Voice Control Intelligent Wheelchair Movement Using CNNs

Mohammad Shahrul Izham Sharifuddin[1], Sharifalillah Nordin[2], Azliza Mohd Ali[3]
[1,2,3]*Faculty of Computer and Mathematical Sciences,*
Universiti Teknologi MARA
Shah Alam, Selangor, Malaysia
[1]mohammadshahrulizham@gmail.com,{[2]sharifa,[3]azliza}@tmsk.uitm.edu.my

***Abstract*— In this paper, we introduced a voice control intelligent wheelchair movement using Convolutional Neural Networks (CNNs). The intelligent wheelchair used four voice commands such as stop, go, left and right to assist disable people to move. Data are collected from google in the wav format. Mel-Frequency Cepstral Coefficient (MFCC) is applied to extract the command voice. The hardware used to deploy the system is Raspberry PI 3B+. The proposed method is using CNNs to classify the voice command and achieved excellent result with 95.30% accuracy. Therefore, the method can be commercialized and hopefully can give benefit to the disable society.**

Keywords— Classification, CNNs, Voice Command, MFCC

I. INTRODUCTION

The wheelchair is the important tool required by the people with mobility impairments due to illness, injury and disability. There are two types of wheelchair; motorized and manual wheelchair. Motorised wheelchair is a wheelchair integrated with electric motor and usually have joystick at the armrest while manual wheelchair is operated manually by a person rear of the chair. Elderly people and children with quadriplegia, spinal cord injuries (SCI) and amputation, require motorized wheelchair rather than the manual wheelchair [1]. Many researches currently focusing on creating motorized wheelchair to help the impaired person. Researchers have conducted many experiments to increase the accuracy of voice recognition using different feature extraction and classification technique like Support Vector Machine (SVM), Hidden Markov Model (HMM), Linear Prediction Coefficients (LPC), Mel Frequency Cepstral Coefficents (MFCC) and etc.

Voice recognition is one of a technique that enables the human voice to control a device. The needs to use buttons and switches are no longer needed; people can easily control appliances with their voice while doing other tasks. Voice command device (VCD) can be found in many home appliances. Lights and washing machine for example, can be controlled using voice. In the 1960 Texas Instruments has started to improve voice or speech identification technology. Ever since that time, voice identification has made aggressive development to be implemented in various domains [2]. There are currently several voice identification software available in the market such as Siri, Google Assistant, Alexa, and Cortana. The software is designed to help user implementing voice recognition in a device.

Different voice properties require different extraction and classification method. Table I shows five properties of voice characteristics used in this project. The properties are type of speech signal, type of speakers, type of language, type of vocabulary, and background noise.

TABLE I. PROPERTIES OF VOICE RECOGNITION

Properties	**Characteristic**
Type of speech signal	Isolated words
Type of speakers	Independent speakers
Type of language	English
Type of vocabulary	Small
Background noise	True

A development board is a small, compact circuit board that contains either microprocessor or microcontroller, or both. It supplies necessary components to hardware and software for bottom-up design and programming [3]. The board is used to process the signal or command and control the device. There are various development boards available in the market, and each development board has its advantages and disadvantages. Among the examples of development board available in the market are Raspberry PI, Banana PI, Arduino, and SK40C. The development board that will be used in this project is Raspberry PI 3B+, because it is easy to programme, cheaper compared to the other Single Board Computer (SBC), and faster in processing.

In this paper we propose an approach to recognize a voice command given by the user and send signal to the motor to control the movement of the wheelchair. This paper is organised as follows. Section II represents the related work on

978-1-7281-3041-5/19/$31.00 ©2019 IEEE.

voice recognition and convolutional neural networks. The proposed methodology is demonstrated in section III. Experimentation details and results are presented in section IV, and, finally, the paper is concluded in section V.

II. Related Works

Voice control wheelchair enables disabilities people move their wheelchair using voice. Electric wheelchair is better than manual when people want to move alone but people still needing hand to control. With the advancement of technology especially in applying artificial intelligence, wheelchair can be move by using voice recognition. Barriuso [4] proposed an agent based intelligent interface for wheelchair movement with smartphone application. Another application proposed by Nasrin [5] applied voice recognition and also GPS for the wheelchair user to track the location where they are. They also produced mobile apps for the wheelchair user. However, the user must have the Wi-Fi connection to activate the GPS features. Meanwhile, Avutu et.al [1] proposed a voice recognition wheelchair with the low cost map which only can be applied to local location. Therefore, the cost will be much cheaper because the application can be used without wifi connection. Another application proposed by [6], the system applied Arduino Mega 2560 as a controller and had some additional devices to enhance security features for example sending emergency messages to the important people. They applied ultrasonic sensors to detect and avoid conflicts with any obstacles. The best part of this system it also can works without internet connection.

Recently, many researches related to voice recognition have applied CNNs. CNNs is one of neural network techniques that are currently popular and achieve better results compared to other supervised techniques in neural network such as backpropagation. Back propagation neural network (BPNN) is the second generation of neural network that calculates error to produce learning [7] similar to CNNs. BPNN consists of three-layer; input layer, hidden layer while CNNs involves one or more convolution layers, pooling layer and fully connected layer [8]. CNNs is a feedforward neural network and have better generalisation than networks with full connectivity between adjacent layers [9]. CNNs has been successfully applied with great success in many others application such as image recognition [10], surveillance [11], and person identification [12][13]. Lei and She [14] applying CNNs to authenticate voice in a noisy environment. The proposed method reduces equal error rate and produce better accuracy in authenticate voice. Guan [15] applying CNNs to optimise performance in speech recognition and the result improved the recognition performance with error rate 13.88%.

III. Proposed Method

CNNs is one of the most powerful techniques used in voice recognition research. CNNs provides high accuracy especially in solving image recognition and speech recognition problem [16]. CNNs operate well in filtering noise and highlight important points in the dataset. CNNs is divided into two layers, where the first layer is the feature extraction and the second layer is classification layer.

A. Feature Extraction Layer

Feature extraction layer consists of two layers, i.e. Convolutional Layer and Pooling Layer. 2D Convolutional Layer and three 2D Max Pooling layers are used in this project.

B. Classification Layer

Classification layer is used to classify the class. There are three layers that need to be set in this project; there are input layer, hidden layer, and output layer.

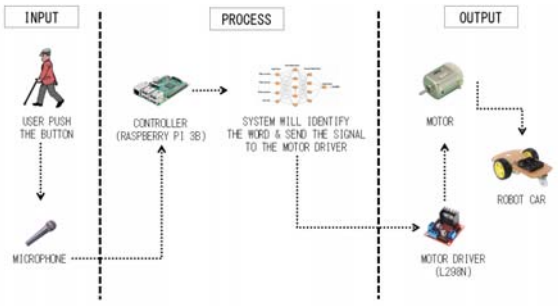

Fig. 1. Flow of the system

Fig 1 shows the flow of the system. Firstly, the user i.e. the disabled people push the button while recording the voice command using the microphone. The recorded voice is then processed inside the Raspberry PI 3B+ using MFCC and CNNs. After the system recognizes the voice, the output signal is sent to the motor driver. The motor driver determines the movement of the motor and moves the robot car.

IV. Experimentations and Results

A. Data Sample

In this paper, we collected four raw voice data which are go, left, right and stop. Other than that, also two types of noises which are urban noise and white noise are also collected. This data were downloaded from Google and for white noise the data are self-recorded. The reason why white noise is self-recorded, it is because white noise audio data are not available online. The raw voice data are saved in WAV format, and each of the voice will be the one-second length. Table II, shows the number of the command that has been downloaded and recorded in this research. Total of collected data are 14,145. 11,845 data are collected from Google and 2,300 data are self-recorded.

B. Mel Frequency Cepstral Coefficients (MFCC)

MFCC feature extraction is very close to the human auditory system and becomes the most suitable technique to be used. The reason behind this, because MFCC put the

frequency band logarithmically, with these it will allow speech to process better [17]. In this step, feature extraction is to get the signal features. Those signal features later will be used in feature matching.The MFCC process will be done using Librosa library. Librosa library is one of the famous libraries that used to extract a feature from the audio signal. In this process, 20th MFCC will be extracted from the raw audio signal. The raw data of MFCC will be in 2-Dimensional shape. Therefore, for different algorithm, the data preparation will be different.

TABLE II. NUMBER OF A SAMPLE OF VOICE COMMAND

Sources	Command	Total of data
Google	Go	2,372
	Stop	2,380
	Left	2,353
	Right	2,367
	Urban Noise	2,373
Self-Recorded	White Noise	2,300
Total of the data		14,145

C. Data Preparation

2-dimensional CNNs is used in this research. Therefore, the dataset needs to be prepared as 2D. Since 2D CNNs already need a 2D dataset, so we do not need to compress it. We just have to take it directly from raw MFCC. Before that, we also need to make sure that all the dataset is in the same shape. For this project, the shape of the data that has been set is (44, 20). The dataset has been set like that because of our data only 1-second length, so that why the shape of the data is not so big. If the shape of the data exceeded the shape that has been set, then the excess will be cut off and if the data shape is not enough it will be pad with zero. Lastly, the channel used is 1 which is mean grayscale. Figure 2 shows the flow of data preparation of CNNs.

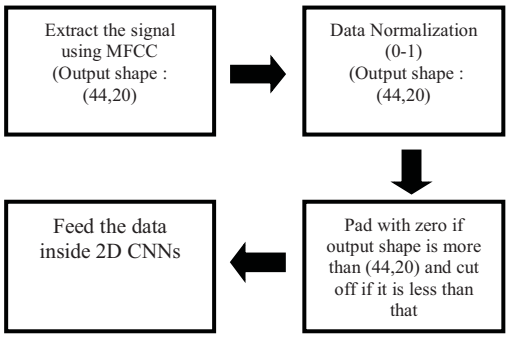

Fig 2 Flow of the data preparation for CNNs

D. BPNN Parameter Tuning

We also train the data using Backpropagation Neural Network to show the comparison between BPNN and CNNs. In BPNN parameter tuning there are 12 experiments have been done. We set number of input layers is 20 nodes, 6 hidden layers with 20 nodes and 6 outputs. Table III shows the best model that have been achieved after 12 experiments have been conducted. In this model, the best accuracy achieved is 74.87% with 0.61270 minute.

TABLE III. BPNN BEST MODEL

Layers	Nodes	Activation	Bias
Input	20	-	-
Hidden Layer	20	relu	True
Hidden Layer	20	relu	True
Hidden Layer	20	relu	True
Hidden Layer	20	relu	True
Hidden Layer	20	relu	True
Hidden Layer	20	relu	True
Output	6	softmax	True
Optimizer	Adam (lr = 0.001)		
Loss Function	Logcosh		
Epochs	200		
Batch Size	40		

E. CNNs Parameter Tuning

In CNNs parameter tuning there are 12 experiments have been done. Table IV shows the best model that have been achieved after 12 experiment have been conducted. In this model, the best accuracy achieved is 95.30% with 4.1672 minute.

TABLE IV. CNNS BEST MODEL

Layers	Kernel Size	Stride	Filters	Padding	Nodes	Bias	Activat -ion
Input	20,44,1						
2D	2	1	16	valid	-	False	relu
2D Max	2	2	-	valid	-	-	-
2D	2	1	32	valid	-	False	relu
2D Max	2	2	-	valid	-	-	-
2D	2	1	64	valid	-	False	relu
2D Max	2	2	-	valid	-	-	-
Flatten	-				-	-	
Output	-				6	False	softplus
Optimizer	Adam						
Loss	mean_squared_error						
Epochs	50						
Batch Size	10						

F. Comparison between BPNN and CNNs

In this project, we can see that CNNs produced 95.30% accuracy higher compared to BPNN which is only 74.87%. In

time comparison, CNNs took 4.1672 minutes to run while BPNN only 0.61270 minutes. This is because in the CNNs, there are feature extraction layer that able to filter the noise of the audio before feed into the classification layer.

TABLE V. MODEL COMPARISON

Model	Accuracy	Time(m)
BPNN	74.87%	0.61270
CNNs	95.30%	4.1672

G. Hardware Tuning

Table VI shows the logic gate used in Motor Driver LN298N, this logic gate is used to control the direction of the motor and use it to move the demo car according to the required command.

TABLE VI. MOTOR DIRECTION

Motor Driver Pin LN298N				Motor Direction
In1	In2	In3	In4	
0	0	0	0	Stop
0	1	0	1	Go
1	0	0	1	Left
0	1	1	0	Right

V. CONCLUSION

In this paper, we developed a prototype that able to classify the command based on features that have been extracted from MFCC. With the highest accuracy of 95.30%, this prototype can be considered reliable and efficient. It is our hope that the prototype can help disabled people to move around without another person help. However, there are several limitations in the study, which can be considered in future enhancements/research such as the quality of the data, number of vocabularies used, and type of language used.

ACKNOWLEDGEMENT

The authors would like to thank to the Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, Malaysia for the support throughout this research.

REFERENCES

[1] S. R. Avutu, "Voice control module for Low cost Local-Map navigation based Intillegent wheelchair," *2017 IEEE 7th Int. Adv. Comput. Conf.*, pp. 609–613, 2017.

[2] S. Khan, H. Akmal, I. Ali, and N. Naeem, "Efficient and unique learning of in-car voice control for engineering education," pp. 1–6, 2017.

[3] R. Lang, M. Lescisin, and Q. H. Mahmoud, "Selecting a development board for your capstone or course project," *IEEE Potentials*, vol. 37, pp. 6–14, 2018.

[4] A. L. Barriuso, J. Pérez-Marcos, D. M. Jiménez-Bravo, G. V. González, and J. F. De Paz, "Agent-Based Intelligent Interface for Wheelchair Movement Control," *Sensors*, vol. 18, no. 1511, pp. 1–31, 2018.

[5] N. Aktar, I. Jahan, and B. Lala, "Voice Recognition based Intelligent Wheelchair and GPS Tracking System," in *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, 2019, pp. 7–9.

[6] Z. Raiyan, "Design of an Arduino Based Voice-Controlled Automated Wheelchair," pp. 21–23, 2017.

[7] T. Bouwmans, S. Javed, M. Sultana, and S. Ki, "Deep neural network concepts for background subtraction : A systematic review and comparative evaluation," *Neural Networks*, vol. 117, pp. 8–66, 2019.

[8] A. A. Amri, A. R. Ismail, and A. A. Zarir, "Comparative Performance of Deep Learning and Machine Learning Algorithms on Imbalanced Handwritten Data," *Int. J. Adv. Comput. Sci. Appl.*, vol. 9, no. 2, pp. 258–264, 2018.

[9] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nat. Int. Wkly. J. Sci.*, vol. 521, pp. 436–444, 2015.

[10] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN features off-the-shelf: An astounding baseline for recognition," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2014, pp. 512–519.

[11] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and F. F. Li, "Large-scale video classification with convolutional neural networks," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1725–1732.

[12] X. Wang, A. Mohd Ali, and P. Angelov, "Gender and Age Classification of Human Faces for Automatic Detection of Anomalous Human Behaviour," in *International Conference on Cybernetics (CYBCONF 2017)*, 2017, pp. 1–6.

[13] A. M. Ali and P. Angelov, "Anomalous behaviour detection based on heterogeneous data and data fusion," *Soft Comput.*, 2018.

[14] L. Lei and K. She, "Identity Vector Extraction by Perceptual Wavelet Packet Entropy and Convolutional Neural Network," 2018.

[15] W. Guan, "Performance Optimization of Speech Recognition System with Deep Neural Network Model," vol. 27, no. 4, pp. 272–282, 2018.

[16] A. Incze, Henrietta-Bernadett Jancso, Z. Szilagyi, A. Farkas, and C. Sulyok, "Bird Sound Recognition Using a Convolutional Neural Network," in *IEEE 16th International Symposium on Intelligent Systems and Informatics*, 2018, pp. 295–300.

[17] S. T. S and C. Lingam, "Speaker based Language Independent Isolated Speech Recognition System," in *2015 International Conference on Communication, Information & Computing Technology (ICCICT)*, 2015, pp. 1–7.