Journal of
Data Acquisition and Processing

ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.777458

AUTOMATIC SPEECH RECOGNITION USING MODIFIED PRINCIPAL COMPONENT ANALYSIS AND ENHANCED CONVOLUTION NEURAL NETWORK

Dr.M. Kathiresh[1], Dr. R. SankaraSubramanian[2]

[1]Assistant Professor, Department of Computer Applications, CMS College of Science and Commerce, Coimbatore, Tamil Nadu, India Email: kathireshcs83@gmail.com.

[2]Principal, Erode Arts and Science College, Erode, Email: rsankarprofessor@gmail.com

Abstract

Research on speech recognitions were initiated by notions of HMIs (human machine interactions). ASR (Automatic voice Recognition) is method that employs implementable algorithms on computers to translate voice signals as strings of words. Systems can understand human speech inputs. Speech signals transmit two crucial forms of information, including speech contents and identities of speakers. Existing system have issues with speech recognition accuracies and feature extractions. To overcome these problems, in this work, ECNN (Enhanced Convolution Neural Networks) is proposed. The main modules are pre-processing, feature extraction and speech recognition. In pre-processing, noises are removed by the application of Wiener filters for obtaining cleaner speeches. Subsequently, MPCA (Modified Principal Component Analysis) is used for feature extractions where most informative features are extracted. Noise corrupt speech feature matrices are the focus of MPCA and it is demonstrated that the generated sparse partitions reveal speech dominant properties. The ECNN algorithm is subsequently used for speech recognitions and thus enhancing speech recognitions with reduced error rates. The experimental results demonstrate in the conclusion that the proposed MPCA+ECNN algorithm provides better values in comparison with other methods in terms of MSE (Mean Square Error) rates, accuracy, specificity and execution times.

Key words: Speech recognition, Enhanced Convolution Neural Network (ECNN), Modified Principal Component Analysis (MPCA)

1. Introduction

ASR has been used in many facets of our daily life, including automatic phone answering, text dictation, and providing speech commands to computers. Speech recognition is one of the most rapidly developing areas of speech science and engineering. It is also the next big innovation in human-computer interaction in computing technology. Speeches are natural forms interactions between human and they are associated with human physiological capabilities [1] [2]. It is the most vital, efficient, and convenient method of exchanging information. Speech processing is a comprehensive subject and a popular research field that covers a wide range of topics.

Speech recognitions methods involve developments of methods and systems for identifications using speech inputs fed into machines. Since speeches are the most common forms of interpersonal communications, studies need to concentrate on voices as tools in HCIs [5]. Hence, ASRs are seen as integral parts HCI interfaces for speech recognitions and aim to achieve applications that are natural, prevalent, and ubiquitous. Speech recognitions can be categorized as continuous or isolated i.e. dependent or independent of speakers. In contrast to speaker independent techniques, which are frequently unworkable, speaker dependent approaches include training systems to recognise vocabulary phrases spoken once or more by a certain group of speakers [3].The most crucial aspect of voice recognition is feature extractions, which differentiate speeches. Utterances can be retrieved using multiple techniques [6], but the extracted features need to meet certain criteria while negotiating speech signals and include;

- Variations between speaking environments should exist
- Features should be even over a period of time
- Features should represent speeches in a natural way
- Simple to measure,
- Resistant to imitations
- Reduced variables from speaking environments
- Routine occurrences and spontaneity

The study in [8,] proposed a hybrid method based on DNN (Deep Neural Networks) and HMM (Hidden Markov Model). Their schema significantly outperformed traditional GMM (Gaussian Mixture Models) clubbed with HMM in speech recognitions. The improved performance is partly due to the DNN's ability to imitate intricate voice feature correlations. In this work, we show how CNNs may be utilised to further lower error rates. We start off by giving a quick rundown of the fundamental CNN and talking about how it may be used for voice recognition. It debuted a limited-weight-sharing modelling method for speech components. To cope with speaker and environmental variability, local connection, weight sharing, and pooling are a few examples of special CNN architectures that exhibit some degree of invariance to minute variations in speech characteristics along the frequency axis.

The goal of this study is to recognise speech using ECNN algorithms. There have been several studies and approaches introduced, yet accuracies of speech recognitions are not greatly enhanced. The present methods have drawbacks such as high error rates and erroneous categorization findings. To address the aforementioned difficulties, the MPCA+ECNN algorithm is proposed in this study to increase overall recognition performance. This study's key contribution is preprocessing, feature extraction, and voice recognition. The proposed method employs effective algorithms to produce more accurate results for the given datasets.

The remainder of the paper is structured as follows: Section 2 provides a brief assessment of some of the literature studies on preprocessing, feature extraction, and recognition approaches for speech signals. Section 3 describes the suggested approach for the MPCA+ECNN system. Section 4 discusses the experimental results and performance analyses. Finally, Section 5 summarises the findings.

2. Related work

Tao et al. (2020) proposed AV-ASR (Audiovisual ASR) system based on MTL (multitask learning) in [9] where AV-VAD (audio visual voice activity) detections were a secondary goal. The study accomplished a generalizable and dependable AV system that was accurate. AV-ASR performance improved when speech activities were detected in segments. AV-VAD aligned information due to its usage of CTC (connectionist temporal classifications) with loss functions. Their approach mined data directly and learned discriminative high-level representations for both spoken and raw audiovisual inputs. The study takes into account the temporal dynamics that occur between the modalities, resulting in an appealing and practical fusion approach. The researchers compared their approaches using a large audiovisual corpus (nearly 60 hours) with different channels and single and multiple jobs. Their experimental findings showed that their approach could attain best performances of ASR under all conditions and provided better AV-ASR performances in speech activity information, two most critical tasks in speech based applications.

Guglani et al. (2020) in [10] enhanced ASR system performances where pitch and voice qualities were examined. Because of the tonal foundation of Punjabi, their ASR systems based on pitch characteristics were investigated. When assessed in terms of word mistake rates, their system performed well, with improvements due to pitch and voice dependant features. The Yin, SAcC, FFV (Fundamental Frequency Variation), and Kaldi pitch aspects of their proposed ASR system were compared in terms of WERs..

Winursito et al. (2018) in [11], combined MFCC feature extractions with PCA (Principal Component Analysis) for improved accuracy of Indonesian speech recognition systems. Their system's accuracy increased and dimensions reduced due to MFCC and PCA. MFCC-based feature extractions and delta coefficients were used to build matrices, while PCA minimized dimensionalities. Data reductions were accomplished by the usage of PCA variations and subsequently classified using KNN (K-Nearest Neighbour). The study used data created from 140 voice recordings from 28 different speakers. In their experimentations, their first variation of PCA reduced features from 26 to 12 while maintaining speech recognition accuracies at 86.43% and equal to conventional MFCC approaches without PCA. Their second variation of PCA reduced feature counts from 26 to 10 with enhanced recognition accuracies of 89.29% from from 86.43% over MFCC without PCA baseline.

Celin et al. (2020) suggested multi-resolution feature extractions in [12] after dual data augmentations on dysarthric speeches utilising microphone array based virtual linear syntheses. Using the augmented speech data, an isolated word hybrid DNN-HMM-based ASR system was trained on authors' UA and Tamil dysarthric speech corpuses. Their ASR system was designed for low and extremely low intelligible speakers with dysarthria and their results showed lower WERs of up to 32.79% against 35.75% when compared to current dysarthric speech recognition data augmentationss.

Aida-zade et al. (2016) in [13] utilised SVMs (Support Vector Machines) to develop acoustic models of Speech Recognitions based on MFCC and LPC characteristics, evaluated on Azerbaijani data. Multilayer ANN (Artificial Neural Networks) was used on the data set to detect speeches. The study used SVMs for Azerbaijan Speech Recognitions. The range of SVM outcomes with various Kernel functions were studied throughout training phases and it was proved by them that Multilayered ANN do not perform as well as SVMs with Radial Basis and Polynomial Kernels in recognizing speeches.

A CNN-BLSTM hybrid architecture was developed by Passricha et al (2019) in [14] to effectively make use of these characteristics and enhance continuous voice recognitions. CNN achieves higher recognition rates by including sharing of weights, ideal hidden unit counts, and apt pooling approaches. The study focussed on number of BLSTM layers that were useful. The study tried to solve CNN's inability to properly represent speaker-adapted properties, which is another drawback. Subsequently, many non-linearities in recognizing speeches both with and without dropouts were are investigated. In comparison to CNN and DNN systems, their experiments show that hybrid architectures with speech features and non-linearities with dropouts, decreased WER by 5.8% and 10%, respectively.

3. Proposed methodology

The MPCA+ECNN algorithm is proposed in this paper to increase voice recognition performance. Preprocessing, feature extraction, and speech recognition are the three primary processes in this work. Figure 1 displays the proposed MPCA+ECNN system’s overall flow.

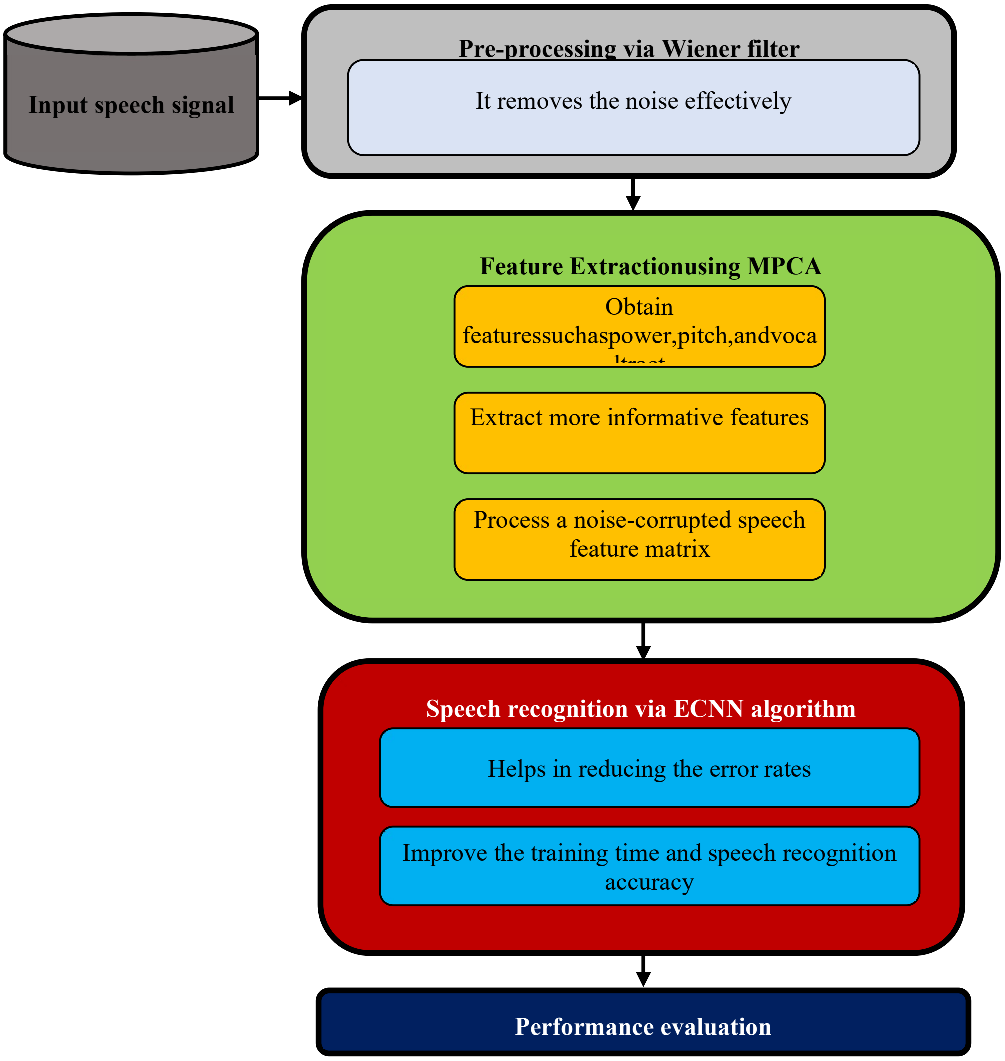
Input speech signal
Pre-processing via Wiener filter
It removes the noise effectively
Feature Extractionusing MPCA
Obtain featuressuchaspower,pitch,andvoca
Extract more informative features
Process a noise-corrupted speech feature matrix
Speech recognition via ECNN algorithm
Helps in reducing the error rates
Improve the training time and speech recognition accuracy
Performance evaluation

Fig 1 Overall flow of the proposed system

3.1 Pre-processing via Wiener filter

In this study, pre-processing is done via Wiener filters.During the early stages of an audio analysis system process, the ability to separate important portions of a spoken signal from a stream of data may be crucial. Signal that are not true signals, are considered as ambient noises which are interferences. In reality, background noises are crucial for determining ASR systems's noise levels. While training and testing data with varying noise levels, performance outcomes of of speech recognitions suffer. SNR (Signal-to-noise ratio) expressed in decibels (dB) measure strengths of appropriate signals in comparison to noise strengths. SNR can be computed as:

$$SNR = 20\log_{10}\frac{V_{signal}}{V_{noise}}$$

(1)

Where V_{signal} is the voltage of correct signal, V_{noise} is the voltage of the noise. Common sources of ambient noises include fans, fluorescent lights, typewriters, computers, backtalk, footsteps, road noise, sirens, bird sounds, and doors opening and closing. Developers of ASR systems frequently have limited control over these disruptions in practical settings. All noise is additive in nature and often steady state, with the exception of impulsive noise generators as typewriters [15]. A close-talk microphone is often used during the training and testing phases to reduce the impact of background noise on speech recognition. The average signal to noise ratio (speech level), when a speaker emits a speech utterance at normal communication level, increases by around 3 dB anytime the microphone filters the speech utterance. To lessen ambient noise, the following filter was used:

$$E_s = 10 * \log_{10}[\epsilon + \frac{1}{N}\sum_{n=1}^{N} S^2(n)]$$

(2)

Where, the $E_s$is log energies of N block samples and ϵ represents minimal positive constants added to avoid log zero computations. $S(n)$ represents nth speech samples in N samples

3.2 Feature Extraction using MPCA

MPCA is used to extract features efficiently in this study while PCA extracts hidden features. Identifying voice signal's properties including powers, pitches, and vocal tract architectures are called feature extractions. The short term amplitude spectrum of the speech wave form really serves as a representation of the information contained in a voice transmission. As a result, we are able to decode speech (phonemes) using the short-term amplitude spectrum. Applying some feature extraction is preferred to reduce the variability. It specifically reduces the impact of a variety of sources of information, including the voice or lack thereof, the periodicity or pitch of the spoken sound, the loudness of the excitation signal, the fundamental frequency, etc.

For economical representations of data, feature space 's intrinsic dimensionalities (observed variables) must be reduced to the level of lesser inherent dimensionalities of the feature spaces (independent variables). This is the case when there is a strong relationship between observable variables. PCA successfully decreases counts of features and projects the dataset in low dimensional subspaces by removing uninteresting components [16]. PCA is a multivariate data analysis used to extract linear features. To efficiently represent datasets of voice signals, the techniques' coefficients are used as feature vectors. PCA may miss important feature information during feature extractions. PCA methods cannot guarantee data regarding pertinent classes are compressed effectively and hence this work uses a modified PCA to handle such shortcomings.

normalising the jth element of ith feature vector, y (y_{ij})with regard to its standard deviation $\sqrt{\lambda_j}$,, impacts of eigenvectors acquiring larger values in MPCA can be minimized and resulting feature vectors y_i' can be written as

$$y_i' = [\frac{y_{i0}}{\lambda_0}, \frac{y_{i1}}{\lambda_1}, \dots . \frac{y_{i(r-1)}}{\lambda_{r-1}}]$$

(3)

A new feature subspace is then created using normalized feature vectors where distances between training and testing features are computed by first normalizing feature vectors by square roots of corresponding eigenvalues.

Linear transformations by (PCA are often depicted by Equation (4):

$$Y = TX \tag{4}$$

where X represent original vectors while Y imply modified vectors, and T stands for transformation matrices generated using Equation (5):

$$(\lambda I - S)U = 0 \tag{5}$$

Where I represents unit square matrices, S stands for original image's covariance matrices, U stands for eigenvectors while λ their eigenvalues. U_j and $\lambda_j (j = 1,2, \dots m)$ are calculated using Equation (2), with eigenvalues ordered as $\lambda_1 \geq \lambda_2 \geq \cdots . \geq \lambda_m$. U (eigenvectors) are expressed as $U = [U_1, U_2, \dots \dots, U_m]$.

Transformed matrices T' are generated by MPCA's selected training samples from input speech signals that are suitable to applications and can be expressed using equations below:

$$Y = T'X \tag{6}$$

$$V_N = b_1 u_1 + b_2 u_2 + \cdots . + b_N u_N \tag{7}$$

$$S = \sum_{i=0}^{1} b_1 u_1; 1 < N \tag{8}$$

The transform matrices and, more significantly, samples used for covariance matrix constructions change between equations (7) and (8) where the former uses training samples, while the latter uses whole hate speech dataset.

PCA is a feature-based approach that extracts noise-resistant speech characteristics. A few of the robust speech representation and voice augmentation techniques that have been successfully implemented using MPCA are briefly described below. It has been demonstrated that the sparse component produced by MPCA on a spectrogram of voice sounds has less noise and is hence noise-resistant [17]. MPCA extracts lowly ranked component of matrices produced by overlapped frames of sub-band signals for processing spectrograms of noise-corrupted signals which are then integrated with RPCA using exemplar based sparse representations for minimized SNR.

The primary advantages of MPCA include minimizations of losses while eliminating extraneous information from data dimensions. The use of statistics and diverse mathematical approaches like Eigen values and vectors, assists processes used by PCA. MPCA is a mathematical process that employs linear transformations to convert data from high to low dimensional spaces. Covariance matrix's eigen vectors can be utilised to identify the low-dimensional space. They have been employed in this study to extract crucial vital acoustic information for speech recognitions due to error minimizations and de-correlations. Voice signal data input contain inputs like means and standard deviations.

(i) Means = total number of data / sum of the number of data
(9)

(ii) Standard deviations: Also known as root-mean square deviations, they represent square roots of squared variations of arithmetic means $\sigma = \surd(\sum(x - x)/n)$
(10)

Algorithm 1: MPCA

```
1.  Start
2.  Find mean values $S'$ of speech signals S
3.  Subtract mean values from S
4.  Generate new matrices A
5.  Covariance obtained from matrices i.e.,$C = AA^T$ Eigen values are obtained from covariance matrixes i.e. $V_1V_2V_3V_4 \dots V_N$
6.  Eigen vectors are computed for covariance matrices $C$
7.  Vectors S are written as linear combinations of Eigen vectors using (7)
8.  Largest eigen values are maintained to reduce dimensionality of the data
9.  Match combinations of features in input speech signals (8)
10. Compute features using mean and Standard deviation (9) & (10)
11. Extract most informative speech features
12. End
```

MPCA extracts greatest and minimal occurring synchrostates, as well as network parameters associated with them. These are subsequently fed into the discriminant. The effectiveness of feature extractions for speech recognition accuracies improves.

3.3 Speech recognition via ECNN algorithm

The ECNN algorithm is used to recognise speech in this paper. The retrieved speech signal is sent into the training and testing phases of the ECNN. The computer system turns the spoken signal into its corresponding written text during the speech recognition process. CNNs are the most powerful deep neural networks, employing multiple hidden layers for convolutions and sub-sampling resulting in extractions of low-high level properties from data inputs. These networks are made up of convolution, sub-sampling or pooling, and completely connected layers. The network's inputs are collections of features and encompass intermediate hidden layers, input layers that receive features as inputs, output layers that create trained outputs. Feature weights are tuned in this proposed ECNN to produce reliable results.

CNN aspects such as local frequency areas, contribute to their optimal performances and strengthen their durability. Moreover, Convolution layer's sparse local connections prevent over fits by using limited parameters to extract lower level attributes from input speech signals. CNNs learn about lower-level properties by processing incoming audio signals with spectro-temporal filters. Differences between the speaker's lips and the microphone, as well as additive disturbances in speech signals, make CNN approaches more effective in analysing distorted voice signals. In convolution layers, different convolution kernels extract feature maps from preceding layers, created by combining learned kernel's input signals.

Convolution and pooling layers are piled, then more fully-connected layers are added on top to create the ECNN. We utilise log mel-filter-bank (with energy term) coefficients with deltas and delta-deltas to retain the spectrogram's local correlations since ECNNs are strong at modelling local structures in inputs.

Convolution layer

Kernels (filters) of sizes a * a are convolved with input voice features of sizes RC in this layer.Blocks of input matrices are convolved with kernels independently to create output pixels. Output features are produced by convolutions of input data with kernels [19]. Filters are commonly referred to as convolution matrix kernels, and feature maps of dimension i * i describe output data features created by convolving kernels and input data. Figure 2 shows a schematic of the ECNN architecture..

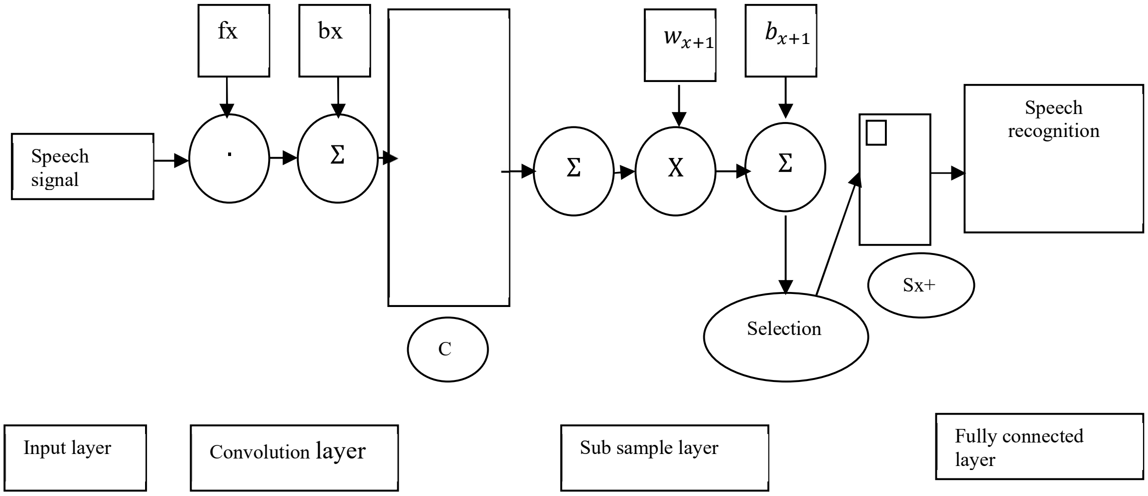

Fig2 Architecture Diagram of ECNN

The feature vector of a CNN is composed of the inputs and outputs of the succeeding Convolution layers, and a CNN may contain numerous Convolution layers. Each convolution layer has a large number of n filters. These filters are convolved with the input, and the number of filters used in the convolution process is equal to the feature map depth (n). Keep in mind that each filter map is regarded as a distinct feature at a specific position in the input image.

The outputs of l-th convolution layers ($C_j^{(l)}$) include feature maps computed using

$$C_i^{(l)} = B_i^{(l)} + \sum_{j=1}^{a_i^{(l-1)}} K_{i,j}^{(l-1)} * C_j^{(l)}$$

(11)

Where, $B_i^{(l)}$ are bias matrices and$K_{i,j}^{(l-1)}$ represent convolution filters or kernel of sizes a∗a that connect the jth feature maps in layers (l − 1) with ith feature mapd in same layers. Output $C_i^{(l)}$ layers consist of feature mapss. The first Convolution layer$C_i^{(l-1)}$ forms input spaces i.e. $C_i^{(0)} = X_i$

The feature map is produced by the kernel. The activation function can be used to apply nonlinear adjustments to the outputs of the convolution layer after it.:

$$Y_i^{(l)} = Y(C_i^{(l)})$$
(12)

Where, $Y_i^{(l)}$ is the output of the activation function and $C_i^{(l)}$is the input that it receives.

Typically used activation functions are sigmoid, tanh, and rectified linear units (ReLUs). In this work, ReLUs which is denoted as $Y_i^{(l)} = \max(0, Y_i^{(l)})$ are used. This function is commonly employed in deep learning models since it aids in the reduction of interaction and nonlinear effects. Unless the input is negative, ReLU returns the same value as the input, in which case the output is set to 0. The advantage of this activation function over other functions is faster training since the error derivative reduces considerably in the saturating area, when weight updates almost disappear. The vanishing gradient issue is what is being discussed here.

Sub sampling Layer

The primary goal of this layer is to minimise the spatial dimensionality of the feature maps produced by the preceding convolution layer. This is done by using a bb-sized mask and conducting a subsampling operation between the mask and the feature maps. It's worth noting that, with the aid of a sub sample layer, the convolution layer may survive rotation and translation of the input pictures. The proposed research study updates optimum weights based on the mean of feature weights, Weighted mean $w_H = \frac{N}{\sum_{i=1}^{N} wx_i}$ (13)

Where,

N − counts of features

w- Feature Weights

x_i- Features\ .b/v/

Pooling layer

Following the element-wise non-linearities, the features will go via a max-pooling layer, which outputs the maximum unit from a set of p adjacent units. Pooling is only done along the frequency axis since it helps to eliminate spectral differences within and between speakers. As a result, the sequence lengths of the feature maps after pooling are the same as those before pooling. After the first Convolution layer, max pooling is performed only once. The assumption is that as more pooling layers are added, units in higher levels will become less discriminative in response to variances in input attributes.

Full Connection

The output layer uses Softmax activation function:

$Y_i^{(l)} = f(z_i^{(l)})$, where $z_i^{(l)} = \sum_{i=1}^{m_i^{(l-1)}} w_H y_i^{(l-1)}$
(14)

where w_HH are weighted Harmonic features are those that must be adjusted by the entire fully linked layer in order to create class representations, and f denotes nonlinearity transfer functions. Rectifier Linear Unit (ReLU) is a piece-wise linear activation function that returns zero in the event of a negative input and the input in the case of a positive input. A single feature map is offered formally. Hello, the definition of a ReLU function is as follows, where H and H are the input and output variables, respectively.

$\widehat{H}_i = \max(0, H_i)$
(15)

Finally, a classifier is employed to connect the completely connected layer and output layer in order to complete the classification selection process. Before creating a classification for a given input, the output probabilities from all ECNN are summed. The average output Si for output i is provided by:

$$S_i = \frac{1}{n}\sum_{j=1}^{n} r_j(i) \qquad (16)$$

where $r_j(i)$ is the output i of network j for a given input pattern.

The method entails assigning a distinct weight to each network. When integrating the findings from the validation set, networks with higher classification accuracy will be given a higher weight. Before making a prediction, the output probabilities from all ECNNs are multiplied by a weight based on some input pattern:

$$S_i = \sum_{j=1}^{n} \alpha_j r_j(i) \qquad (17)$$

Weighted mean is used to calculate the weight in this proposed study work. The weight calculation is as follows:

$$\alpha_k = \frac{A_k}{\sum_{i=1}^{n} A_i} \qquad (18)$$

Where, $A_k$is accuracy in the validation set for the network k, and i runs over the n. The ECNN network's average output indicates that incoming speech signals are recognised more accurately.

Algorithm 2: ECNN
Input: Speech signal
Output: Recognized speech

1. Start
2. Get input as audio signal
3. For all input signal, describe speech recognition ∈ dataset do
4. Perform pre-processing to eliminate noises from original signals
5. Extract the features using MPCA
6. Train the ECNN algorithm
7. Sublayer the input by converting it.
8. Find the properties of the speech signal
9. Using vocal duration and pitch information, extract more detailed and pertinent elements.
10. Conduct training and testing on the provided dataset.
11. Copy each feature's preset speech feature label according to the input dataset.
12. Provide better results for speech recognition

4. Experimental result

All SA entries are eliminated and the TIMIT corpus and standard 462-speaker training set are utilised. A 50-speaker development set is utilised for early pausing. The 192 sentences in the core test set are used for evaluation. Original audios get converted as 40-dimensional log mel-filter-banks (with energy term) coefficients with deltas and delta-deltas, which are then transformed into 123 dimensional characteristics. Each dimension is normalised throughout the training set to have a zero mean and unit variance. We use 61 phone labels for training and a blank label for scoring, and the output is mapped to 39 phonemes..

In terms of accuracy, MSE, specificity, and execution time, the proposed MPCA+ECNN technique outperforms the existing SVM [21] and CNN [22] methods for speech recognition systems.

Accuracy

Accuracies are defined as model's overall correctness and computed as total actual classification parameters $(\mathrm{T_p} + \mathrm{T_n})$ divided by sums of classification parameters $(\mathrm{T_p} + \mathrm{T_n} + \mathrm{F_p} + \mathrm{F_n})$. Accuracy can be calculated as:

$$\text{Accuracy } = \frac{\mathrm{T_p}+\mathrm{T_n}}{(\mathrm{T_p}+\mathrm{T_n}+\mathrm{F_p}+\mathrm{F_n})} \quad (19)$$

Where $\mathrm{T_p}$ is True positive, $\mathrm{T_n}$is true negative, $\mathrm{F_p}$is false positive and $\mathrm{F_n}$is false negative

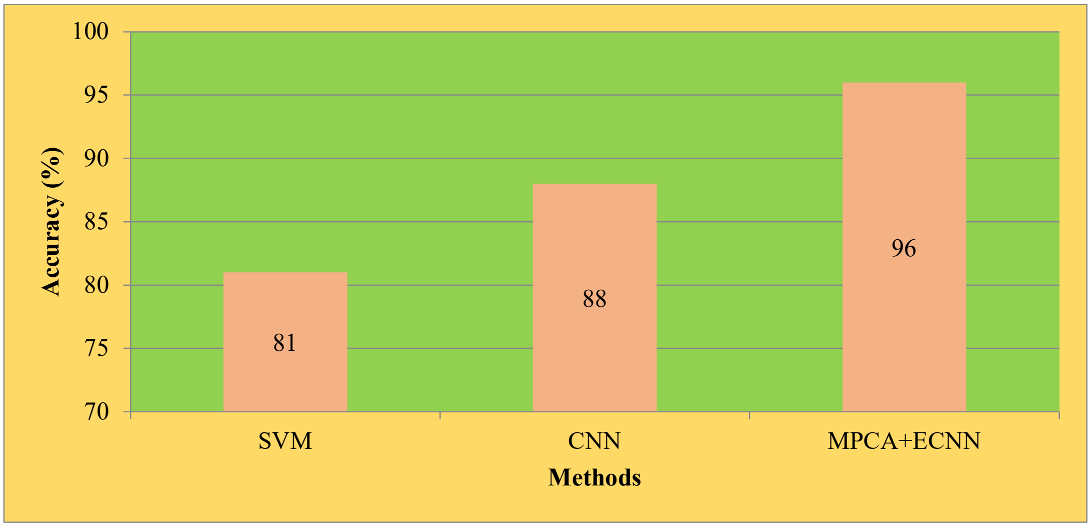

Fig 3 Accuracy

Fig. 3 examines accuracies of existing and new approaches. Accuracy values are represented on the y-axis, with datasets and techniques forming the x-axis. Existing approaches such as SVM and CNN algorithms provide less accuracy for given speech signals, however the suggested MPCA+ECNN algorithm provides more accuracy. The pre-processing method improves classification accuracy by removing noise. As a result, the proposed MPCA+ECNN algorithm improves speech recognition accuracy through feature extraction.

Specificity

Specificity (also known as the real negative rates) are percentages of actual negatives that are accurately identified (for example percentage of healthy people correctly identified as people with illness).

$$\text{Specificity} = \frac{\mathrm{T_n}}{\mathrm{T_n}+F_p} \tag{20}$$

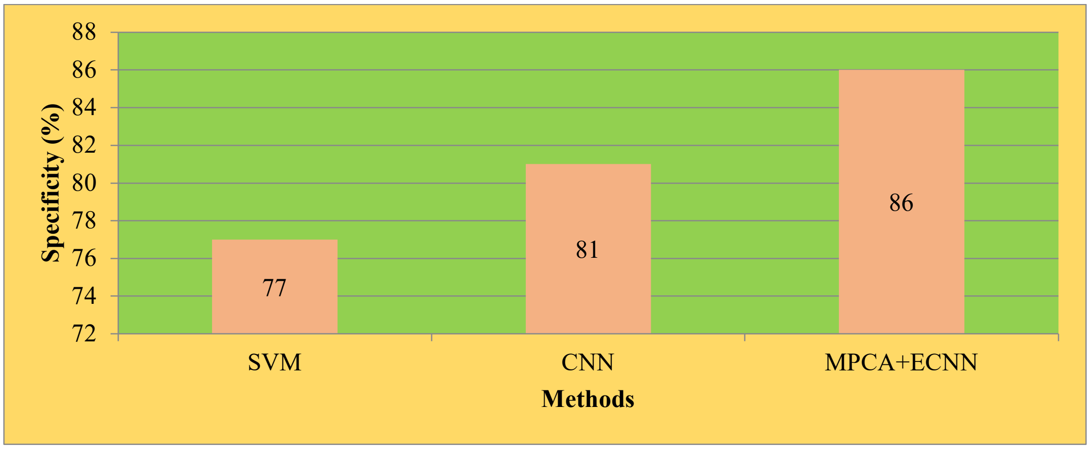

Fig 4 Specificity

The comparison measure is assessed for specificity using both established and new methodologies, as shown in Fig. 4. For the presented dataset, existing methods like SVM and

CNN algorithms offer lesser specificity, however the suggested MPCA+ECNN algorithm provides more specificity. As a result, the suggested MPCA+ECNN improves the performance of the voice recognition system by efficient information extraction.

MSE

MSE of estimators (processes for estimating unobserved variables) in statistics measure averages of squares of errors or average squared difference between estimated and actual values.

$$MSE = 1/n \sum_{i=1}^{n} (Y_i - \widehat{Y_\iota})^2 \qquad (21)$$

n is number of samples, Y_i is observed values and $\widehat{Y_\iota}$ is predicted values

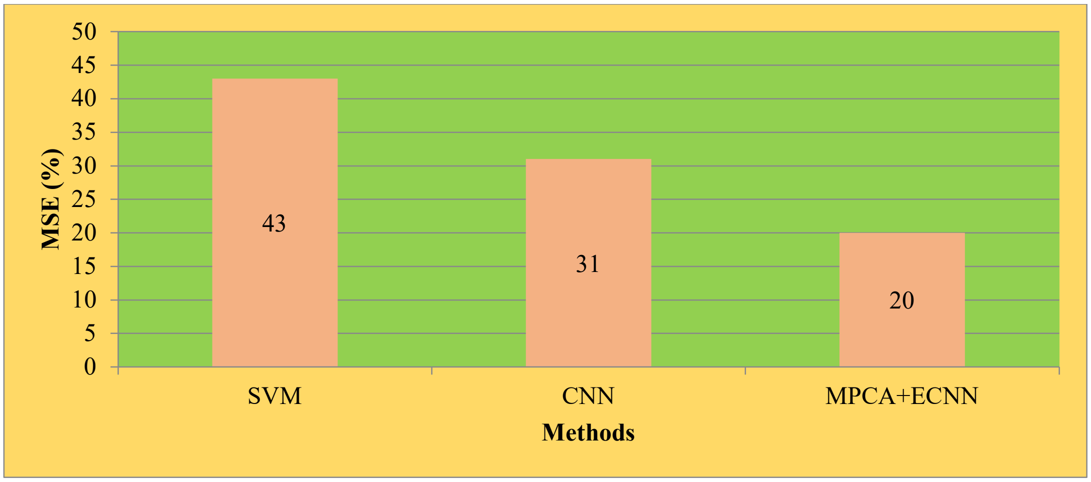

Fig 5 MSE

The comparison measure is assessed in terms of MSE using both the existing and suggested approaches, as shown in Fig. 5. The techniques are shown on the x-axis, while the MSE value is shown on the y-axis. Existing approaches, such as the SVM and CNN algorithms, yield a higher MSE for the given dataset. however the suggested MPCA+ECNN algorithm delivers lower MSE. As a result, the suggested MPCA+ECNN improves speech recognition accuracy through an effective feature extraction technique.

Execution time

When the proposed method executes in less time, the system improves.

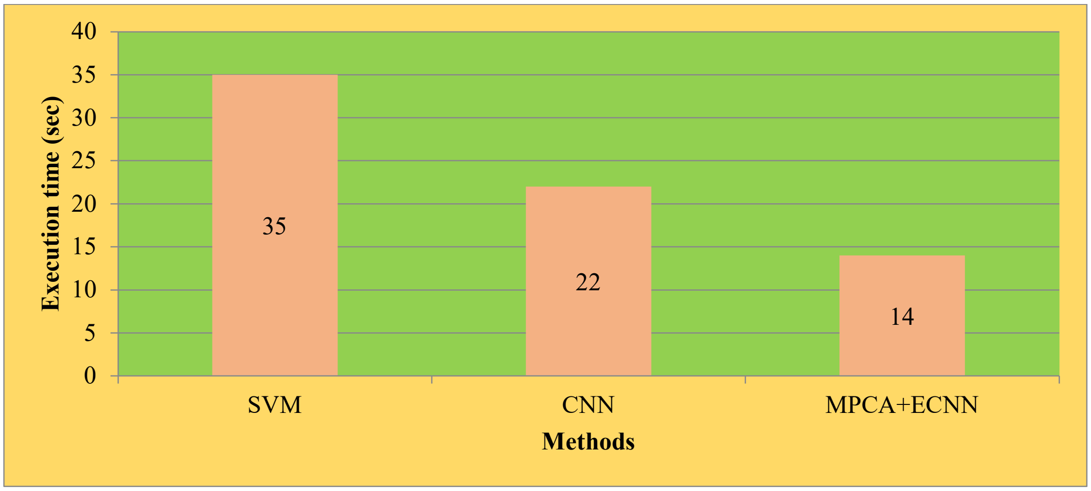

Fig 6 Execution time

As illustrated in Fig. 6, the comparative measure is calculated in terms of execution time using both the existing and recommended methodologies. The execution time value is shown on the y-axis, while the approaches are shown on the x-axis. For the given dataset, existing techniques like SVM and CNN algorithms take longer to run, however the suggested MPCA+ECNN algorithm has a shorter execution time. As a result, the suggested MPCA+ECNN improves speech recognition accuracy through an effective feature extraction technique.

5. Conclusion

The MPCA+ECNN approach is recommended in this research to improve the performance of the speech recognition system for the given dataset. Speech recognition, feature extraction, and pre-processing are the three main sections of this study. Pre-processing reduces noise levels in an effort to improve dataset quality. Then, feature extraction is carried out utilising MPCA in order to extract more useful features. Last but not least, voice recognition using the ECNN algorithm enhances the continuous speech recognition problem. With ECNN, error rates are significantly lower and voice recognition accuracy is increased. The experimental findings showed that the proposed MPCA+ECNN algorithm outperforms existing strategies in terms of better values in MSE, accuracy, specificity, and execution time. Future research can use a larger number of speakers utilising the scaled dataset. Furthermore, by creating a hybrid model that uses many machine learning techniques, the work may be developed in the future to enhance the performance of the system.

References

1. Schönherr, Lea, et al. "Imperio: Robust over-the-air adversarial examples for automatic speech recognition systems." *Annual Computer Security Applications Conference*. 2020.

2. Avram, Andrei-Marius, P. A. I. S. Vasile, and Dan Tufis. "Towards a Romanian end-to-end automatic speech recognition based on Deepspeech2." *Proc. Rom. Acad. Ser. A*. Vol. 21. 2020.

3. Park, Daniel S., et al. "Specaugment: A simple data augmentation method for automatic speech recognition." *arXiv preprint arXiv:1904.08779* (2019).

4. Tamazin, Mohamed, Ahmed Gouda, and Mohamed Khedr. "Enhanced automatic speech recognition system based on enhancing power-normalized cepstral coefficients." *Applied Sciences* 9.10 (2019): 2166.

5. Li, J., Gadde, R., Ginsburg, B., &Lavrukhin, V. (2018). Training neural speech recognition systems with synthetic speech augmentation. *arXiv preprint arXiv:1811.00707*.

6. Gupta, Harshita, and Divya Gupta. "LPC and LPCC method of feature extraction in Speech Recognition System." *2016 6th international conference-cloud system and big data engineering (confluence)*. IEEE, 2016.

7. Yang, Chao-Han Huck, et al. "Decentralizing feature extraction with quantum Convolution neural network for automatic speech recognition." *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021.

8. Palaz, Dimitri, Mathew Magimai-Doss, and Ronan Collobert. "End-to-end acoustic modeling using Convolution neural networks for HMM-based automatic speech recognition." *Speech Communication* 108 (2019): 15-32.

9. Tao, Fei, and Carlos Busso. "End-to-end audiovisual speech recognition system with multitask learning." *IEEE Transactions on Multimedia* 23 (2020): 1-11.

10. Guglani, Jyoti, and A. N. Mishra. "Automatic speech recognition system with pitch dependent features for Punjabi language on KALDI toolkit." *Applied Acoustics* 167 (2020): 107386.

11. Winursito, Anggun, RisanuriHidayat, and AgusBejo. "Improvement of MFCC feature extraction accuracy using PCA in Indonesian speech recognition." *2018 International Conference on Information and Communications Technology (ICOIACT)*. IEEE, 2018.

12. Celin, TA Mariya, T. Nagarajan, and P. Vijayalakshmi. "Data augmentation using virtual microphone array synthesis and multi-resolution feature extraction for isolated word dysarthric speech recognition." *IEEE Journal of Selected Topics in Signal Processing* 14.2 (2020): 346-354.

13. Aida-zade, Kamil, AnarXocayev, and Samir Rustamov. "Speech recognition using support vector machines." *2016 IEEE 10th international conference on application of information and communication technologies (AICT)*. IEEE, 2016.

14. Passricha, Vishal, and Rajesh Kumar Aggarwal. "A hybrid of deep CNN and bidirectional LSTM for automatic speech recognition." *Journal of Intelligent Systems* 29.1 (2019): 1261-1274.

15. Shrawankar, Urmila, and Vilas Thakare. "Noise estimation and noise removal techniques for speech recognition in adverse environment." *Intelligent Information Processing V: 6th IFIP TC 12 International Conference, IIP 2010, Manchester, UK, October 13-16, 2010. Proceedings 6*. Springer Berlin Heidelberg, 2010.

16. Taguchi, Y. H., and Yoshiki Murakami. "Principal component analysis based feature extraction approach to identify circulating microRNA biomarkers." *PloS one* 8.6 (2013): e66714

17. Hung, Jeih-weih, Jung-Shan Lin, and Po-Jen Wu. "Employing robust principal component analysis for noise-robust speech feature extraction in automatic speech recognition with the structure of a deep neural network." *Applied System Innovation* 1.3 (2018): 28.

18. Zhang, Ying, et al. "Towards end-to-end speech recognition with deep Convolution neural networks." *arXiv preprint arXiv:1701.02720* (2017).

19. Han, Wei, et al. "Contextnet: Improving Convolution neural networks for automatic speech recognition with global context." *arXiv preprint arXiv:2005.03191* (2020).

20. Chouhan, Kuldeep, et al. "Structural support vector machine for speech recognition classification with CNN approach." *2021 9th International Conference on Cyber and IT Service Management (CITSM)*. IEEE, 2021.

21. Alsobhani, Ayad, Hanaa MA ALabboodi, and Haider Mahdi. "Speech Recognition using Convolution Deep Neural Networks." *Journal of Physics: Conference Series*. Vol. 1973. No. 1. IOP Publishing, 2021.