

Article

Steering a Robotic Wheelchair Based on Voice Recognition System Using Convolutional Neural Networks

Mohsen Bakouri ^{1,2,*}, Mohammed Alsehaimi ¹, Husham Farouk Ismail ³, Khaled Alshareef ¹, Ali Ganoun ⁴, Abdulrahman Alqahtani ¹ and Yousef Alharbi ⁵

¹ Department of Medical Equipment Technology, College of Applied Medical Science, Majmaah University, Al Majma'ah 11952, Saudi Arabia; m.alsehaimi90@gmail.com (M.A.); k.alshareef@mu.edu.sa (K.A.); a.alqahtani@mu.edu.sa (A.A.)

² Department of Physics, College of Arts, Sebha University, Traghen 71340, Libya

³ Department of Biomedical Equipment Technology, Inaya Medical College, Riyadh 13541, Saudi Arabia; hushamfarouk@inaya.edu.sa

⁴ Department of Electrical Engineering, College of Engineering, Tripoli University, Tripoli 22131, Libya; a.ganoun@uot.edu.ly

⁵ Department of Medical Equipment Technology, College of Applied Medical Sciences, Prince Sattam Bin Abdulaziz University, Al-Kharj 16278, Saudi Arabia; y.alharbi@psau.edu.sa

* Correspondence: m.bakouri@mu.edu.sa; Tel.: +966-533231905



Citation: Bakouri, M.; Alsehaimi, M.; Ismail, H.F.; Alshareef, K.; Ganoun, A.; Alqahtani, A.; Alharbi, Y. Steering a Robotic Wheelchair Based on Voice Recognition System Using Convolutional Neural Networks. *Electronics* **2022**, *11*, 168. <https://doi.org/10.3390/electronics11010168>

Academic Editors: Katarzyna Antosz, Jose Machado, Yi Ren, Rochdi El Abdi, Dariusz Mazurkiewicz, Marina Ranga, Pierluigi Rea, Vijaya Kumar Manupati, Emilia Villani and Erika Ottaviano

Received: 5 December 2021

Accepted: 3 January 2022

Published: 5 January 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Abstract: Many wheelchair people depend on others to control the movement of their wheelchairs, which significantly influences their independence and quality of life. Smart wheelchairs offer a degree of self-dependence and freedom to drive their own vehicles. In this work, we designed and implemented a low-cost software and hardware method to steer a robotic wheelchair. Moreover, from our method, we developed our own Android mobile app based on Flutter software. A convolutional neural network (CNN)-based network-in-network (NIN) structure approach integrated with a voice recognition model was also developed and configured to build the mobile app. The technique was also implemented and configured using an offline Wi-Fi network hotspot between software and hardware components. Five voice commands (yes, no, left, right, and stop) guided and controlled the wheelchair through the Raspberry Pi and DC motor drives. The overall system was evaluated based on a trained and validated English speech corpus by Arabic native speakers for isolated words to assess the performance of the Android OS application. The maneuverability performance of indoor and outdoor navigation was also evaluated in terms of accuracy. The results indicated a degree of accuracy of approximately 87.2% of the accurate prediction of some of the five voice commands. Additionally, in the real-time performance test, the root-mean-square deviation (RMSD) values between the planned and actual nodes for indoor/outdoor maneuvering were 1.721×10^{-5} and 1.743×10^{-5} , respectively.

Keywords: wheelchair; voice recognition; Raspberry Pi; Android; convolutional neural network

1. Introduction

Many patients still depend on others to help them move their wheelchairs, and patients with limited mobility still face significant challenges when using wheelchairs in public and in other places [1]. Statistics also indicate that 9–10% of patients who were trained to operate power wheelchairs could not use them for daily activities, and 40% of limited mobility patients reported that it was almost impossible to steer and maneuver a wheelchair [2]. Moreover, it was reported that approximately half of the 40% of patients with impaired mobility could not control a powered wheelchair [3]. Furthermore, the same study determined that over 10% of patients that use traditional power wheelchairs not equipped with any sensors have accidents after 4 months [3]. However, using an electric wheelchair equipped with an automatic navigation and sensor system, such as

a smart wheelchair, would be beneficial in addressing a significant challenge for several patients. The smart wheelchair is an electric wheelchair equipped with a computer and sensors designed to facilitate the efficient and effortless movement of patients [4–7]. These wheelchairs are considered safer and more comfortable than conventional wheelchairs because they introduce new control options, which include navigation systems (GPS) and other technologies, such as saving places on the user's map [8,9].

Various sensors can be used in smart wheelchairs, such as ultrasound, laser, infrared, and input cameras. These wheelchairs adopt computers that process input data from the sensors and produce a command that is sent to the motor to spin the wheels of the chair [10]. One of the most important developments in this field is the introduction of the joystick control system. This system drives the wheelchair via an intelligent control unit [11]. However, patients with impaired upper extremities cannot operate the joystick flexibly and smoothly. This leads to fatal accidents when situations require rapid action in motion. Therefore, the conventional joystick system needs to be replaced with advanced technologies [12]. The human–computer interface (HCI) is a method for controlling a wheelchair using a signal or a combination of different signals, such as electroencephalogram (EEG), electrooculogram (EOG), and electromyogram (EMG) [13–15]. Brain–computer interfaces (BCIs) are one of the most researched CIs that translate brain signals into action to control a device [16]. Regarding EEG-BCI, it has some limitations, including low spatial resolution and a low signal-to-noise ratio (SNR). A hybrid BCI (hBCI) that combines EEG with EOG exhibited improved accuracy and speed. Despite the HCI results, some limitations to the application of BCI systems still exist. EEG devices are relatively expensive, and bio-potential signals are affected by artifacts. Furthermore, although hBCI can address some of these challenges, it is not efficient and flexible in its simultaneous control of speed and direction [17–19].

Speech is the most important means of communication between humans and in human communication, speech is the most important mode of communication. By employing a microphone sensor, speech can be used to interact with a computer and serve as a potential method for human–computer interactions (HCI). These sensors are being used in quantifiable voice recognition research in human–computer interactions (HCI), which has applications in a variety of areas such as human–computer interactions (HCI), controlling wheelchairs, and health-related applications. Therefore, smart or intelligent wheelchair developments—based on voice recognition techniques—have increased significantly [20]. For instance, Aktar et al. [21] developed an intelligent wheelchair system using a voice recognition technique with a GPS tracking model. The voice commands were converted into hexadecimal numeral data to control the wheelchair in three different speed stages via a Wi-Fi module. The system also used an infrared radiation (IR) sensor to detect obstacles and used a mobile app to detect the location of the patient. Similarly, Raiyan et al. [22] developed an automated wheelchair system based on the Arduino and Easy VR3 speech recognition module. In this study, the authors claim that the implemented system is less expensive and does not require any wearable sensor or complex signal processing. In an advanced study, an adaptive neuro-fuzzy has been designed to drive a powered wheelchair. The system implementation was based on real-time control signals generated by the voice commands' classification unit. The proposed system used a wireless sensor network to track the wheelchair [23]. Despite highly sophisticated approaches presented by researchers in this area, high cost, accuracy in distinguishing, classifying, and identifying the patient's voice remain the most critical challenges.

To overcome the lack of accuracy for distinguishing and classifying patients' speech, many researchers have used the convolutional neural network (CNN) technique [24,25]. This technique relies on converting voice commands into spectrogram images before being fed into CNN. This method has proven to be helpful in the level of accuracy for speech recognition. In this context, Huang et al. [26] proposed a method to analyze CNN for speech recognition. In this method, visualizing the localized filters learned in the convolutional layer was used to detect automatic learning. The authors claim that this method has advantages of identifying four domains of CNN over the fully connected

method. These domains are distant speech recognition, noise robustness, low-footprint models, and channel-mismatched training–test conditions. In addition, Korvel et al. [27] analyzed 2D feature spaces for voice recognition based on CNN. The analysis used the Lithuanian word recognition task to feature maps. The results showed that the highest rate of word recognition was achieved using spectral analysis. Moreover, the Mel scale and spectral linear cepstra and chroma are outperformed by cepstral feature spaces.

The driving of smart wheelchairs using voice recognition technologies with CNN has attracted many researchers [28]. For instance, Sutikno et al. [24] proposed a voice control method for wheelchairs using long short-term memory (LSTM) and CNN. This method used Sox Sound Exchange and Sound Recorder Pro to achieve the objective. The accuracy level of this method was above 97.80%. Another study was conducted by Ali et al. [29], who designed an algorithm for smart wheelchairs using CNN to help people with disabilities in detecting buses and bus doors. The method was implemented based on accurate localization information and used CPU for fast detecting. However, the use of CNN in smartphones is still under development due to associated complex calculations to achieve high accuracy predictions [30].

This paper develops a new powerful, low-cost system based on voice recognition and CNN approaches to drive a wheelchair for disabled users. The method proposes the use of a network-in-network (NiN) structure for mobile applications [31]. The system used smartphones to create an interactive user interface that can be easily controlled by sending a voice command via the mobile application to the system's motherboard. A mobile application, voice recognition model, and CNN model were developed and implemented to achieve the main goal of this study. In addition, all safety issues were considered during driving and maneuvering at indoor and outdoor locations. Results showed that the implemented system was robust in time response and had accurate execution of all orders without time delay.

The paper is organized as follows: Section 2 illustrates the materials and methods used in this study. Section 3 addresses the experimental procedure. Section 4 shows the results of the study. Section 5 discusses the results. Section 6 concludes this study. Finally, Section 7 shows future work.

2. Materials and Methods

Figure 1 illustrates the implementation of the system architecture of the proposed system. This system is divided into two stages. The first stage is the set of hardware devices used to control the movement of the wheelchair reliably. These devices include a standard wheelchair, Android smartphone (Huawei Y9—CPU: Octa-core, 4 × 2.2 GHz), DC electric motors, batteries, relay model, Raspberry Pi4, and an emergency push button in case of an abnormal system response. The second stage focuses on the software development of the mobile application, voice recognition model, and CNN model. The software was designed and implemented to control the wheelchair using the five voice commands mentioned in Table 1. The main components for controlling the chair were connected via offline Wi-Fi.

In this work, the mobile app was built based on Flutter software [32,33]. The design process includes creating a user flow diagram for each screen, creating and drawing wireframes, selecting design patterns and color palettes, creating mock-ups, creating an animated app prototype, and designing final mock-ups to prepare the final screens for coding to be initiated. Usually, the app appears on the application list, and after it has been opened, it displays the enlisted words that we have trained in our model. After permitting the application to use our microphone, it attempts the words and highlights them in the interface recognition, as shown in Figure 1.

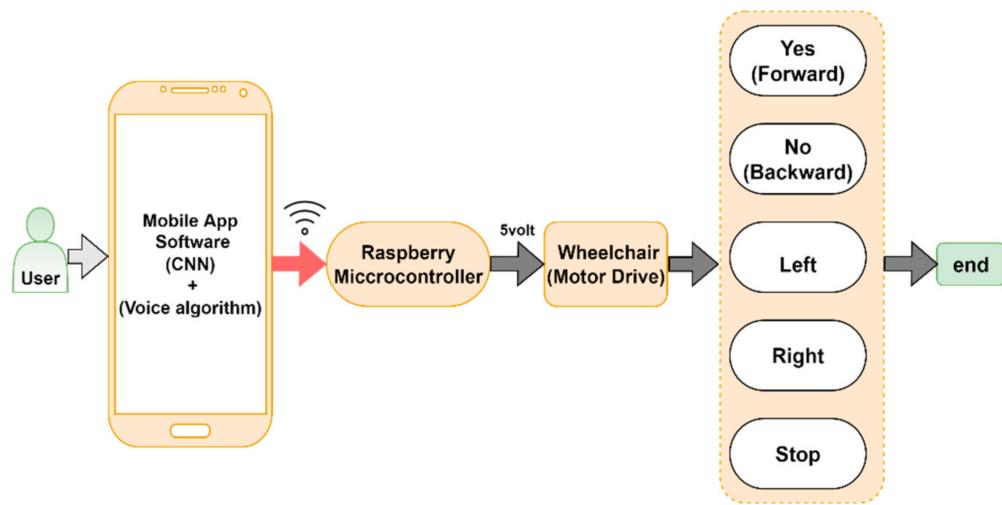


Figure 1. Overall system architecture.

Table 1. File of voice command.

Voice Command	Yes	No	Left	Right	Stop
Direction	Moving forward	Moving backward	Turning to the left	Turning to right	Not moving

2.1. Voice Recognition Model Development

Each audio file signal is subjected to feature extraction to create a map that shows how the signal changes in frequency over time. Therefore, the Mel frequency cepstral coefficients (MFCC) were used in speech analysis systems to extract this information [34]. The initial step in character extraction is to emphasize the signal by passing it through a one-coefficient digital filter (finite impulse response (FIR) filter) to prevent numerical instability as:

$$y(n) = x(n) - \beta x(n-1) \quad (1)$$

where $x(n)$ is the original voice signal, $y(n)$ is the output of the filter, n is the number of sampling, and β is a constant such that $0 < \beta \leq 1$.

To keep the samples in frame and reduce signal discontinuities, the framing and windowing [$w(n)$] are employed as:

$$w(n) = \begin{cases} (1 - \alpha) - \alpha \cos\left(\frac{2\pi n}{N-1}\right), & n = 0, 1, \dots, N-1 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

where α is a constant and N is the number of frames. For spectral analysis, fast Fourier transform (FFT) is applied to calculate the spectrum of magnitude for each frame as:

$$y(k) = \sum_{n=0}^{N-1} y(n) e^{-j2\pi kn/N}, \quad k = 0, \dots, N-1 \quad (3)$$

The spectrum is then processed by a bank of filters according to MFCC, where the Mel filter bank can be written as:

$$H_m[f_k] = \frac{f_k - f[m-1]}{f[m] - f[m-1]} \quad (4)$$

If we consider f_l and f_h to be lowest and highest on the filter bank in hertz and frequency, then the boundary points $f[m]$ can be written as:

$$f[m] = \left(\frac{N}{F_s} \right) B^{-1} \left(B(f_l) + m \frac{B(f_h) - B(f_l)}{M+1} \right) \quad (5)$$

where N is the size of the FFT, M is the number of filters, and B is the Mel scale which is given by:

$$B(f) = 1125 \ln \left(1 + \frac{f}{700} \right) \quad (6)$$

To eliminate noise and spectral estimation errors, we applied approximate homomorphic transform as:

$$S[m] = \ln \left[\sum_{k=0}^{N-1} |y[k]^2 H_m[k]| \right], \quad 0 < m \leq M \quad (7)$$

The logarithmic energy operation $\log(\sum|\cdot|^2)$ and the inverse of discrete cosine transformer (DCT) are used in the final step of MFCC processing. The use of DCT has features for high decorrelation, and partial decorrelation can be given as:

$$c_l[n] = \sqrt{\frac{2}{M}} \sum_{m=1}^M S_l[m] \cos \left[\frac{n\pi}{M} \left(m - \frac{1}{2} \right) \right], \quad n = 0, 1, \dots, L < M \quad (8)$$

To obtain the feature map, we take the first and second derivatives of (8) to obtain:

$$\Delta c_l[n] = \frac{\sum_{p=1}^p P(c_{l+p}[n] - c_{l-p}[n])}{2 \sum_{p=1}^p P^2} \quad (9)$$

$$\Delta^2 c_l[n] = \frac{\sum_{p=1}^p P(\Delta c_{l+p}[n] - \Delta c_{l-p}[n])}{2 \sum_{p=1}^p P^2} \quad (10)$$

This is applied to all recordings that have been made; the database was thus created and used by the CNN.

2.2. CNN Implementation Model

Here, we adopted the network-in-network (NIN) structure as the foundational architecture for mobile application development [35,36]. NIN is a CNN technique that does not include fully connected (FC) layers and, in addition, can accept images of any size as inputs to the network by employing global pooling rather than fixed-size pools. This is useful for mobile applications because users may adjust the balance between speed and accuracy without affecting the network weights.

To contrast CNN, we adopt a multi-threading technique. In this technique, the smartphone has four CPU cores that easily allow dividing a kernel matrix into four sub-matrices along with the row. Therefore, four generic matrix multiplication (GEMM) operations are carried out in parallel to obtain the output feature maps of the target convolution layer. Our method adopted cascaded cross channel parametric pooling (CCCPP) to compensate for the FC layers' elimination. Therefore, our CNN model consists of input, output layers, twelve convolution layers, and two consecutive layers, as shown in Figure 2.

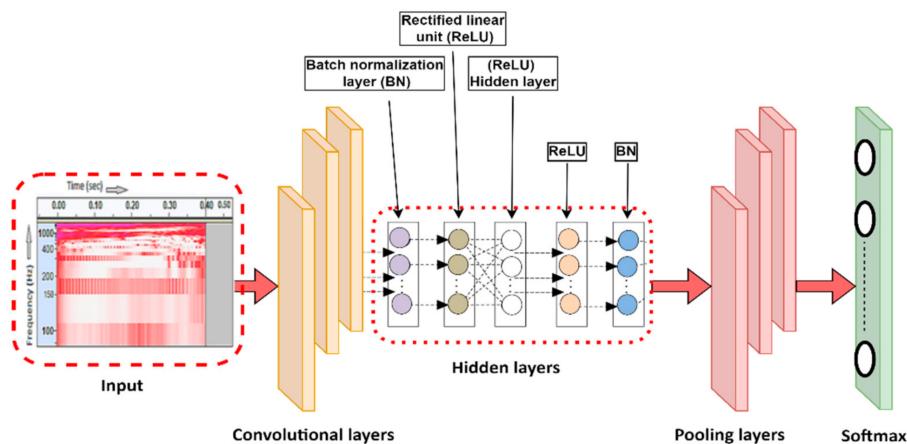


Figure 2. Diagram of the proposed NIN architecture of the neural network.

2.3. DC Motor Control Drive

The drive wheels are powered by motors at the rear and front ends of the chair. The rear motors correspond to the rear wheel movement, which is used to drive the wheels forward, and the front wheel (freewheels) corresponds to different chair movements. The two motors were connected to the driver via four power lines. The motor speed was predefined at approximately 1 km/h. To move forward or backward, both wheels will move clockwise and anti-clockwise, respectively. However, to turn right or left, one motor uses the entire free gear and the other moves forward. If one needs to turn left, the left wheel uses the free gear and the right one moves forward, thereby causing the wheelchair to move in the opposite direction. The movement table of the wheelchair is presented in Table 2.

All wheelchair movements are controlled by a relay module. The relay module provides four relays that are rated for 15–20 mA at 5 Vdc. Each relay has a normally closed (NC) and normally open (NO) contact. Each relay is controlled by a corresponding pin that originates from the microcontroller. The relays are optically isolated, and each motor is controlled by two relays: one relay is used to switch (on), and the other remains at the first position (off) by means (ground), which will cause the (on) motor to turn in a clockwise or opposite direction, based on the (on) or (off) state of the relay. Then, the command is sent by the microcontroller program, and the relay coil operates at 5 V.

Figure 2 presents the complete electronic circuit diagram for the wheelchair movement. In this diagram, the polarity across a load for the four relay modules can be altered in both directions. Terminals are connected between the common poles of the two relays and the DC motor. Normally open terminals are connected to the positive terminal, whereas a normally closed terminal of both relays is connected to a current driver circuit (ULN2033) to protect the pins of the controller from any abrupt sinking current. The current driver circuit can support approximately 500 mA, which is sufficient for the relay module. Furthermore, the diode connects to each relay to ensure protection from voltage spikes when the supply is disconnected.

Table 2. Truth table of the wheelchair movement (−5 v means activate the relay).

Voice Command	Active Relay				Active Motor	Movement Type
	Relay-10	Relay-11	Relay-12	Relay-13		
Yes	+5 v	−5 v	+5 v	−5 v	Left/right motor	Clockwise
No	−5 v	+5 v	−5 v	+5 v	Left/right motor	Anti-clockwise
Left	+5 v	−5 v	+5 v	+5 v	Right motor	Clockwise
Right	+5 v	+5 v	+5 v	−5 v	Left motor	Clockwise
Stop	+5 v	+5 v	+5 v	+5 v	No movement	No movement
Emergency	+5 v	+5 v	+5 v	+5 v	No movement	No movement

2.4. Mechanical Assembly

Figure 3 illustrates the mechanical assembly of a wheelchair. This wheelchair was purchased from the market, and no mechanical modifications were made to the basic design of the original chair. In our proposed design, an electro-mechanical motor was attached directly to the frame of the wheelchair. A wheelchair's maneuverability depends on the position of the steering wheels, which significantly affects the space required for the chair to turn, including the way the chair moves in narrow spaces. Owing to their small 360-degree turning circumference and tight turning radius (20–26 in), mid-wheel drives are the most maneuverable, making them excellent indoor wheelchairs. Table 3 summarizes the hardware specifications of all parts that are used in this work. Figure 4 presents a flowchart of the complete wheelchair system.

Table 3. Hardware specifications.

Parameter	Value
Mechanical Parts	
Wheelchair	Standard wheelchair
Motor pair	3.13.6LST10 24 v DC 120 rpm
Acid battery	NP7-12 12 v 7ah lead acid battery
Control Unit	
Raspberry pi4	SoC: Broadcom BCM2711B0 quad-core A72 (ARMv8-A) 64-bit @ 1.5 GHz. GPU: Broadcom Video Core VI. Networking: 2.4 GHz and 5 GHz 802.11b/g/n/ac wireless LAN. RAM: 4 GB LPDDR4 SDRAM. Bluetooth: Bluetooth 5.0, Bluetooth Low Energy (BLE) GPIO: 40-pin GPIO header, populated. Storage: microSD
Relay Module	5 V 4-channel relay interface board. 15–20 mA signal drive current. TTL logic compatible. High-current AC250 V/10 A, DC30 V/10 A relay.
Android Smartphone (Huawei Y9)	
CPU Core	Octa-core (4 × 2.2 GHz Cortex-A73 and 4 × 1.7 GHz Cortex-A53)
Memory	128 GB 6 GB RAM
Operation System	Android 8.1
Display	1080 × 2340 pixels, 19.5:9 ratio

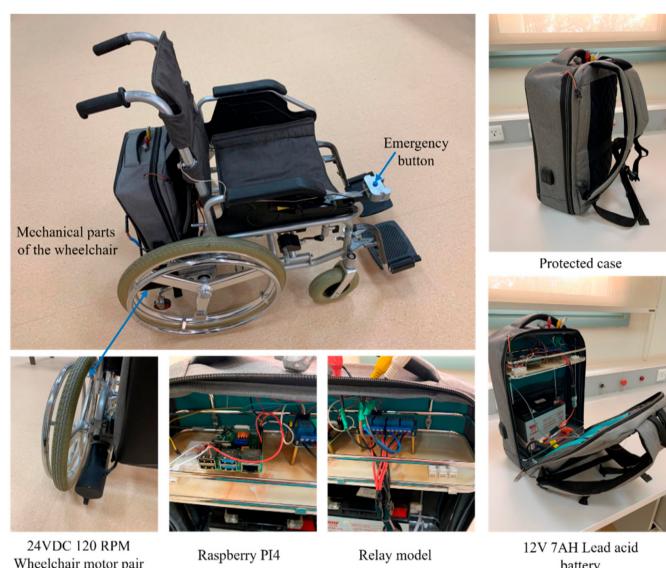


Figure 3. Mechanical assembly of wheel chair.

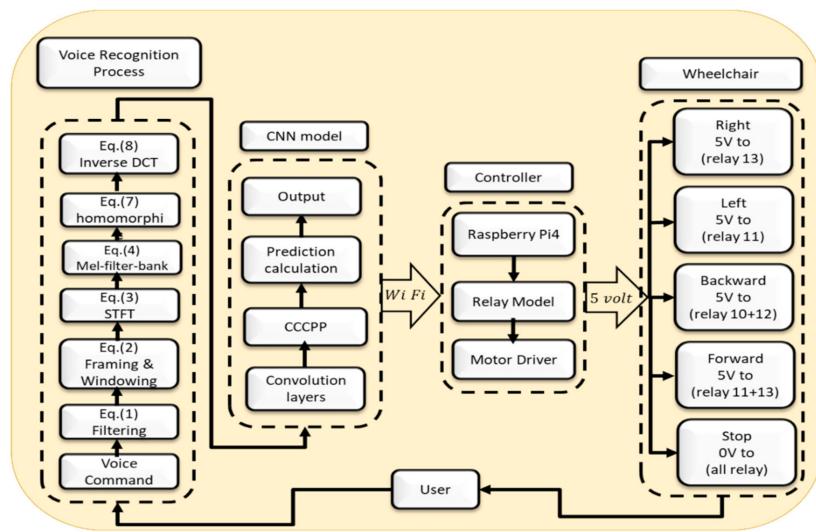


Figure 4. Flowchart of complete system for wheelchair movement.

3. Experimental Procedure

We evaluated our system on the English speech corpus for isolated words, which was conducted at the Health and Basic Sciences Research Center, Majmaah University. A total of 2000 utterances of five words are contained inside this collection, which was created by 10 native Arabic speakers. At a sample rate of 20 kHz and a 16-bit resolution, the corpus was recorded. Then, that data set was augmented by creating extra speech signals using a method of augmentation. The additional data set contains 2000 utterances by changing pitch, speed, dynamic range, adding noise, and forward and backward shift in time. The new dataset (original and augmented) contains 4000 utterances is divided into two parts: a training set (training and validation) with 80% of the samples (3200) and the test set with the remaining 20% of samples (800).

To evaluate the accuracy and the quality of prediction of the proposed system, we calculate the *F-score* as:

$$F-score = 2 \left(\frac{P * R}{P + R} \right) \quad (11)$$

where *P* and *R* represent precision and recall, respectively, and are stated by the following:

$$P = \frac{T_p}{T_p + F_p} \quad (12)$$

$$R = \frac{T_p}{T_p + F_N} \quad (13)$$

Here, T_p is the true positive, F_p is the false positive, and F_N is the false negative.

To evaluate the right prediction of each voice command during the classification, the percentage difference ($\%d$) equation was used as:

$$\%d = \frac{|V_1 - V_2|}{\left(\frac{V_1 + V_2}{2}\right)} * 100\% \quad (14)$$

where V_1 and V_2 represent the first and second observations during the comparison process, respectively.

The method also evaluates the real-time performance of indoor/outdoor navigation. This test (Video S1) describes the indoor/outdoor navigation performance, when the user controlled the wheelchair via voice commands, and the path is around and inside the mosque, with the coordinates 24.893374, 46.614728.

4. Results

In this work, the audio file was recorded and trained for five words to test the application performance until it reached the required prediction ratio. These words were chosen mainly based on the ease of pronunciation and circulation in the Arab countries and the significant variation among each other in the phonemic outlets. Figures 5–9 illustrate the recognizable faces of the voice command “Yes, No, Left, Right, and Stop”. Each figure includes the sound waveform (a) two-dimensional long-term spectrum with frequency band (b), spectrogram (c), and voice command prediction ratio in the mobile app (d). Table 4 summarizes the resizing and normalization phases for each voice command. The program also displays the predicted weight of the spoken word for the user. It is always a one-voice command and has more weight than other words, and this indicates that an incorrect decision cannot be made during the classification process.

Table 4. Resizing and normalization phases for the voice commands.

Voice Command	Yes	No	Left	Right	Stop
Training time after the normalization	0.520 s	0.302 S	0.398 s	0.321 s	0.317
Non-trainable samples	7	9	15	6	13
Total samples	88,347	81,307	74,276	56,348	80,667
Long-term spectrum; frequency range	19.600 kHz	16.650 kHz	21.00 kHz	16.831 kHz	21.264 kHz

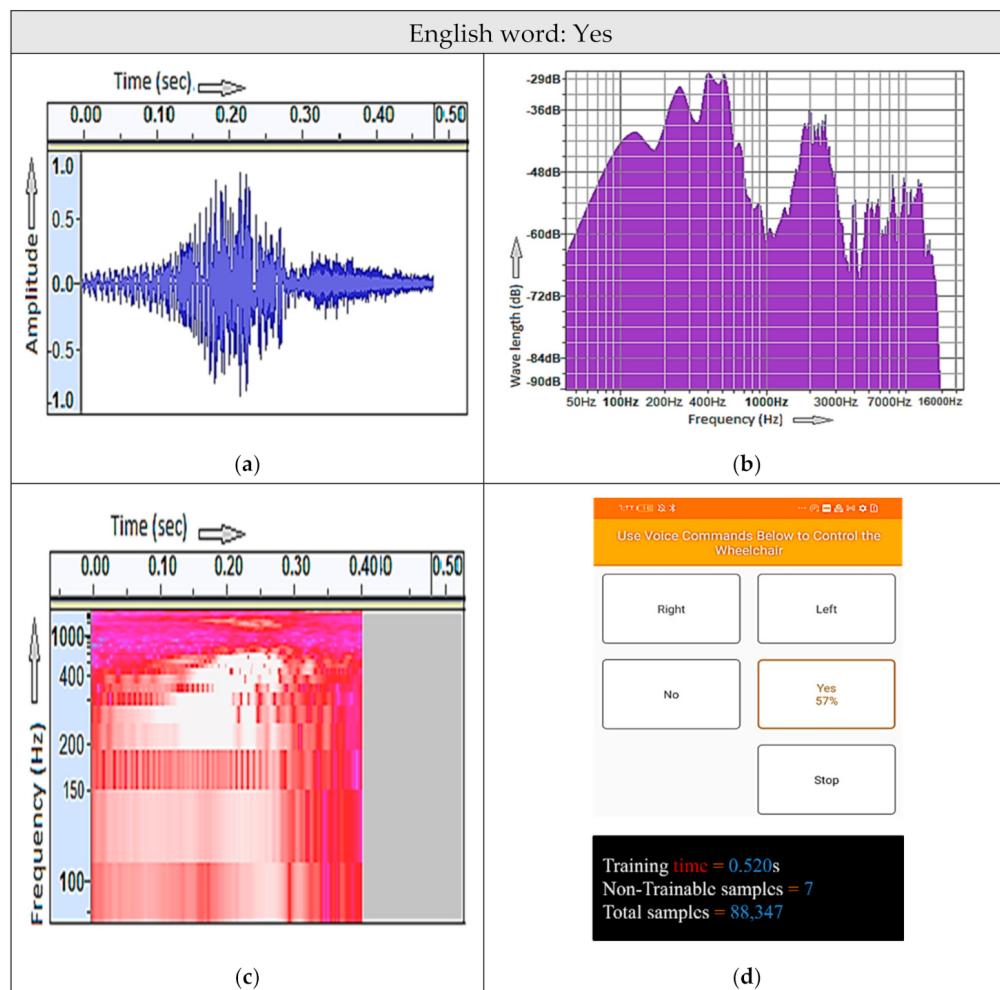


Figure 5. Recognition of the “Yes” voice command. (a) Audio shape; training time = 0.520 s; (b) long-term spectrum; frequency range = 19.6 kHz; (c) spectrogram; (d) screenshot on the mobile app.

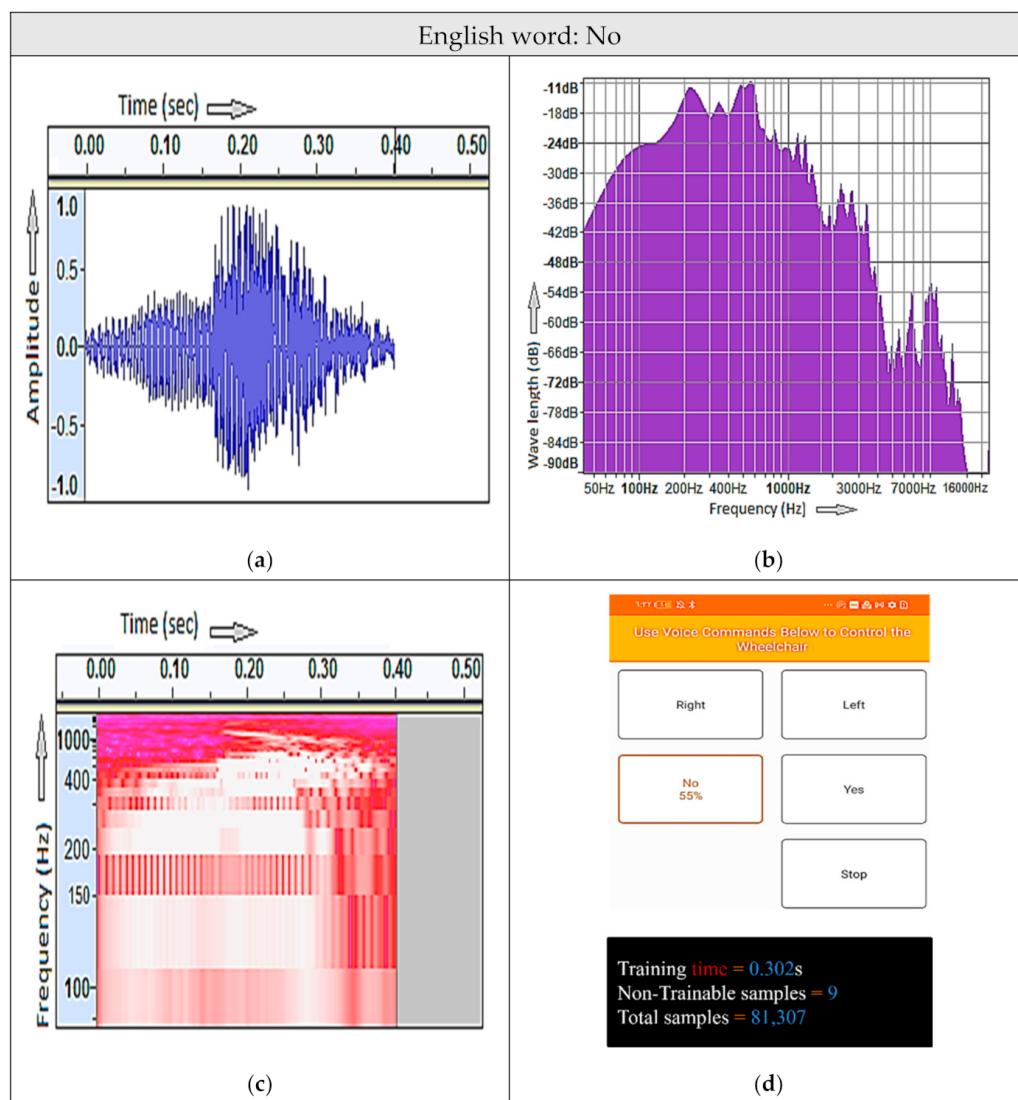


Figure 6. Recognition of the “No” voice command. (a) Audio shape; training time = 0.302 s; (b) long-term spectrum; frequency range = 16.65 kHz; (c) spectrogram; (d) screenshot of the mobile app.

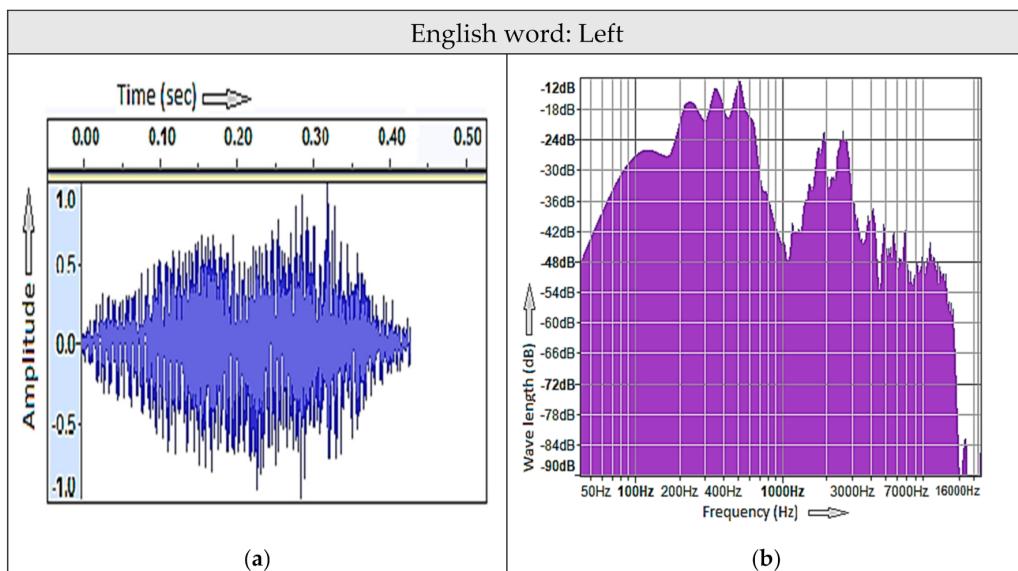


Figure 7. Cont.

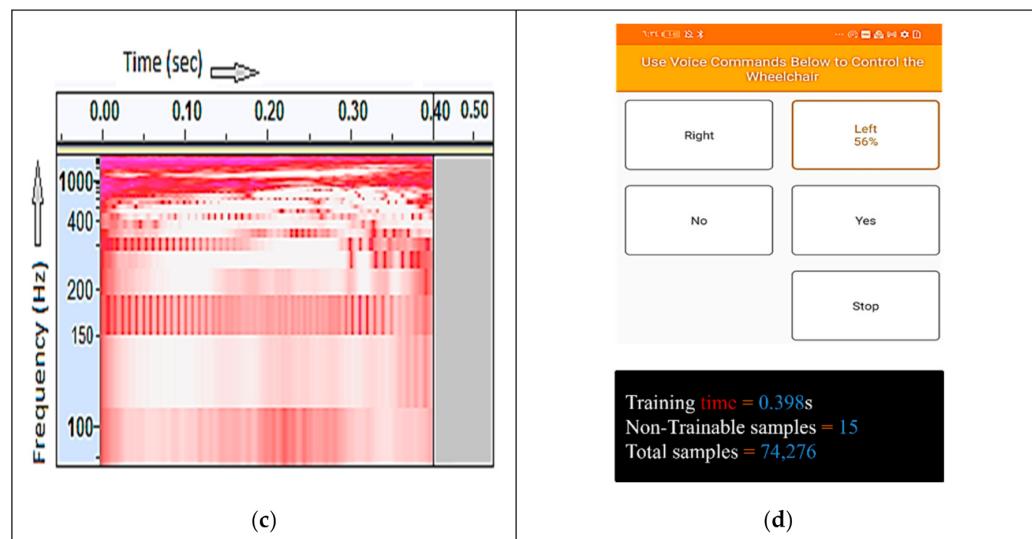


Figure 7. Recognition of the “Left” voice command. (a) Audio shape; training time = 0.398 s; (b) long-term spectrum; frequency range = 21kHz; (c) spectrogram; (d) screenshot of the mobile app.

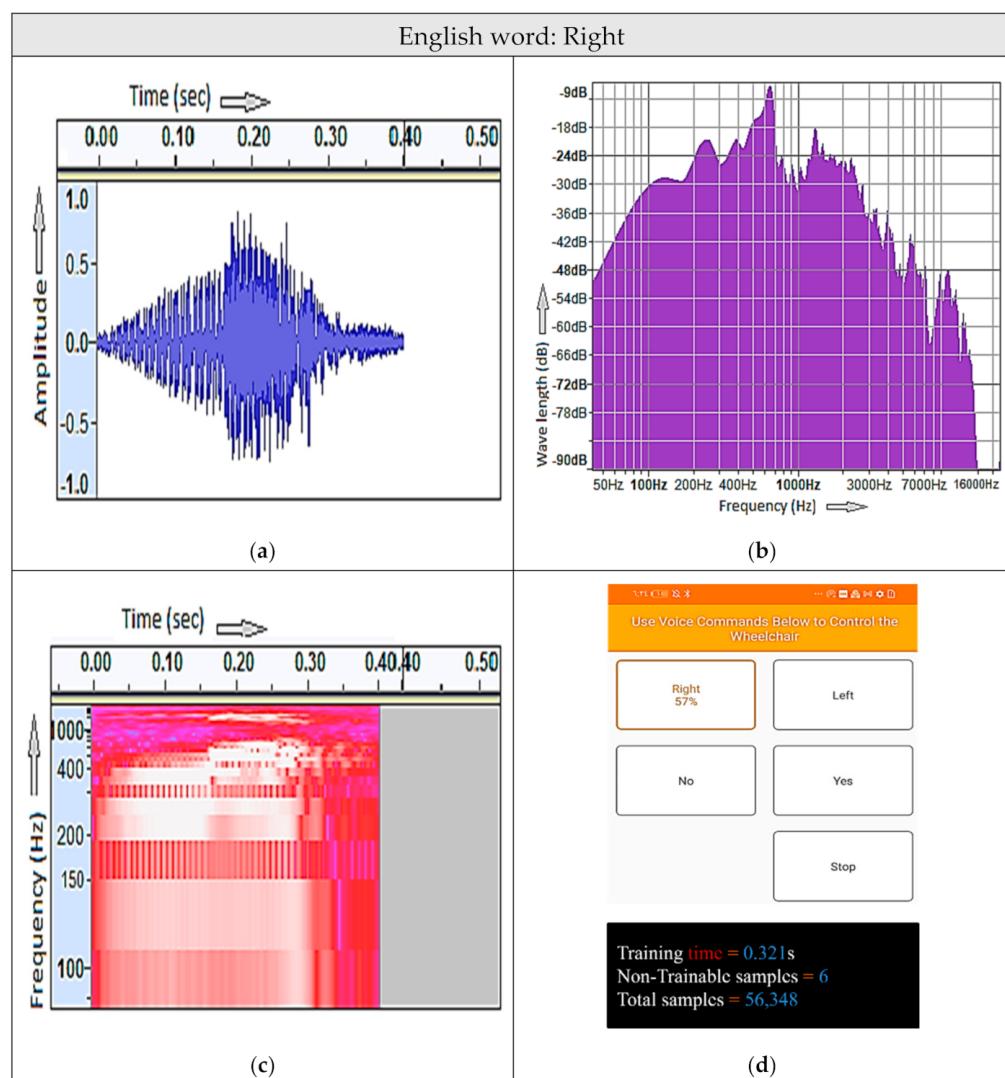


Figure 8. Recognition of the “Right” voice command. (a) Audio shape; training time = 0.321 s; (b) long-term spectrum; frequency range = 16.83 kHz; (c) spectrogram; (d) screenshot of the mobile app.

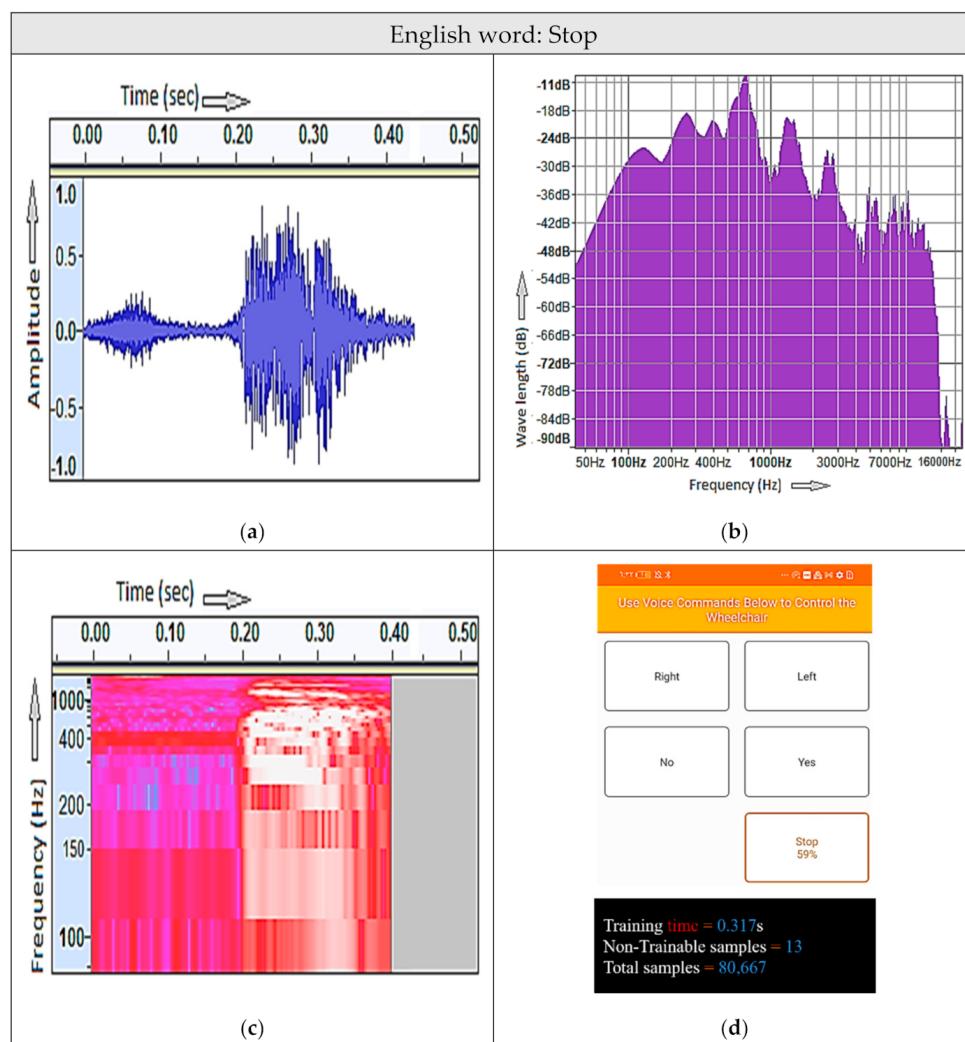


Figure 9. Recognition of the “Stop” voice command. (a) Audio shape; training time = 0.317 s; (b) long-term spectrum; frequency range = 21.26 kHz; (c) spectrogram; (d) screenshot of the mobile app.

Based on the previous results, the confusion matrix was calculated as shown in Table 5. The accuracy of voice command “yes” was approximately 87.2% of the true prediction for the five voice commands. Regarding the classification tasks, we adopted the terms—true positives, true negatives, false positives, and false negatives. Tables 6 and 7 present the calculations of the voice-command prediction ratio, accuracy, and precision. In terms of calculating percentage difference when comparing one command with other commands, the example displays “STOP” against other commands. This indicates a slight possibility of making an incorrect choice during the classification. The difference between the percentage of true and false predictions is markedly high, which indicates a negligible probability of making wrong predictions, and the difference reached more than 150%, as presented in Table 7.

Table 5. Normalized confusion matrix.

		Actual Voice Command					
		Class	Yes	No	Left	Right	Stop
Prediction ratio %	Yes	57%	7%	19%	9%	8%	
	No	6%	55%	18%	12%	9%	
	Left	4%	11%	56%	16%	13%	
	Right	5%	4%	23%	57%	11%	
	Stop	6%	18%	11%	6%	59%	

Table 6. Accuracy, precision, recall, and F-score of voice commands.

Class	Accuracy	Precision	Recall	F-Score
Yes	87.2%	0.57	0.73	0.64
No	83%	0.55	0.58	0.56
Left	77%	0.56	0.44	0.49
Right	82.8%	0.57	0.57	0.57
Stop	83.6%	0.59	0.59	0.59

Table 7. Percentage difference between “stop” command and other commands.

Voice Command	Yes	No	Left	Right	Stop
Prediction ratio	8%	9%	13%	11%	59%
Percentage difference	152%	147%	127%	137%	—

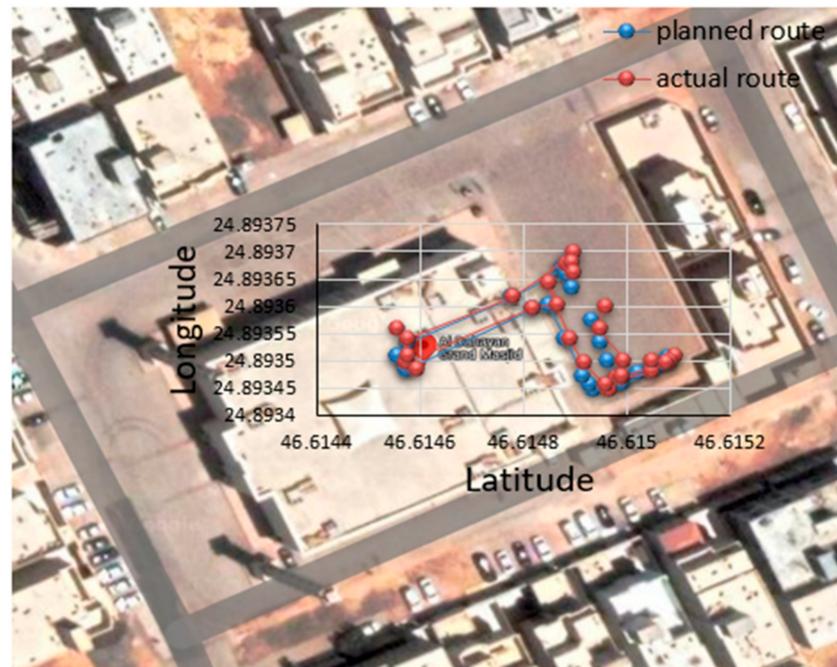
The real-time performance of indoor/outdoor navigation was evaluated in public places, as shown in Figure 10, which presents the planned route navigation versus the actual route (outbound navigation). Table 8 presents the coordinate nodes of the planned and actual paths while navigating. The root-mean-square deviation (RMSD) was adopted to represent the differences between the planned and actual nodes of this experiment. RMSD appears to be equal to 1.721×10^{-5} and 1.743×10^{-5} for latitude and longitude coordinates, respectively.

Table 8. Navigation planned path compared with the actual path (outdoor navigation).

Coordinate	Planned Longitude	Planned Latitude	Actual Longitude	Actual Latitude
Go	24.89337	46.614728	24.893383	46.61478
	24.89357	46.61493	24.893598	46.61496
	24.89353	46.614947	24.89356	46.614947
	24.8935	46.614963	24.8935	46.614995
	24.89347	46.61498	24.893468	46.61498
Right	24.89347	46.614989	24.893469	46.614989
Go	24.89348	46.61502	24.89348	46.61505
	24.89349	46.615048	24.893498	46.615048
	24.89351	46.61508	24.893508	46.615092
Right	24.8935	46.615083	24.8935	46.615083
Go	24.89349	46.615056	24.893497	46.615076
	24.89346	46.61499	24.89347	46.614991
	24.89345	46.614936	24.893448	46.614966
Right	24.89346	46.614928	24.893457	46.614958
Go	24.89347	46.614915	24.893496	46.614918
	24.89354	46.614881	24.893541	46.614891
	24.8936	46.614847	24.893642	46.614847
Left	24.8936	46.614847	24.893602	46.614867
Go	24.8936	46.614818	24.893597	46.614819
	24.89348	46.614572	24.893495	46.614572
Left	24.89348	46.614572	24.893483	46.614592
	24.89351	46.614557	24.893558	46.614557
Right	24.89351	46.614557	24.893509	46.614578
Go	24.89353	46.614578	24.89354	46.614578
	24.89362	46.614776	24.893619	46.614779
	24.89367	46.614877	24.893698	46.614897

Table 8. Cont.

Coordinate	Planned Longitude	Planned Latitude	Actual Longitude	Actual Latitude
Right	24.89367	46.614877	24.893675	46.614882
Go	24.89366 24.89363	46.614884 46.614895	24.89368 24.89366	46.614894 46.614898

**Figure 10.** Navigation planned route versus actual route.

5. Discussion

The objective of this study was to design and implement a low-cost and powerful system to drive a powered wheelchair system using a built-in voice recognition app on a smartphone. This design was achieved to facilitate substantial independence among disabled people and, consequently, improving their quality of life. The proposed design of the smart wheelchair increases the capabilities of the conventional joystick-controlled design by introducing novel smart control systems, such as voice recognition technology and GPS navigation systems. Owing to the significant advancements in smartphones, accompanied by high technology for voice recognition and the use of wireless headphones, the voice recognition technology for controlling wheelchairs has become widely adopted [4,29,30].

In general, the proposed system is characterized by the ease of installing the proposed electrical and electronic circuits, along with low economic cost and low energy consumption. Figure 4 shows a simple structure of the electronic circuit connection inside the installed protection case. The design used is highly effective and low cost in terms of the materials and techniques used and their ability to be configured, customized, and subsequently transferred to the end-user. The average response time for processing a single task is approximately 0.5 s, which is sufficient to avoid accidents. All programs and applications in this smart wheelchair can operate offline without requiring access to the Internet. In addition, the proposed program works under conditions of external noise with high accuracy.

This study investigated the robustness of the voice recognition model by examining the percentage difference between true words, predictions, and false predictions. The experimental results exhibited a significantly high difference between the percentage values

of different categories, which indicates a very low probability of wrong predictions. According to Table 3, the difference between the true and false predictions was approximately over 150%. The second experiment was adopted to evaluate the performance of indoor and outdoor navigation. The user controlled the chair via voice command, and the RMSD was employed to represent the errors in navigation.

In general, the technology of the speech recognition module in Android has become widely used in recent years. In this regard, there are many free or commercially online license software available in the market and suitable for our proposed model, such as Google Cloud Speech API, Kaldi, HTK, and CMUSphinx [37–39]. However, wheelchairs require more studies in terms of static, motion, and moment of inertia. These studies make the system more suitable for different users. In addition, the current voice recognition model did not implement a speaker identification algorithm. Identifying a speaker could improve the safety of wheelchair users by only accepting specific instructions from the authorized person.

Comparing our study with others in terms of efficacy, reliability, and cost, we believe that our design has overcome many complexities. For example, a recent study conducted by Abdulghani et al. implemented and tested an adaptive neuro-fuzzy control to track powered wheelchairs based on voice recognition. To perform a robust accuracy, the design needs to implement a wireless network where the wheelchair is considered a node within the network. Furthermore, this controller is dependent on real-time data obtained from obstacle avoidance sensors and a voice recognition classifier to function appropriately and efficiently [28]. A different study used an eye and voice-controlled human-machine interface technique to drive a wheelchair. In this technique, the authors incorporated a voice-controlled mode with a web camera to achieve congenial and reliable performance for the controller, in which this camera was used to capture real-time images [22].

6. Conclusions

In this study, a low-cost and robust method was used for designing a voice-controlled wheelchair and subsequently implemented using an Android smartphone app to connect microcontrollers via an offline Wi-Fi hotspot. The hardware used in this design consisted of an Android smartphone (Huawei Y9: CPU—Octa-core, 4 × 2.2 GHz), DC electric motors, batteries, relay model, Raspberry Pi4, and an emergency push button in case of an abnormal system response. The system controlled the wheelchair via a mobile app that was built based on Flutter software. A built-in voice recognition model was developed in combination with the CNN model to train and classify five voice commands (yes, no, left, right, and stop).

The experimental procedure was designed and implemented with a total of 2000 utterances of five words that were created by 10 native Arabic speakers. The maneuverability, accuracy, and performance of indoor and outdoor navigation were evaluated in the presence of various disturbances. Normalized confusion matrix, accuracy, precision, recall, and F-score of all voice commands were calculated. Results obtained from real experiments demonstrated that the accuracy of voice recognition commands and wheelchair maneuvers was high. Moreover, the calculated RMSD between the planned and actual nodes at indoor/outdoor maneuvering was shown to be accurate. Importantly, the implemented prototype has many benefits, including its simplicity, low cost, self-sufficiency, and safety. In addition, the system has an emergency push button feature to ensure the safety of the disabled individual and the system.

7. Future Work

The system can be adapted with GPS location technology, and the user can use this technology to create their own path, i.e., building a manual map. By introducing ultrasound sensors for safety purposes, this system will activate and ignore the user's command if the chair arrives near an obstacle that could lead to an accident. In addition, we were able to investigate the users' preference for a voice-controlled interface against a brain-

controlled interface. Moreover, a speaker identification algorithm can be added to the voice recognition model to ensure the safety of the disabled person by only accepting commands from a specific user.

Supplementary Materials: The following are available online at <https://www.mdpi.com/article/10.3390/electronics11010168/s1>, Video S1: The recorded video for indoor/outdoor maneuvering test.

Author Contributions: Conceptualization, M.B.; formal analysis, M.B. and Y.A.; funding acquisition, M.B., K.A., H.F.I. and A.G.; methodology, M.B.; software, H.F.I. and M.A.; validation, M.A. and H.F.I.; visualization, A.G., K.A., A.A. and Y.A.; writing—original draft, M.B.; writing—review and editing, all authors equally contributed to this. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the deputyship for Research and Innovation, Ministry of Education, Saudi Arabia, grant number IFP-2020-31.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data was generated during the study at Health and Basic Sciences Research Center, Majmaah University.

Acknowledgments: The authors extend their appreciation to the deputyship for Research and Innovation, Ministry of Education, Saudi Arabia for funding this research work through grant number IFP-2020-31.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Vignier, N.; Ravaud, J.F.; Winance, M.; Lepoutre, F.X.; Ville, I. Demographics of wheelchair users in France: Results of national community-based handicaps-incapacités-dépendance surveys. *J. Rehabil. Med.* **2008**, *40*, 231–239. [[CrossRef](#)] [[PubMed](#)]
2. Zeng, Q.; Teo, C.L.; Rebsamen, B.; Burdet, E. A collaborative wheelchair system. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2008**, *16*, 161–170. [[CrossRef](#)] [[PubMed](#)]
3. Carlson, T.; Demiris, Y. Collaborative control for a robotic wheelchair: Evaluation of performance, attention, and workload. *IEEE Trans. Syst. Man Cybern. Syst. Part B* **2012**, *42*, 876–888. [[CrossRef](#)] [[PubMed](#)]
4. Pineau, J.; West, R.; Atrash, A.; Villemure, J.; Routhier, F. On the feasibility of using a standardized test for evaluating a speech-controlled smart wheelchair. *Int. J. Intell. Syst.* **2011**, *16*, 124–131.
5. Sharmila, A.; Saini, A.; Choudhary, S.; Yuvaraja, T.; Rahul, S.G. Solar Powered Multi-Controlled Smart Wheelchair for Disabled: Development and Features. *J. Comput. Theor. Nanosci.* **2019**, *16*, 4889–4900. [[CrossRef](#)]
6. Hartman, A.; Nandikolla, V.K. Human-machine interface for a smart wheelchair. *J. Robot.* **2019**, *2019*, 4837058. [[CrossRef](#)]
7. Tang, J.; Liu, Y.; Hu, D.; Zhou, Z. Towards BCI-actuated smart wheelchair system. *BioMed. Eng. OnLine* **2018**, *17*, 111. [[CrossRef](#)]
8. Tomari, M.R.; Kobayashi, Y.; Kuno, Y. Development of smart wheelchair system for a user with severe motor impairment. *Procedia Eng.* **2012**, *41*, 538–546. [[CrossRef](#)]
9. Bourhis, G.; Horn, O.; Habert, O.; Pruski, A. An autonomous vehicle for people with motor disabilities. *IEEE Robot. Autom. Mag.* **2001**, *8*, 20–28. [[CrossRef](#)]
10. Simpson, R.C. Smart wheelchairs: A literature review. *J. Rehabil. Res. Dev.* **2005**, *42*, 423. [[CrossRef](#)]
11. Desai, S.; Mantha, S.S.; Phalle, V.M. Advances in smart wheelchair technology. In Proceedings of the International Conference on Nascent Technologies in Engineering (ICNTE), Vashi, Navi Mumbai, India, 27–28 January 2017; pp. 1–7.
12. Rabhi, Y.; Mrabet, M.; Fnaiech, F. Intelligent control wheelchair using a new visual joystick. *J. Healthc. Eng.* **2018**, *2018*, 6083565. [[CrossRef](#)]
13. Yathunanthan, S.; Chandrasena, L.U.; Umakanthan, A.; Vasuki, V.; Munasinghe, S.R. Controlling a wheelchair by use of EOG signal. In Proceedings of the 4th International Conference on Information and Automation for Sustainability, Colombo, Sri Lanka, 12–14 December 2008; pp. 283–288.
14. Wieczorek, B.; Kukla, M.; Rybarczyk, D.; Warguła, Ł. Evaluation of the Biomechanical Parameters of Human-Wheelchair Systems during Ramp Climbing with the Use of a Manual Wheelchair with Anti-Rollback Devices. *Appl. Sci.* **2020**, *10*, 8757. [[CrossRef](#)]
15. Kukla, M.; Wieczorek, B.; Warguła, Ł. Development of methods for performing the maximum voluntary contraction (MVC) test. *Proc. MATEC Web Conf.* **2018**, *157*, 05015. [[CrossRef](#)]
16. Li, Y.; Pan, J.; Wang, F.; Yu, Z. A hybrid BCI system combining P300 and SSVEP and its application to wheelchair control. *IEEE Trans. Biomed. Eng.* **2013**, *60*, 3156–3166.
17. Hosni, S.M.; Shedeed, H.A.; Mabrouk, M.S.; Tolba, M.F. EEG-EOG based virtual keyboard: Toward hybrid brain computer interface. *Neuroinformatics* **2019**, *17*, 323–341. [[CrossRef](#)]

18. Olesen, S.D.; Das, R.; Olsson, M.D.; Khan, M.A.; Puthusserypady, S. Hybrid EEG-EOG-based BCI system for Vehicle Control. In Proceedings of the 9th International Winter Conference on Brain-Computer Interface (BCI) 2021, Gangwon, Korea, 22–24 February 2021; pp. 1–7.
19. Al-Qays, Z.T.; Zaidan, B.B.; Zaidan, A.A.; Suzani, M.S. A review of disability EEG based wheelchair control system: Coherent taxonomy, open challenges and recommendations. *Comput. Methods Programs Biomed.* **2018**, *164*, 221–237. [[CrossRef](#)]
20. Leaman, J.; La, H.M. A comprehensive review of smart wheelchairs: Past, present, and future. *IEEE Trans. Hum.-Mach. Syst.* **2017**, *7*, 486–499. [[CrossRef](#)]
21. Aktar, N.; Jaharr, I.; Lala, B. Voice recognition based intelligent wheelchair and GPS tracking system. In Proceedings of the IEEE International Conference on Electrical, Computer and Communication Engineering (ECCE), Cox’s Bazar, Bangladesh, 7–9 February 2019; pp. 1–6.
22. Raiyan, Z.; Nawaz, M.S.; Adnan, A.A.; Imam, M.H. Design of an arduino based voice-controlled automated wheelchair. In Proceedings of the IEEE Region 10 Humanitarian Technology Conference (R10-HTC), Dhaka, Bangladesh, 21–23 December 2017; pp. 267–270.
23. Abdulghani, M.M.; Al-Aubidy, K.M.; Ali, M.M.; Hamarsheh, Q.J. Wheelchair Neuro Fuzzy Control and Tracking System Based on Voice Recognition. *Sensors* **2020**, *20*, 2872. [[CrossRef](#)]
24. Anam, K.; Saleh, A. Voice Controlled Wheelchair for Disabled Patients based on CNN and LSTM. In Proceedings of the 2020 4th International Conference on Informatics and Computational Sciences (ICICoS), Semarang, Indonesia, 10–11 November 2020; pp. 1–5. [[CrossRef](#)]
25. Sharifuddin, M.S.I.; Nordin, S.; Ali, A.M. Comparison of CNNs and SVM for voice control wheelchair. *IAES Int. J. Artif. Intell.* **2020**, *9*, 387. [[CrossRef](#)]
26. Huang, J.T.; Li, J.; Gong, Y. An analysis of convolutional neural networks for speech recognition. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), South Brisbane, Australia, 19–24 April 2015; pp. 4989–4993.
27. Korvel, G.; Treigys, P.; Tamulevicius, G.; Bernataviciene, J.; Kostek, B. Analysis of 2d feature spaces for deep learning-based speech recognition. *J. Audio Eng. Soc.* **2018**, *66*, 1072–1081. [[CrossRef](#)]
28. Lee, W.; Seong, J.J.; Ozlu, B.; Shim, B.S.; Marakhimov, A.; Lee, S. Biosignal Sensors and Deep Learning-Based Speech Recognition: A Review. *Sensors* **2021**, *21*, 1399. [[CrossRef](#)]
29. Ali, S.; Al Mamun, S.; Fukuda, H.; Lam, A.; Kobayashi, Y.; Kuno, Y. Smart robotic wheelchair for bus boarding using CNN combined with hough transforms. In Proceedings of the international Conference on Intelligent Computing, Wuhan, China, 15–18 August 2018; pp. 163–172.
30. Martinez-Alpiste, I.; Casaseca-de-la-Higuera, P.; Alcaraz-Calero, J.M.; Grecos, C.; Wang, Q. Smartphone-based object recognition with embedded machine learning intelligence for unmanned aerial vehicles. *J. Field Robot.* **2020**, *37*, 404–420. [[CrossRef](#)]
31. Han, X.; Dai, Q. Batch-normalized Mlpconv-wise supervised pre-training network in network. *Appl. Intell.* **2018**, *48*, 142–155. [[CrossRef](#)]
32. Kuzmin, N.; Ignatiev, K.; Grafov, D. Experience of Developing a Mobile Application Using Flutter. In *Information Science and Applications (ICISA)*; Springer: Singapore, 2020; pp. 571–575.
33. Napoli, M.L. *Beginning Flutter: A Hands on Guide to App Development*; John Wiley & Sons: Hoboken, NJ, USA, 2019.
34. Ghaffar, M.S.B.A.; Khan, U.S.; Iqbal, J.; Rashid, N.; Hamza, A.; Qureshi, W.S.; Tiwana, M.I.; Izhar, U. Improving classification performance of four class FNIRS-BCI using Mel Frequency Cepstral Coefficients (MFCC). *Infrared Phys. Technol.* **2021**, *112*, 103589. [[CrossRef](#)]
35. Alaeddine, H.; Jihene, M. Deep network in network. *Neural Comput. Appl.* **2021**, *33*, 1453–1465. [[CrossRef](#)]
36. Kim, Y.J.; Bae, J.P.; Chung, J.W.; Park, D.K.; Kim, K.G.; Kim, Y.J. New polyp image classification technique using transfer learning of network-in-network structure in endoscopic images. *Sci. Rep.* **2021**, *11*, 3605. [[CrossRef](#)]
37. Matarneh, R.; Maksymova, S.; Lyashenko, V.; Belova, N. Speech recognition systems: A comparative review. *J. Comput. Eng.* **2017**, *19*, 71–79.
38. Anggraini, N.; Kurniawan, A.; Wardhani, L.K.; Hakiem, N. Speech recognition application for the speech impaired using the android-based google cloud speech API. *Telkomnika* **2018**, *16*, 2733–2739. [[CrossRef](#)]
39. Kępuska, V.; Bohouta, G. Comparing speech recognition systems (Microsoft API, Google API and CMU Sphinx). *Int. J. Eng. Res. Appl.* **2017**, *7*, 20–24. [[CrossRef](#)]