Journal of Robotics and Control (JRC)
Volume 4, Issue 4, 2023
ISSN: 2715-5072, DOI: 10.18196/jrc.v4i4.15918

Development of Speech Command Control Based TinyML System for Post-Stroke Dysarthria Therapy Device

Bambang Riyanta [1*], Henry Ardian Irianta [2], Berli Paripurna Kamiel [3]
[1, 2, 3] Department of Mechanical Engineering, Universitas Muhammadiyah Yogyakarta, Yogyakarta, Indonesia
Email: [1] bambangriyanta@umy.ac.id, [2] henry.ardian.2015@ft.umy.ac.id, [3] berlikamiel@umy.ac.id
*Corresponding Author

***Abstract*—Post-stroke dysarthria (PSD) is a widespread outcome of a stroke. To help in the objective evaluation of dysarthria, the development of pathological voice recognition and technology has a lot of attention. Soft robotics therapy devices have been received as an alternative rehabilitation and hand grasp assistance for improving activity daily living (ADL). Despite the significant progress in this field, most soft robotic therapy devices use a complex, bulky, lack of pathological voice recognition model, large computational power, and stationary controller. This study aims to develop a portable wirelessly multi-controller with a simulated dysarthric vowel speech in Bahasa Indonesia and non-dysarthric micro speech recognition, using tiny machine learning (TinyMl) system for hardware efficiency. The speech interface using INMP441, compute with a lightweight Deep Convolutional Neural network (DCNN) design and embedded into ESP-32. Feature model using Short Time Fourier Transform (STFT) and fed into CNN. This method has proven useful in micro-speech recognition with low computational power in both speech scenarios with a level of accuracy above 90%. Realtime inference performance on ESP-32 using hand prosthetics, with 3-level household noise intensity respectively 24db,42db, and 62db, and has respectively resulted from 95%, 85%, and 50% Accuracy. Wireless connectivity success rate with both controllers is around 0.2 - 0.5 ms.**

Keywords—Post Stroke Dysarthria; Dysarthic Speech Recognition; SCR; ASR; Micro Speech; KWS; TinyML; Edge Controller Devices; STFT-CNN.

I. INTRODUCTION

Strokes are a type of illness where parts of the brain are damaged from blood clots, resulting in various after-effects depending on the affected area and, in many cases, leading to death [1, 2, 3]. Research has revealed that strokes impact approximately 16 million individuals worldwide each year, and about 57% to 69% of stroke patients experience speech disorders, including dysarthria and aphasia or both [4]. Post-stroke dysarthria (PSD) is a common and persistent aftermath of a stroke, affecting most of all acute stroke cases [5, 6, 7, 8], but this group has received limited research attention, especially particularly pronounced in Indonesian speakers. Non- invasive brain stimulation (NIBS), Lee Silverman voice treatment (LSVT) and other traditional dysarthria assessments primarily depend on invasive testing with different speech impairment treatments due to subject severity classes, leading to significant variations clinical improvement and efficacy [9-13]. To provide an objective basis for diagnosis, treatment, and assistive, pathological voice recognition technology has emerged as a helpful device in distinguishing transdisciplinary program for dysarthtria treatment. This has led to increased interest and attention from both society and the industry for clinical rehabilitation, with the aid of integrated robotic devices like *Amadeo*^{TM}, *HandMentor, HomeRehab, ReJoyce, Robotherapist 2d,* and others [14, 15, 16, 17]. Enriching patients's motivation to do their basic needs for activity daily living (ADL), and increasing level of independency for the sufferer.

A researcher developed an integrated robotic control-based rehabilitation that makes it easier for medical personnel to carry out renewable methods that accompany clinical therapy sessions to be able to do it independently, and ergonomically [18, 19]. Exo-glove poly device development became popular among researchers to strive for ergonomics, safety, portability, and reliability for users. The choosen of materials [20, 21, 22, 23] with a different mechanical approach, and a variety of control also have a significant change for a user to get the best user experience.

The underactuated motor tendon-driven mechanism concept uses a servo motor system, with fewer cables transmission. Control and navigation system that switches from *pneunets* or combustion driver actuation that is heavy and bulky, to an underactuated servo control mechanism [22-30] that has a lot of room for portability. Due to come with a novel for enhanced control experience, this study develops a more comfortable way to control such therapy devices, like Exo gloves with an *edge* or *TinyML* system, using a simulated dysarthric speech interface with keyword spotting system (KWS) for a better and more reliable user experience.

Researchers use deep neural networks (DNN) to conduct further analysis of pathological voices, due to its exceptional capability in learning features for identifying or operating a device [31-37]. However, most existing method have limitations in detecting pathologhical voice with hardware efficiency. This is because they rely on classical features such as Mel-Frequency cepstrum coefficient (MFCC), which have limitations in fully capturing the deep characteristics of pathological sounds, and require a significant amount of computational power and RAM to run.

David et al. [31] Research related to ASR in dysarthria with KWS uses the CNN model, developing a software ecosystem for full recognition of dysarthric speech for certain needs of patient assistance running using personal computer. The results of the implementation accuracy in modeling on average for all target classes are 86%.

Y.-Y. Lin et al. [32] compared speech feature models for dysarthric speech preprocessing and proposed a Convolutional Neural Network with Phonetic Posteriorgram (CNN-PPG) model. They also compared this model with a CNN model using Mel-frequency Cepstral Coefficients (CNN-MFCC) and an ASR-based system. The experimental results showed that the CNN-PPG system had 93.49% accuracy, which was better than both the CNN-MFCC (65.67%) and the ASR-based systems (89.59%). Furthermore, the CNN-PPG model had a smaller size, consisting of only 54%.

Yakoub et al. [33], developed dysarthric speech recognition using empirical mode, EMDH-CNN approach, combining empirical mode decomposition and Hurst-based mode selection (EMDH) with a CNN for enhancing dysarthric speech recognition. The EMDH technique acts as a preprocessing step to improve speech quality, followed by feature extraction using MFCC from the processed speech, and overall consisting around 62.7% accuracy.

Vachhani et al. [34] conducted a study on the use of data augmentation through temporal and speed modifications on healthy speech to simulate dysarthric speech. They evaluated the results with both DNN-HMM based Automatic Speech Recognition (ASR) and Random Forest classification. The synthetic dysarthric speech was classified for its severity level using a Random Forest classifier trained on real dysarthric speech samples. The results showed an improvement of 4.24% and 2% in the ASR performance when using tempo-based and speed-based data augmentation respectively, compared to using healthy speech alone for training.

Shamiri et al. [35] focused on presenting the Speech Vision (SV), which addresses by recognizing word shapes using visual-feature representations. Additionally, SV deals with limited dysarthric speech samples by applying data augmentation, generating synthetic speech, and utilizing transfer learning. Evaluations showed SV achieving average Accuracy rate of 61.11% and 64.71%.

Joshy et al. [36] discover present a comparative study on the classification of dysarthria severity levels using a pre-trained DeepSpeech-1 ASR model. Trained with 1000 hours of data, achieved comparable results to the simple MFCC-DNN with an unbalanced small dataset and achieved an average accuracy of 70.52%.

Jiang et al. [37] present a paper that suggests a hybrid recognition model that combines 1DCNN and Double-LSTM (DLSTM) networks based on MFCC features of pathological voices. To build the model, a database of syllable pronunciations was created using Mandarin-speaking participants including normal adults and patients with PSD. The 1DCNN network was then used to process the MFCC features of the syllable pronunciations and extract deeper hidden features. The results of the experiments showed that the further processing of MFCC features by the 1DCNN network significantly improved the performance of the DLSTM-based recognition model, with an accuracy of 82.1% at the syllable level and 97.4% at the speaker level.

Another variety of interfacing pursued in supporting servo control for therapy devices, such as in studies that focus on bio-electric sensors using electromyography (EMG), electroencephalography (EEG), electrocardiography (ECG), and phyisichal sensory eg, Flex sensor, leap motion, Intertial measurement unit sensory (IMU), even multisensory approach and inference models on hand rehabilitation devices [24, 25, 38, 39]. Thus sensor will have a high level of accuracy when using a multichannel design with a sensitive, relatively expensive, and larger acquisition device and hig computational power. Hence, the speech command recognition (SCR) approach with lightweight CNN is used in this study for ergonomic control and produce a smaller resolution by applying the micro speech method or KWS model [40-47], which uses STFT to generate spectrogram images for CNN modeling features.

This research contribution is to present detailed process development of dysarthic and non dysarthic SCR based using TinyML system, and proposed KWS algorithm model with limited dataset, and low stake perform of augmentation using ambient noise and normalise voice data. Trained model converted and inference for low computational power in to microcrontroller, then wirelessly navigate underactuated servo motor for ADL activity on theraphy device which simulate using a hand prosthetic. Also, pre build for Bahasa corpus for Indonesia dysasrthic speech dataset. This paper will contain the following sections in a sequence:

- Briefly proposed prototype method for designing dysarthric and non-dysarthic SCR and TinyML based controller devices includes limited dataset collection, preprocessing, data augmentation, CNN Conv2D modeling, rendering, testing evaluation, inference process and interpreter with ESP32 microcontroller based on KWS model.
- Proposed unicast wireless protocol using ESP NOW, multi-task module using FreeRTOS, interfacing using INMP441 and momentary button, and electrical design.
- Results and discussion show the spectrogram feuture analysis from audio data feuture extraction, wireless and proposed prototype response time, and Range of motion (ROM) for ADL practical experiment result using hand prosthetic.
- Finally, the conclusion section concludes the paper and future work advice for the next research.

II. PROPOSED METHOD

This study's proposed Multi control design uses voice and button control simultaneously. Fig. 1 indicates the block diagram of this study. The proposed voice control design includes limited dataset collection, preprocessing, lightweight CNN modeling, and rendering with a local

machine, then inference with ESP32 which relatively low cost and powerful to use for TinyMl embedded [48-55] and single-channel INMP441 as microphone modules. The modeling will be generated and converted for tiny machine learning (TinyMl) requirements on the ESP32 microcontroller using Jupyternotebook and the python language, along with libraries and modules.

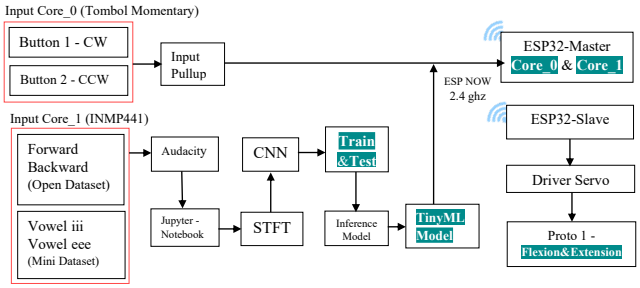


Fig. 1. The proposed dysarthia speech command TinyML block diagram

The button control design uses two buttons using internal pullups on the ESP32, which is configured through C/C++ programming, with a normally closed button and loop system to control servo per increment continuously. The inferencing model and final firmware development with the ESP32 microcontroller will also be compiled using the C/C++ programming language. The compiled firmware were develop using transfer learning method from open framework, and pre trained model by microspeech firmware from TensorFlow [42].

III. TINYML CONTROLLER DESIGN

In this study, feautured engineering for neural net has been employed. Feautured data using multiple sources of datasets, and preprocessing by transforming voice signals in the time domain into spectrogram images as in this paper [56-61]. Deep learning using lightweight architecture CNN-STFT with 2 convolution layer, and adam optimizer [62-70]. Conversion and quantization of neural net model or inferencing the model for ESP32 using tf.lite converter, also designing button controller, wireless and multi-task protocol also describe in this section.

A. Preparation of the Dataset

This study SCR approach to dysarthric speech as in Fig. 2 which refers to Ref. [16, 17], used Indonesian vocal vowel sound recordings, namely "eee" and "iii" on healthy subjects mimicking a dysarthric way of speech by two healthy subjects. Selection of vowels "eee" and "iii" to reduce false negatives and false positives in models with vowel phonemes that have a probability close to the production of other phonemic sounds or the resonant characteristics of voice, such as "eee" and "aaa" or "eee " and "ooo" based on linguistic research based on the Vowel Space Area (VSA) conducted by previous researcher [71]. The recording was performed with parameters based on speech_command v2 datasets [72], as shown in Table I. Recordings vowels dataset were taken 100 times each, with a total of 500 recordings for dysarthric vowel mimicking speech "aaa"," iii,", "uuu", "eee" and "ooo". The audio format used was *WAV, 16-bit, mono, and a sampling rate of 16 kHz, for 1s, at a noise around 35db, and recorded using a smartphone.

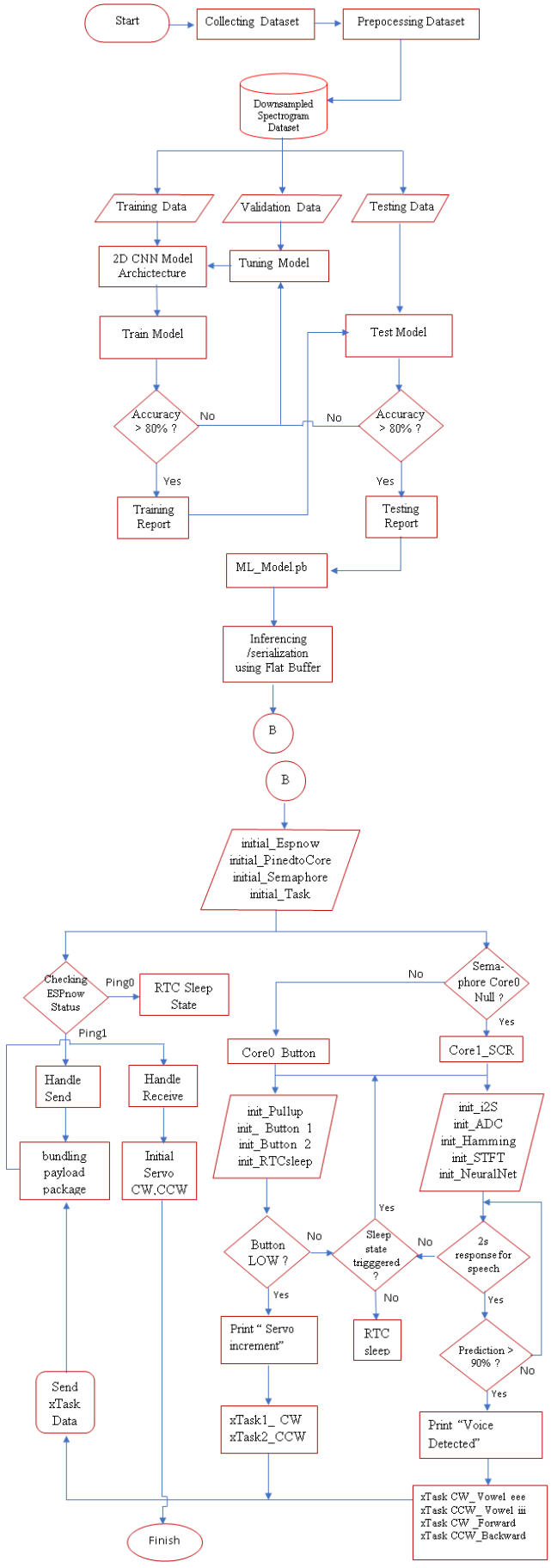


Fig. 2. Flowchart SCR based TinyML proposed system

The final dataset was grouped into one folder with a ratio of 80:10:10% for training, validation, and testing, which will be extracted into a spectrogram as an image transformation from sound waves for CNN modeling features. The amount of data is 200 recordings of vowel speech, namely "eee" and "iii" from TORGO dataset and single subject with mimicking dysarthic speech, two types of speech from *speech_command v2* with a total, 3221 recordings, namely "forward and backward", six types of background noise in addition to data augmentation, and 33 other types of speech with a total of 102,608 recordings from folders *speech_command v2* becomes invalid class. The total dataset used for model training is 106,035 recorded data with 5 target classes, namely, vocal "eee", "iii", "forward", "backward" and invalid, for overall information from the dataset variables shown in Table II.

TABLE I. COMMAND WORDS AND DATASET RATIO

Command	No of Speakers	No of Utterances	Training (80%)	Testing (10%)	Validation (10%)
Forward	390	1,557	1,245	125	125
Backward	416	1,664	1,331	166	166
Vowel iii	2	100	80	10	10
Vowel eee	2	100	80	10	10
Invalid	1,812	102,608	1,245	82,086	82,086

TABLE II. VOICE RECORDING PARAMETERS BASED ON *SPEECH_COMMAND V2* DATASETS

Dataset Variable	Information
Dysarthic Speech	Vowel "iii " and " eee "
Non Dysarthic Speech	'Forward" and "Backward."
Number of total Samples	6000
Duration	1s
Recording location	Indoor room
Sample Rate	16.000 Hz
Bit Depth	16 bit
Channel	1 / Mono
Bitrate	256 Kbit/s
Byterate	32 Kbyte/s
Format	*.wav (PCM)

B. Short Time Forier Transform (STFT)

Short Time Fourier Transform (STFT) is a mathematical transformation that decomposes a function in the time domain into its constituent frequencies [73, 74, 75, 76]. This transformation is widely used, especially in signal processing [77]. Defined in equation (1), the Fourier transforms a function in the time domain f(t) into another function in the frequency domain f(x).

$$f(x) = \int_{-\infty}^{\infty} f(t) e^{-i \times t}\, dt \quad (1)$$

Other transformations related to the *Fourier* are also used to convert from time to frequency domain for both continuous and discrete data [78]. STFT is a discrete *Fourier transform* (DFT) used to determine the sinusoidal frequency in the local part of the signal as the signal changes with time. In other words, STFT is a Fourier transform in a *windowed signal*, as shown in Fig. 3.

STFT provides localized information (concerning time) of the frequency component, in contrast to the standard Fourier transform, which includes frequency information over time intervals [74]. STFT is formulated as the product of the signal with a weight which is called the window x(t), where is the Spectro-temporal index as in equation (2).

$$x(\tau, \omega) = \int_{-\infty}^{\infty} x(t).\omega(n-\tau).e^{-i \times t}\, dt \quad (2)$$

Where — Input signal at time n, w(n) window function used, Xm(ω) — DTFT result of window per sample (mR), R — Hop, on sample window w, can be selected from a wide variety of existing sinusoidal functions. In this study, the Hamming Window as the weight, shown in equation (3).

$$w(n) = 0.54 - 0.46\, cos\left(2\pi \frac{\pi}{N}\right), 0 \leq n \leq N \quad (3)$$

where w = Hamming window, n = time index, N = sampling rate. In its application, converting the sample of sound waves in the time domain into the combined time and frequency domain, using equation (2) to decompose the bin frequency into a spectral quantity (magnitude), represented by 2-dimensional color output. The spectogram parameter approach as a general image is used to build a short speech recognition system using the CNN model.

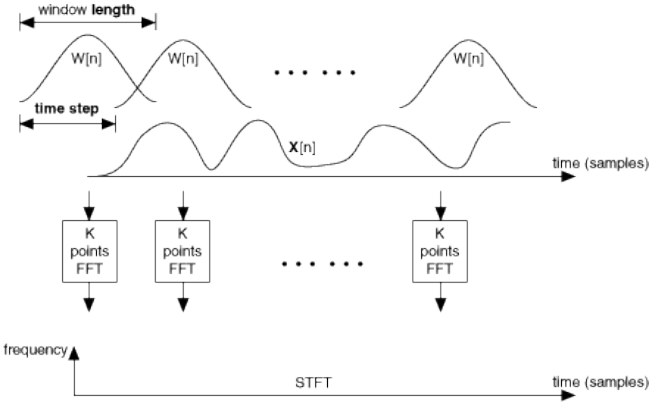


Fig. 3. Time domain voice signal transformation with STFT with hamming window

In this study, computational configuration with STFT was carried out using the NumPy and SciPy library in python, with the function configuration w(n) using a hamming window, 50% overlap in each frame or window size, sampling rate using 16 kHz, a window size of 320 points, amplitude for the duration of the sample per frame blocking is 20ms without overlap, with 50% overlap in each blocking frame value is obtained stride in the amplitude, or the stride size is 160 amplitude points, sampled 10ms for each bin frequency. Each configuration windowed for the internal time of the voice signal for 1 second. Representation bin frequency through computation, to be transformed by the STFT algorithm into a spectral quantity in the form of a spectrogram.

Fig. 4 is the downsampling with logarithmic scale of spectrogram (log-spectrogram) was employed, using average pooling and pixel reduction with a 1x6 kernel was intended to reduce the number of initial pixels of the spectrogram to 99x43 pixels as the primary input to the CNN model. This preprocessing method was conducted from previous research and has significant result to make spectrogram feature convertible for small form factor [79-85].

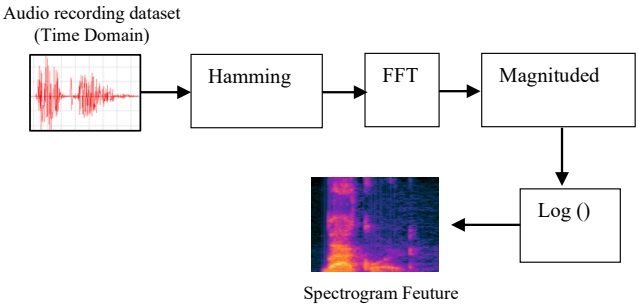


Fig. 4. The proposed STFT with 1x downsampling spectrogram for model features

C. Convolutional Neural Network

Lightweight CNN architecture is one of a deep learning commonly used in image data processing in a small form factor neural net [86-90]. In this research, can be seen in Fig. 5 which is the architecture consisting of convolution, neurons with weights and biases, backpropagation, and activation functions. The CNN method has two stages: the feedforward, and the learning stage using the backpropagation neural network (BPNN). CNN's algorithm is similar to a multilayer perceptron (MLP), but each neuron on CNN is presented in two-dimensional form. Unlike the case with MLP, where each neuron only has one dimension.

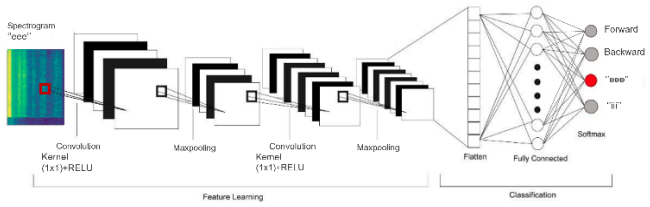


Fig. 5. A novel 2DCNN architectural diagram of processing dysarthic vowel using only 2 convolution layers with downsampling spectrogram feuture

In this study, the architecture model was developed using a *Jupyternotebook* with *Tensorflow* and *Keras* libraries and inspired by a pre-trained model from TensorFlow [42].

Fig. 6 is the flowchart of architecture 2DCNN then uses a sequential method with several layers including 2 convolution layers with 4 filters with a 3x3 kernel, 2 max-pooling for downsampling output shape with pool size 2x2 and ReLU activation, the next layer is a fully connected layer or MLP including, flatten layer, dense layer with several neurons 80-unit nodes and dropout of a neuron weight about 10% or 0.1, activation function using *Softmax*, and then BPNNs layer.

Dataset augmentation and configuration on the MLP uses a kernel tuning to determine and limit the value of the neuron weights in the convulsion process or weight initializer, one of which is the regulation of kernel weights or filters with L2 regulators using a threshold with kernel size 1x1 on the convoulution process in addition to Prevention of overfitting [91].

The spectrogram image in the final dataset or feature for modeling is 99 x 43 pixels. Then the model fits with 10 epochs and is carried out per 30 batches, and the following are the results map with a summary of the algorithm sequence:

- ***Input Feature***, input for CNN architecture is an image spectrogram with a size of [99 x 43] pixel. The input image represents the voice signal taken from the STFT operation with a total duration of 1s and the previously explained dataset parameter.

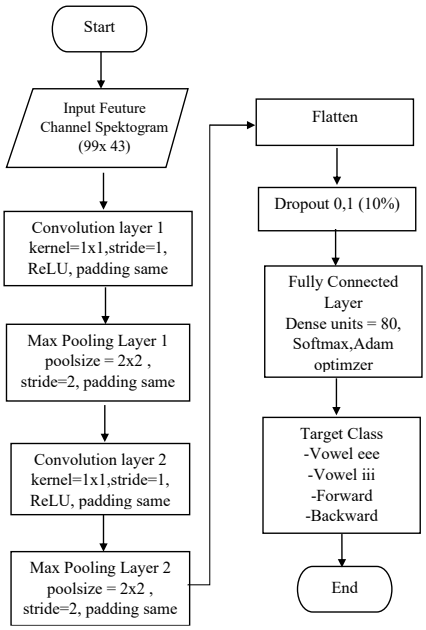


Fig. 6. Proposed flowchart 2DCNN architecture

- ***First Convolution Layer***, in this study, every convolution using a single channel performed in each layer to produce a feature map with a mathematical model (4). In the first Convulsion Layer, the input image will be convoluted by a kernel with dimension [1 x 1], with the number of filters 4, stride [1 x 1], and padding "same" with a value of 0. If the padding is "same" the system will apply the zero padding to the input matrix. The value of zero padding can be determined using equation (5).

$$(I \times K)_{ij} = \sum_{m=0}^{k1-1} \sum_{n=0}^{k2-1} K_{m,n}\, I_{i+m,j+n\,+} b \tag{4}$$

$$(P) = \frac{K-1}{2} \tag{5}$$

Due to the filter size used by the l2 regularization, from the equation, the output shape for convolution on (6) is obtained, and the stride size is [1 x 1]. The output shape of the first convulsion layer is the same as the size input shape of the input image, namely [99 x 43]. The activation function uses the ReLU layer, which aims to change the minus value at the output of the convolution process to zero due to non-linearity.

$$(O) = \frac{W - K + +2P}{S} + 1 \tag{6}$$

- ***First Maxpooling Layer***, the number of input shapes for the first convolution is the output shape for the first max-pooling layer. In this layer, the image will be

downsampled using a pool size of [2 x 2], stride [2 x 2], with padding mode 0. The padding value in this layer can be calculated using equation (1) which can also be found in the first convolution layer. While output size on the max-pooling layer uses the following equation (6), the output of the max-pooling layer based on equations (6) and (7) is [49 x 21].

$$[h\ w] = \frac{w - Ps + (2.P)}{2} + 1 \quad (7)$$

- ***Second convolution*** process is to continue the pooling result by inputting an image matrix of [49 x 21] with the same configuration as the first convolution so that the *output shape* remains [49 x 21].
- ***Second Maxpooling layer***, the number of output shapes on the second convolution is the input shape, using the same configuration as the first max-pooling and producing an output shape [24 x 10].
- And the last, ***fully connected layer***, calculate the final dot product, weights, and biases using the BPNN technique, with loss cross-entropy autocorrelation dot product and stochastic gradient descent (SGD) operations, as well as weight transformation with the flatten, at this stage, it is used to convert the output pooling layer into a 1-dimensional vector. In the propagation and classification process or predicting images, a dropout regulation technique selects several neurons randomly and will not be used during the training process; in other words, the neurons are discarded randomly. This process aims to reduce overfitting during the training [91, 92, 93, 94]. Furthermore, another fully connected process uses dense with 80-unit nodes and the layer using *SoftMax* activation function, and this layer becomes the last layer that will calculate the probability of the input image against all target classes, which is possible and will then determine the target class based on the given input compact.

D. Model Conversion and Quantization

The conversion process is required for inferencing a pre-trained model to every compact embedded microcontroller. This study used a library and an API interpreter provided by TensorFlow, which adjusts the data type for the C array on ESP32 to store the modeling conversion via ESP32 read-only memory (ROM) so that quantization and transformation when using the function of tf. lite converter for microcontrollers, only a tiny amount of memory was needed [42]. Quantize model is carried out one time of 8bit quantization according to the capacity of ESP32. After conversion and quantization, the modeling weight value will be hex code in a C array with *.cc* format.

E. Button Control

Button control using two tactile buttons, or momentary buttons, with an internal pullup on the ESP32, then connected to 2 analog GPIOs on the ESP32 with an 8-bit unsigned char integer data type, 0-255 with a normally open gate, and for loop sequence for 0–180-degree servo increments. The button sequence sub-program will be combined with the last stage of *TinyMl* firmware.

F. Wireless Protocol with ESP-NOW

Wireless communication is provided by ESP32 using Wi-Fi infrastructure with an organization identifier marked on the MAC address number on the microcontroller device known as ESP-NOW, so peer-to-peer communication must register between microcontroller devices and enable to have repairing process efficiently. The control topology used in this study is unicast, as shown in Fig. 7.

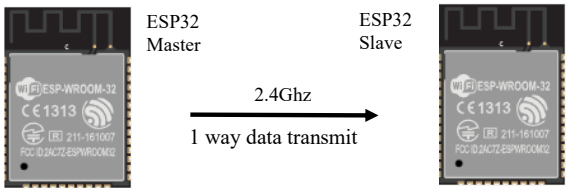


Fig. 7. ESP NOW unicast protocol transmitting command words to prosthethic devices

G. Final Embedded TinyML Master–Slave

The final multi-control firmware using 2 separate firmwares. The master and slave firmware for each of the 2 ESP32; the ESP32-Master, as shown in Fig. 8, installed on the remote control, ESP32-Master firmware combined button control and SCR into one framework – with subprograms, headers, and supporting libraries, namely, ESP NOW master register peer & ESP NOW slave callback, Freertos Xtask with semaphore, pulse width modulation (PWM) servo, analog to digital (ADC) compiler, The ESP32-Slave, as shown in Fig. 9, was mounted on the servo controller in this case as a substitution of flexion and extension movement patterns for the Exo-glove poly using a hand prosthetic and servo mechanism from proto 1 [95], using underactuated pulley mechanism with DC servo as artificial MCP joint or flexion and extension to articulate finger movement which has similar as exo-glove poly servo mechanism [96, 97, 98, 99].

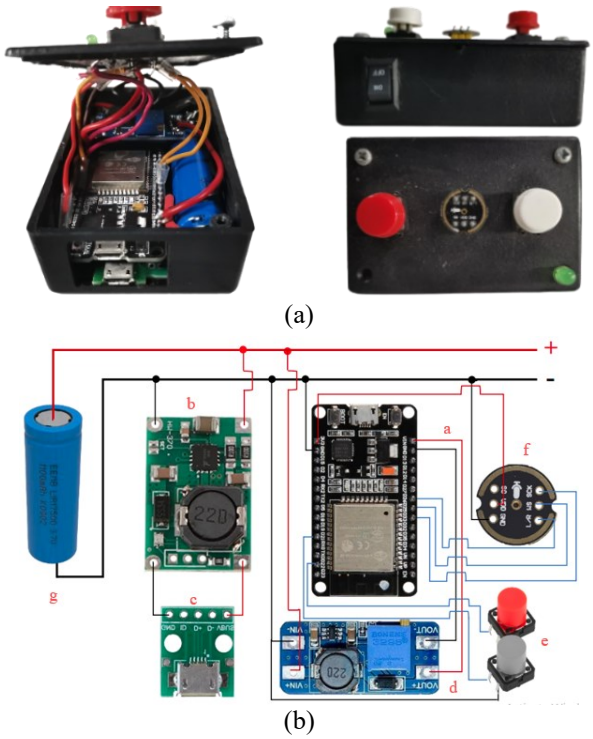


Fig. 8. (a) ESP32-master controller. (b) ESP32-master schematic

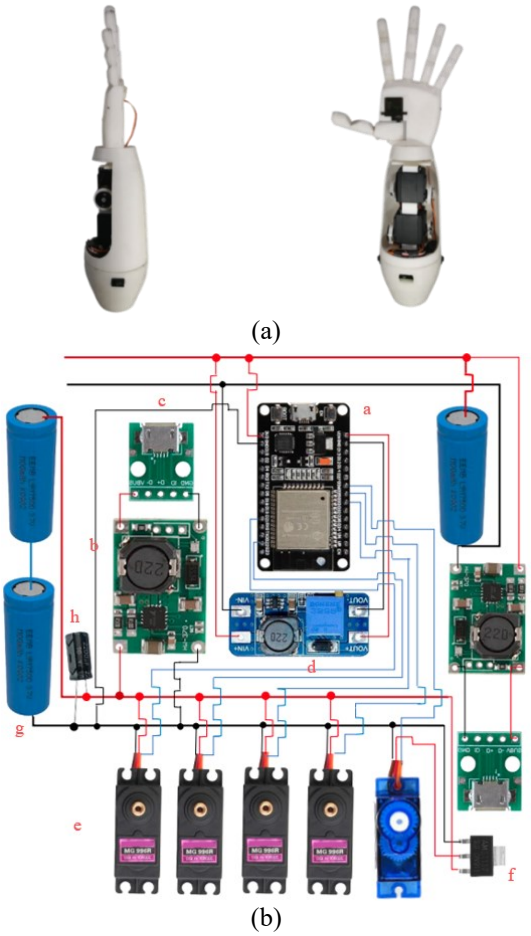


Fig. 9. (a) ESP32-slave prosthetic. (b) ESP32-slave schematic

The final firmware will be built with C++ programming with a .cpp extension like a pre-trained model design. The ESP32-Slave will be a receiver, that will callback data threshold sent by the ESP32-Master, then control HIGH or LOW for the 4 GPIO output pins connected to the each 4 servo running in series, based on the threshold activation xtasks, respectively.

IV. EXPERIMENTAL RESULT

A. Spectrogram Image Result

Data in this study are shown; only the initial and the final processes of the STFT mechanism, which are signals in the time domain, and the results of STFT signal transformation for the features model in a total time of 1 second. As shown in the graphs below, feature representation as a spectrogram was computed without downsampling, and it can be identified by spectral pattern or from the pixel image value for the ML model. Characteristics of each signal will be different by determining the value of the envelope amplitude or frequency bin on each signal so that the convolution layer, BPNN learning in the MLP layer, or fully connected until the output layer will get a prediction of neuron weight in target class its respective.

Extracted Speech on the word "Forward" is shown in Fig. 10. The high-amplitude activity is in the 50-2000 Hz bin frequency interval at 200-550ms when viewed from the spectrogram image. The second speech sample is in the frequency interval bin 50-2000 Hz at a time of 200-850ms. There is a slight difference in the magnitude of the amplitude with respect to time, but it still looks identical. Meanwhile, other amplitude activities at different times also represent the word "forward" characteristics when computing the spectrogram to be used in modeling. So that when compared between 2 samples of "forward" on the plot of spectrogram-1 and spectrogram-2, it is visually different and able to spot even without using ML modeling. The characteristics of the sound signal character can be seen in the output shape.

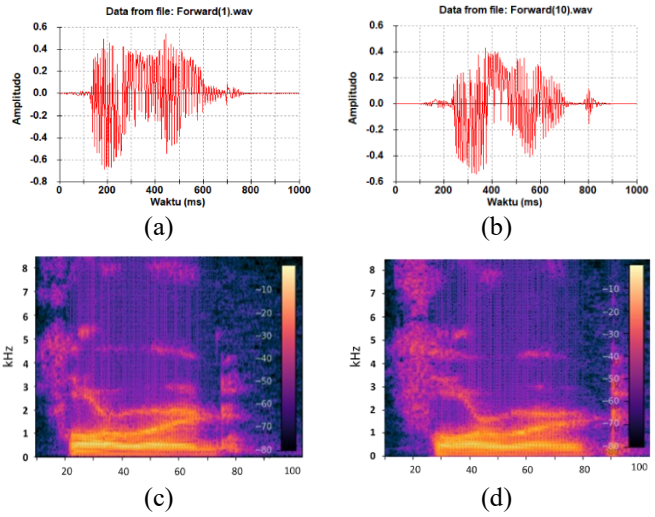


Fig. 10. (a) Time domain signal “Forward-1”. (b) Time domain signal “Forward-2”. (c) Spectrogram “Forward-1”. (d) Spectrogram “Forward-2”

The first sample of the dysarthric vocal speech "eee" shown in Fig. 11 can't be easily distinguished by a pattern rather than the non-dysarthric speech vowel.

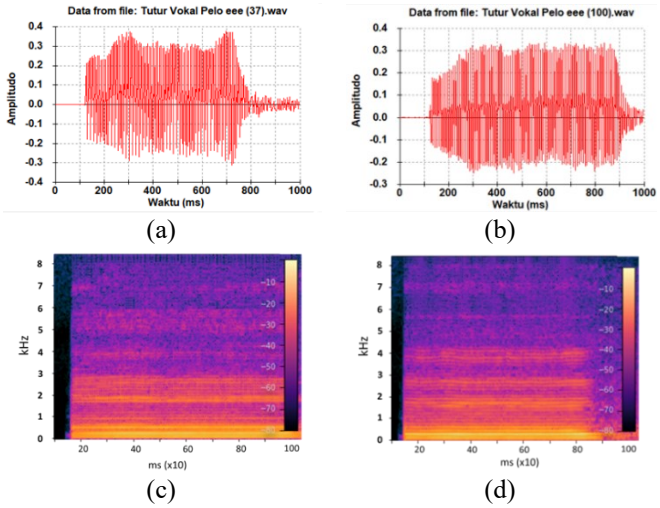


Fig. 11. (a) Time Domain Signal “vowel eee-1”. (b) Time Domain Signal “vowel eee-2”. (c) Spectrogram “vowel eee-1”. (d) Spectrogram “vowel eee-2”

Fig. 11 is the pattern looks constant because single phonemes are pronounced as steady or continuous at a particular frequency level, with a vocal intensity that can be aligned, depending on the speaker's condition. However, the amplitude activity in the first speech, produce the highest frequency in the bin frequency interval of 10-256 Hz at 200-800 ms. The second speech sample is at the bin frequency

interval of 10-200 Hz at 20-80 ms. There is a slight difference in the amplitude at a particular frequency indicates the intensity of the speaker, but identical amplitude activity can be seen between frequencies of 256Hz and 3kHz.

For dysarthric vowel speech "iii" as shown in Fig. 12, the high-amplitude activity is in the 50-400 Hz bin frequency interval at 200-800 ms. The 2nd speech sample is in the 50-400 Hz bin frequency interval at 200-800 ms. Pattern for Non-dysarthric speech can be distinguished without scaling at a particular frequency, but dysarthric vowel speech has a distinctive identity at each bin frequency when looking at specific logarithmic scaling, as shown in Fig. 13.

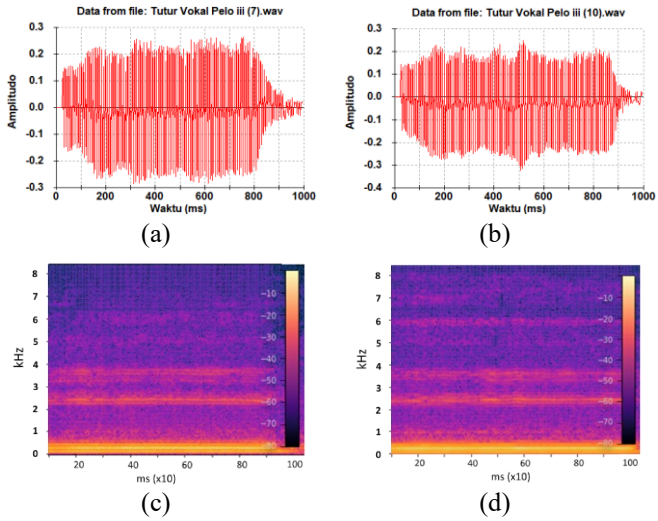


Fig. 12. (a) Time domain signal "vowel iii-1". (b) Time Domain Signal "vowel iii-2". (c) Spectrogram "vowel iii-1". (d) Spectrogram "vowel iii-2"

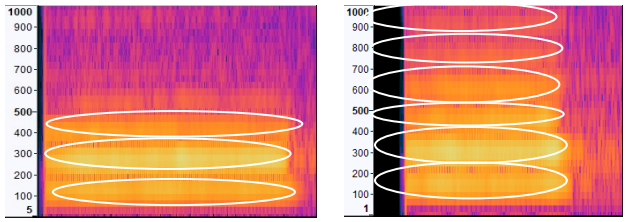


Fig. 13. Differences in amplitude activity of vowels iii in logarithmic scale

B. Training and Accuracy 2DCNN Modelling Result

The results training was carried out in this research by each batch, epochs, or iterations, with a total of 10 epochs, and doing data sate divided batch size by 32 states in each epoch.

It is known that the results in Table III above and the following Fig. 14 are the accumulated results of training data and validation data with a ratio of 80:10% in each epoch iteration, which shows the value of the validation data is not greater than the value of the training data or represented in the Table III value as accuracy, and loss, while the validation data is "validation-loss" and "validation-accuracy", and it states that the modeling with the architecture used in addition to getting the highest of accuracy is 94%, with the smallest loss value of 3.2%, meaning that the modeling on the training results seems does not experience overfitting.

TABLE III. OBTAINED ACCURACY AND VALIDATION ACCURACY OF PROPOSED 2DCNN MODEL

Epoch	Loss (%)	Val-loss (%)	Accuracy (%)	Val-Accuracy
0	0.4498	0.2881	0.8707	0.9318
1	0.3522	0.2861	0.9080	0.9334
2	0.3418	0.2790	0.9119	0.9345
3	0.3372	0.2742	0.9130	0.9367
4	0.3347	0.2667	0.9146	0.9373
5	0.3300	0.264	0.915	0.9354
6	0.3286	0.2659	0.9159	0.9350
7	0.3267	0.2611	0.9172	0.9378
8	0.3260	0.2730	0.9177	0.9349
9	0.3255	0.2608	0.9177	0.9400

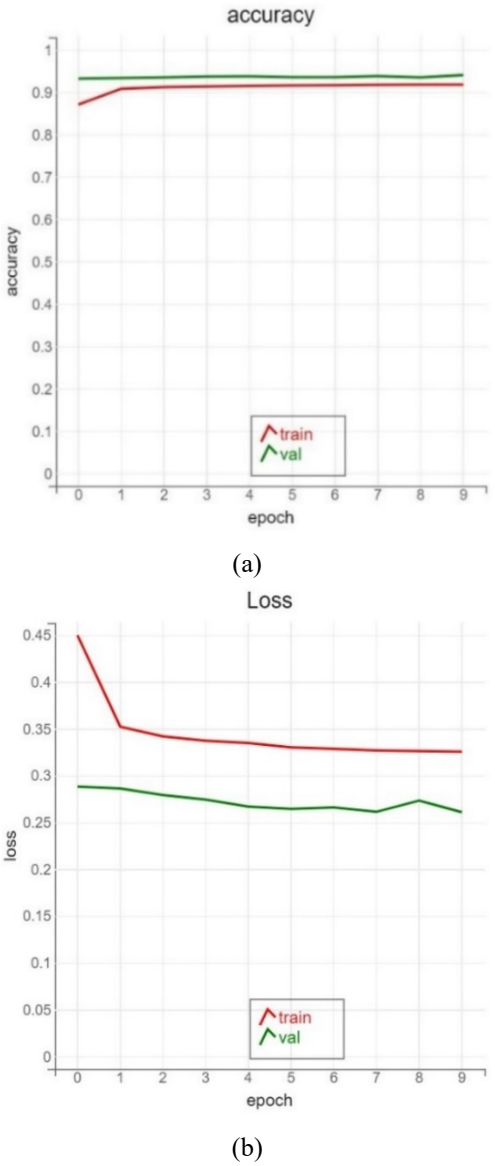


Fig. 14. (a) The proposed model training accuracy (accuracy vs epochs). (b) Model loss (loss vs epochs)

C. Testing 2DCNN Modelling Result

As mentioned in the dataset preparation sub-page, the training, validation, and testing data composition was divided into a ratio of 80:10:10%. However in this research,

limited dataset for dysarthic vowel speech with single subject utterances were obtained based on small dataset approach that conducted in previous research [100-102]. In this research, 100 records for each target class for testing data and only ten records data were tested as new data. Then the testing results is shown in the confusion matrix Fig. 15 using testing data. Dysarthric vocal speech appears to have 100% accuracy without missing data.

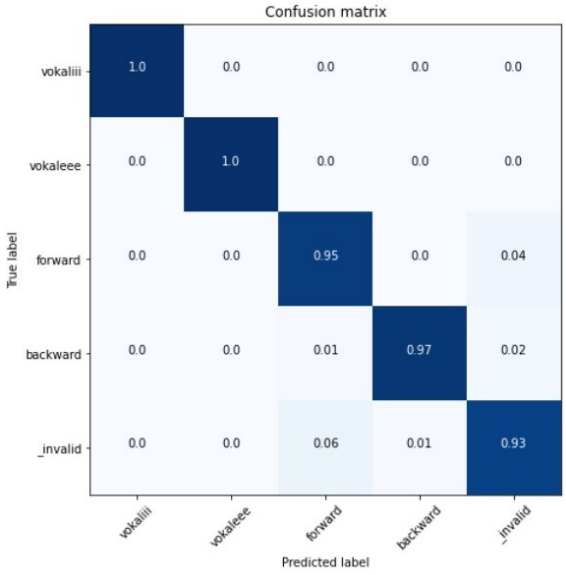


Fig. 15. The confusion matrix of testing model results. The y-axis represents the true label and the x-axis represents the predicted label for SCR classes

This phenomenon is declared overfitting in the testing process with identical new data, or the model is too memorized from the speaker's subject, or this phenomenon is also often called audio fingerprinting. However, for non-dysarthric speech utterances with better quantitative and qualitative datasets taken directly from the TensorFlow library dataset or *speech _data v2*, the accuracy of the prediction results is very relevant in the 93-97% range with 1-7% spatial data.

The accuracy comparison model shown in Table IV, with the proposed architecture and limited dataset, the result shown a reliable performance. However, this proposed model could accept dysarthic vowel speech and non dsyarthic command word on this study implementation, and perform well according table shown in the next following section.

TABLE IV. TRAINING ACCURACY COMPARISON FROM PROPOSED METHOD WITH EXISTING LITERATURE

Article	Algorithm	Accuracy (%)
David et al. [31]	CNN	86%
Y.-Y. Lin et al. [32]	CNN – MFCC	65.67%
Yakoub et al. [33]	CNN – EMDH- MFCC	62.7%
Shamiri et al. [35]	SV – CNN	71 %
Jiang et al. [37]	DLSTM – MFCC	82.1 %
[Proposed]	2DCNN – STFT	93 %

D. TinyML Controller Speech Test Results

The speech test was conducted with the noise measured with a sound pressure level (SPL) meter software, or decibel meter using the *"decibel x"* apps available on Playstore, with sensibility performance and calibration of noise in a relatively representative indoor scenario, which ranges from 24db, 42db and 62db as shown in Table V, Table VI, and Table VII. The distance and position on this test were around ± 25cm from the sound source with a speech intensity ranging from 60-70db. Proto-1 will perform flexion and extension servo mechanism activities as a substitute for the Exo glove during a duty cycle or fully opening and closing hands sequence.

TABLE V. SPEECH TEST WITH NOISE LEVEL ±24DB - ENCLOSED ROOM

Command Word	No of test	True	Percentage (%)
Vowel ''iii''	20	18	90
Vowel ''eee''	20	17	85
Forward	20	20	100
Backward	20	20	100

TABLE VI. SPEECH TEST WITH NOISE LEVEL ±42DB – HOUSEHOLD

Command Word	No of test	True	Percentage (%)
Vowel ''iii''	20	16	80
Vowel ''eee''	20	15	75
Forward	20	18	90
Backward	20	18	90

TABLE VII. SPEECH TEST WITH NOISE LEVEL ±62DB – CROWD IN THE ROOM

Command Word	No of test	True	Percentage (%)
Vowel ''iii''	20	5	25
Vowel ''eee''	20	13	65
Forward	20	9	40
Backward	20	11	55

In the first test, the noise level of about 24dB was tested in a closed room scenario and the system had a good performance with level accuracy ranging and consistent around 85-100%. In the second scenario, which is about 42dB of noise generated by environmental or surrounding noise around the house, such as motorbikes, birds chirping, kitchen activities, and other natural noises. The accuracy of speech recognition decreases around -10% level from the latest value, and the noise scenario is 62db which was tested among a crowd of conversations, and the performance decline is up to -70% so that the design SCR system in this research is temporarily better if applied to noise levels ranging from 20-42 dB.

E. Button Test Results

Testing for button controller with an on/off sequence, which moves every 5 increments, or 5 degrees on the Proto - 1 servos, then observes its activity in a serial monitor, along with the response of the data packet transmission, measured on each button pressed for 1-meter range. The measured time response from the controller was about 45ms per increment, and packet size transmission activity was observed for a total of 1000ms period as shown in Table VIII and Table IX.

The button bounce phenomenon was recorded by looking at packet size activity per 1000ms and had no significant bounce. The number of buttons pressed for movement of flexion or extension until it is intact, or 100% duty cycle measured, was 12 times, and the maximum bounce was around at 20 increments or 4 times the trigger button sequence but does not interfere with high spike,

peaking, or ripple on servo voltage. Meanwhile, the limitation of data transmission parameters measured in this study is the number per frame/complete packet, which is 100-111 packets/second.

TABLE VIII. WIRELESS BOUNCING BUTTON TEST – FLEXION SEQUENCE

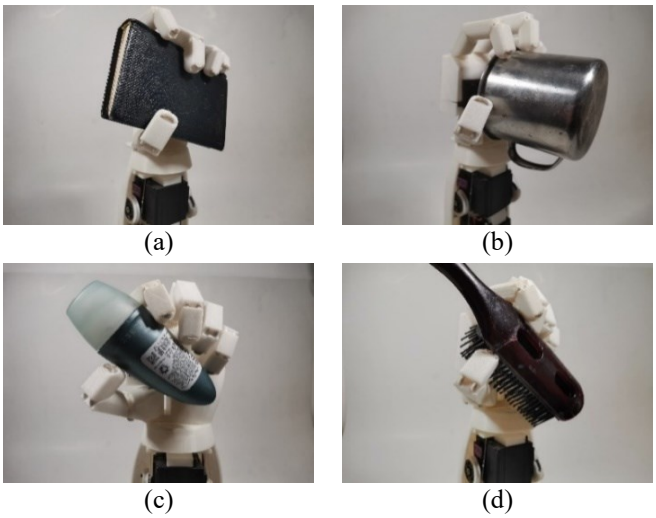
Flexion				
Push Order	**Increment (0-180)**	**Time (ms)**	**Packet size**	**Degree (°)**
1	5-20	45	109	0-20
2	25-30	48	100	20-30
3	35-45	47	108	30-45
4	50-60	47	102	45-60
5	65-75	47	108	60-75
6	80-85	46	106	75-85
7	90-95	47	105	85-95
8	100 - 105	47	105	95-105
9	110-130	94	107	105-130
10	135-145	46	110	130-145
11	150-165	95	105	145-165
12	170-180	47	111	165-180

TABLE IX. WIRELESS BOUNCING BUTTON TEST – EXTENSION SEQUENCE

Extension				
Push Order	**Increment (0-180)**	**Time (ms)**	**Packet size**	**Degree (°)**
1	180-165	47	117	180-165
2	160 - 150	46	117	165-150
3	145-130	48	108	150-130
4	130-125	96	108	130-125
5	125-120	50	100	125-120
6	115-100	93	109	120-100
7	95-80	93	111	100-80
8	75-60	608	105	80-60
9	55-45	92	111	60-45
10	45-40	94	110	45-40
11	35-20	46	110	40-20
12	15-0	46	112	0

F. Range of Motion (ROM) Proto-1

Fig. 16 is the ROM activities in this study were carried out to indicate projection performance of flexion and extension activities using hand prosthetics and validate the implementation of a multi-control system with servo performance for grip activities on selected objects according to the study [9].



(a) (b) (c) (d)

Fig. 16. Photographs ROM Activities, demonstrate Proto-1 grasping objects of various shapes. (a–d) Grasping a Qur'an, grasping a metal mug, grasping a deodorant, and grasping a comb

V. CONCLUSION

Speech recognition system for dysarthric and non-dysarthric uses the CNN-STFT technique to represent the sound signal captured by the microphone sensor, then processes it into a spectrogram log image, which will be classified into image patterns by the CNN algorithm. The total dataset used was 102,124 recordings, combined with the *speech_command v2* dataset and a small dataset for dysarthric speakers. Testing Accuracy for dysarthric and non-dysarthric speech control is quite good, "forward" speech get 95% accuracy, for "Backward" speech 97%, for vowel dysarthric speech "eee" and "iii" both have overfitted phenomenon 100% (overfit) cause of lack quantity, quality and augmented method for a small dataset but in real life testing, had a quite good performance. Implementing neural net modeling on ESP32 with the INMP441 mic sensor module as voice signal input was conducted with three scenarios of noise levels 24db, 42db, and 62db, each tested 20 times, and produced an average accuracy of 95%, 85%, 50% respectively. The results of the wireless data distribution response with ESP-NOW using a button control, the on-off sequence for the remote was about 60ms, while for the full duty cycle it was about 800ms. For hardware or electrical circuit design, there is a bouncing phenomenon in momentary button, use filters in the circuit to reduce this phenomenon, adding an H-bridge for the efficiency of servo output current, and using PID or fuzzy logic control for the safety open loop mechanism of the servo. In feature engineering or dataset engineering, overfitting occurs in modeling, especially in large dysarthic spoken language datasets, so increasing the quantity and quality of datasets, such as using datasets from severe subjects or related open datasets, and performing preprocessing variations like using mel-spectrogram, log mel spectrogram, L2M spectrogram, Chroma and others, can be used as a comparison for upcoming research.

REFERENCES

[1] C. M. J. M. Dourado Jr, S. P. P. da Silva, R. V. M. da Nóbrega, A. C. da S. Barros, P. P. R. Filho, and V. H. C. de Albuquerque, "Deep learning IoT system for online stroke detection in skull computed tomography images," *Computer Networks*, vol. 152, pp. 25–39, Apr. 2019, doi: 10.1016/j.comnet.2019.01.019.

[2] S. Wang, H. Zhai, L. Wei, B. Shen, and J. Wang, "Socioeconomic status predicts the risk of stroke death: A systematic review and meta-analysis," *Preventive Medicine Reports*, vol. 19, p. 101124, Sep. 2020, doi: 10.1016/j.pmedr.2020.101124.

[3] F. Herpich and F. Rincon, "Management of Acute Ischemic Stroke," *Critical Care Medicine*, vol. 48, no. 11, pp. 1654–1663, Oct. 2020, doi: 10.1097/ccm.0000000000004597.

[4] R. Chiaramonte and M. Vecchio, "A Systematic Review of Measures of Dysarthria Severity in Stroke Patients," *PM&R*, vol. 13, no. 3, pp. 314–324, Oct. 2020, doi: 10.1002/pmrj.12469.

[5] K. Brown and K. Spencer, "Dysarthria following Stroke," *Seminars in Speech and Language*, vol. 39, no. 1, pp. 015–024, Jan. 2018, doi: 10.1055/s-0037-1608852.

[6] Z. Mou, Z. Chen, J. Yang, and L. Xu, "Acoustic properties of vowel production in Mandarin-speaking patients with post-stroke dysarthria," *Scientific Reports*, vol. 8, no. 1, Sep. 2018, doi: 10.1038/s41598-018-32429-8.

[7] M.-Y. Liaw *et al*., "Respiratory muscle training in stroke patients with

respiratory muscle weakness, dysphagia, and dysarthria – a prospective randomized trial," *Medicine*, vol. 99, no. 10, p. e19337, Mar. 2020, doi: 10.1097/md.0000000000019337.

[8] R. Chiaramonte, P. Pavone, and M. Vecchio, "Speech rehabilitation in dysarthria after stroke: a systematic review of the studies," *European Journal of Physical and Rehabilitation Medicine*, vol. 56, no. 5, Nov. 2020, doi: 10.23736/s1973-9087.20.06185-7.

[9] R. Islam, M. Tarique, and E. Abdel-Raheem, "A Survey on Signal Processing Based Pathological Voice Detection Techniques," *IEEE Access*, vol. 8, pp. 66749–66776, 2020, doi: 10.1109/access.2020.2985280.

[10] S. Pinto *et al*., "Treatments for dysarthria in Parkinson's disease," *The Lancet Neurology*, vol. 3, pp . 547-556, 204, doi: doi: 10.1016/S1474-4422(04)00854-3.

[11] A. F. Rumbach, E. Finch, and G. Stevenson, "What are the usual assessment practices in adult non-progressive dysarthria rehabilitation? A survey of Australian dysarthria practice patterns," *Journal of Communication Disorders*, vol. 79, pp. 46–57, May 2019, doi: 10.1016/j.jcomdis.2019.03.002.

[12] P. Balzan, C. Tattersall, and R. Palmer, "Non-invasive brain stimulation for treating neurogenic dysarthria: A systematic review," *Annals of Physical and Rehabilitation Medicine*, vol. 65, no. 5, p. 101580, Sep. 2022, doi: 10.1016/j.rehab.2021.101580.

[13] M. Icht, "Improving speech characteristics of young adults with congenital dysarthria: An exploratory study comparing articulation training and the Beatalk method," *Journal of Communication Disorders*, vol. 93, p. 106147, Sep. 2021, doi: 10.1016/j.jcomdis.2021.106147.

[14] I. Díaz et al., "Development of a robotic device for post-stroke home tele-rehabilitation," *Advances in Mechanical Engineering*, vol. 10, no. 1, Jan. 2018, doi: 10.1177/1687814017752302.

[15] A. A. Khan, S. K. Ranjha, M. U. Akram, S. G. Khawaja, and A. Shaukat, "Neurotransmission cognitive theory: A novel approach for non-invasive brain stimulation using mechanical vibrations for the rehabilitation of neurological patients," *Medical Hypotheses*, vol. 143, p. 110078, Oct. 2020, doi: 10.1016/j.mehy.2020.110078.

[16] R. S. Calabrò *et al*., "Does hand robotic rehabilitation improve motor function by rebalancing interhemispheric connectivity after chronic stroke? Encouraging data from a randomised-clinical-trial," *Clinical Neurophysiology*, vol. 130, no. 5, pp. 767–780, May 2019, doi: 10.1016/j.clinph.2019.02.013.

[17] I. Boukhennoufa, X. Zhai, V. Utti, J. Jackson, and K. D. McDonald-Maier, "Wearable sensors and machine learning in post-stroke rehabilitation assessment: A systematic review," *Biomedical Signal Processing and Control*, vol. 71, p. 103197, Jan. 2022, doi: 10.1016/j.bspc.2021.103197.

[18] K. Nuckols *et al*., "Proof of Concept of Soft Robotic Glove for Hand Rehabilitation in Stroke Survivors," *Archives of Physical Medicine and Rehabilitation*, vol. 100, no. 12, p. e195, Dec. 2019, doi: 10.1016/j.apmr.2019.10.099.

[19] C. Proulx, D. Gagnon, and J. Higgins, "Perceived Usability and Acceptability of a Soft Robotic Glove for Rehabilitation of Adults With Hand Hemiparesis: A Mixed-Method Study Among Occupational Therapists in Stroke Rehabilitation," *Archives of Physical Medicine and Rehabilitation*, vol. 101, no. 11, p. e101, Nov. 2020, doi: 10.1016/j.apmr.2020.09.308.

[20] J. D. Setiawan, M. Ariyanto, S. Nugroho, M. Munadi, and R. Ismail, "A Soft Exoskeleton Glove Incorporating Motor-Tendon Actuator for Hand Movements Assistance," *International Review of Automatic Control (IREACO)*, vol. 13, no. 1, p. 1, Jan. 2020, doi: 10.15866/ireaco.v13i1.18274.

[21] B. B. Kang, H. Choi, H. Lee, and K.-J. Cho, "Exo-Glove Poly II: A Polymer-Based Soft Wearable Robot for the Hand with a Tendon-Driven Actuation System," *Soft Robotics*, vol. 6, no. 2, pp. 214–227, Apr. 2019, doi: 10.1089/soro.2018.0006.

[22] C.-Y. Chu and R. M. Patterson, "Soft robotic devices for hand rehabilitation and assistance: a narrative review," *Journal of NeuroEngineering and Rehabilitation*, vol. 15, no. 1, Feb. 2018, doi: 10.1186/s12984-018-0350-6.

[23] P. Tran, S. Jeong, S. L. Wolf, and J. P. Desai, "Patient-Specific, Voice-Controlled, Robotic FLEXotendon Glove-II System for Spinal Cord Injury," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 898–905, Apr. 2020, doi: 10.1109/lra.2020.2965900.

[24] A. Dwivedi, L. Gerez, W. Hasan, C.-H. Yang, and M. Liarokapis, "A Soft Exoglove Equipped With a Wearable Muscle-Machine Interface Based on Forcemyography and Electromyography," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3240–3246, Oct. 2019, doi: 10.1109/lra.2019.2925302.

[25] A. Foroutannia, M.-R. Akbarzadeh-T, and A. Akbarzadeh, "A deep learning strategy for EMG-based joint position prediction in hip exoskeleton assistive robots," *Biomedical Signal Processing and Control*, vol. 75, p. 103557, May 2022, doi: 10.1016/j.bspc.2022.103557.

[26] F. Wang, Y. Chen, Y. Wang, Z. Liu, Y. Tian, and D. Zhang, "A soft pneumatic glove with multiple rehabilitation postures and assisted grasping modes," *Sensors and Actuators A: Physical*, vol. 347, p. 113978, Nov. 2022, doi: 10.1016/j.sna.2022.113978.

[27] Q. Liu, J. Zuo, C. Zhu, and S. Q. Xie, "Design and control of soft rehabilitation robots actuated by pneumatic muscles: State of the art," *Future Generation Computer Systems*, vol. 113, pp. 620–634, Dec. 2020, doi: 10.1016/j.future.2020.06.046.

[28] M. V. M. Neves, L. Furlan, F. Fregni, L. R. Battistella, and M. Simis, "Robotic-Assisted Gait Training (RAGT) in Stroke Rehabilitation: A Pilot Study," *Archives of Rehabilitation Research and Clinical Translation*, vol. 5, no. 1, p. 100255, Mar. 2023, doi: 10.1016/j.arrct.2023.100255.

[29] P. Caliandro et al., "Exoskeleton-assisted gait in chronic stroke: An EMG and functional near-infrared spectroscopy study of muscle activation patterns and prefrontal cortex activity," *Clinical Neurophysiology*, vol. 131, no. 8, pp. 1775–1781, Aug. 2020, doi: 10.1016/j.clinph.2020.04.158.

[30] T. Triwiyanto, S. Luthfiyah, I. Putu Alit Pawana, A. Ali Ahmed, and A. Andrian, "Bilateral mode exoskeleton for hand rehabilitation with wireless control using 3D printing technology based on IMU sensor," *HardwareX*, vol. 14, p. e00432, Jun. 2023, doi: 10.1016/j.ohx.2023.e00432.

[31] D. Mulfari, G. Meoni, M. Marini, and L. Fanucci, "Towards a Deep Learning Based ASR System for Users with Dysarthria," *Computers Helping People with Special Needs*, pp. 554–557, 2018, doi: 10.1007/978-3-319-94277-3_86.

[32] Y.-Y. Lin *et al*., "A Speech Command Control-Based Recognition System for Dysarthric Patients Based on Deep Learning Technology," *Applied Sciences*, vol. 11, no. 6, p. 2477, Mar. 2021, doi: 10.3390/app11062477.

[33] M. S. Yakoub, S. Selouani, B.-F. Zaidi, and A. Bouchair, "Improving dysarthric speech recognition using empirical mode decomposition and convolutional neural network," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2020, no. 1, Jan. 2020, doi: 10.1186/s13636-019-0169-5.

[34] B. Vachhani, C. Bhat, and S. K. Kopparapu, "Data Augmentation Using Healthy Speech for Dysarthric Speech Recognition," *Proc. Interspeech 2018*, pp. 471-475, Sep. 2018, doi: 10.21437/interspeech.2018-1751.

[35] S. R. Shahamiri, "Speech Vision: An End-to-End Deep Learning-Based Dysarthric Automatic Speech Recognition System," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 29, pp. 852–861, 2021, doi: 10.1109/tnsre.2021.3076778.

[36] A. A. Joshy and R. Rajan, "Automated Dysarthria Severity Classification: A Study on Acoustic Features and Deep Learning Techniques," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 30, pp. 1147–1157, 2022, doi: 10.1109/tnsre.2022.3169814.

[37] W. Ye, Z. Jiang, Q. Li, Y. Liu, and Z. Mou, "A hybrid model for pathological voice recognition of post-stroke dysarthria by using 1DCNN and double-LSTM networks," *Applied Acoustics*, vol. 197, p. 108934, Aug. 2022, doi: 10.1016/j.apacoust.2022.108934.

[38] B. A. D. la C. Sánchez, M. A. Montiel, and E. L. González, "EMG-controlled hand exoskeleton for assisted bilateral rehabilitation," *Biocybernetics and Biomedical Engineering*, vol. 42, no. 2, pp. 596–614, Apr. 2022, doi: 10.1016/j.bbe.2022.04.001.

[39] F. Putri, W. Caesarendra, E. D. Pamanasari, M. Ariyanto, and J. D. Setiawan, "Parkinson Disease Detection Based on Voice and EMG Pattern Classification Method for Indonesian Case Study," *Journal of Energy, Mechanical, Material and Manufacturing Engineering*, vol.

3, no. 2, p. 87, Dec. 2018, doi: 10.22219/jemmme.v3i2.6977.

[40] J. Hou, Y. Shi, M. Ostendorf, M.-Y. Hwang, and L. Xie, "Region Proposal Network Based Small-Footprint Keyword Spotting," *IEEE Signal Processing Letters*, vol. 26, no. 10, pp. 1471–1475, Oct. 2019, doi: 10.1109/lsp.2019.2936282.

[41] A. Ghandoura, F. Hjabo, and O. A. Dakkak, "Building and benchmarking an Arabic Speech Commands dataset for small-footprint keyword spotting," *Engineering Applications of Artificial Intelligence*, vol. 102, p. 104267, Jun. 2021, doi: 10.1016/j.engappai.2021.104267.

[42] R. Bhalley, "TensorFlow Basics," *Deep Learning with Swift for TensorFlow*, pp. 143–169, 2021, doi: 10.1007/978-1-4842-6330-3_4.

[43] V. J. Reddi *et al*., "Widening Access to Applied Machine Learning with TinyML," *Harvard Data Science Review*, Jan. 2022, doi: 10.1162/99608f92.762d171a.

[44] A. M. Rostami, A. Karimi, and M. A. Akhaee, "Keyword spotting in continuous speech using convolutional neural network," *Speech Communication*, vol. 142, pp. 15–21, Jul. 2022, doi: 10.1016/j.specom.2022.06.001.

[45] E. van der Westhuizen, H. Kamper, R. Menon, J. Quinn, and T. Niesler, "Feature learning for efficient ASR-free keyword spotting in low-resource languages," *Computer Speech & Language*, vol. 71, p. 101275, Jan. 2022, doi: 10.1016/j.csl.2021.101275.

[46] L. Liu, M. Yang, X. Gao, Q. Liu, Z. Yuan, and J. Zhou, "Keyword spotting techniques to improve the recognition accuracy of user-defined keywords," *Neural Networks*, vol. 139, pp. 237–245, Jul. 2021, doi: 10.1016/j.neunet.2021.03.012.

[47] S. Cai *et al*., "A Voice-Activated Switch for Persons with Motor and Speech Impairments: Isolated-Vowel Spotting Using Neural Networks," *Proc. Interspeech 2021*, pp. 4823-4827, Aug. 2021, doi: 10.21437/interspeech.2021-330.

[48] K. Dokic, D. Mandusic, and B. Radisic, "Analysis of ESP32 SoC for Feed-Forward Neural Network Applications," *Innovation in Information Systems and Technologies to Support Learning Research*, pp. 165–175, Dec. 2019, doi: 10.1007/978-3-030-36778-7_18.

[49] M. Z. H. Zim, "TinyML: analysis of sekar Xtensa LX6 microprocessor for neural network applications by ESP32 SoC," *Machine Learning*, Jun. 2021, doi: arXiv:2106.10652.

[50] R. S. Iborra and A. F. Skarmeta, "TinyML-Enabled Frugal Smart Objects: Challenges and Opportunities," *IEEE Circuits and Systems Magazine*, vol. 20, no. 3, pp. 4–18, Aug. 2020, doi: 10.1109/mcas.2020.3005467.

[51] S. Asutkar, C. Chalke, K. Shivgan, and S. Tallur, "TinyML-enabled edge implementation of transfer learning framework for domain generalization in machine fault diagnosis," *Expert Systems with Applications*, vol. 213, p. 119016, Mar. 2023, doi: 10.1016/j.eswa.2022.119016.

[52] M. M. Shibl, L. S. Ismail, and A. M. Massoud, "A machine learning-based battery management system for state-of-charge prediction and state-of-health estimation for unmanned aerial vehicles," *Journal of Energy Storage*, vol. 66, p. 107380, Aug. 2023, doi: 10.1016/j.est.2023.107380.

[53] P. P. Ray, "A review on TinyML: State-of-the-art and prospects," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 4, pp. 1595–1623, Apr. 2022, doi: 10.1016/j.jksuci.2021.11.019.

[54] H. Rahman *et al*., "IoT enabled mushroom farm automation with Machine Learning to classify toxic mushrooms in Bangladesh," *Journal of Agriculture and Food Research*, vol. 7, p. 100267, Mar. 2022, doi: 10.1016/j.jafr.2021.100267.

[55] D. M. Matilla, Á. L. Murciego, D. M. J. Bravo, A. S. Mendes, and V. R. Q. Leithardt, "Low-cost Edge Computing devices and novel user interfaces for monitoring pivot irrigation systems based on Internet of Things and LoRaWAN technologies," *Biosystems Engineering*, vol. 223, pp. 14–29, Nov. 2022, doi: 10.1016/j.biosystemseng.2021.07.010.

[56] Lu, Xugang, Sheng Li, and M. Fujimoto, "Automatic speech recognition," *Speech-to-speech translation*, pp. 21-38, 2020.

[57] M. Yankayiş, "Performance Evaluation of Feature Extraction and Modeling Methods for Speaker Recognition," *Annals of Reviews & Research*, vol. 4, no. 3, Nov. 2018, doi: 10.19080/arr.2018.04.555639.

[58] B. Ustubioglu, G. Tahaoglu, and G. Ulutas, "Detection of audio copy-move-forgery with novel feature matching on Mel spectrogram," *Expert Systems with Applications*, vol. 213, p. 118963, Mar. 2023, doi: 10.1016/j.eswa.2022.118963.

[59] G. Parisi, A. Coluccia, and A. Fascista, "On time-frequency correlation in spectrogram samples with application to target detection," *Signal Processing*, vol. 200, p. 108648, Nov. 2022, doi: 10.1016/j.sigpro.2022.108648.

[60] S. Jothimani and K. Premalatha, "MFF-SAug: Multi feature fusion with spectrogram augmentation of speech emotion recognition using convolution neural network," *Chaos, Solitons & Fractals*, vol. 162, p. 112512, Sep. 2022, doi: 10.1016/j.chaos.2022.112512.

[61] D. Kim and J. Lee, "Predictive evaluation of spectrogram-based vehicle sound quality via data augmentation and explainable artificial Intelligence: Image color adjustment with brightness and contrast," *Mechanical Systems and Signal Processing*, vol. 179, p. 109363, Nov. 2022, doi: 10.1016/j.ymssp.2022.109363.

[62] T. Alam and A. Khan, "Lightweight CNN for Robust Voice Activity Detection," *Lecture Notes in Computer Science*, pp. 1–12, 2020, doi: 10.1007/978-3-030-60276-5_1.

[63] Sutikno, K. Anam, and A. Saleh, "Voice Controlled Wheelchair for Disabled Patients based on CNN and LSTM," *2020 4th International Conference on Informatics and Computational Sciences (ICICoS)*, pp. 1-5, Nov. 2020, doi: 10.1109/icicos51170.2020.9299007.

[64] J. Kwon and D. Park, "Hardware/Software Co-Design for TinyML Voice-Recognition Application on Resource Frugal Edge Devices," *Applied Sciences*, vol. 11, no. 22, p. 11073, Nov. 2021, doi: 10.3390/app112211073.

[65] A. Suryarasmi, C. -C. Chang, R. Akhmalia, M. Marshallia, W. -J. Wang, and D. Liang, "FN-Net: A lightweight CNN-based architecture for fabric defect detection with adaptive threshold-based class determination," *Displays*, vol. 73, p. 102241, Jul. 2022, doi: 10.1016/j.displa.2022.102241.

[66] C. Chen, H. Seo, and Y. Zhao, "A novel pavement transverse cracks detection model using WT-CNN and STFT-CNN for smartphone data analysis," *International Journal of Pavement Engineering*, vol. 23, no. 12, pp. 4372–4384, Jun. 2021, doi: 10.1080/10298436.2021.1945056.

[67] S. Duan, H. Zheng, and J. Liu, "A Novel Classification Method for Flutter Signals Based on the CNN and STFT," *International Journal of Aerospace Engineering*, vol. 2019, pp. 1–8, Apr. 2019, doi: 10.1155/2019/9375437.

[68] J. Huang, B. Chen, B. Yao, and W. He, "ECG Arrhythmia Classification Using STFT-Based Spectrogram and Convolutional Neural Network," *IEEE Access*, vol. 7, pp. 92871–92880, 2019, doi: 10.1109/access.2019.2928017.

[69] S. M. Beeraka, A. Kumar, M. Sameer, S. Ghosh, and B. Gupta, "Accuracy Enhancement of Epileptic Seizure Detection: A Deep Learning Approach with Hardware Realization of STFT," *Circuits, Systems, and Signal Processing*, vol. 41, no. 1, pp. 461–484, Jul. 2021, doi: 10.1007/s00034-021-01789-4.

[70] A. Pandey and D. Wang, "A New Framework for CNN-Based Speech Enhancement in the Time Domain," in *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 7, pp. 1179-1188, July 2019, doi: 10.1109/TASLP.2019.2913512.

[71] Y. Lee, H. J. Park, I. H. Bae, and G. Kim, "Resonance Characteristics in Epiglottic Cyst: Formant Frequency, Vowel Space Area, Vowel Articulatory Index, and Formant Centralization Ratio," *Journal of Voice*, Oct. 2021, doi: 10.1016/j.jvoice.2021.09.008.

[72] P. Warden, "Speech commands: A dataset for limited-vocabulary speech recognition," *arXiv preprint arXiv:1804.03209*, Apr. 2018, doi: https://doi.org/10.48550/arXiv.1804.03209.

[73] M. M. Goodwin, "The STFT, Sinusoidal Models, and Speech Modification," *Springer Handbook of Speech Processing*, pp. 229–258, 2008, doi: 10.1007/978-3-540-49127-9_12.

[74] S. A. Alim and N. K. A. Rashid, *Some Commonly Used Speech Feature Extraction Algorithms*. London, UK: IntechOpen, 2018.

[75] X. Wang, T. Ying, and W. Tian, "Spectrum Representation Based on STFT," *2020 13th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pp. 435-438, 2020, doi: 10.1109/CISP-BMEI51763.2020.9263516.

[76] J. Benesty, J. Chen, and E. A. P. Habets. *Speech enhancement in the STFT domain*. Springer Science & Business Media, 2011.

[77] L. Rabiner and B. W. Juang. *Fundamentals of speech recognition*. Prentice-Hall, Inc., 1993.

[78] N. Kehtarnavaz, "Frequency Domain Processing," *Digital Signal Processing System Design*, pp. 175–196, 2008, doi: 10.1016/b978-0-12-374490-6.00007-6.

[79] K. Bhangale and K. Mohanaprasad, "Speech Emotion Recognition Using Mel Frequency Log Spectrogram and Deep Convolutional Neural Network," *Lecture Notes in Electrical Engineering*, pp. 241–250, Oct. 2021, doi 10.1007978-981-16-4625-6_24.0

[80] M. Loughlin, Z. Xie, Y. Song, H. Phan, and R. Palaniappan, "Time–Frequency Feature Fusion for Noise Robust Audio Event Classification," *Circuits, Systems, and Signal Processing*, vol. 39, no. 3, pp. 1672–1687, Jul. 2019, doi: 10.1007/s00034-019-01203-0.

[81] M. T. Nguyen, W. W. Lin, and J. H. Huang, "Heart Sound Classification Using Deep Learning Techniques Based on Log-mel Spectrogram," *Circuits, Systems, and Signal Processing*, vol. 42, no. 1, pp. 344–360, Aug. 2022, doi: 10.1007/s00034-022-02124-1.

[82] H. Xu, J. Zhang, and L. Dai, "Differential Time-frequency Log-mel Spectrogram Features for Vision Transformer Based Infant Cry Recognition," *Proc. Interspeech 2022*, pp. 1963-1967, Sep. 2022, doi: 10.21437/interspeech.2022-18.

[83] D. Gao, X. Tang, M. Wan, G. Huang, and Y. Zhang, "EEG driving fatigue detection based on log-Mel spectrogram and convolutional recurrent neural networks," *Frontiers in Neuroscience*, vol. 17, Mar. 2023, doi: 10.3389/fnins.2023.1136609.

[84] M. M. Oo and L. L. Oo, "Fusion of Log-Mel Spectrogram and GLCM Feature in Acoustic Scene Classification," *Studies in Computational Intelligence*, pp. 175–187, Jul. 2019, doi: 10.1007/978-3-030-24344-9_11.

[85] H. Meng, T. Yan, F. Yuan, and H. Wei, "Speech Emotion Recognition From 3D Log-Mel Spectrograms With Deep Learning Network," *IEEE Access*, vol. 7, pp. 125868–125881, 2019, doi: 10.1109/access.2019.2938007.

[86] Z. Diao, J. Yan, Z. He, S. Zhao, and P. Guo, "Corn Seedling Recognition Algorithm Based on Hyperspectral Image and Lightweight-3d-Cnn," *SSRN Electronic Journal*, 2022, doi: 10.2139/ssrn.4162664.

[87] O. Attallah, "CerCan·Net: Cervical cancer classification model via multi-layer feature ensembles of lightweight CNNs and transfer learning," *Expert Systems with Applications*, vol. 229, p. 120624, Nov. 2023, doi: 10.1016/j.eswa.2023.120624.

[88] J. Yang, L. Zhang, X. Tang, and M. Han, "CodnNet: A lightweight CNN architecture for detection of COVID-19 infection," *Applied Soft Computing*, vol. 130, p. 109656, Nov. 2022, doi: 10.1016/j.asoc.2022.109656.

[89] H. I. Hussein, A. O. Mohammed, M. M. Hassan, and R. J. Mstafa, "Lightweight deep CNN-based models for early detection of COVID-19 patients from chest X-ray images," *Expert Systems with Applications*, vol. 223, p. 119900, Aug. 2023, doi: 10.1016/j.eswa.2023.119900.

[90] Y. Wang, S. Li, H. Zhang, and T. Liu, "A lightweight CNN-based model for early warning in sow oestrus sound monitoring," *Ecological Informatics*, vol. 72, p. 101863, Dec. 2022, doi: 10.1016/j.ecoinf.2022.101863.

[91] K. Sanjar, A. Rehman, A. Paul, and K. JeongHong, "Weight Dropout for Preventing Neural Networks from Overfitting," *2020 8th International Conference on Orange Technology (ICOT)*, Dec. 2020, doi: 10.1109/icot51877.2020.9468799.

[92] L. Li and M. Spratling, "Understanding and combating robust overfitting via input loss landscape analysis and regularization," *Pattern Recognition*, vol. 136, p. 109229, Apr. 2023, doi: 10.1016/j.patcog.2022.109229.

[93] O. İrsoy and E. Alpaydın, "Dropout regularization in hierarchical mixture of experts," *Neurocomputing*, vol. 419, pp. 148–156, Jan. 2021, doi: 10.1016/j.neucom.2020.08.052.

[94] S. H. Khan, M. Hayat, and F. Porikli, "Regularization of deep neural networks with spectral dropout," *Neural Networks*, vol. 110, pp. 82–90, Feb. 2019, doi: 10.1016/j.neunet.2018.09.009.

[95] Q. K. Pham, T. V. Vo, and P. T. Tran, "On the Implementation of a Low-Cost Mind-Voice-and-Gesture-Controlled Humanoid Robotic Arm Using Leap Motion and Neurosky Sensor," *Journal of Electrical Engineering & Technology*, vol. 17, no. 1, pp. 665–683, Sep. 2021, doi: 10.1007/s42835-021-00903-5.

[96] W. Batayneh, E. Abdulhay, and M. Alothman, "Comparing the efficiency of artificial neural networks in sEMG-based simultaneous and continuous estimation of hand kinematics," *Digital Communications and Networks*, vol. 8, no. 2, pp. 162–173, Apr. 2022, doi: 10.1016/j.dcan.2021.08.002.

[97] J. Ramirez, A. Rubiano, and P. Castiblanco, "Soft Driving Epicyclical Mechanism for Robotic Finger," *Actuators*, vol. 8, no. 3, p. 58, Jul. 2019, doi: 10.3390/act8030058.

[98] J. Park, I. Hwang, and W. Lee, "Wearable Robotic Glove Design Using Surface-Mounted Actuators," *Frontiers in Bioengineering and Biotechnology*, vol. 8, Sep. 2020, doi: 10.3389/fbioe.2020.548947.

[99] D. Kim *et al*., "Eyes are faster than hands: A soft wearable robot learns user intention from the egocentric view," *Science Robotics*, vol. 4, no. 26, Jan. 2019, doi: 10.1126/scirobotics.aav2949.

[100] J. Shor *et al*., "Personalizing ASR for Dysarthric and Accented Speech with Limited Data," *Proc. Interspeech 2019*, pp. 784-788, Sep. 2019, doi: 10.21437/interspeech.2019-1427.

[101] A. Jalali, R. Mallipeddi, and M. Lee, "Sensitive deep convolutional neural network for face recognition at large standoffs with small dataset," *Expert Systems with Applications*, vol. 87, pp. 304-315, 2017, doi: 10.1016/j.eswa.2017.06.025.

[102] E. Li, L. Wang, Q. Xie, R. Gao, Z. Su, and Y. Li, "A novel deep learning method for maize disease identification based on small sample-size and complex background datasets," *Ecological Informatics*, vol. 75, p. 102011, Jul. 2023, doi: 10.1016/j.ecoinf.2023.102.