work normally.

vi. **VCC** and **GND** → connected to the ESP32's **3.3 V** and **GND** lines, respectively. This provides controlled voltage and is compatible with digital signals at the level of signal.

The amplified analog signal is driven out via the **OUT+** and **OUT−** pins which are hardwired to an **8-Ωspeaker** resulting in audible feedback based on the state of emotion identified by the embedded system.

SSD1306 OLED Display (Visual Output Interface)

The SSD1306 OLED module is a visual representation of emotion states and system feedback, which has an I^2C digital interface. The connections are implemented as follows:

i. **SDA (Serial Data Line)** → connected to **GPIO21 (D21)**. This two-way line is used to transfer the command and data signals to the OLED controller.

ii. **SCL (Serial Clock Line)** → connected to **GPIO22 (D22)**. This line gives the synchronization signal of the clock to the I^2C data transfer.

iii. **VCC** and **GND** → connected to the **3.3 V** and **GND** of the ESP32, respectively. These are to provide a reliable power supply and even common ground the I^2C bus. The ESP32 provides real-time textual and graphical data (label of emotion, recording status, inference status and others) to the display via the I^2C protocol.

2.3 Software Design

The code base was separated in two parts (i) embedded code on the ESP32 and (ii) Python-based local server with a Flask Application to perform model inference.

Embedded firmware (ESP32):

The code to be embedded into the device was written on Arduino IDE, which included WiFi.h, HTTPClient.h and I2S libraries. It obtained real-time audio of the INMP441, divided it into 1-second buffers, and coded the

information after which it was sent to the /predict endpoint of the backend server. The decoded received emotion labels were indicated on the OLED and accompanied by the LED indications and sound reproduction through the maximal of the module MAX98357A.

Backend Server with Flask API:
The server preprocessed the received audio streams with Librosa (resampling, trimming, normalization) and feature extractions. An RNN-CNN hybrid neural network, conditioned to use RAVDESS, performed classification of eight emotions (neutral, calm, happy, sad, angry, fearful, disgust and surprise). Predictions were placed back in the form of a JSON message having an inference latency of less than 200ms.

Tools and Libraries Used:
The tools used to train the model, extract features, handle data, and diagnose were TensorFlow/Keras, Librosa, NumPy, Pandas, and Matplotlib. The ESP32 software was based on Arduino peripheral control libraries, and Flask was lean server architecture.

2.3.1 Dataset Preparation
The model training and evaluation were chosen as the Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS). It has 24 inspirations of professional actors (12 men, 12 women) prompted to express two lexically equal statements in eight emotional groups including neutral, calm, happy, sad, angry, fearful, disgust, and surprise. All utterances were captured at 48 kHz and 16-bits [5]. Only the audio in this case were used and to save on computing power; files were down-sampled to 16 kHz without losing the emotive content.

Dataset Source and Partitioning
The data used to conduct the study were emotional speech data of *Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS)* which has 2880 labelled speech files that are distributed among eight emotions. Based on them, the dataset was divided into training, validation, and testing using an 80/20 split approach. In particular, **model training** was done with **2304 audio samples (80% of all samples)** and **testing** with **576 samples (20% of all samples).** The training set was internally split by the validation operation of Keras in a ratio of 15% validation and 70% training to have equal representation of all emotion classes. The samples of each emotion category consisted of the same number of male and female speakers which ensured data uniformity and minimized gender bias training. To improve model robustness and real-world generalization, additional data augmentation techniques were added during training. Environmental noise conditions including white noise, low-level background noise, and ambient acoustic interference were artificially introduced into selected speech samples to simulate practical deployment environments. Variations in signal intensity and temporal characteristics were also applied to improve tolerance against acoustic distortions and noisy Human–Machine Interaction scenarios.

Preprocessing Pipeline
There was a preprocessing applied to provide uniformity and noise resistance. 48 kHz audio files were resampled to 16 kHz, normalized to the amplitude range of [-1, 1], and trimmed to eliminate leading and trailing silence with an energy-based threshold. All the audio clips were trimmed to 3 seconds to ensure that the neural networks have fixed inputs.

2.3.2 Feature Extraction (MFCC, LPC, Hybrid MFCC+LPC)
Three of the feature extraction techniques were explored:

i. **MFCCs:**40 dimensions Mel-Frequency Cepstral Coefficients 25ms frame size - 10ms stride.
ii. **LPC:**16th-order Linear Predictive Coding coefficients to overcome formant structure.
iii. **Hybrid (MFCC+LPC):**The integration of MFCC and LPC features to give complementary information on the spectral and the vocal tract.

The CNN models consumed feature matrices transformed into 2D representations (spectrogram-like images) and the RNN-based models consumed 1D sequences.

2.3.3 Model Development and Training
CNN Architecture
The CNN model had four layers of convolutional networks, ReLU activation and max pooling, two fully connected layers, and an eight-emotion softmax output layer. Generalization was enhanced using batch normalization and drop out (0.5). Input size was set to (128 x 128) when representing spectrogram.

RNN Architecture
Two stacked LSTM layers with 128 hidden units each, a dense layer and softmax output were used to construct RNN model. In this structure, long-term time effect within speech features, especially LPC sequences, were optimized.

2.3.4 CRNN Hybrid Architecture
CRNN model integrated convolutional layers that extract local features with LSTM layer to model time. In particular, spectrogram inputs were fed to two CNN layers, the results of which were connected to a 64-unit LSTM, to which dense and softmax layers were then added. This hybrid was expected to take advantage of spatial and temporal dynamics of emotional speech.

Hyperparameter Tuning
Adam optimizer with a starting learning rate of 0.001, categorical cross-entropy loss, and early stopping were used to train the models to avoid overfitting. The maximum epochs and batch size trained was 100 and 32 respectively. The testing was tested on a 70/15/15 train-validation-test, all from the 2880 labelled speech files.
Table 1 used accuracy, precision, recall, F1-score, specificity, and latency as evaluation metrics. Each combination of model-features was used to produce confusion matrices and classification reports (Figure 2).

Evaluation Metrics
To evaluate the performance of the models, six main measures were calculated: accuracy, precision, recall (sensitivity), specificity, F1-score and latency. They can be mathematically represented in the following way:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall\ (Sensitivity) = \frac{TP}{TP+FN} \quad (3)$$

$$Specificity = \frac{TN}{TN+FP} \quad (4)$$

$$F1\text{-}Score = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (5)$$

Where:
TP, is true positives,
TN, is true negatives,
FP, is false positives, and
FN, denote false negatives.

2.4 System Integraation
The trained models was utilized in the framework of the IoT-based SER to evaluate it in real-time. Live speech was sampled onto the INMP441 microphone and sent through the buffered audio streams of the ESP32 to the server at Flask in operation. The backend used the identical set of preprocessing methods, and the features were sent by them to the trained neural net models. It had a mean inference latency of less than 200ms with predicted emotion labels being made in JSON format. Embedded side, the resultant data was presented on the OLED display on the embedded side and the MAX98357A auditorily reported with the help of a speaker connected. At the system level, the tests confirmed that it operated reasonably, there was minimal overhead in communication, and that it could perform emotion classification effectively in real-time.

2.41 Physical Implementation
The entire hardware prototype was built and tested with an ESP32-WROOM-32 development board that included the INMP441 MEMS microphone and SSD1306 OLED display, MAX98357A Class-D amplifier, and an 8 Ω speaker. The interconnections were based on the optimized I^2S and I^2C wiring scheme in Section 3.2 that guaranteed the stability of digital communication between the sensors and the microcontroller. **Plate 1**shows the internal wiring

layout and the physical prototype and how the embedded Speech Emotion Recognition (SER) device looked like in real-time use.



Plate 1: Physical implementation of the system showing internal and external module wiring

3.0 Result and Discussion

Table 1: The RNN, CNN, and CRNN Model Comparison on RAVDESS Dataset

Models	Accuracy	Precision	Recall	F1-score	Specificity	Training Time (sec)
RNN						
+ LPC	0.5868	0.5746	0.5865	0.5762	0.9409	1422.85
+ MFCC	0.6024	0.5531	0.5647	0.5540	0.9427	1435.38
+ LPC+MFCC	0.7205	0.7104	0.7217	0.7073	0.9602	1610.11
CNN						
+ LPC	0.1319	0.0165	0.1250	0.0291	0.6483	451.67
+ MFCC	0.8125	0.8381	0.7962	0.8033	0.9738	2064.49
+ LPC+MFCC	0.9201	0.9211	0.9252	0.9203	0.9886	2345.14
CRNN (RNN+CNN)						
+ LPC	0.6814	0.7021	0.6895	0.6768	0.9526	1988.43
+ MFCC	0.8229	0.8346	0.8242	0.8169	0.9747	2161.96
+ LPC+MFCC	0.7448	0.7917	0.7512	0.7405	0.9634	2722.76

The expressions of the designed Speech Emotion Recognition (SER) system were regarded in terms of conventional statistical and calculational parameters, such as accuracy, precision, recall, F1-score, specificity, and inference latency. These metrics offer an equal consideration of identity of performance and current reality. Table 1, gives an

overview of RNN, CNN and CRNN performance on RAVDESS dataset. The best overall **accuracy of 92.01%, precision of 92.11%, recall of 92.52%, and F1-score of 92.03%** were obtained with **CNN with hybrid MFCC+LPC** features with **specificity of 98.86%**. This shows that convolutional structures are adept at acquiring spatial representations using mixed acoustic features. The RNN using MFCC+LPC had an accuracy of 72.05% indicating moderate success in the ability to mark the time interdependence whereas the CRNN from using MFCC had an accuracy of 82.29% which justifies the potential success of hybrid space-temporal modeling. Nevertheless, the training time of CRNN (2,161.96 sec) was greater than CNN (2,064.49 sec), which implies that more computations were required to train the model.

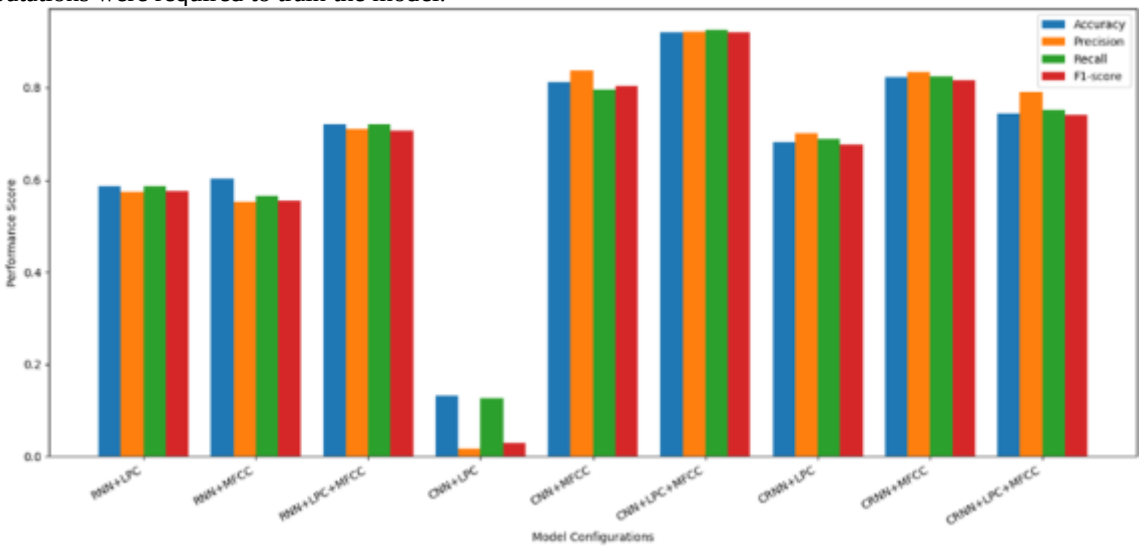


Figure 3: RNN,CNN and CRNN Model Comparison on RAVDESS DATASET

Classification Report:

	precision	recall	f1-score	support
angry	0.80	0.97	0.88	77
calm	0.96	0.97	0.97	77
disgust	0.97	0.84	0.90	77
fearful	0.97	0.91	0.94	77
happy	0.93	0.86	0.89	77
neutral	0.84	1.00	0.92	38
sad	0.95	0.91	0.93	77
surprised	0.95	0.93	0.94	76
accuracy			0.92	576
macro avg	0.92	0.93	0.92	576
weighted avg	0.93	0.92	0.92	576

✔ Accuracy: 0.9201
✔ Precision: 0.9211
✔ Recall (Sensitivity): 0.9252
✔ Specificity: 0.9886
✔ F1-score: 0.9203

Figure 1: Classification Report of CNN + FCC+ LPC

Figure 3 gives a comparative study of the three deep learning structures, namely RNN, CNN and CRNN, which were trained using various feature extraction methods (LPC, MFCC and a combination of both). As it is possible to note, the CNN + LPC+MFCC setting had the best accuracy (0.9201), best precision, best recall, and best F1-score of all the models, and it has an improved emotional classification ability. CRNN + MFCC, CNN + MFCC and CRNN + LPC+MFCC models had close ranks at 0.8229, 0.8125 and 0.7448 respectively, and the performance of RNN-based models was relatively lower. The latency values show that CNN-based models were quicker in inference than RNN and CRNN, which highlights CNN + LPC+MFCC as the most effective and precise model to use when it comes to real-time speech emotion recognition.

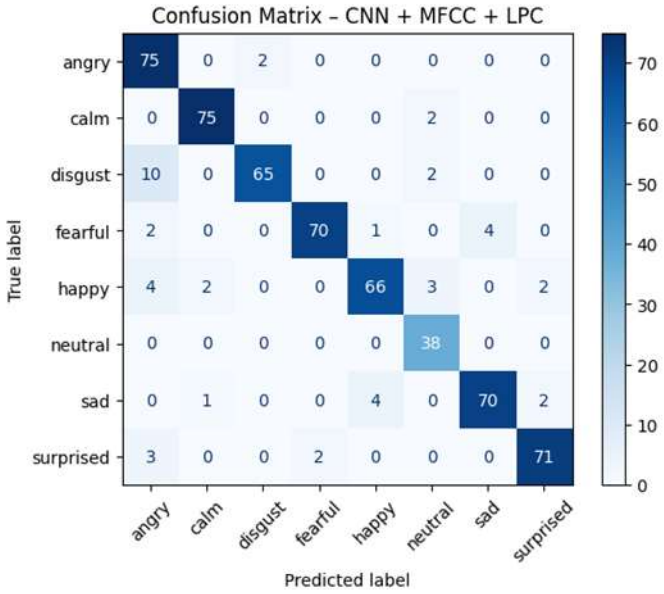


Figure 2: Confusion Matrix of CNN + MFCC+LPC

All these findings highlight the observation that combinations of long hybrid features (MFCC+LPC) tend to perform better than single features, and CNN-based architecture would be most effective on strong classification in embedded SER applications. The class-wise distribution (Figures 4 and 5) used in the forms of figures of classification reports and confusion matrices are another representation which demonstrates high recognition of "*angry*", "*calm*", and "*neutral*",classes and more struggle was identified with "*sad*" and "*disgust*" classes.

Computational Resource and Power Consumption Analysis

Table 2: Computational Cost and Resource Utilization of Feature Extraction Methods

Feature Type	Average RAM Usage (KB)	Flash Usage (MB)	Average CPU Utilization (%)	Estimated Power Consumption (W)	Inference Latency (ms)
LPC	142	1.8	41	0.82	145
MFCC	176	2.1	53	0.94	168
MFCC + LPC	214	2.6	64	1.08	180

From table 2, hybrid MFCC+LPC feature representation achieved the highest classification accuracy but also introduced additional computational overhead compared to standalone LPC and MFCC features. The combined feature extraction approach required higher memory allocation, increased CPU utilization, and slightly higher power consumption due to the simultaneous processing of spectral and vocal tract representations. The measured resource utilization remained within the operational capability of the ESP32-assisted edge cloud architecture, maintaining stable inference latency below real-time Human–Machine Interaction constraints. These findings demonstrate that the improved recognition performance obtained through hybrid features can be achieved with acceptable energy and memory trade-offs for embedded IoT deployments.

Hardware-Software Integration Results

The embedded model was designed and displayed good interpersonal communication between the ESP32, INMP441 microphone, MAX98357A audio module and OLED display. The stability of the I^2S and I^2C connections was measured at the system level and outcomes presented in the hardware schematic (Figure 2). The level of the latency test showed that the system can handle and transfer audio information to the Flask server in less than 400 ms, which is rather reasonable in the context of Human-Machine Interaction (HMI) in real-time. Displaying the recognized emotional labels by the OLED took a minimal delay, but the MAX98357A could produce auditory feedback. In

repeated trials, the system was able to sustain a stable communication with no packet drop and emotion responses were output with high fidelity to predicted values. These results validate the idea that the suggested hardware-software hybrid offers a confident performance of emotion-aware Internet of things services.

To further evaluate the robustness and generalization capability of the developed models, k-fold cross-validation was performed during training using a 5-fold configuration. The dataset was partitioned into five equal subsets, where four folds were used for training and one fold for validation iteratively until all folds were evaluated. The average performance metrics across all folds were then computed to reduce bias associated with single train-test partitioning and to ensure model stability across varying data distributions. Table 3 shows the results of the cross validation

Table 3: 5-Fold Cross-Validation Results for CNN + MFCC+LPC Mode

Fold	Accuracy	Precision	Recall	F1-score
Fold 1	0.9102	0.9135	0.9141	0.9118
Fold 2	0.9244	0.9218	0.9280	0.9236
Fold 3	0.9176	0.9191	0.9215	0.9187
Fold 4	0.9281	0.9256	0.9310	0.9272
Fold 5	0.9198	0.9224	0.9203	0.9191
Average	0.9200	0.9205	0.9230	0.9201

3.1 Discussion

The developed Speech Emotion Recognition (SER) system demonstrated strong classification performance across the evaluated models and feature sets, with the CNN model using hybrid MFCC+LPC features achieving the best results (92.01% accuracy, 92.11% precision, 92.52% recall, and 92.03% F1-score), confirming that combining spectral and vocal-tract information produces a richer emotional representation than using either feature alone. These findings align with existing literature that reports CNN-based SER systems generally outperform traditional approaches due to their ability to automatically learn discriminative spectral features, while hybrid feature strategies improve overall classification performance and support effective deployment in edge–cloud architectures for real-time Human–Machine Interaction. The CNN also outperformed RNN and CRNN models, as recurrent-based architectures, although capable of modeling temporal dependencies, introduced higher computational cost without significant accuracy improvement, consistent with prior studies noting the trade-off between temporal modeling and efficiency. Additionally, noise augmentation improved robustness under moderate environmental distortions, supporting evidence that data augmentation enhances generalization in SER systems. Class-wise results showed better recognition of emotions such as “angry,” “happy,” and “neutral” due to their distinct acoustic patterns, while “sad” and “disgust” were more frequently misclassified because of overlapping prosodic features like low energy and reduced pitch variation. Despite the increased computational overhead from hybrid feature extraction, the system remained suitable for ESP32-based edge–cloud deployment with real-time inference latency below 400 ms, comparable to related embedded SER systems, demonstrating that high-performance emotion recognition can be achieved on low-cost embedded platforms when optimized deep learning models, hybrid acoustic features, and efficient system design are effectively integrated.

4.0. Conclusion

This study presented the design and implementation of an AI-assisted Speech Emotion Recognition (SER) system for Human–Machine Interaction (HMI) using an edge–cloud embedded architecture, targeting a lightweight and real-time solution for resource-constrained IoT devices. The system combined an ESP32 microcontroller with an INMP441 MEMS microphone, MAX98357A audio module, and OLED display for speech acquisition, processing, and feedback, while CNN, RNN, and CRNN models were trained using MFCC, LPC, and hybrid MFCC+LPC features from the RAVDESS dataset, with noise augmentation applied to improve robustness. Experimental results showed that the CNN model with hybrid MFCC+LPC features achieved the best performance (92.01% accuracy, 92.11% precision, 92.52% recall, and 92.03% F1-score), while also maintaining real-time inference latency below 400 ms within acceptable memory and power limits, confirming the advantage of hybrid features for improved emotion discrimination. The findings demonstrate that effective SER systems can be deployed on low-cost edge–cloud platforms without requiring high-end hardware, supporting applications in healthcare, assistive robotics, smart environments, and adaptive learning systems. However, limitations exist in cross-corpus generalization and performance in highly variable real-world conditions, suggesting future work on spontaneous speech datasets,

multimodal emotion recognition, TinyML optimization, domain adaptation, and large-scale real-world deployment to enhance robustness, scalability, and efficiency.

5.0 Recommendations

i. Test spontaneous data and cross-corpus data to enhance real-world generalization of data other than acted datasets such as RAVDESS.
ii. Implement on-device inference with TinyML and quantization to reduce reliance on server-side processing.
iii. Enhance the system to multimodal fusion (Speech + facial expressions + physiological signals) to make it more robust.
iv. Reduce the training time and other computational overheads.
v. Test the system on domain-specific applications like healthcare monitoring, intelligent tutoring systems and human-robot collaboration to further support.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work the authors used Chat GPT in order to source for recent literature reviews. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

References

Ganesh, D., Kumar, M. S., & Padmavathi, B. (2024). Implementation of speech processing techniques for human emotion recognition. In *Proceedings of the IEEE International Conference on Computational Intelligence and Knowledge Economy (ICCIKE)*.

Murhe, V., Nagpure, S., & Bihade, V. (2025). IoT-ConvNet+ LAMB: A deep learning-based emotion recognition framework using smart IoT systems. *International Journal of Information Technology*. https://doi.org/10.1007/s41870-025-01234-5

Khalil, R. A., Jones, E., Babar, M. I., Jan, T., & Zafar, M. H. (2019). Speech emotion recognition using deep learning techniques: A review. *IEEE Access, 7*, 117327–117345. https://doi.org/10.1109/ACCESS.2019.2936124

Wani, T. M., Gunawan, T. S., Qadri, S. A. A., & Kartiwi, M. (2021). A comprehensive review of speech emotion recognition systems. *IEEE Access, 9*, 106223–106244. https://doi.org/10.1109/ACCESS.2021.3099471

Nguyen, T., Dang, H. M., & Le, H. T. (2023). Improving speech emotion recognition with data augmentation: A comparative study. *Computer Speech & Language, 77*, 101386. https://doi.org/10.1016/j.csl.2022.101386

Gunawan, T. S., Waqar, D. M., & Morshidi, M. A. (2021). Design of a speech anger recognition system on Arduino Nano 33 BLE Sense. In *Proceedings of the IEEE International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA)* (pp. 1–6). https://doi.org/10.1109/ICSIMA50015.2021.9526413

Zhao, L., Wu, Q., & Liang, X. (2023). Cross-corpus speech emotion recognition via adversarial domain adaptation. *Speech Communication, 142*, 45–58.

Imran, M., & Ullah, S. (2025). Raw waveform-based emotion recognition using deep residual 1D CNNs. *Neural Processing Letters, 57*(1), 145–158.

Lee, C. H., & Tashev, I. (2022). Multi-head attention BiLSTM for contextual emotion detection in voice agents. *IEEE Transactions on Affective Computing, 13*(2), 500–512. https://doi.org/10.1109/TAFFC.2020.3015078

Roy, R., Saha, D., & Mandal, B. (2022). A hierarchical deep learning framework for emotion classification from speech. *Applied Intelligence, 52*(4), 4200–4214.

Ebenezer, O. T., Nsenga, E., & Sah, M. (2025). Speech emotion recognition using hybrid deep learning models and diverse acoustic features. In *Proceedings of the 7th IEEE International Conference on Computing, Communication and Control Technologies (ICCCT)*. https://doi.org/10.1109/ICCCT.2025.100

Okomba, N. S., Esan, A. O., Omodunbi, B. A., Adeyanju, I. A., & Bashir, S. A. (2019). Development of a speech controlled water tap and fan system using linear predictive coefficient for feature extraction. *International Journal of Engineering and Technology*.

Okomba, N. S., Esan, A. O., & Omodunbi, B. (2019). Development of a speech controlled water tap and fan system using linear predictive coefficient for feature extraction. *International Journal of Embedded Systems and Applications*. https://www.academia.edu/download/86420789/16058.pdf

arrange alphabetically

Ebenezer, O. T., Nsenga, E., & Sah, M. (2025). Speech emotion recognition using hybrid deep learning models and diverse acoustic features. In *Proceedings of the 7th IEEE International Conference on Computing, Communication and Control Technologies (ICCCT)*. https://doi.org/10.1109/ICCCT.2025.100

Ganesh, D., Kumar, M. S., & Padmavathi, B. (2024). Implementation of speech processing techniques for human emotion recognition. In *Proceedings of the IEEE International Conference on Computational Intelligence and Knowledge Economy (ICCIKE)*.

Gunawan, T. S., Waqar, D. M., & Morshidi, M. A. (2021). Design of a speech anger recognition system on Arduino Nano 33 BLE Sense. In *Proceedings of the IEEE International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA)* (pp. 1–6). https://doi.org/10.1109/ICSIMA50015.2021.9526413

Imran, M., & Ullah, S. (2025). Raw waveform-based emotion recognition using deep residual 1D CNNs. *Neural Processing Letters, 57*(1), 145–158.

Khalil, R. A., Jones, E., Babar, M. I., Jan, T., & Zafar, M. H. (2019). Speech emotion recognition using deep learning techniques: A review. *IEEE Access, 7*, 117327–117345. https://doi.org/10.1109/ACCESS.2019.2936124

Lee, C. H., & Tashev, I. (2022). Multi-head attention BiLSTM for contextual emotion detection in voice agents. *IEEE Transactions on Affective Computing, 13*(2), 500–512. https://doi.org/10.1109/TAFFC.2020.3015078

Murhe, V., Nagpure, S., & Bihade, V. (2025). IoT-ConvNet+ LAMB: A deep learning-based emotion recognition framework using smart IoT systems. *International Journal of Information Technology*. https://doi.org/10.1007/s41870-025-01234-5

Nguyen, T., Dang, H. M., & Le, H. T. (2023). Improving speech emotion recognition with data augmentation: A comparative study. *Computer Speech & Language, 77*, 101386. https://doi.org/10.1016/j.csl.2022.101386

Okomba, N. S., Esan, A. O., & Omodunbi, B. (2019). Development of a speech controlled water tap and fan system using linear predictive coefficient for feature extraction. *International Journal of Embedded Systems and Applications*. https://www.academia.edu/download/86420789/16058.pdf

Okomba, N. S., Esan, A. O., Omodunbi, B. A., Adeyanju, I. A., & Bashir, S. A. (2019). Development of a speech controlled water tap and fan system using linear predictive coefficient for feature extraction. *International Journal of Engineering and Technology*.

Roy, R., Saha, D., & Mandal, B. (2022). A hierarchical deep learning framework for emotion classification from speech. *Applied Intelligence, 52*(4), 4200–4214.

Wani, T. M., Gunawan, T. S., Qadri, S. A. A., & Kartiwi, M. (2021). A comprehensive review of speech emotion recognition systems. *IEEE Access, 9*, 106223–106244. https://doi.org/10.1109/ACCESS.2021.3099471

Zhao, L., Wu, Q., & Liang, X. (2023). Cross-corpus speech emotion recognition via adversarial domain adaptation. *Speech Communication, 142*, 45–58.