Int. j. inf. tecnol. (April 2021) 13(2):483–492
https://doi.org/10.1007/s41870-020-00586-7

ORIGINAL RESEARCH

Hindi speech recognition in noisy environment using hybrid technique

Ashok Kumar[1] · Vikas Mittal[2]

Received: 19 February 2020 / Accepted: 20 November 2020 / Published online: 1 January 2021

Abstract Automatic speech recognition is generally analyzed for two types of word utterances; isolated and continuous-words speech. Continuous-words speech is almost natural way of speaking but is difficult to be recognized through machines (speech recognizers). It is also highly sensitive to environmental variations. There are various parameters which are directly affecting the performance of automatic speech recognition like size of datasets/corpus, type of data sets (isolated, spontaneous or continuous) and environment variations (noisy/clean). The performance of speech recognizers is generally good in clean environments for isolated words, but it becomes typical in noisy environments especially for continuous words/sentences and is still a challenge. In this paper, a hybrid feature extraction technique is proposed by joining core blocks of PLP (perceptual linear predictive) and Mel frequency cepstral coefficients (MFCC) that can be utilized to improve the performance of speech recognizers under such circumstances. Voice activity and detection (VAD)-based frame dropping formula has been used solely within the training part of ASR (automatic speech recognition) procedure obviating its need in actual implementations. The motivation to use this formula is for removal of pauses and distorted elements of speech improving the phonemes modeling further. The proposed method shows average improvement in performance by 12.88% for standard datasets.

Keywords Automatic speech recognition · Mel frequency cepstral coefficients · Perceptual linear predictive · Voice activity and detection

1 Introduction

Automatic speech recognition (ASR) is a practice to convert a sequence of words spoken by human being into text by means of machines. Automatic speech recognition can be divided in many forms depending on the type of utterances like Isolated words, connected words, continuous speech and spontaneous speech. Continuous-words speech recognition is almost near to natural way of speaking. So, several studies have been conducted to improve the accuracy of recognition of this speech format, but it is very difficult task to design such recognizers due to absence of efficient techniques for the detection of start and end points of such speech [1].

The selection of language is priority while designing a speech recognizer that is suitable for a country/region. For the wide acceptance of speech recognition system, it is required to be designed in local language. In country like India, Hindi language has wide acceptance to connect people across the country.

Most of speech recognition systems have been designed in foreign languages like English and Japanese etc. In India, majority of the people live in villages and adopt farming to earn their livelihoods. Indian government is running many schemes to benefit villagers for overall

✉ Ashok Kumar
ashokgiri010@gmail.com

Vikas Mittal
vikasmittalkkr@gmail.com

[1] Department of Electronics and Communication Engineering,
National Institute of Technology,
Kurukshetra 136119, Haryana, India

[2] Department of Electronics and Communication Engineering,
National Institute of Technology,
Kurukshetra 136119, Haryana, India

development of the country, but most of the people are not aware about them due to lack of education and availability of computer knowledge in English. Hence, automatic speech recognition in Hindi has the potential to solve their problems [2].

The objective of ASR is to behave as a medium between man and machine and it is expected to remain robust in varying environments [3]. This task becomes too complex when continuous speech recognizers are designed for noisy environments. It is a serious problem to define speech boundaries in noisy environments due to the occurrence of non-speech events [4]. Speech recognition systems which are trained in noisy environments are often affected by ambient acoustic noise thereby reducing their performance. This dilapidation is generally due to the gap between clean acoustic models and noisy speech data. Significant research efforts have undergone to lessen this mismatch and recover recognition accuracy in noisy conditions [5].

In automatic speech recognition (ASR), noise robustness is ensured by several methods. First method is to train the system directly on a noise that interferes during the recognition phase. This kind of system is known as matched system. This system is probably far better compared to several noise compensation methods. Changing the system to adapt for new type of noises is a complex and time-consuming task since its re-training needs a lot of time. A more real-world substitute to the matched training is multi-condition training, in which the system is trained right on noisy speech occurring in the most common noise environments.

Therefore, the necessity for re-training the system each time the background noise changes can be avoided [3]. In [6], the authors explained the effects of noise in communication systems according to the sources of noise, the numbers and the types of talkers and listener's hearing ability and provided research guidance for effective recognition in noisy environments. The authors in [7], outlined a set of challenges where optimization formulations and algorithms play an important role. Authors have also described various approaches in speech recognition and their optimization. The authors of [8], proposed the optimization technique based on Stochastic Gradient Descent algorithm to upgrade the performance of speech recognizers in noisy environments.

In [9], authors developed an algorithm based on binary masking to separate speech from noise. This binary masking was different from the ideal binary mask which needs priori information about premixed signals. The authors in [10], addressed the problem of distant speech recognition in noisy environment and proposed non-negative matrix factorization (NMF) enhancement method to improve the robustness of Automatic Speech Recognition (ASR) systems. The Modified K-NN based algorithm was suggested for classification of the large database to improve the accuracy of detection by reducing the impact of noise in [11]. The author of [12], designed a real time educational software for signal and speech processing applications developed using MATLAB. In [13], the authors compared different feature extraction methods in noisy environments for isolated words. Kalman filter was used to remove the background noise and enhance the speech signal. The authors in [14], developed a noise robust distributed speech recognizer for real world applications using cepstral mean normalization (CMN) for robust feature extraction. The authors presented a modified framework using support Vector Machine algorithm to detect different keyloggers installed or available on PC for security purpose of information and datasets in [15]. In [16], authors analysed the influence of window length and frame shift on speech recognition. It was concluded that a window length of 10 ms with the frame shift between (7.5–10) ms can increase the recognition rate up to 2.5%. The authors of [17], provided the comparison of different feature extraction techniques using neural network as classifier. The self -adapted diversity-based parameters were applied to particle swarm optimization algorithm to obtain improved form of clusters for better detection in [18]. In [19], authors showed that signal acquired through throat microphone can improve the speech recognition in noisy environment as compared to conventional microphone. The authors in [20], presented a comparative analysis of different feature extraction techniques for isolated words in noisy environments. In [21], authors reported Social Spider Algorithm for global optimization between class variance to get improved thresholding. The authors of [22], proposed a novel method of speech segregation for unlabelled stationary noisy audio signals using the deep belief network (DBN) model. The proposed method successfully segregates a music signal from noisy audio streams. The local featured-based supervised learning was presented using Support Vector Machine classification to recognize Thai character in [23]. In [24], authors provided a thorough study of work done using deep learning between 2006 and 2018 in field of automatic speech recognition.

It is observed from above that not much work has been done in ASR of Hindi speech in noisy environment. Most of the reported researches were based on Hidden Markov Model (HMM), Gaussian Mixture Model and their Hybridizations. Therefore, there is an ample scope of using Deep Neural Network (DNN) with hybrid features to further improve the accuracy of automatic speech recognition.

There are various parameters which effect the performance of automatic speech recognition. Noise is one such parameter. It is a tedious task to improve the recognition rate in noisy environments especially for continuous speech, since here each word is highly dependent on each other to produce its meaning. So, their recognition degrades further in the presence of noise. Speech presence probability (SPP) [25] based noise assessment method is preferred for noise power spectrum estimation. It is a good estimator to find presence of speech in stationary and non-stationary environments. First 20 frames are usually considered to get the power spectrum for better approximation of the speech. The objective of this study is to increase the stability of speech recognition systems in real-time reverberant environments. MFCC and PLP are widely used to estimate the concept of human auditory system in automatic speech recognition. Both shows almost comparable results for small parameters. But, PLP performs better for large number of parameters [26]. In this paper, coefficients of these two sets are combined to get better results.

This paper is organized in four different sections. First section introduces different types of speech recognition systems and discusses the present state-of-the-art. Second section presents the proposed methodology. Third section covers results and discussion. Fourth section concludes the findings followed by future scope.

2 Proposed methodology

Proposed methodology constitutes acquisition of data sets along with noisy ones, feature extraction and proposed algorithm as presented in the following subsections.

2.1 Speech datasets

Hindi speech signals of different speakers are recorded by using Audacity 2.3.2. It is an open source software for audio editing and recording applications. For this purpose, 3 males and 3 females of different age group are selected. The acquired datasets are described in Table 1.

A total of 600 voice samples were recorded. These voice samples were divided into two sets; first set consisting of 75% of the total speech samples that is used for training and remaining 25% of the samples are considered for testing purpose. A WO Mic client interface was used to connect with Audacity tool for sound recording. Audacity is a free sound recording tool with various options for clipping, storing and mixing of sounds. Speech signals were recorded using the following parameters; Sampling frequency = 16 kHz, Coding Technique PCM, Mode of recording Mono and bit rate = 16 bits/s.

Waveforms of a speech signal in Audacity (with many specifications of parameters) and MATLAB are shown in Figs. 1 and 2, respectively. The speech data collected is mixed with various noises that usually exist in the environment (e.g., car, fan and diesel engine noise) to study their effects on ASR. Therefore, different noise files are obtained from online sources as explained in next subsection.

2.2 Noisy database

Different type of noise samples from car, diesel engine and fan are obtained from www.freesound.org.

This website has different types of sounds that can be used for research purpose. Noises from different sources is mixed with speech samples from clean environment to obtain noisy speech having SNR value lying between (0–15) db at equal intervals of 5 db. These noisy speech signals are used to train and test the performance of speech recognizers. Waveform of various noises are shown in Figs. 3, 4 and 5. Noise reduction option under effect menu bar of Audacity is used to change the Signal to Noise ratio.

A portion of noise can be added to signal to obtain noisy speech data using two separate windows for the two. Voice Activity and Detection (VAD) is applied as a filter to separate speech from non-speech signals and noise. After filtering speech data, its features are extracted to train the recognizer using Deep neural network (DNN) that is used as a classifier in MATLAB. MFCC and PLP features are utilized individually and collectively to analyze the speech recognition rate with and without VAD (Voice Activity and Detection). The steps to extract different features are presented in next subsection.

2.3 Feature extraction

Mel frequency cepstral coefficient (MFCC) and perceptual linear perceptron features are used collectively in this paper

Table 1 Speech datasets

S. No	Gender	Age	No. of sample (S)	No. of repetition (R)	Total samples (T = S*R)
1	M	40	50	2	100
2	M	17	50	2	100
3	M	16	50	2	100
4	F	17	50	2	100
5	F	18	50	2	100
6	F	30	50	2	100

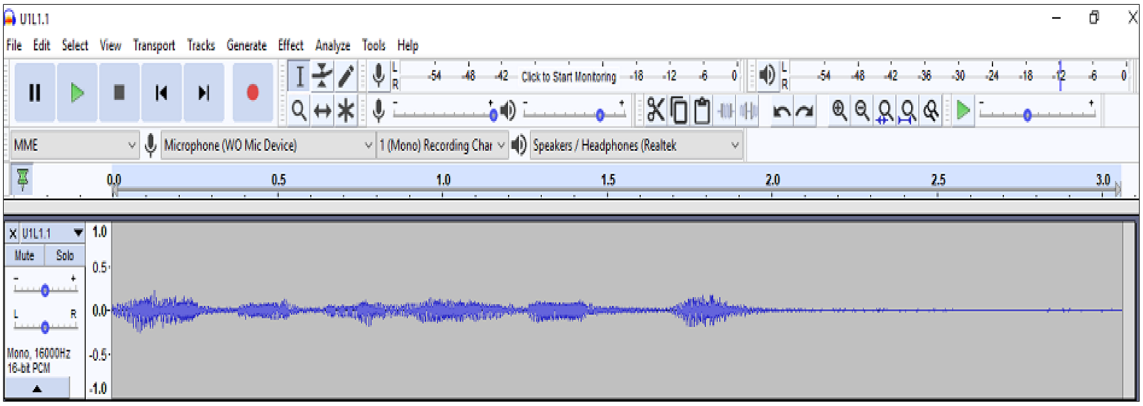
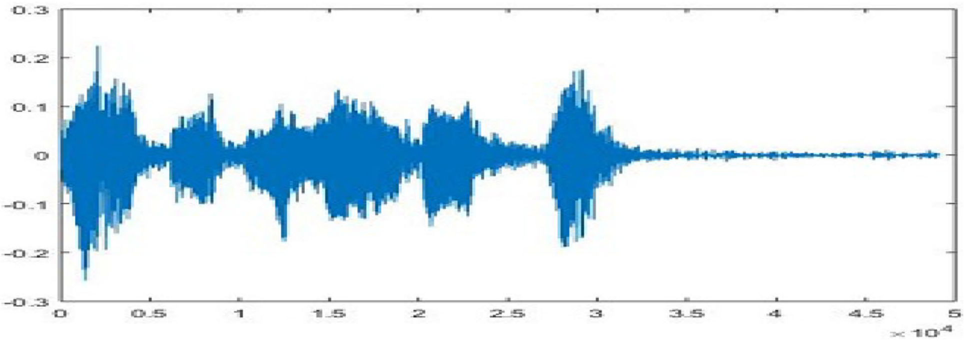

U1L1.1
File Edit Select View Transport Tracks Generate Effect Analyze Tools Help
Click to Start Monitoring
MME
Microphone (WO Mic Device)
1 (Mono) Recording Char
Speakers / Headphones (Realtek
0.0 0.5 1.0 1.5 2.0 2.5 3.0
U1L1.1
Mute Solo
Mono, 16000Hz
16-bit PCM
1.0 0.5 0.0 -0.5 -1.0

Fig. 1 Waveform of speech signal in Audacity with different parameters

Fig. 2 Waveform of speech signal (‘Bharat Ek Mahaan Desh Hai’) in MATLAB



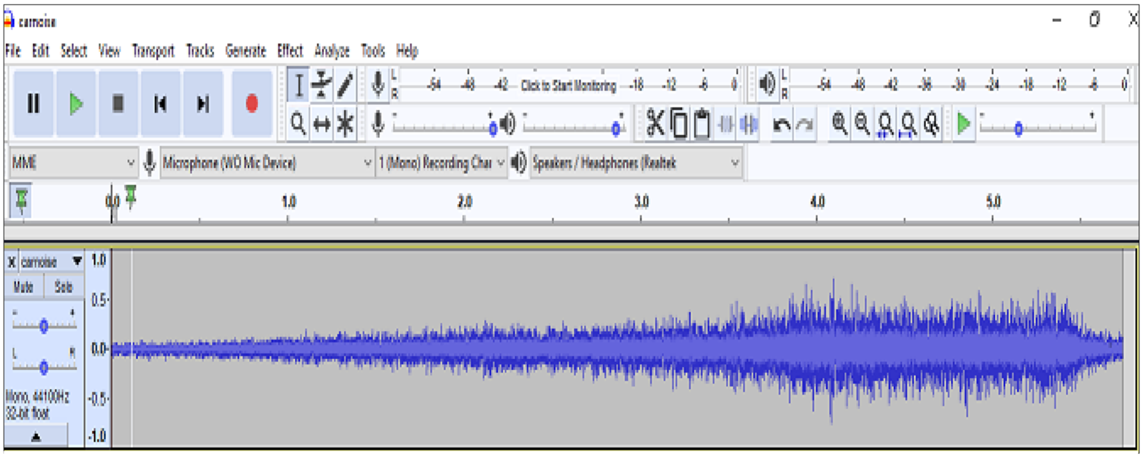

carnoise
File Edit Select View Transport Tracks Generate Effect Analyze Tools Help
Click to Start Monitoring
MME
Microphone (WO Mic Device)
1 (Mono) Recording Char
Speakers / Headphones (Realtek
0.0 1.0 2.0 3.0 4.0 5.0
carnoise
Mute Solo
Mono, 44100Hz
32-bit float
1.0 0.5 0.0 -0.5 -1.0

Fig. 3 Waveform of car noise using audacity tool

for feature extraction. Both types of feature extraction techniques behave well to map human auditory system. The mechanism to extract these features is presented in the following subsections.

2.3.1 Mel frequency cepstral coefficients (MFCC)

Mel frequency cepstral coefficient (MFCC) is widely accepted frequency domain feature extraction technique to map human auditory system [27]. Human speech is not linear in nature. It is linear below 1 Khz and non-linear above it and can be well estimated using Mel-scale as shown by Fig. 6.

Mel-scaled frequency domain features provide better modeling as compared to time domain features [28, 29]. MFCC features are extracted using the following steps;

1. Pre-emphasis: The recorded speech signal needs pre-emphasis to boost the energy of signal at high

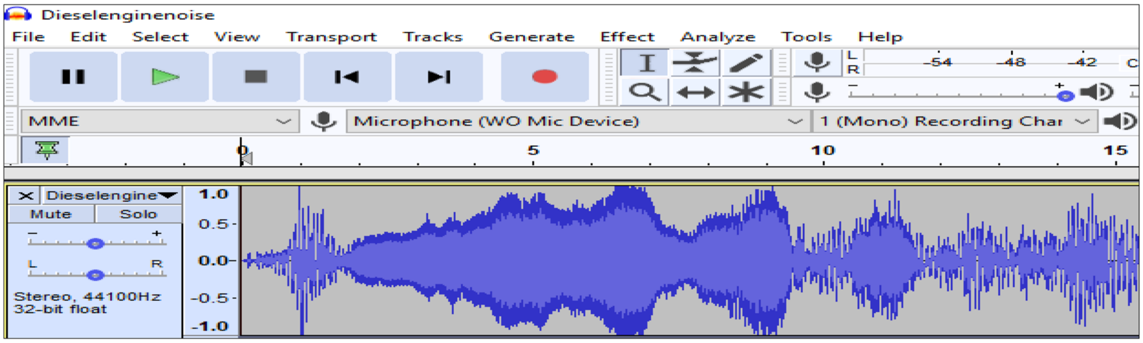


Fig. 4 Waveform of diesel engine noise using audacity tool

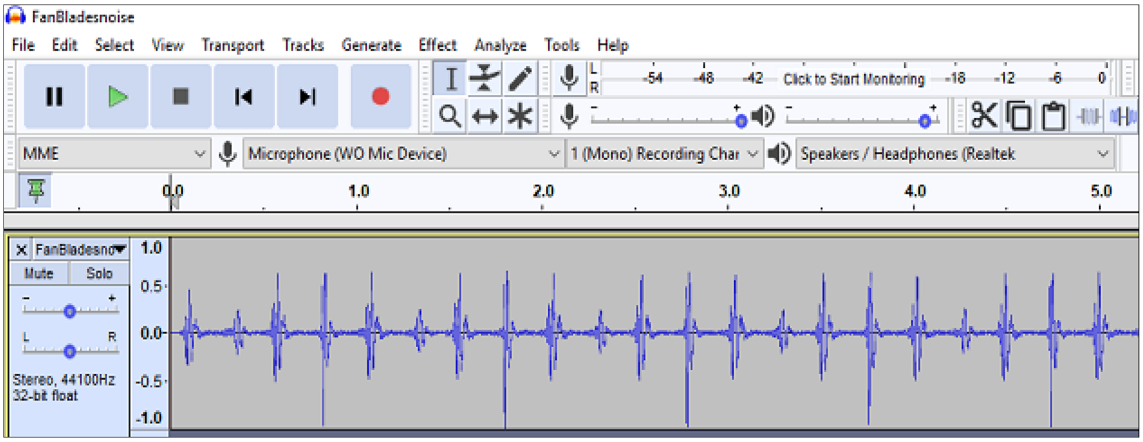

FanBladesnoise
File Edit Select View Transport Tracks Generate Effect Analyze Tools Help
-54 -48 -42 Click to Start Monitoring -18 -12 -6 0
MME
Microphone (WO Mic Device)
1 (Mono) Recording Char
Speakers / Headphones (Realtek
0.0 1.0 2.0 3.0 4.0 5.0
FanBladesno
Mute Solo
Stereo, 44100Hz
32-bit float
1.0 0.5 0.0 -0.5 -1.0

Fig. 5 Waveform of fan noise using audacity tool

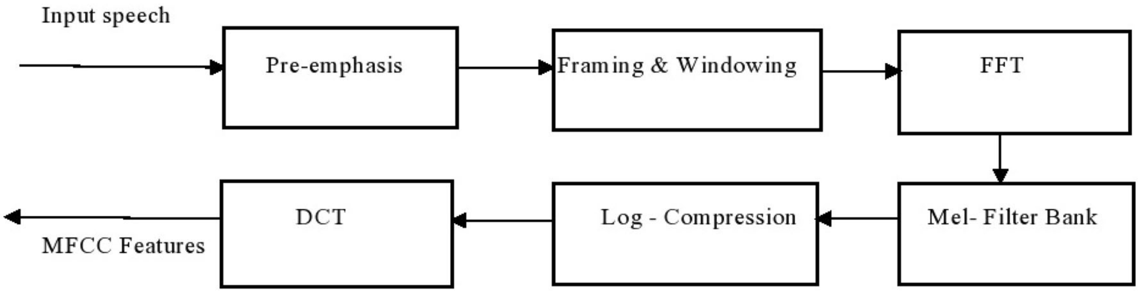


Fig. 6 Extraction of MFCC features

frequencies. The high frequency components are more affected by noise as compared to lower ones. Hence, a proper high pass filter is needed to maintain signal to noise ratio at high frequencies. The transfer function of this filter in z domain H(z) is given as

$$H(z) = 1 - 0.97z^{-1}. \tag{1}$$

Here 0.97 is Pre-emphasis factor.

2. Framing and windowing: The speech signal is non-stationary in nature. It is stationary for short interval of time so framing is required to resolve it into small overlapping pieces known as frames. Then windowing is performed to eliminate discontinuities at edges. The Hamming Window [30] performs this action as

$$W(n) = \begin{cases} 0.54 - 0.46\cos\left(\dfrac{2\pi n}{N} - 1\right) & 0 \le n \le N \\ 0 & \text{otherwise,} \end{cases} \tag{2}$$

where W(n) is Hamming Window, n is considered samples out of total N samples.

3. Fast Fourier Transform (FFT): The Fast Fourier Transform is applied for transformation into frequency domain as $X(K)$ defined by

$$X(K) = \sum_{n=0}^{N-1} x(n)W^{nk}.....0 \leq n \leq N-1, \quad (3)$$

where N is the size of FFT and x(n) is input signal.

Mel-scale conversion: Mel-scale has better adaptability to human auditory system so frequency is converted to Mel-scale filter bank as

$$M(f) = 1127\ln\left(1 + \frac{f}{700}\right), \quad (4)$$

where f is frequency on linear scale and M(f) is Mel-scale frequency.

Discrete Cosine Transform (DCT): DCT is performed on Log Mel- spectrum of previous output to decorrelate the filter outputs. Its filter coefficients are grouped with log energy coefficients for final preparation of vector coefficients. The DCT transform is given as

$$MFFC(k) = \sqrt{\frac{2}{M}}\sum_{m=1}^{M} X(m)\cos\left[\frac{\Pi k}{M}(m-0.5)\right], \quad (5)$$

where M is number of Triangular filter and the m is number that lies between 1 to M (1 < m < M).

A set of 13 features (coefficients) are generated from the above steps out of which 12 features are computed using DCT transform and one energy feature is appended to it. Next 13 feature are obtained from delta method which are produced from first order derivative. In this way, a feature set of 26 coefficients is framed. The delta coefficients help to map dynamic nature of speech [31]. The 20 coefficients are finally selected for analysis purpose to reduce the complexity and define non -uniform nature of the speech.

2.3.2 Perceptual linear perceptron (PLP)

The perceptual linear prediction (PLP) is another feature extraction technique which follows the concept of psychophysics of hearing. It is similar to LPC with a difference that it uses its spectral characteristics to map the human auditory system using three steps: 1. critical band analysis 2. equal loudness curve 3. intensity loudness (power–law relation). Speech signal cannot be estimated on linear scale as done by LPC, therefore, PLP is preferred [32–34].

PLP involves the following computational steps as shown in Fig. 7

- Initially steps similar to (1), (2) and (3) used for MFCC are also followed for PLP.
- After that Band pass filtering is done to approximate the power spectrum of each frequency band as

$$P(\omega) = \mathrm{Re}(s(\omega))^2 + Im(s(\omega))^2 \quad (6)$$

- Then audio frequency is converted to Bark Scale for better mapping of human auditory process as

$$f(Bark) = 6\ln\left[\frac{f}{600} + \left[\left(\frac{f}{600}\right)^2 + 1\right]^{0.5}\right] \quad (7)$$

The Bark filter bank is used for better sense of hearing under equal loudness operation. Further, these matched values are boosted according to the power Law.

- Finally, LP Model is applied to predict the feature coefficients by mapping the power spectrums $P(\omega)$ and $P'(\omega)$ as

$$\frac{1}{M}\sum_{m=1}^{M}\frac{P(\omega)}{P'(\omega)} = 1 \quad (8)$$

where $P(\omega)$ and $P'(\omega)$ are input and predicted power spectrum of the speech signal.

The initial steps such as windowing and Fourier transform are same for both MFCC and PLP. with the difference of use of Mel-scale in MFCC and bark scale in PLP. The equal loudness function is applied before linear prediction to amplify weak signal components. In PLP, trapezoidal filters are incorporated in place of triangular filters used in MFCC. The recursive cepstrum computation is applied to compute first 13 coefficients of PLP features. This process is similar to that used for MFCC to extract a separate 13 feature vector as explained above under MFCC. The 20 features are selected from this combined feature vector to reduce the computational complexity.

2.4 Proposed algorithm

This section presents the proposed algorithm. The performance of speech recognition system mainly depends upon the efficiency of VAD (voice activity and detection) to sort speech using different steps.

Step 1 Apply silence indicator to find signal idleness and the noise scales is updated for the duration of these stages.

Step 2 Apply Short-time Fourier Transform (STFT) for transforming time domain signal to frequency domain. STFT is trailed by a magnitude operator. Speech signal is processed for short interval of time (10-50 ms) so DFT is performed after windowing, which is also known as STFT as

$$X_n\left(e^{j\omega}\right) = \sum_{m=-\infty}^{\infty} x(m)w(n-m)e^{-jwm}, \quad (9)$$

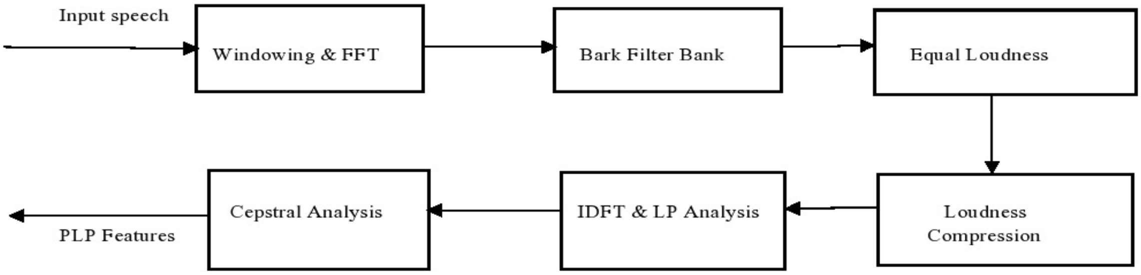


Fig. 7 Extraction of PLP features

Table 2 Accuracy [%] without VAD using MFCC

Type of noise	Signal to noise ratio (db)				Average (15–0 db)
	15	10	5	0	
Car	84.8	66.0	42.1	22.2	53.775
Fan	83.2	66.1	39.7	21.1	52.525
Diesel engine	81.5	60.8	37.1	10.8	47.55

Table 3 Accuracy [%] with VAD using MFCC

Type of noise	Signal to noise ratio (db)				Average (15–0 db)
	15	10	5	0	
Car	88.8	77.1	55.2	35.3	64.10
Fan	91.1	77.8	55.1	33.8	64.45
Diesel engine	90.2	74.2	51.25	30.1	61.43

Table 5 Accuracy [%] with VAD using PLP

Type of noise	Signal to noise ratio (db)				Average (15–0 db)
	15	10	5	0	
Car	89.3	77.25	55.8	35.8	64.537
Fan	91.4	75.45	55.7	34.2	64.187
Diesel engine	90.4	74.5	51.8	30.3	61.75

where w(n − m) is window which select the portion of input x(n) for further computation.

Step 3. Apply High pass filter (HPF) to reduce the noise variance. This is essential to decrease the misrepresentations due to noise deviations. The function of High Pass Filter is to suppress the noise and increase the energy at high frequency given in (1).

Step 4. Apply post-processor for eliminating the misrepresentations by spectral subtraction.

Step 5. Apply an Inverse Short-time Fourier Transform (ISTFT) for transforming the treated signal back to the time domain as

$$x(n) = \frac{1}{2\pi w(0)} \int_{-\pi}^{\pi} X_n(e^{jw}) e^{j\omega n} d\omega, \tag{10}$$

where x(n) is time domain signal and w (0) is real window sequence.

Step 6. A grouping rule is applied to categorize a signal's section as speech or non-speech. The grouping rule compares the output obtained from the VAD using threshold defined in terms of speech parameters. If value exceeds the threshold, it indicates a speech signal otherwise it belongs to a non-speech category. A value close to threshold is uncertain that reduces the performance of speech recognizer.

Step 7. Compute a set of features to distinguish speech and non-speech.

Step 8. Combine the evidence from the features in a classifier for classification.

The multilayer training is provided on the data set by dividing it in to three subsets. The first subset is training set that is used to compute gradient and biases of network. The second set is validation set. It is used to monitor validation error. The third subset is test set error. The validation error and test set error are compared to choose the appropriate value. The default value of training, validation and testing are 0.7, 0.15 and 0.15. But in this paper, values used are 1.0, 0.15 and 0.15, respectively. The value of test set error varies according to the different iteration number this may happen due to the poor division of dataset. The function of this regression layer is used to compute mean squared error to get targeted output [35].

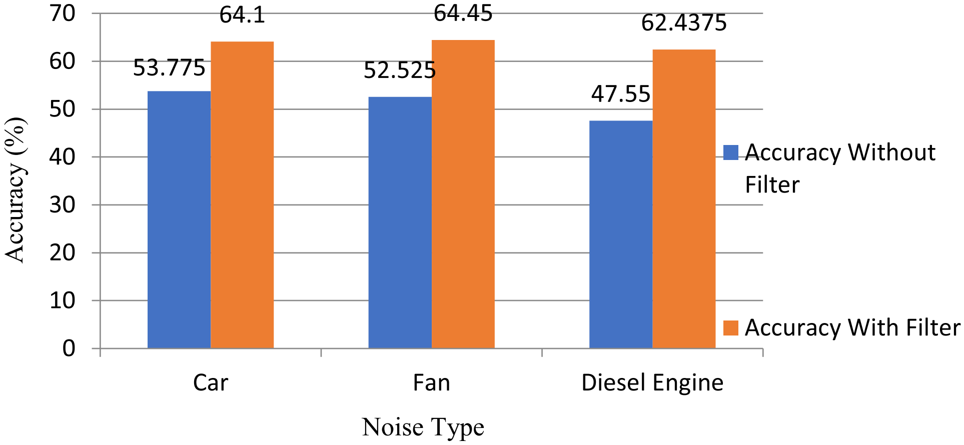


Fig. 8 Performance evaluation of VAD using MFCC

Table 4 Accuracy [%] without VAD using PLP

Type of noise	Signal to noise ratio (db)				Average (15–0 db)
	15	10	5	0	
Car	85.3	66.2	42.4	22.5	54.1
Fan	83.5	66.5	40.2	21.8	53.0
Diesel engine	82.5	61.5	37.3	11.5	48.2

Table 6 Accuracy [%] without VAD using MFCC_PLP

Type of noise	Signal to noise ratio (db)				Average (15–0 db)
	15	10	5	0	
Car	87.2	68.5	44.6	24.4	56.175
Fan	86.3	67.7	42.5	22.9	54.85
Diesel engine	85.3	62.5	38.2	12.4	49.6

3 Results and discussion

The accuracy of speech recognizer is computed in three stages. In first part, MFCC features are used with and without VAD to classify the speech signal for various noises at different signal to noise ratio. Secondly, PLP features are utilized and finally, the two features are combined to form a hybrid feature set. Voice Activation and Detection (VAD) is universally used in all cases.

3.1 Performance analysis using MFCC

The accuracy of speech recognition system using MFCC feature with VAD and without VAD is shown below in Tables 2 and 3. It is observed from the table that as signal

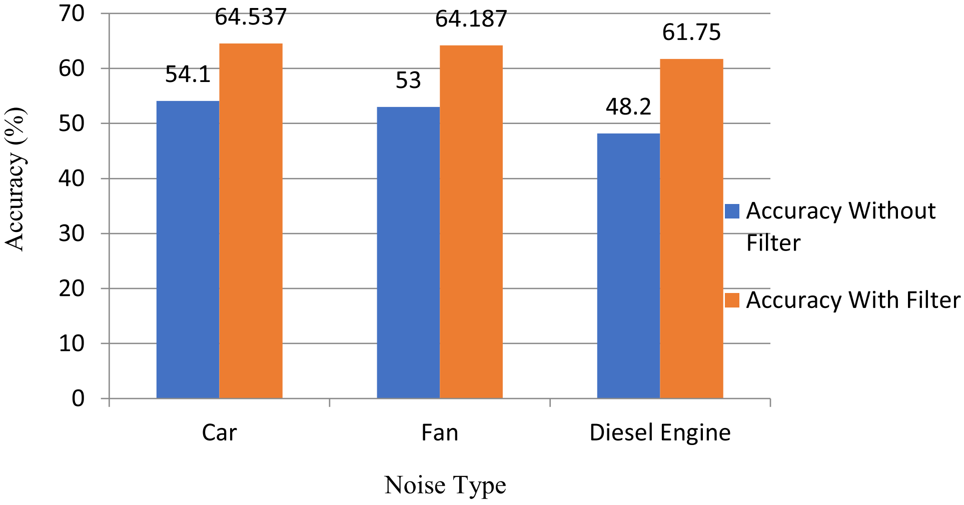


Fig. 9 Performance evaluation of VAD using PLP

Table 7 Accuracy with VAD using MFCC-PLP

Type of noise	Signal to noise ratio (db)				Average (15–0 db)
	15	10	5	0	
Car	91.2	78.5	56.2	36.2	65.525
Fan	93.5	76.5	56.5	34.5	65.25
Diesel engine	92.4	75.5	52.5	31.5	62.975

to noise ratio increases, the performance of recognition also improves. The recognition rate is maximum in the presence of car noise and minimum for the case of diesel engine noise. The average recognition rate between 0 to 15 db also follow the same pattern.

It is observed from Fig. 8 that there is a considerable improvement in accuracy of speech signal using VAD. Maximum recognition is noted from fan noise at 10 and 15 dB. It is also noticed that (0–5) dB and (5–10) dB show almost similar improvement. But after 10 dB improvement in performance is comparatively low. It is found from Fig. 8 and Table 4 that Diesel Engine noise shows maximum improvement in accuracy by 13.88%. and car has least one. The accuracy of speech recognition system in presence of fan noise lies in accuracy of these two noise sources.

3.2 Performance analysis using PLP (perceptual linear perceptron)

It is observed from Tables 2, 3, 4, 5 and Fig. 9, that MFCC follow the same pattern as PLP. But the performance of PLP in Noisy environment is somewhat higher than the MFCC both with and without VAD. The average recognition in case of PLP is 51.7% as compared to 51.28% without using VAD in MFCC. On the other hand, using VAD this performance increases to 63.48% as compared to 63.31% for MFCC. In case of PLP, there is an increase by 11.78% as compared to 12.03% for MFCC.

3.3 Performance analysis using hybrid features i.e., MFCC_PLP

It is observed from Tables 2, 3, 4, 5, 6, 7 and Fig. 10, that MFCC_PLP hybrid feature extraction has better performance as compared to MFCC and PLP used individually. The average performance in this case is 53.54% with VAD as compared to 51.7% for PLP and 51.28% for MFCC. The average percentage increases to 64.58% as compared to 63.48% for PLP and 63.31% for MFCC. The Proposed methodology improves the recognition by 11.78% in PLP, 12.03% in MFCC and 12.88% on an average as compared to system without VAD.

4 Conclusion

A hybrid MFCC and PLP features based ASR has been implemented successfully for Hindi speech in noisy environment. Both of this techniques show comparable results in noise free environments when applied individually. But in noisy environment, PLP provides slightly better results as compared to MFCC.

VAD can differentiate well between speech and non-speech data in noisy environments. The proposed hybrid technique based on VAD increases the efficiency of ASR system in noisy environments. The proposed methodology shows average improvements by 12.88% with VAD as compared to the case without it. This work can be further extended by integrating VAD and Deep Neural Networks with some evolutionary algorithms like particle swarm optimization (PSO), differential evolution (DE) etc. to further improve the system performance by optimizing number of filterbanks.

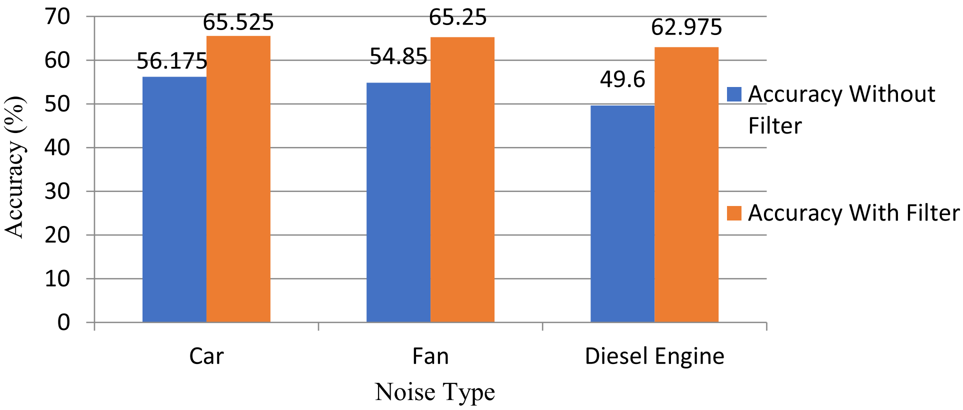


Fig.10 Performance evaluation of VAD using MFCC_PLP

Acknowledgements A special note of thanks is due to ECE Department, NIT Kurukshetra, Haryana, India for providing the required infrastructure and research environment.

References

1. Kurzekar PK, Desmukh RR, Waghmare VB, Shrishrimal P (2014) Continuous speech recognition system: a review. Asian J Comput Sci Inform Technol (AJCSIT) 4:(6): 62–66
2. Agarwal RK, Dave M (2008) Implementing a speech recognition interface for Indian Languages. In: Proceedings of the IJCNLP-08 Workshop on NLP for Less Privileged Languages. pp. 105–112
3. Keronen S, Remes U, Palomaki KJ, Virtanen T, Kurimo M (2010) Comparison of noise robust methods in large vocabulary speech recognition. In: 18th European Signal Processing Conference (EUSIPCO-2010), 1973–1977
4. Li Q, Zheng J, Tsai A, Zhou Q (2002) Robust endpoint detection and energy normalization for real-time speech and speaker recognition. IEEE Trans Speech Audio Process 10(3):146–157
5. Cui X, Alwan A (2005) Noise robust speech recognition using feature compensation based on polynomial regression of utterance SNR. IEEE Trans Speech Audio Process 13(6):1161–1172. https://doi.org/10.1109/TSA.2005.853002
6. Le Prell CG, Clavier OH (2017) Effects of noise on speech recognition: Challenges for communication by service members, www.elsevier.com/locate/heares. Hearing Res 349:76–89
7. Wright SJ, Kanevsky D, Deng L, He X, Heigold G, Li H (2013) Optimization algorithms and applications for speech and language processing. IEEE Trans Audio Speech Lang Process 21(11):2231–2243
8. Nasef A, Marjanovic-Jakovljevic M, Njegus A (2017) Optimization of the speaker recognition in noisy environments using a stochastic gradient descent. Intern Sci Conf Inform Technol Data Relat Res Sinteza 2017:369–373
9. Healy EW, Yoho SE, Wang Y, Wang D (2013) An algorithm to improve speech recognition in noise for hearing-impaired listeners. J Acoust Soc Am 134(4):3029–3038. https://doi.org/10.1121/1.4820893
10. Geiger JT, Weninger F, Gemmeke JF, Wollmer M, Schuller B, Rigoll G (2014) Memory-enhanced neural networks and NMF for robust ASR. IEEE/ACM Trans Audio Speech Lang process 22(6):1037–1046. https://doi.org/10.1109/TASLP.2014.2318514
11. Sahu SK, Kumar P, Singh AP (2018) Modified K-NN algorithm for classification problems with improved accuracy. Intern J Inform Technol 10:65–70. https://doi.org/10.1007/s41870-017-0058-z
12. Bouafif L, Ouni K (2012) A speech tool software for signal processing applications. In: 6th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT). pp. 788–791
13. Sumithra MG, Ramya MS, Thanuskodi K (2011) Speech recognition in noisy environment using different feature extraction techniques. Intern J Computat Intell Telecommun Syst 2(1):57–62
14. Rahman MM, Saha SK, Hossain MK, Islam MB (2012) Performance evaluation of CMN for Mel-LPC based speech recognition in different noisy environments. Intern J Comput Appl 58(10):6–10. https://doi.org/10.5120/9316-3548
15. Pillai D, Siddavatam I (2019) A modified framework to detect keyloggers using machine learning algorithm. Int J Inf Technol 11:707–712. https://doi.org/10.1007/s41870-018-0237-6
16. Eringis D, Tamulevicius G (2014) Improving speech recognition rate through analysis parameters. Electr Contr Commun Eng 5(1). https://doi.org/10.2478/ecce-2014-009
17. Dave N (2013) Feature extraction methods LPC PLP and MFCC in speech recognition. Intern J Adv Res Eng Technol 1(6):1–5
18. Patil S, Anandhi RJ (2020) Diversity based self-adaptive clusters using PSO clustering for crime data. Int J Inf Technol 12:319–327. https://doi.org/10.1007/s41870-019-00311-z
19. Dekens T, Verhelst W, Capman F, Beaugendre F (2010) Improved speech recognition in noisy environments by using a throat microphone for accurate voicing detection. In: 18th European Signal Processing Conference (EUSIPCO-2010), 1978–1982
20. Sharma K, Sinha HP, Agarwal RK (2010) Comparative study of speech recognition system using various feature extraction techniques. Intern J Inform Technol Knowl Manage 3(2):695–698
21. Rahkar Farshi T, Orujpour M (2019) Multi-level image thresholding based on social spider algorithm for global optimization. Intern J Inform Technol 11:713–718. https://doi.org/10.1007/s41870-019-00328-4
22. Qazi KA, Nawaz T, Mehmood Z, Rashid M, Hafiz AH (2018) A hybrid technique for speech segregation and classification using a sophisticated deep neural network. PLoS ONE 13:e0194151. https://doi.org/10.1371/journal.pone.0194151
23. Joseph FJJ (2020) Effect of supervised learning methodologies in offline handwritten Thai character recognition. Int J Inf Technol 12:57–64. https://doi.org/10.1007/s41870-019-00366-y
24. Nassif AB, Shanin I, Attili I, Azzeh M, Shaalan K (2019) Speech recognition using deep neural networks: a systematic review. IEEE Access 7:19143–19165
25. Gerkmann T, Hendriks RC (2011) Noise power estimation based on the probability of speech presence. In: 2011 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, pp. 145–148
26. Psutka J, Muller L, Psutka JV (2001) Comparison of MFCC and PLP Parameterizations in the speaker independent continuous speech recognition task, Eurospeech 2001, Scandinavia
27. Xie L, Liu ZQ (2006) A comparative study of audio features for audio to visual cobversion in MPEG-4 compliant facial animation. In: Proc. of ICMLC, Dalian, 13–16 Aug-2006
28. Leong ATK (2003) A music identification system based on audio content similarity. In: Thesis of Bachelor of Engineering, Division of Electrical Engineering, The School of Information Technology and Electrical Engineering, The University of Queensland, Queensland
29. Murugappan M, Selvaraj J (2012) DWT and MFCC based human emotional speech classification using LDA. In: International Conference on Biomedical Engineering (ICoBE), Penang, pp. 203–206
30. Prithvi P, Kumar TK (2016) Comparative analysis of MFCC, LFCC, RASTA-PLP. In: International Journal of Scientific Engineering and Research (IJSER) 4(5): 4–7
31. Dua M, Agarwal RK, Biswas M (2018) Performance evaluation of hindi speech recognition using optimized filter banks. Eng Sci Technol Intern J 21(2018):389–398. https://doi.org/10.1016/j.jestch.2018.04.005
32. Hermansky H (1990) Perceptual linear predictive (PLP) analysis for speech. J Acoust Soc Am 87(4):1738–1752. https://doi.org/10.1121/1.399423
33. Hermansky H., Hanson B. and Wakita H (1985) Perceptually based linear predictive analysis of speech, acoustics, speech, and signal processing. In: IEEE International Conference on ICASSP 85, 10:509–512
34. Hermansky H, Morgan N, Bayya A, Kohn P (1991) The challenge of inverse-E: the RASTA-PLP method. IEEE 2:800–804. https://doi.org/10.1109/ACSSC.1991.186557
35. Kim Phil, MATLAB Deep Learning. https://doi.org/10.1007/978-1-4842-2845-6