JOSTIR
VOLUME 1 SPECIAL ISSUE | DECEMBER, 2025
JOURNAL OF SCIENCE, TECHNOLOGY AND INNOVATION RESEARCH
SCHOOL OF POSTGRADUATE STUDIES
FEDERAL UNIVERSITY OF TECHNOLOGY, AKURE

Journal of Science, Technology and Innovation Research
Volume 1 Special Issue | December 2025

Spoken Command Recognition System for Hands-Free Door Control

Adetoyi, O. E., and Folami, O. P.

Department of Electrical and Electronic Engineering, University of Ibadan, Ibadan, Nigeria

Correspondence: yemi.ade.oe@gmail.com

ABSTRACT

This paper presents the development of a Spoken Command Recognition System (SCRS) that utilise a lightweight convolutional neural network (CNN) for hands-free door access control. The system interprets voice commands, specifically "open" and "close", to activate a solenoid that locks or unlocks a door. An audio dataset was created by recording 1,000 open and close commands each and 1000 varied environmental noise. The dataset was augmented from 3,000 to 15,000 using a time mask, frequency mask, combined time and frequency mask, and noise addition. Mel-Frequency Cepstral Coefficients (MFCCs) were extracted from the audio samples; 80% of this was used for training and the remaining 20% for validation. The accuracy of the validation set is nearly 100%. The trained model was deployed on an ESP32-S3 microcontroller using TensorFlow Lite, and the system was packaged in a 155 mm × 155 mm × 56 mm air-vented metal case. The deployed system achieved a recognition accuracy of 97.5% with a real-time response time of 3000ms for opening and closing door operations. The developed SCRS offers a low-power, real-time, and robust noise-tolerant solution for voice-activated door operations. It also offers an affordable solution with a cost of ₦71,500.

Keywords: Command Recognition, CNN, Embedded Systems, ESP32-S3, Door Operational Control.

Introduction

Voice command recognition for device control has experienced a convergence of high-accuracy and efficient architectures in recent years, thereby offering innovative ways to enhance human interaction with technological advancements (Pereira *et al*., 2023; Ng *et al*., 2023). Command recognition systems have found application in enabling hands-free and intuitive control of systems across various domains, including home automation, security, assistive technologies for individuals with disabilities, industrial applications, curtailing pandemics, gaming, and many more (Arifin & Sarno, 2018; Soltanian *et al*., 2021; Waqar *et al*., 2021; Ayache *et al*., 2021; López-Espejo *et al*., 2022; Abdulghani *et al*., 2022; Chumuang *et al*., 2024; Li *et al*., 2023; Liang *et al*., 2022; Changchun, 2021). The aim is to provide a seamless and efficient means of interacting naturally and conveniently with technology, rather than relying on physical inputs such as keys, buttons, switches, or remote controls.

In addition to the significant success in image recognition, deep neural networks have also demonstrated outstanding performance in speech recognition (Ayache *et al*., 2021; Waqar *et al*., 2021; Chen *et al*., 2024). However, research is ongoing on running speech command recognition models on resource-constrained devices (Ng *et al*., 2023; Soltanian *et al*., 2021). Such research explores one-dimensional or depth-wise convolutions and aggressive pruning to fit models on microcontrollers (Pereira *et al*., 2023). Quantum computing was

doi.org/10.51459/jostir.2025.1.Special-Issue.067

explored in (Qi & Tejedor, 2022; Yang *et al*., 2023) to reduce the computation cost of DNN, while strategies like parameter sharing, local receptive fields, and sparse connectivity were used to minimise the CNN model complexity in (Chen *et al*., 2024).

Noise robustness and real-world testing are also growing research areas in command recognition. Ng *et al*. (2023) designed a multi-channel keyword spotting framework that is robust in noisy and far-field environments while maintaining a small footprint for on-device applications. Another technique involves adding noise to the training datasets to enhance the accuracy of the models in a noisy environment (Pereira *et al*., 2023).

However, no single existing model provides high accuracy, low latency, robustness against noise, and easy customisation. A hands-free door-control system could fill these gaps by integrating an efficient CNN architecture with robust training on multi-condition augmentation and few-shot keyword learning. This would build on recent advances by addressing the identified architectural and capability limitations.

Materials and Methods

The development of the system involves interfacing the ESP32-S3 microcontroller with other hardware components to enable command recognition for door control operations. The block diagram of the hardware components required to achieve real-time voice command recognition and door control execution is shown in Figure 1.

Hardware Design

The core of the system's architecture is the ESP32-S3 microcontroller, selected for its dual-core processor and support for machine learning inference using TensorFlow Lite. This device serves as the primary computing unit, handling voice data acquisition, model inference, and control of the connected hardware elements in Figure 1. The schematic diagram of Figure 2 shows the interconnection of the system's components.

The INMP441 omnidirectional microphone provides a speech input signal to the controller. This microphone supports the Inter-IC Sound protocol (I2S), enabling high-quality, low-latency data transfer from the microphone to the ESP32-S3. Voice commands

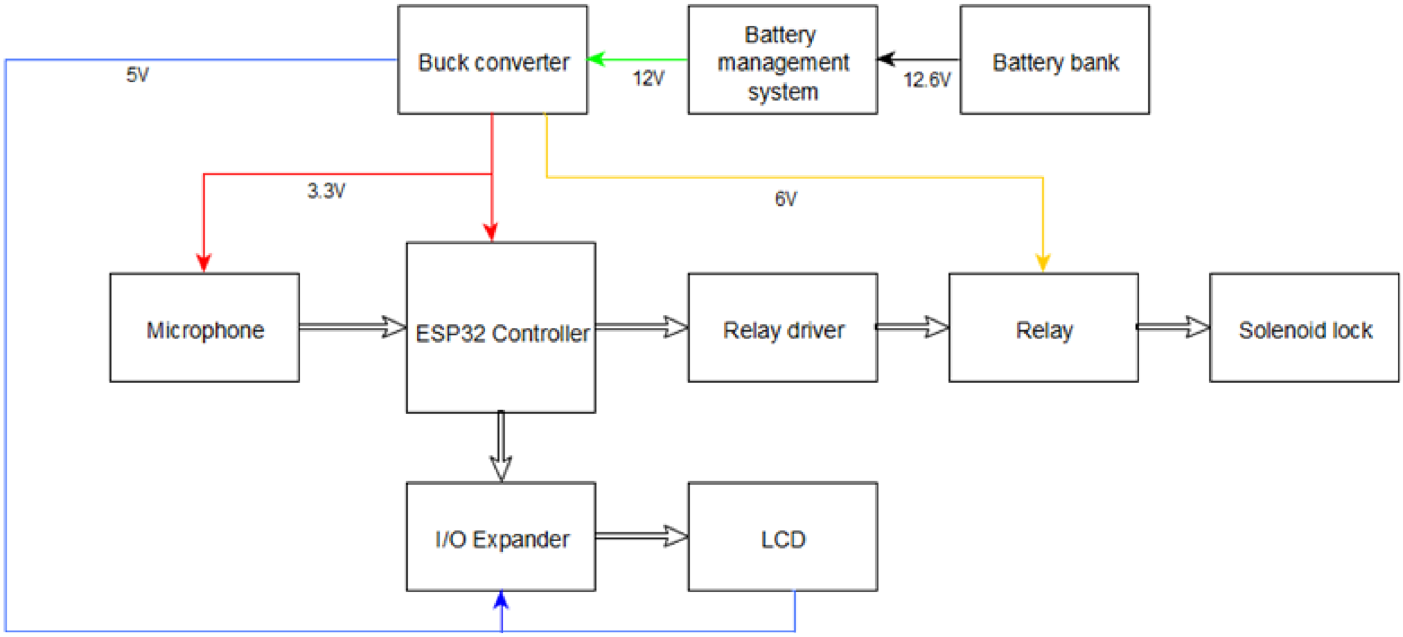

Figure 1. Block diagram of the voice-controlled door system

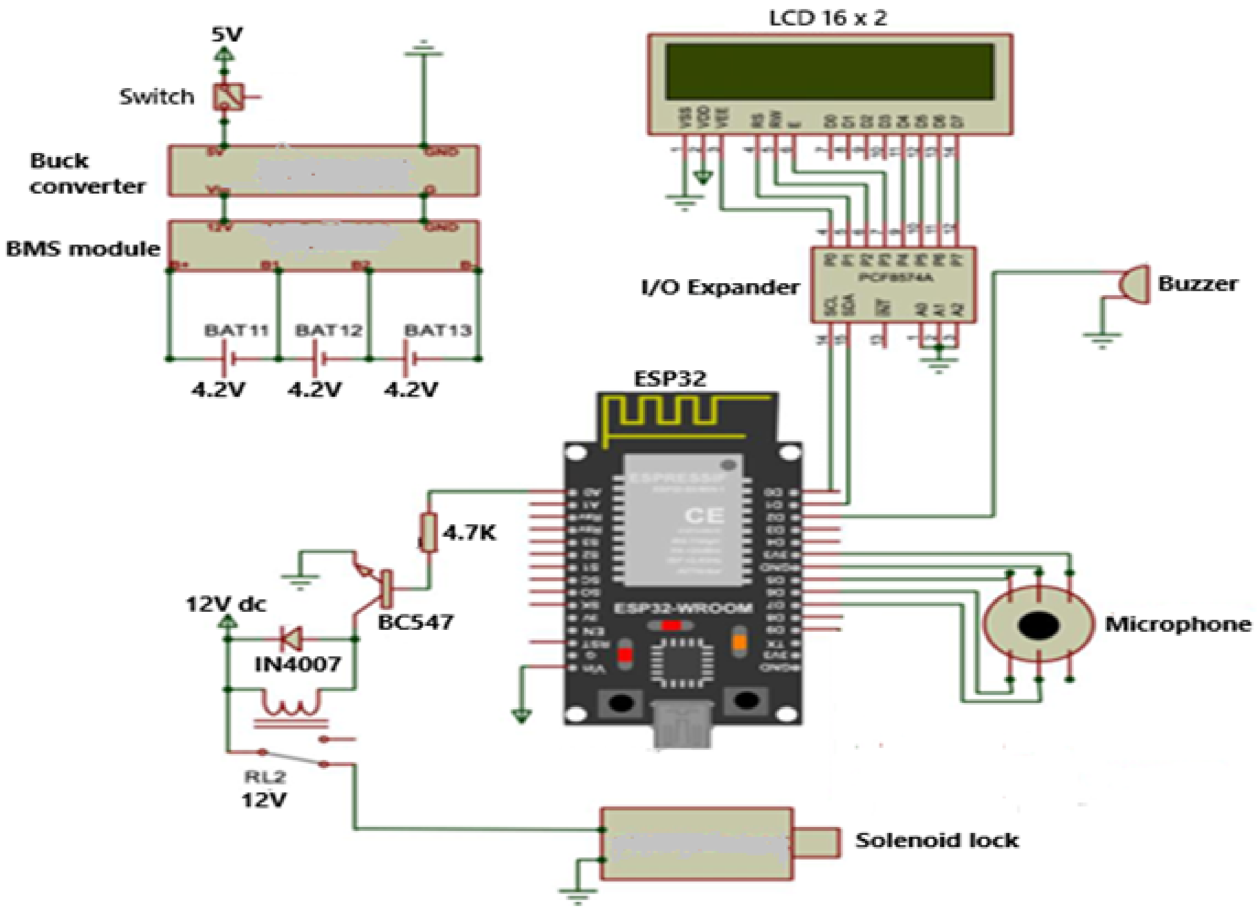

Figure 2. Schematic diagram showing the interconnection of the system

spoken by users, which are limited to "open" and "close", are picked up by this microphone and sent to the microcontroller for processing.

To facilitate user interaction and system feedback, an LM016L Liquid Crystal Display (LCD) was integrated. It communicates with the microcontroller via two-wire serial communication protocol (I2C) and displays system status messages, including initialisation, command recognition, and access confirmation. This makes the system intuitive and informative, especially in scenarios where audio confirmation may be inadequate or impractical. It was connected to the microcontroller via a PCF8574A integrated circuit, which is an 8-bit input/output (I/O) expander.

The locking mechanism was implemented using a 6V DC solenoid lock, controlled through a 5A relay module. The relay acts as an intermediary switch that boosts generated transistor current to enable the triggering of the solenoid in accordance with the model's inference results by the ESP32-S3. When a recognised command such as "open" is detected, the relay is activated to unlock the door, and vice versa for the "close" command.

Power for the entire system is supplied by a 12.6V lithium-ion battery pack, which is configured with a Battery Management System (BMS). The voltage is regulated using an adjustable DC-DC buck converter to match the voltage requirements of the ESP32-S3 and the connected peripherals. This power setup ensures portability and independence from fixed power sources, making the system suitable for embedded and real-world deployments.

Audio data acquisition

A high-performance, omnidirectional Micro Electro-Mechanical Systems (MEMS) microphone with a digital I2S interface was used for recording the audio samples. The microphone can support a speech range of about 60Hz - 15KHz. Fifteen samples of each command were acquired from 40 speakers, and 200 samples of each command from 2 speakers, making a total of 1,000 samples per command. Additionally, 1000 environmental noises were acquired. Each audio file was augmented in turn with an environmental noise, a time mask, a frequency mask, and a combined mask, thereby increasing the dataset to 15,000 and enhancing the model's ability to distinguish noise from spoken commands in real-life scenarios. The unaugmented dataset can be accessed at https://drive.google.com/drive/folders/18LEQgPOPh5mymAoLce4_fUh5nB_tZKWs?usp=sharing

Audio conversion to Mel-Frequency Cepstral Coefficients (MFCCs)

Each 1D speech wave of approximately 1s length was split into 20ms frames with a 10ms stride. Thus, the number of frames in each audio clip was obtained as:

$$\text{Number of frames} = \left(\frac{clip\ length - window\ size}{window\ step}\right) + 1 \qquad (1)$$

where *clip length = 1s, window size* = 0.02s, *window step* = 0.02s. A Hamming window was used to reduce the leakage caused by the overlapped frames. The Fast Fourier Transform (FFT) of each frame was performed at a sampling rate of 16 kHz, and the power spectrum was obtained by calculating the squared magnitude of the FFT. The power spectrum is then filtered using 32 triangular filter banks spaced equally on the Mel scale. A Discrete Cosine Transform (DCT) was finally applied to the logarithm of the filter-bank output energies to obtain the Mel Frequency Cepstral Coefficients. Since the model is intended for a resource-constrained device, only 13 coefficients, which have sufficient audio information for the classification, are retained in each window. The number of MFCC features extracted from each audio clip is then given by:

$$Number\ of\ features = Number\ of\ frames \times Number\ of\ coefficients \qquad (2)$$

Hence, each audio clip was split into 50 frames, and each frame contributes 13 features, resulting in a total of 650 extracted MFCC features per clip.

Convolutional Neural Network (CNN) Architecture

Convolutional Neural Networks are deep-learning architectures that use stacked linear and nonlinear transformations to learn from hierarchical feature representations of input data. CNNs are quite effective for data containing spatial or temporal correlations, such as images, audio signals, and sensor readings. The extracted time-frequency audio features were applied to the CNN architecture in Figure 3, which has a reshape layer to ensure compatibility with the input layer. A 1D convolution layer applies a filter to extract local time-frequency patterns and a RELU activation function to learn complex feature patterns. A maximum pooling layer is then applied to reduce the feature map dimension. The overfitting of the model was prevented by setting one out of four inputs to zero. Further feature extraction was performed by the second 1D convolution and pooling layer, and an additional dropout layer was added to prevent overfitting. A flattening layer was then added to reorder the pooled feature map into a single column that is made available at the output layer. A dense layer was added to map the extracted features to the output categories. The output layer used a softmax activation function to assign probabilities to each category (e.g., "open," "close," and "noise"). The number of neurons in the output layer corresponded to the three categories.

Inference Model Development and Deployment

The CNN was trained with MFCC features extracted from 80% of the preprocessed 2,070 audio data. The MFCC of the remaining 20% data was used to evaluate the developed model's performance in classifying audio samples into one of the predefined categories (open, close, noise). After training and evaluation, the model was built using the TensorFlow Lite for Microcontrollers framework, which provides tools for designing lightweight and efficient neural networks suitable for deployment on memory-constrained devices such as ESP32-S3. The entire system was packaged in a 155 mm × 155 mm × 56 mm (L × W × H) metal enclosure having circular air vents. The command recognition process for the deployed model is depicted in Figure 4.

Results and Discussion

The effects of data augmentation on a sample audio file are shown in Figure 5. The raw audio is shown in Figure 5a, and the noise-augmented version is shown in Figure 5b, which exhibits haziness. Figure 5c shows the MFCC spectrum of the raw audio, while Figures 5d, 5e, and 5f show the effect of the combined time and frequency mask, time mask, and frequency mask on the raw audio, respectively. It can be seen that time masking hides a continuous block of time frames in the spectrogram, while frequency masking hides a continuous block of frequency bins in the spectrogram. The confusion matrix in Figure 6 shows that 3 open commands out of 1025 were misclassified as noise, 1 close command out of 1004 was misclassified as open, and 1 noise each out of 981 were misclassified as open and close, respectively. This indicates that the CNN model achieves nearly 100% recognition accuracy for open, close, and noise audio inputs. The model's high recognition accuracy is due to the availability of large training data, made possible through data augmentation techniques. Figure 7 shows the packaged view of the developed command-controlled door system. When the power

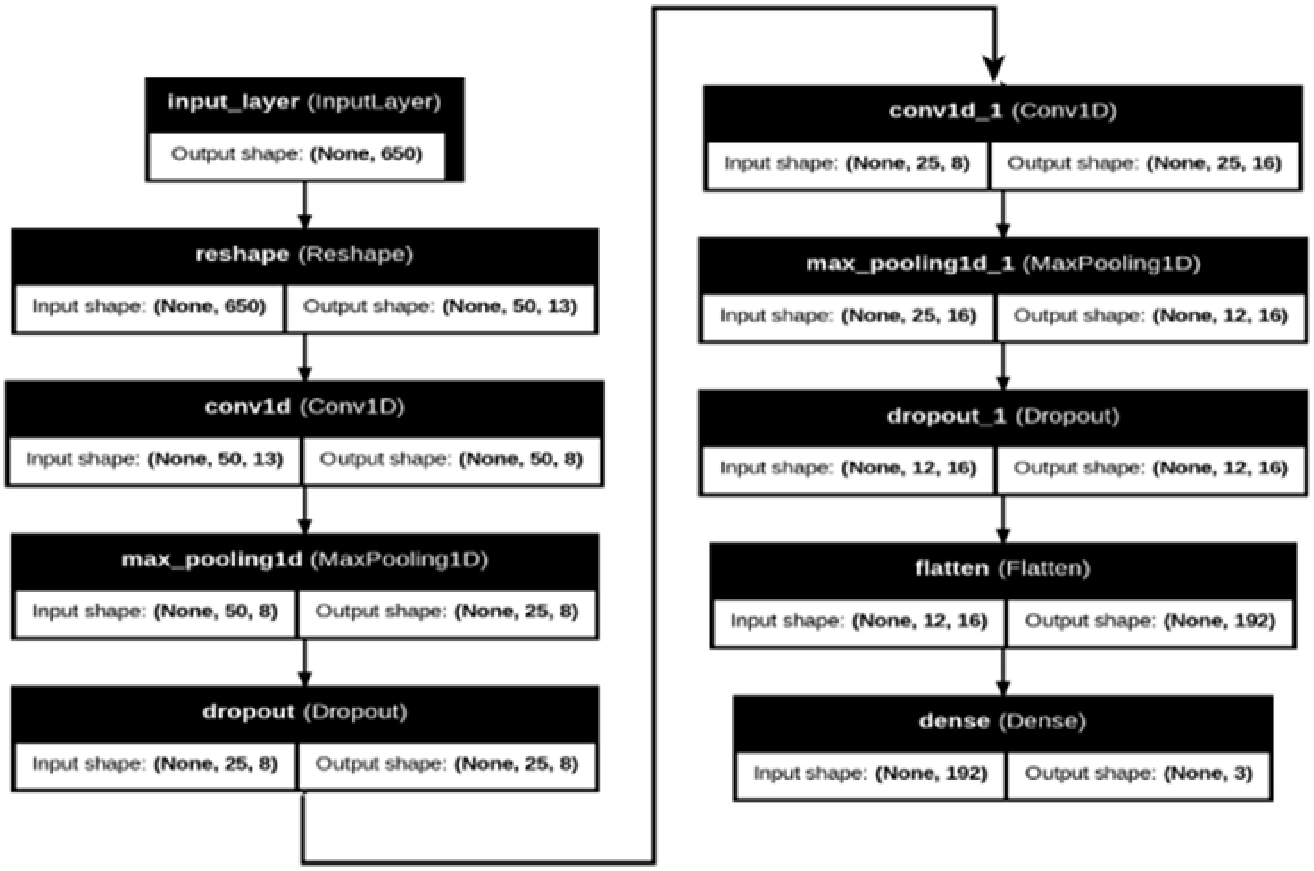
input_layer (InputLayer)
Output shape: (None, 650)
reshape (Reshape)
Input shape: (None, 650)
Output shape: (None, 50, 13)
conv1d (Conv1D)
Input shape: (None, 50, 13)
Output shape: (None, 50, 8)
max_pooling1d (MaxPooling1D)
Input shape: (None, 50, 8)
Output shape: (None, 25, 8)
dropout (Dropout)
Input shape: (None, 25, 8)
Output shape: (None, 25, 8)
conv1d_1 (Conv1D)
Input shape: (None, 25, 8)
Output shape: (None, 25, 16)
max_pooling1d_1 (MaxPooling1D)
Input shape: (None, 25, 16)
Output shape: (None, 12, 16)
dropout_1 (Dropout)
Input shape: (None, 12, 16)
Output shape: (None, 12, 16)
flatten (Flatten)
Input shape: (None, 12, 16)
Output shape: (None, 192)
dense (Dense)
Input shape: (None, 192)
Output shape: (None, 3)

Figure 3. Architecture of the CNN model

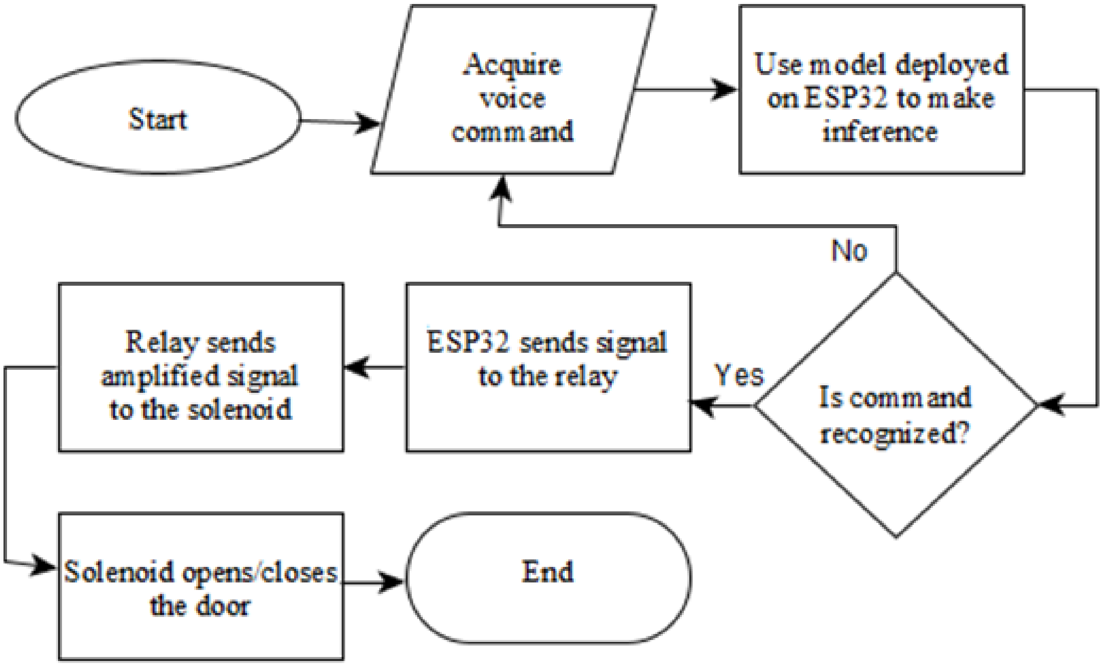
Start
Acquire voice command
Use model deployed on ESP32 to make inference
Is command recognized?
No
Yes
ESP32 sends signal to the relay
Relay sends amplified signal to the solenoid
Solenoid opens/closes the door
End

Figure 4. Flowchart of the model's command inference process

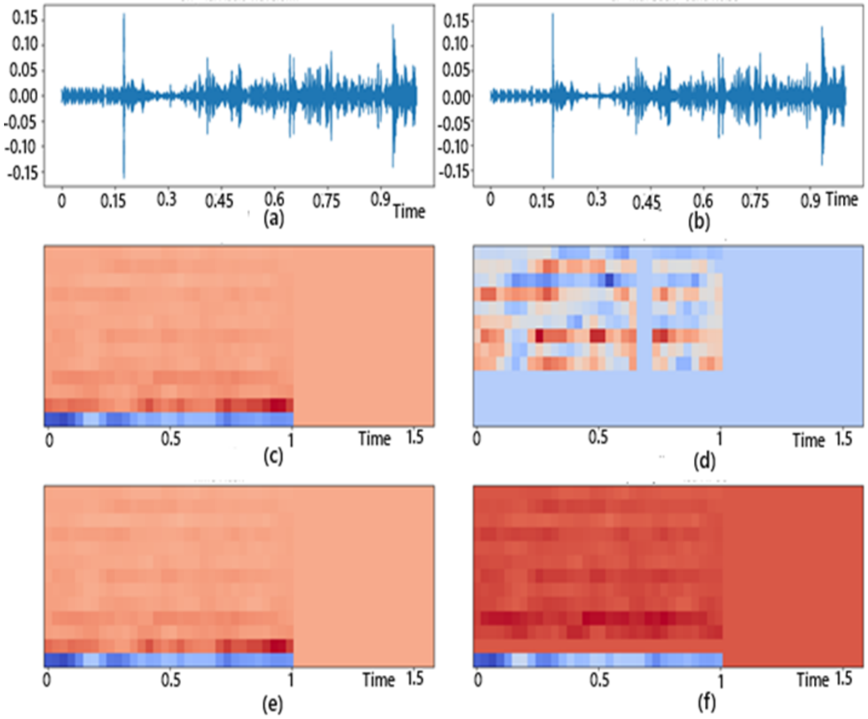

Figure 5. (a) Sample audio waveform (b) Effect of added noise on the sample audio waveform (c) MFCC of the sample audio waveform (d) Effect of combined time and frequency mask on the sample audio MFCC (e) Effect of time mask on the sample audio MFCC (f) Effect of frequency mask on the sample audio MFCC

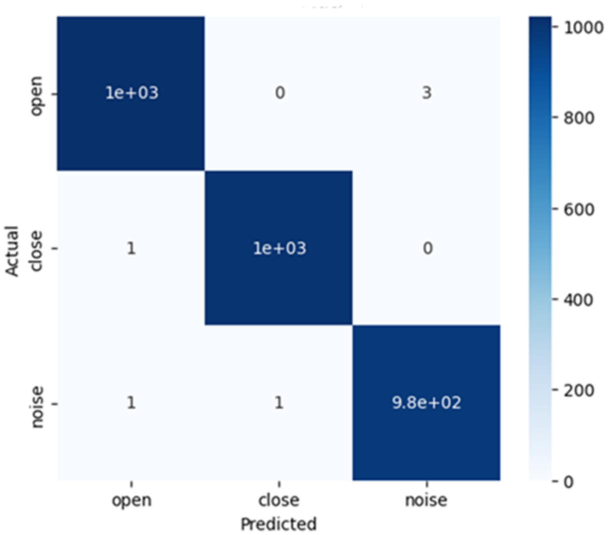

Figure 6. Confusion matrix of the CNN model inference.

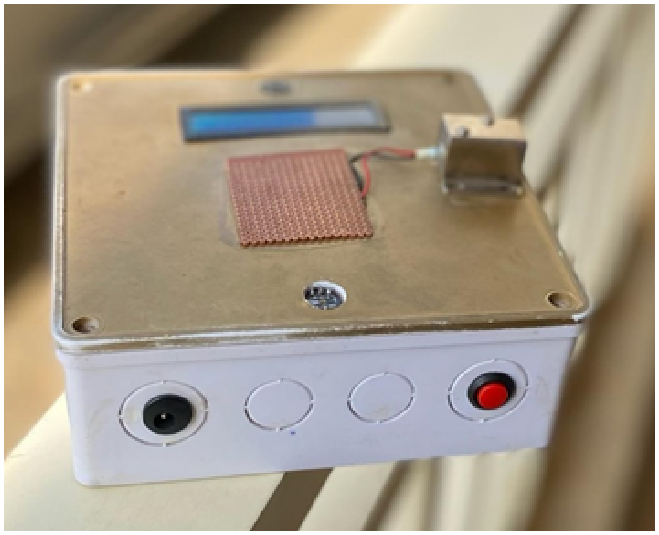

Figure 7: Package view of the developed voice-controlled door system.

button of the device is turned on, the LCD shows “Voice Control Initialising”, which is followed by “Ready!”. When the "open" command is recognised, the system confirms the unlocked door operation by displaying **"Door is OPENED"** on the LCD. Similarly, when the "close" command is recognised, the LCD will display **"Door is CLOSED"** to confirm that the system has locked the door. In case of an unrecognised command, the buzzer sounds and the LCD shows “**Unrecognised command, speak again**”.

Conclusion and Recommendations

The developed CNN model processes real-time audio, classifies it into one of the predefined categories (open, close, noise), and triggers the appropriate hardware response to control the door. This integrated setup of embedded hardware, signal processing, and machine learning allows the system to offer reliable door control based on voice commands. The device can be used in automated home environments, assisted living facilities for the elderly and persons with disabilities, and public buildings, especially during pandemic situations. The recognition of the model can be improved by introducing background noise in the audio clips of the commands and increasing the noise data to have a balanced dataset.

References

Abdulghani, M. M., Walters, W. L., & Abed, K. H. (2022). Autonomous Voice Recognition Wheelchair Control System. *2022 International Conference on Computational Science and Computational Intelligence (CSCI)*, 226–230. https://doi.org/10.1109/CSCI58124.2022.00043

Arifin, R. D. H., & Sarno, R. (2018). Door automation system based on speech command and PIN using Android smartphone. *2018 International Conference on Information and Communications Technology (ICOIACT)*, 667–672. https://doi.org/10.1109/ICOIACT.2018.8350715

Ayache, M., Kanaan, H., Kassir, K., & Kassir, Y. (2021). Speech Command Recognition Using Deep Learning. *International Conference on Advances in Biomedical Engineering, ICABME*, *2021-Octob*(September), 24–29. https://doi.org/10.1109/ICABME53305.2021.9604862

Changchun, Y. (2021). Design of smart home control system based on wireless voice sensor. *Journal of Sensors*, *1*, 1–11. https://doi.org/10.1155/2021/8254478

Chen, J., Teo, T. H., Kok, C. L., & Koh, Y. Y. (2024). A Novel Single-Word Speech Recognition on Embedded Systems Using a Convolution Neuron Network with Improved Out-of-

Distribution Detection. *Electronics (Switzerland)*, *13*(3), 1–16. https://doi.org/10.3390/electronics13030530

Chumuang, N., Ganokratanaa, T., Pramkeaw, P., Ketcham, M., Suvil, C., & Yimyam, W. (2024). Voice-activated assistance for the elderly: Integrating speech recognition and iot. *2024 IEEE International Conference on Consumer Electronics (ICCE)*, 1–4. https://doi.org/10.1109/ICCE59016.2024.10444265

Li, S.-A., Liu, Y.-Y., Chen, Y.-C., Feng, H.-M., Shen, P.-K., & Wu, Y.-C. (2023). Voice Interaction Recognition Design in Real-Life Scenario Mobile Robot Applications. *Applied Sciences*, *13*(5), 3359. https://doi.org/10.3390/app13053359

Liang, X., Batsis, J. A., Zhu, Y., Driesse, T. M., Roth, R. M., Kotz, D., & MacWhinney, B. (2022). Evaluating voice-assistant commands for dementia detection. *Computer Speech & Language*, *72*, 101297. https://doi.org/10.1016/j.csl.2021.101297.

López-Espejo, I., Tan, Z.-H., Hansen, J. H. L., & Jensen, J. (2022). Deep Spoken Keyword Spotting: An Overview. *IEEE Access*, *10*, 4169–4199. https://doi.org/10.1109/ACCESS.2021.3139508

Ng, D., Xiao, Y., Yip, J. Q., Yang, Z., Tian, B., Fu, Q., Chng, E. S., & Ma, B. (2023). Small Footprint Multi-channel Network for Keyword Spotting with Centroid Based Awareness. *INTERSPEECH 2023*, 296–300. https://doi.org/10.21437/Interspeech.2023-1210

Pereira, P. H., Beccaro, W., & Ramirez, M. A. (2023). Evaluating Robustness to Noise and Compression of Deep Neural Networks for Keyword Spotting. *IEEE Access*, *11*, 53224–53236. https://doi.org/10.1109/ACCESS.2023.3280477

Qi, J., & Tejedor, J. (2022). Classical-To-Quantum Transfer Learning for Spoken Command Recognition Based on Quantum Neural Networks. *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 8627–8631. https://doi.org/10.1109/ICASSP43922.2022.9747636

Soltanian, M., Malik, J., Raitoharju, J., Iosifidis, A., Kiranyaz, S., & Gabbouj, M. (2021). Speech Command Recognition in Computationally Constrained Environments with a Quadratic Self-Organized Operational Layer. *2021 International Joint Conference on Neural Networks (IJCNN)*, 1–6. https://doi.org/10.1109/IJCNN52387.2021.9534232

Waqar, D. M., Gunawan, T. S., Kartiwi, M., & Ahmad, R. (2021). Real-Time Voice-Controlled Game Interaction using Convolutional Neural Networks. In *2021 IEEE 7th International Conference on Smart Instrumentation, Measurement and Applications, ICSIMA 2021* (pp. 76–81). https://doi.org/10.1109/ICSIMA50015.2021.9526318

Yang, C.-H. H., Li, B., Zhang, Y., Chen, N., Sainath, T. N., & Siniscalchi, S. M. (2023). A Quantum Kernel Learning Approach to Acoustic Modeling for Spoken Command Recognition. *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1–5. https://doi.org/10.1109/ICASSP49357.2023.10095142.