Voice Command Intelligent System (VCIS) for Smart Home Application using Mel-frequency cepstral coefficients and linear prediction coefficients

Yusnita Mohd Ali1, Nor Fadzilah Mokhtar1, Emilia Noorsal1, Aida Zulia Zulhanip1, Asmalia Zanal1, Siti Zubaidah Md Saad1 and Nur Aliza Abdul Samad2

1 Faculty of Electrical Engineering, Universiti Teknologi MARA, Cawangan Pulau Pinang, 13500 Permatang Pauh, Pulau Pinang, Malaysia
2 Inari P21, Plot 201, Phase 3, Bayan Lepas FTZ, Lebu Kampung Jawa, Kawasan Perindustrian Bayan Lepas, 11900 Bayan Lepas, Pulau Pinang

*yusnita082@uitm.edu.my, norfadzilah105@uitm.edu.my

Abstract. Voice recognition, command and control systems have played a vital role to facilitate our modern lifestyle and routine operations through hands-free applications. Working people are busy with their routines, simple tasks such as to on/off electrical appliances at home are often overlooked leads to wastage of energy. Besides, the elderly and handicapped often face difficulties to control the electrical appliances due to their physical disability. To overcome, this project aims to develop a voice command intelligent system (VCIS) as an aid to these routine tasks such as to switch on/off a lamp, control a motor and control other modes of operation of home appliances using voice interface. In particular, VCIS involves several development processes such as database collection, pre-processing, feature extraction algorithms and classification modeling for application in smart home. This paper proposes three hierarchical stages in VCIS. The experimental results show that Mel-frequency cepstral coefficients (MFCC) surpassed linear prediction coefficients (LPC) to correlate with the voice commands in all classification stages using artificial neural network. The classification rates were successfully accomplished, 99.12% (MFCC) and 95.23% (LPC). In conclusion, the proposed VCIS can be employed as an efficient method for better quality of life.

1. Introduction

Smart home applications or well known as home automation system (HAS) has become widespread through the advent of computer technologies, devices and sensors such as Internet of Things (IoT) [1]. Three elements that feature a smart home are networking, intelligent control and home automation [2] to control electrical appliances and electronic gadgets at home for making human lifestyle become more sophisticated, efficient and comfortable. Multiple interaction modes are available in HAS such as gesture, eye movement, remote control using joystick, smart phone or other screen based interaction, up to the highest level of human intelligence using voice control. The popularity of voice command interfaces becomes more evident and emerge as a new paradigm in HAS due to vast availability of smart devices in the market, and supported by the online voice recognition engines [3, 4]. Voice interfaces is a natural communication means to aid disabled, elderly, and normal people. Voice commands can ease their activities just by speaking and serve as assistive technology to their physical limitations.
Rizvi et al [5] proposed multiple techniques using GSM for sending short text messages or using android application for long distance, and Bluetooth with android application or Google voice tool for short distance to operate remotely home appliances such as fan, TV and light as assistive system for the elderly, handicapped and blind people. The system was built up using Arduino UNO for controlling the appliances. Another work using voice command to automate the operation of doors and windows was reported in [6] using Google cloud speech API, Bluetooth module HC-05 as communication module between a smartphone and Arduino UNO. Several voice recognition engines are available as compared in [7] such as Jasper platform, Google speech API, Alexa voice service and Bing speech API to integrate with HAS architecture. Similar framework of smart room using AMR voice application was proposed in [8] to control appliances operation through relay and/or infrared emitter based on keyword search to make the system more flexible.

This paper proposes a different framework named as voice command intelligent system (VCIS) for smart home applications based on three-tier classification stage i.e. device selection, power option, and mode of operation for the devices. This project has recorded sufficient database of specific voice commands to be extracted using two feature extractors and a voice command classification algorithm.

2. Speech Database and Experimental Setup

2.1. Database
The speech corpus was recorded from 25 male and 25 female volunteers. Each speaker pronounced ten voice commands. None of them had a history of smoking, voice disorder, allergies and symptoms of cold and flu. Subjects were undergraduate students of Universiti Teknologi Mara (UiTM) Penang Branch aged from 18 to less than 30 years old. The recordings were conducted in a quiet room to get the best quality of speech sounds using a headset microphone and a laptop computer sound card and MATLAB software.

2.2. Speech Commands
The subjects were asked to pronounce ten words commands with loudness level at a comfortable pitch. Sampling rate and bit resolution were set to 16 kHz and 16 bps respectively during recording sessions. Each word was repeated five times by each speaker to increase the number of samples per speaker, to increase precision and to provide an estimate of the experimental error per word. This collections of utterances amounted to 2,500 speech samples recorded from a total of 50 volunteers equally from male and female volunteers. The list of ten command words were All, Motor, Led, Red, Blue, Green, Fast, Slow, On, and Off.

3. Methodology
This section explains on the design and development process of the proposed VCIS. Speech database established for this work has been explained in previous section. This section focuses on extraction of MFCC and LPC feature vectors as input to the neural network model as the classification engine.

3.1. Pre-processing
This process includes normalization, pre-emphasis, frame blocking, overlapping and windowing. Normalization is required to scale the audio vector into a range between -1.0 and +1.0 and to prevent volume differences among speakers [9]. Pre-emphasis filtering is applied to the speech signal with coefficient of 0.9375 [10] to compensate the attenuation using a simple first order high-pass finite impulse response (FIR) filter. The speech was frame blocked into 32 msec short-time frame in order to ensure pseudo-stationary property and then applied with Hamming window function. The complete feature will not be emphasized in one analysis window due to windowing process and the solution for this problem is to apply overlapping between successive frames by 50% hop size [11]. This overlapping will make sure that audio features away from the peak window will be emphasized in the subsequent frame. The function of windowing is to avoid or minimize the spectral distortion.
3.2. Linear Prediction Coefficients
LPC is a technique used in speech analysis in certain manner such that the future speech samples are estimated from a linearly weighted summation of past $p$-samples using method of least squares. LPC analysis determines the coefficients of a forward linear predictor by minimizing the prediction error in the least square sense [12, 13]. The least square function is given in equation (1) to obtain the estimated speech.

$$\tilde{x} = - \sum_{k=1}^{p} a(k)x(n - k) \tag{1}$$

where $x(n)$ and $\tilde{x}$ are speech samples and their estimates, and $a(k) = [a(1),a(2),...a(p)]^T$ is the LPC parameters and $p$ is the linear predictive (LP) filter order. The autocorrelation function (ACF) of each frame can be computed [10] by equation (2) below.

$$r(i) = -\sum_{n=0}^{N-1-i} X^F(n)X^F(n+i). \tag{2}$$

where $r(i) = [r(0),r(1),r(2)...r(p)]^T$ is the ACF of a frame samples denoted as $\tilde{x}(n)$ consists of $N$ sample points. Next, the Yule-Walker equations are solved using the Levinson-Durbin recursive algorithm to obtain the coefficients of the prediction filter as in equation (3).

$$\sum_{k=1}^{p} a(k)R(i - k) = -r(i). \tag{3}$$

where $1 \leq i \leq p$ and $R(i - k)$ forms the ACF matrix ($p$-by-$p$) which is a symmetric Toeplitz matrix. Hence, $a(k)$ can be solved efficiently by taking the inverse of ACF matrix multiplied by ACF vector such that in equation (4).

$$a = -R^{-1}r. \tag{4}$$

3.3. Mel-frequency Cepstral Coefficients
The working principle of MFCC processor is based on a set of filter banks constructed from several band pass filter in a form of triangular-shape window functions [14] using Mel-scale warped frequency domain. The transformation formula from Mel-scale, to Hz is related by equation (5). The center frequencies of the series bandpass filters are designed according to this perceptually motivated scale, the known variation of the human ear’s critical bandwidths.

$$Mel = 2595 \log_{10} \left(1 + \frac{f}{700}\right) \tag{5}$$

Equation (5) reveals that both frequency in Hz and in Mel-scale are almost linearly related below 1 kHz and logarithmically related for frequency above 1 kHz. Mel-scale filtering is used to focus on the lower frequency components that are more important in speech analysis. By summing all the product of log-energy spectrum using Fourier transform from each individual bandpass filter and then applying discrete cosine transform (DCT), the cepstral coefficients of these filter banks are generated as in equation (6).

$$C_m = \sum_{k=1}^{N} E_k \cos[m(k - 0.5)\pi/N] \tag{6}$$
where the variables $C(.)$ and $E(.)$ represent the $m$-th cepstral coefficient and $k$-th log-energy respectively. $N$ is the number of filters in the filter banks and the cepstrum takes in this order, $m = 1, 2, \ldots, M$. The details of MFCC design in this work can be referred to [15]. The slowly varying part of the spectrum were represented by lower order cepstrum while spikes in the series correspond to the harmonic series of the vocal folds. A few lower order coefficients commonly are taken to represent the vocal tract shape, namely $M = 12$ or 13 is sufficient to leave out the pitch property of the speech signal [11].

3.4. Voice Commands Classifier

In this work, artificial neural network (ANN) is used to perform the non-linear task of the voice signal. ANN is learning machines built from many different processing elements. For most classification problems, error back-propagation algorithm is the most popular used to train feed forward Multilayer Perceptron (MLP) [16]. In this project, two multilayer was used to classify the input features. Levenberg-Marquardt was used as learning algorithm to train the network. For binary outputs problems, threshold and margin criterion was used [17]. Mean squared error (MSE) was adopted in this project as an objective criterion for successful learning of the task. The mathematical expression for the activation function is computed in equation (7). In the learning process, the input was normalized between 0.1 to 0.9.

3.5. The Proposed System (VCIS)

Figure 1 shows the block diagram of VCIS. The input has two modes of operation either from real time voice commands or from recorded voice commands database. The input undergoes pre-processing, feature extraction and classification stages. The test signal will be compared with a number of reference models in order to classify which voice command is spoken by the user of the system.

![Block Diagram of VCIS](image)

Figure 1. The block diagram of the proposed Voice Commands Intelligent System (VCIS).

Figure 2 shows a hierarchical three-tier classification proposed in this paper for the users to select voice command words. By default, all devices is in off state. In the first stage, VCIS will ask the user to select between devices: Led, Motor or both (All). In the second stage, the user has to select the power option of the selected appliance(s) either On or Off. In last stage, the user can control the mode of operation of the devices. For motor, user can choose Fast or Slow mode. For Led, user can choose either Blue, Green or Red.
4. Results and Discussion

This section reports the findings of the proposed VCIS performance. Two performance criteria used were accuracy and epoch. The latter is the learning time required to train the dataset for establishing a model until the objective criteria is met. The performance of VCIS are modeled using LPC and MFCC features and ANN as the classifier. Optimal parameter settings were performed in VCIS. Due to the randomness purpose of ANN, the experiments were conducted ten times for each set of parameters. This paper adopted independent test for performance evaluation using 70% of total data samples for training, and the remaining data for testing purposes.

4.1. Varying the number of coefficients in feature extractors

In this experiment, the effect of varying the number of coefficients was investigated by fixing the value of hidden layer to 10. Figure 3 shows the average classification rate (CR) using LPC parameters by varying the number of coefficients from 8 to 20 in each classification stage. In general, it was found that the accuracy increases as the number of coefficients increases. It is found that 16 coefficients yielded the best performance for all classification stages. The improvement was not encouraging after 16 coefficients for all classification stages. Thus, optimum LPC was fixed to order-16 (LPC-16). The average values of CR for overall accuracy in each stage were 95.23% for All_Led_Motor, 94.82% for Blue_Green_Red, 91.93% for Off_On, and 94.29% for Fast_Slow respectively.

The experiment was repeated for MFCC in the same manner and the results are depicted in Figure 4. From this graph, the results yielded 12 coefficients as the best performance for all classification stages. Overall, it was found that the accuracy increases as the number of coefficients increases. However, the performance was almost stagnant and some started to drop after order-12. Thus, optimum MFCC was fixed to order-12 (MFCC-12). Thus, optimum LPC was fixed to order-12 (MFCC-12) were 98.74% for All_Led_Motor, 97.54% for Blue_Green_Red, 98.53% for both Off_On and 99.12 for Fast_Slow stages.
4.2. Varying the number of hidden neurons in classifier

Next, the effect of varying the number of hidden neurons in ANN was investigated by fixing the order of LPC and MFCC to 16 and 12 respectively. Figure 5 shows the performance of LPC-16 for number of neurons varied from 5 to 30. Overall, the best results were obtained at 10 hidden neurons and mostly started to drop at higher number of hidden neurons. For instance in Off-On classification stage, its performance dropped significantly. The epoch became almost doubled for hidden neurons higher than 20. As for MFCC-12, the optimal number was found to be 30 in general for all classification stages as depicted in Figure 6. In addition there was insignificant difference in the number of epoch as we increased the hidden neurons. Thus the hidden neurons were fixed at 10 and 30 for LPC and MFCC respectively.

The performance of LPC and MFCC was compared in Figure 7. It was found that MFCC features outperformed LPC features in all classification stages by average CR of 2.8% to 6.7%. Furthermore MFCC required less epoch by 17 to 45 epochs to train the ANN models compared to LPC as depicted in Figure 8 to make the earlier more accurate and faster to be used as VCIS models.
Figure 5. LPC-16 performance for different hidden neurons.

Figure 6. MFCC-12 performance for different hidden neurons.

Figure 7. Comparison of performance of VCIS using LPC and MFCC in terms of accuracy.
Figure 8. Comparison of performance of VCIS using LPC and MFCC in terms of epoch.

5. Conclusions
This paper has presented a three-stage classification which includes ten voice commands from male and female speakers using LPC and MFCC as features and ANN for classification algorithm. It is found that MFCC outperformed LPC features by 2.8% to 6.7% of accuracy rate in all classification stages. The time of training the models using MFCC was also faster than LPC by the measured epoch between 17 and 45 epochs. In short, the performance of MFCC with ANN classifier provides highly promising results up to 99.12%. These results conclude that VCIS has been successfully modeled using three-stage classification method and can be further tested in embedded systems.

References
[1] Hamdan O, Shanableh H, Zaki I, Al-Ali A R and Shanableh T 2019 IEEE Int. Conf. on Consumer Electronics (ICCE) (Last Vegas) p 1
[2] Mittal Y, Toshniwal P, Sharma S, Singhal D, Gupta R and Mittal V K 2015 Annual IEEE India Conf. (INDICON) (New Delhi) p 1
[3] Arpipin N and Othman M B 2014 Electrical Power, Electronics, Communicatons, Control and Informatics Seminar (EECCIS) (Malang) p 142
[4] MilivojÅ¾a S, IvanoviÄ‡ S, EriÄ‡ T, AntiÄ‡ M and SmiljkoÄ‡ N 2017 Implementation of voice control interface for smart home automation system IEEE 7th Int. Conf. on Consumer Electronics (Berlin, Germany) pp 263-4
[5] Rizvi S, Sohail I, Saleem M M, Irtaza A, Zafar M and Syed M 2018 A Smart Home Appliances Power Management System for Handicapped, Elder and Blind People 4th Int. Conf. on Computer and Information Sciences (Kuala Lumpur, Malaysia) pp 1-4
[6] Pakpahan R L H, Sudiharto D W and Satwiko A G P 2016 The prototype of automated doors and windows by using voice commands Int. Seminar on Application for Technology of Information and Communication (Semarang, Indonesia) pp 323-6
[7] EriÄ‡ T, IvanoviÄ‡ S, MilivojÅ¾a S, MatiÄ‡ M and SmiljkoÄ‡ N 2017 Voice control for smart home automation: Evaluation of approaches and possible architectures IEEE 7th Int. Conf. on Consumer Electronics (Berlin, Germany) pp 140-2
[8] Nath P and Pati U C 2018 Low-Cost Android App Based Voice Operated Room Automation System 3rd Int. Conf. for Convergence in Technology (Pune, India) pp 1-4
[9] McLoughlin I 2009 *Applied Speech and Audio Processing* (New York: Cambridge University Press)
[10] Rabiner L and Juang B H 1993 *Fundamentals of speech recognition* (New Jersey: Prentice Hall) vol 103
[11] Yusnita M A, Paulraj M P, Sazali Y, Shahriman A B and Fadzilah M N 2013 Statistical Band Selection for Descriptors of MBSE and MFCC-based Features for Accent Classification of Malaysian English *J. Electrical Electron. Syst. Res. (IEESR)* 6 31-46
[12] Yusnita M A, Paulraj M P, Yaacob S, Shahriman A B and Saidatul A 2011 Malaysian English accents identification using LPC and formant analysis *IEEE Int. Conf. on Control System, Computing and Engineering (ICCSCE)* (Penang, Malaysia) pp 472-476
[13] Makhoul J 1975 Linear prediction: A tutorial review *Proc. of the IEEE* vol 63 pp 561-580
[14] Davis S and Mermelstein P 1980 Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences *IEEE Trans. Acousti., Speech Signal Process.* 28 357-66
[15] Yusnita M A, Paulraj M P, Yaacob S, Shahriman A B and Yusuf R 2012 Analysis of Accent-sensitive Words in Mel-Frequency Cepstral Coefficients for Speaker Accent Identification of Malaysian English *4th Int. Conf. on Noise, Vibration and Comfort (NVC2012)* (Kuala Lumpur, Malaysia) pp 120-5
[16] Priddy K L and Keller P E 2005 *Introduction to Artificial Neural Networks* (San Francisco: SPIE Press) vol TT68
[17] Fahlman S E (1988) An empirical study of learning speed in back-propagation networks Retrieved from https://pdfs.semanticscholar.org/e843/7b98079a58473b0cc1858a0f98dfecc4ea8f.pdf