HMM-based phoneme speech recognition system for control and command of industrial robots

Adwait Naik
(K J Somaiya College of Engineering, Mumbai, India)
adwaitnaik2@gmail.com

Abstract- Speech recognition is a prominent technology, which helps us to develop a Natural language interface through speech for the Human-Robot interaction (HRI). It allows the computer to take the spoken instructions, interpret it, and generate text from it. In this paper, we propose a phoneme based speech recognition system to control industrial robots. Speech recognition has become one of the popular interfaces when it comes to reducing robot operator's efforts to control and command the robot. This paper intends to investigate the potential of Linear Predictive coding technique to develop a stable and robust phoneme speech recognition system for robotics applications. Our system is divided into three segments: a microphone array, a voice module, and a 3-DOF robotic arm. To validate our approach, we have performed tests with simple and complex sentences for various robotics activities like manipulating a cube and pick and place tasks. Moreover, we also analyzed the test result to rectify the problems and limitations in our approach. The paper presents all the test results which we have achieved through conducting experiments on our project.

Keywords - Speech recognition, Natural language, phoneme, Voice User Interface, robotics Human-Robot interaction (HRI), Linear Predictive Coding (LPC)

Figure. 1 A 3-DOF robotic arm

1. INTRODUCTION

An industrial robot is a robot system primarily used for welding, painting, assembly, and pick and place tasks. In the context of general robotics, most types of robots would fall into the category of robotic arms exhibiting varying degrees of autonomy. For an industrial robot, control is the most challenging part [1, 2]. Conventional methods to control the robots include linking the robot controller to a laptop or desktop computer. Teaching pendant and offline programming are also used extensively in the industries. These methods come with trade-offs like reduced accuracy, low endurance, and improper material handling. On the contrary, modern techniques like voice and gesture control give better accuracy, high precision, and greater compliance .

Although there is a lot of research taking place in the field of human-robot interaction, the smoothness of the interaction remains the biggest challenge to be achieved. Presently, it has not reached the expected level but the robots can recognize the voice commands given by the operator [2, 3].

Honda's ASIMO, a humanoid robot, is a perfect example of the recent breakthrough in the field of speech recognition. It is equipped with the ability to understand at most three human operators giving commands at a time, thanks to HARK software. The HARK system uses speech isolation technique to separate individual voices, before passing it onto speech-recognition software to decode. The array of 8 microphones help ASIMO accurately detect and isolate simultaneous voices.

Eliminating background noise is another challenge in this field. To solve this problem, the researchers developed a machine learning framework that decomposes the processed sound into sets of frequencies using neural networks. One of the best features of this system is, minimized reverberation generated by the robot's joints and motors while performing tasks involving high payloads [3, 4, 5]. This dictates the requirement of using Automatic Speech Recognition (ASR) to lessen the effect of background noise.

Existing speech recognition systems require sophisticated hardware and hundreds of lines of code for implementation which is time-consuming. In comparison to these approaches, our approach is simple, easy-to-implement, and economically feasible. It requires simple hardware that is readily available and easy to train.

This project is developed with the objective of:

- Achieving wireless control over the robotic arm using voice commands to perform the pick and place operation.

- Perform tasks with greater accuracy, higher precision, and better compliance.

- Eliminate challenges like background noise by ensuring smooth interaction between the robot and the operator.

The remainder of this paper is organized as follows. Section 2 discussed the System design and Phoneme extraction process. In section 3, the training and model deployment is outlined. Section 4 describes the testing and hardware deployment stage. Section 5 summarises the robot actuation process. Section 6 describes the experimental results followed by section 7 which summarizes the observation. Section 8 concludes the manuscript.

2. SYSTEM DESIGN

The speech recognition process is divided into two stages: Training is the initial stage and Recognition is the final stage.

2.1 TRAINING STAGE

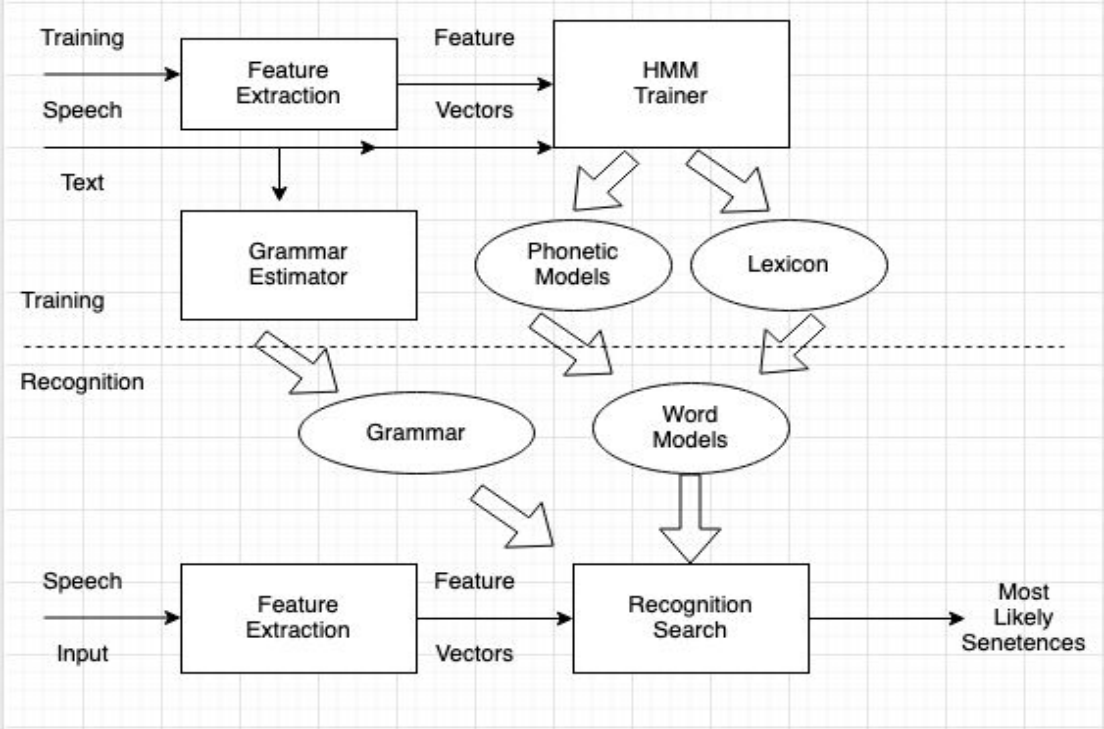

Figure. 2 Speech recognition process (Makhoul et al. 1995)

The training stage includes corpus preparation, extracting the feature vectors to train the Hidden Markov Model(HMM) as illustrated in Figure. 1.

2.2 DATA PREPARATION

The corpus is a dataset of commands recorded to train the (HMM) trainer. Commands used during training are shown in Table. 1 below.

Table 1 List of commands

	Primitive Commands for robot control	**Description**
1	**Start**	This command initiates the movement in the servo motor attached to the base.
2	**Stop**	This command halts the movement of the servo motor attached to the base
3	**Rotate the base clockwise**	This command rotates the servo motor attached to the base by 180 degrees in the clockwise direction
4	**Rotate the shoulder**	This command rotates the servo motor attached to the shoulder by 180 degrees in the clockwise direction
5	**Open gripper**	This command rotates the servo motor attached to the gripper in the clockwise direction thereby opening the gripper.
6	**Close gripper**	This command rotates the servo motor attached to the gripper in an anticlockwise direction thereby closing the gripper.
7	**Lift the shoulder up**	This command rotates the servo motor attached to the shoulder by 90 degrees upwards.
8	**Put the shoulder down**	This command rotates the servo motor attached to the shoulder by 90 degrees downwards.

The text corpus is a collection of eight primitive commands used to control the robot. For each command, a separate program had to be written in C++ which was fed to the Arduino manager to be transferred to the Arduino microcontroller to control the servo motors.

2.3 RECORDING THE COMMANDS

The text corpus was used for recording the commands using 2 speakers at a dual-frequency channel of 44.1 kHz at -6db peak and 256 Kbps data rate using the Audacity® software which is an open-source application used for digital audio editing, mixing, and recording. Commands were recorded in three groups, each having four commands. The recording was exported in .wav format as shown in Figure. 3 below.

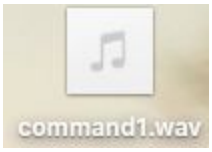

Figure. 3 recorded command in .wav format

To build models using the recorded commands in .wav format Hidden Markov Model Toolkit (HTK) software was used.

2.4 FEATURE EXTRACTION

Feature extraction is an essential step in the process of speech recognition as it segregates the speaker's voice from all other voices and generates observational vectors. It reduces the magnitude of the speech signal responsible for causing damage to the power of the speech signal [5]. In this case, the input given is an audio signal. Various techniques for feature extraction like MFCC, Linear Predictive coding, FFT, and RASTA. We preferably used the Linear Predictive Coding (LPC) technique as shown in Figure. 4 below, for the following advantages:

- High computation speed and robustness [7].
- Bit rate requirement is less for transmission [7, 9].

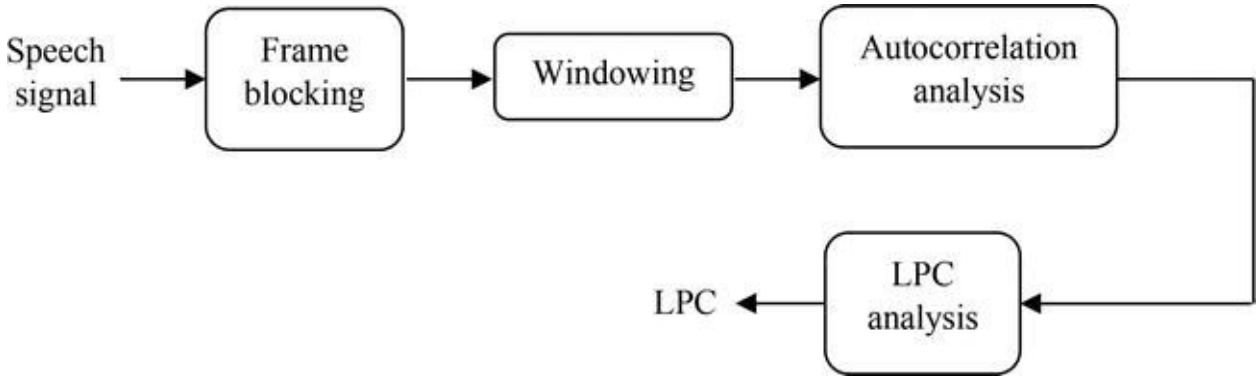

Figure. 4 Block diagram for the LPC technique (Sabur Ajibola Alim et al. 2018)

2.4.1 LINEAR PREDICTIVE CODING

Linear prediction coding, in its operation, resembles the human vocal tract [7]. Also known as a format estimation technique [8, 9], it is used to estimate the formants and reduce their effects on the signal. Here, the formants are peaks or local maximum occurring in the spectrum as a result of resonance. The frequencies where the formants appear are defined as formant frequencies. The location of the formats in the spectrum can be deduced by calculating the linear predictive coefficients [9, 10].

LPC is based on the principle of reducing the mean square error (shown in Equation 2) between the input speech and estimated speech [8]. The speech sample at any time interval is expressed as a linear weighted aggregation of preceding samples [9]. The linear predictive model of speech creation is given as [10]:

$$\hat{S}(n) = \sum_{k=1}^{p} a_k s(n-k) \quad (1)$$

where $\hat{S}$ is the predicted sample, s is the input speech sample, and p is the predictor coefficients.

The prediction error is given as [7]:

$$e(n) = S(n) - \hat{S}(n) \quad (2)$$

After the speech signal is pre-processed, it is passed for frame blocking as shown in the block diagram in Figure. 4 above. Each frame is autocorrelated and the highest autocorrelation value is chosen for the linear predictive analysis [11, 12]. In linear predictive analysis, the coefficients are calculated which are given by [7, 8]:

$$a_m = (\log[1-k_m] / \log[1+k_m]) \quad (3)$$

where a_m is the linear prediction coefficient and k_m is the reflection coefficient.

Furthermore, LPC is used to very accurately estimate the vocal tract properties from the speech signal and hence is a very effective technique employed in the tonal analysis of string instruments like violin and guitar [8].

3. TRAINING HMMs

In this stage, to create the final trained model, we developed the speech recognition system using the Hidden Markov Model-based toolkit HTK Version 3.2 in the Mac OS (Official site of HTK toolkit, htk.eng.cam.ac.uk).

Table 2 Parameters for feature extraction using LPC

S. No	Parameters	Value of parameters
1	Features Extracted	LPC
2	Window used	Hamming
3	Window length	15 ms
4	Frame count	12
5	Pre-emphasis (Pre-processing)	0.67
6	Number of coefficients (a_k)	16
7	linear prediction cepstral coefficients	28

The parameters listed in Table 2 were calculated during the feature extraction from the speech signal using the LPC process [10, 11].

3.1 MODEL TRAINING AND DEPLOYMENT

In HTK, **HRest**, **HInit**, and **HERest** programs as shown in Figure. 5 are used to make the acoustic model [16]. The parameters are initialized by the HInit program using the Viterbi Extraction algorithm. HRest estimates the parameters shown in Table 2 above, using the Baum-Welch algorithm [16]. On comparing the performance HRest is outperformed by HERest in a noisy environment [16].

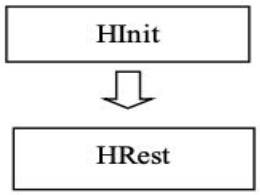

Figure. 5 HTK programs for model creation

4. TESTING

The **HVite** program was used for testing the recorded commands. It uses the Token passing algorithm to perform offline testing using the recorded database. HVITE takes as input a network describing the allowable word sequences, a dictionary defining how each word is pronounced and a set of HMMs [16, 17, 18].

4. HARDWARE

After training the Hidden Markov Model, it was deployed on the Geetech speech recognition module as shown in Figure.5 (Official site Geetech module).

Figure. 6 speech recognition module

The voice module as shown in Figure. 5, is one of the key components of this system. It works on the principle of serial data transfer when connected to the Arduino board. Equipped with a Digital Signal Processor of SC57X series based on SHARC (Super Harvard Architecture Single-Chip Computer) architecture. It comes with the ARM® Cortex-A5 system control capability, which provides high performance for complex applications demanding the latest advanced algorithms.

5. ROBOT ACTUATION AND RECOGNITION

In Figure. 7 speech recognition processes are shown. When the speaker gives a command, for example - “Open Gripper”, every phoneme in this command is isolated and matched with the commands used to train the Hidden Markov Model. If the command matches then the servo motor attached to the particular joint viz. Shoulder, elbow or the gripper actuates as a result of which the movement takes place [18].

The comparison stage contains the trained model that is responsible for carefully matching the commands given by the speaker. Based on the parameters calculated in Table 2, the decision by comparison stage is made. In case of mismatch, background noise, and wrong pronunciation of command the speech recognition system fails to identify the phonemes in the command, as a result, the robot is not actuated [18].

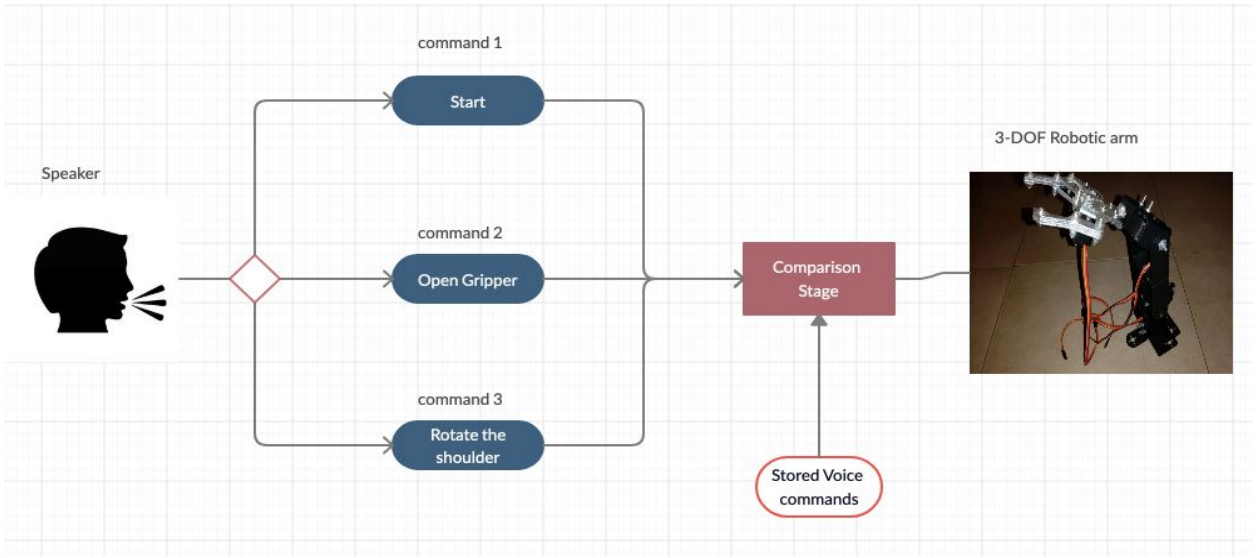

Figure. 7 Speech recognition to control the robot

As mentioned above, a program in C++ was designed for each command listed in Table 1 above. The commands are used to form functions, to which the parameters like time, angle, and the

number of steps are passed; so when the program is executed, the speech signal actuates the servo motor attached to the respective joint. Here the commands are allotted transmission channels on the Arduino board, for example- Channel 1 is designated for "Start" command, Channel 2 is designated for "Stop" command, etc.

6. EXPERIMENTAL RESULTS

The training and testing for the speech recognition system were done using the commands recorded. The database used for testing the speech recognition system consists of 200 speech samples from different speakers involved in the testing process. As mentioned above, the Linear Predictive Coding (LPC) technique used in double differentiation mode. Here, we have used the HTK toolkit to train the acoustic model and deploy it.

In the experiments, we have evaluated Word Error Rate, Recognition Score, and Word Accuracy Rate which are shown in Table 3 below.

Word Error Rate:

Word Error Rate (WER) is the measure of the difference between the recognized word sequence and input word sequence. It is the most commonly used performance metrics for speech recognition systems. Its computation is based on the Levenshtein distance. WER is calculated on the Phoneme level. WER is given by

$$WER = (S + D + I) / N \quad (4)$$

Where *S* is the number of substitutions,

- *D* is the number of deletions,
- *I* is the number of insertions,
- *C* is the number of correct words,
- *N* is the number of words in the reference (N=S+D+C)

Word Accuracy Rate:

Word Accuracy Rate (WAcc) is the percent word accuracy is defined as %WAcc = 100 - %WER. It should be noted that the word accuracy can be negative. WAcc is given by

$$WAcc = (N - S - D - I) / N = (H - I) / N \quad (5)$$

Where H = No. of words that are correctly recognized.

Table 3 WER and WAcc calculation

Technique Used		**Rate in %**	
Linear Prediction Coding	***Word Error Rate***	***Word Accuracy Rate***	***Recognition Score****
Male speaker 1	12.56	87.44	83
Male speaker 2	10.52	89.48	86.42
Female Speaker 1	7.65	92.35	96
Female Speaker 2	5.96	94.04	98.32

*The recognition score is calculated programmatically.

7. OBSERVATIONS

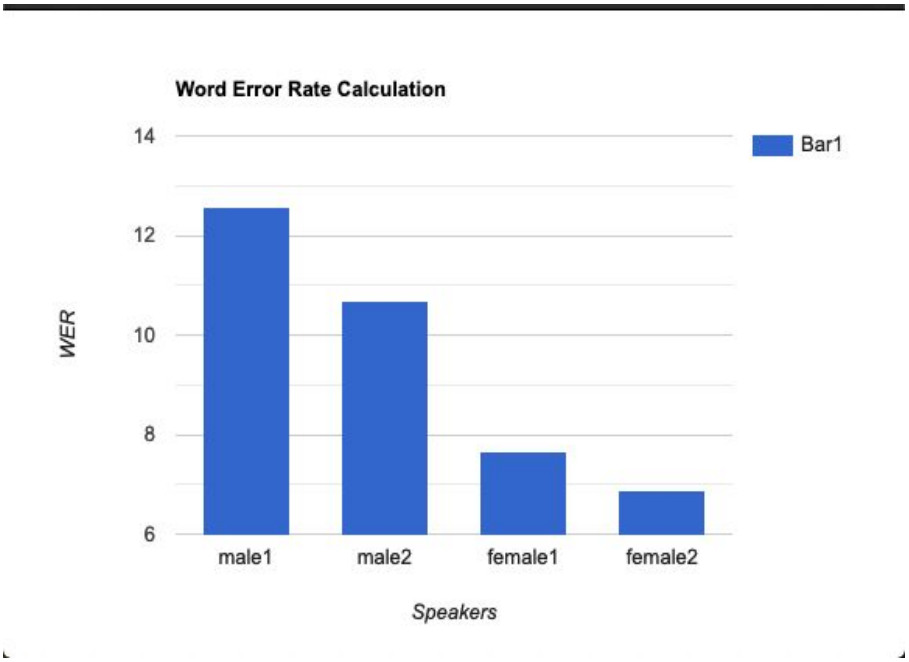

Figure. 8 WER graph

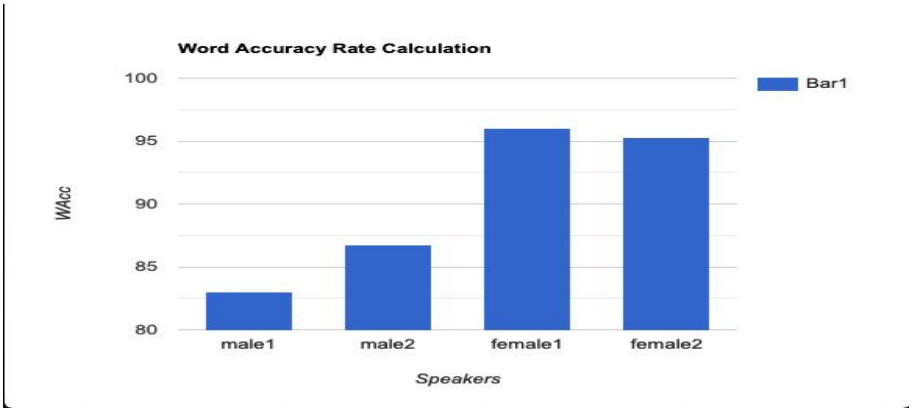

Figure. 9 WAcc graph

We observed that the error rate (WER) of the male speakers is considerably higher than that of the female speakers. Also, the accuracy percentage (WAcc) of male speakers was lower than the accuracy percentage of female speakers. This observation signifies that the speech recognition system is dependent on the voice type, voice pattern and not only the parameters calculated above in Table 2. We have also tested for evaluating the recognition scores at the Phoneme level shown in Table 3.

Table 4 Recognition scores

Words	Phonemes	Avg. phoneme recognition%	Recognition%
Start	/St /a /r /a /t	74.48	74
Stop	/St /o /p	86.62	87
Rotate the base clockwise	/Ro /t /ae /t /ae /th /b /ae /s /cl /o /k /v /i /s	60.32	61
Rotate the shoulder	/Ro /t /ae /t /ae /th /sh /o /wel /dur	74.28	75
Open gripper	/o /pn /ghr /ri /pr	85.63	86
Close gripper	/cl /o /s /ghr /ri /pr	84.43	85
Lift the shoulder up	/l /if /t /th /sh /o /wel /dur /up	55.63	56
Put the shoulder down	/poo /t /th /sh /o /wel /dur /da /un	44.57	45

8. CONCLUSION

This work is aimed at developing a robust and accurate HMM-based phoneme speech recognition system. We have presented the phoneme recognition with detailed analysis by calculating the metrics like Word Error Rate (WAR), Word Accuracy percentage (WAcc), Recognition score, and the Average Phoneme recognition percentage. The results in the study indicate that the speech recognition system is fairly affected by the voice patter and voice type also. Average recognition scores were observed to change rapidly with the change in the distance between the microphone and the speaker from 70% to 55.6%. With the experiments conducted we conclude that the background noise is one of the key factors contributing towards a significant reduction in the Accuracy percentage and recognition scores. Experimental findings reveal that the technique employed for feature extraction from speech input, the Linear Prediction Coding technique is a robust, computationally fast algorithm. During the testing phase, we observed that the pronunciation greatly affects the speech recognition process; such that if two speakers pronounce the same word in a different manner, the accuracy differs considerably.

REFERENCES

[1] Hu, K., Bruguier, A., Sainath, T. N., Prabhavalkar, R., & Pundak, G. (2019). Phoneme-Based Contextualization for Cross-Lingual Speech Recognition in End-to-End Models. arXiv preprint arXiv:1906.09292.

[2] Kashif, K., Wu, Y., & Michael, A. (2019, September). Consonant Phoneme Based Extreme Learning Machine (ELM) Recognition Model for Foreign Accent Identification. In Proceedings of the 2019 The World Symposium on Software Engineering (pp. 68-72).

[3] Nielsen, Jens Bo, and Torsten Dau. "A Danish nonsense word corpus for phoneme recognition measurements." Acta Acustica united with Acustica 105, no. 1 (2019): 183-194.

[4] Makhoul J, Schwartz R (1995) State of the art in continuous speech recognition. Proc Natl Acad Sci 92(22):9956–9963

[5] Bansal P, Dev A, Jain SB (2008) Optimum HMM combined with vector quantization for Hindi speech word recognition. IETE J Res 54(4):239–243

[6] Ben J, Wan WG, Yu XQ (2003) Phoneme-based speaker-independent English command recognition. J Shanghai Univ (English Edition) 7(2):163–167

[7] Sabur Ajibola Alim, Nahrul Khair Alang Rashid (2018) Some Commonly Used Speech Feature Extraction Algorithms. IntechOpen. http://dx.doi.org/10.5772/intechopen.80419

[8] Agrawal S, Shruti AK, Krishna CR. Prosodic feature based text dependent Speaker recognition using machine learning algorithms. International Journal of Engineering Science and Technology. 2010;2(10):5150-5157

[9] Gill AS. A review on feature extraction techniques for speech processing. International Journal Of Engineering and Computer Science. 2016;5(10):18551-18556

[10] Kumar R, Ranjan R, Singh SK, Kala R, Shukla A, Tiwari R. Multilingual speaker recognition using neural network. In: Proceedings of the Frontiers of Research on Speech and Music, FRSM. 2009. pp. 1-8

[11] Paulraj MP, Sazali Y, Nazri A, Kumar S. A speech recognition system for Malaysian English pronunciation using neural network. In: Proceedings of the International Conference on Man-Machine Systems (ICoMMS). 2009

[12] Tan CL, Jantan A. Digit recognition using neural networks. Malaysian Journal of Computer Science. 2004;17(2):40-54

[13] Kurzekar PK, Deshmukh RR, Waghmare VB, Shrishrimal PP. A comparative study of feature extraction techniques for speech recognition system. International Journal of Innovative Research in Science, Engineering and Technology. 2014;3(12):18006-18016

[14] Mosa GS, Ali AA. Arabic phoneme recognition using hierarchical neural fuzzy Petri-net and LPC feature extraction. Signal Processing: An International Journal (SPIJ). 2009;3(5): 161

[15] Kumar P, Chandra M. Speaker identification using Gaussian mixture models. MIT International Journal of Electronics and Communication Engineering. 2011;1(1):27-30

[16] Young S, Evermann G, Gales M, Hain T, Kershaw D, Liu X, et al. The HTK Book, Version 3.4. Cambridge, United Kingdom: Cambridge University; 2006

[17] Shobha Bhatt, Amita Dev, Anurag Jain. Confusion analysis in phoneme based speech recognition in Hindi. Journal of Ambient Intelligence and Humanized Computing. 2020. pp. 1-26

[18] N. Saravanan, R. Sivaramakrishnan. Command and control of industrial manipulator through speech-based interfaces in indic Languages. The Journal of Supercomputing. 2019. pp. 1-12. https://doi.org/10.1007/s11227-019-02790-0

ACKNOWLEDGEMENTS

The authors of the paper would like to express their gratitude to **Prof. Annu Abraham** and **Dr. J H Nirmal** for their guidance and moral support.

CONFLICT OF INTEREST

The authors have no conflicts of interest to declare regarding or related to the contents of the manuscript.

ETHICAL DECLARATIONS

The authors of the manuscript declare that no animals were involved in the experiments performed during the study.

DISCLOSURE OF FUNDING

The authors of this paper would like to mention that no specific funding was received for this project.

Author's biography

Adwait P Naik graduated from K. J. Somaiya College of Engineering, affiliated to the University of Mumbai with B.tech (Hons) in Electronics Engineering in 2019. Currently, working as a research intern at HTIC, IIT-Madras. Author's research interest spans over various fields including Robotics, Artificial Intelligence, Speech Recognition, and Machine learning.