Optimization of Convolutional Neural Network Architectures for High-Accuracy Spoken Digit Classification Using Mel-Frequency Cepstral Coefficients

Pratibha Rashmi[1], Manu Pratap Singh[2], Punj Prakash[3]

[1]Department of Computer Science, Dr. BhimaraoAmbedkar University, Agra, UP, India, Email: pratibha.rashmi@gmail.com
[2]Department of Computer Science, Dr. BhimaraoAmbedkar University, Agra, UP, India, Email: manu_p_singh@hotmail.com
[3]American Airlines, TX76155, United States, Email: punjprakash@gmail.com

Received: 16.04.2024 Revised : 11.05.2024 Accepted: 24.05.2024

ABSTRACT
Sound recognition is the ability of machine learning to identify spoken words. Different approaches have led to various attempts to implement automatic sound recognition systems. In recent years, convolutional neural networks (CNNs) have gained acceptance for performing various classification tasks in computer vision and voice assistants due to their capability of adaptation & learning and to overcome the accuracy issues in traditional machine learning methods. In this present paper, we are optimizing the convolutional neural network architecture and the number of parameters in the network with the improvement in accuracy for sound classification of spoken digits. We train the convolutional neural networks using the feature extraction of Mel-frequency cepstral coefficients (MFCCs). We examine the two different CNN models. In the first model, the three convolutional blocks with max-pooling layers are used, followed by a fully connected network with a classification layer. In the second model, global pooling is used after the convolutional blocks and passes the feature map directly to the classification layer. The experimental results from both networks are obtained and analyzed on the existing datasets of spoken digits. The analysis of the obtained results indicates that the taxonomic accuracy of the proposed optimized architectures of CNN surpasses the existing pre-trained CNN models with MFCC feature extraction for the classification of spoken digits.

Keywords: Convolutional Neural Network, Spoken Digits Recognition, MFCC, Delta MFCC, Global Max Pooling, Deep Learning

1. INTRODUCTION

Anautomatic speech recognition (ARS) or speech-to-text conversion explores the platform to applying machine learning techniques for the sound recognition of individual being from spoken words [1]. Automatic speech recognition has wide range of applications in various domains. In the recent development, Speech recognition techniques are used for human-machine interaction [2]. It generated a massive impact on the speech recognition diligencefor the smart houses, automatic driving, in gaming & entertainment. Generally, the voice recognition of any human being depends on the physiological and behavioural characteristics of the speaker and also on the linguistic pattern. Speech recognition system consists of audio signals. These signals are converted into digital form and the different machine learning methods are used to understand the template pattern information in the input stimuli. Thus, Automatic speech recognition is the capability of a computer machine to understand speech and to classify it as per pre-defined labels or classes. Thus, classification process of sound signals can be shown in the Fig 1.

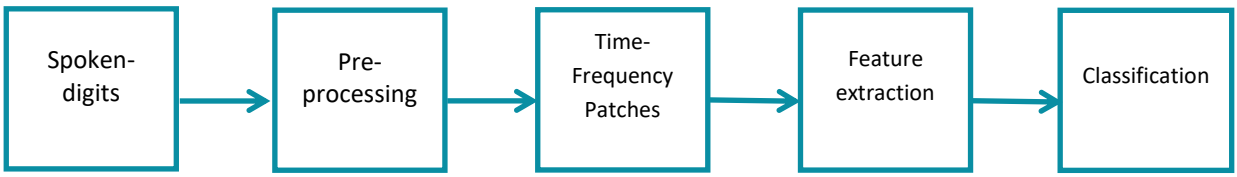

Figure 1. Classification Process for Spoken Digits

In this process, some techniques involve parameter transformations, which entail transforming obtained features into signal parameters using a separation and concatenation procedure [3]. In most of the sound recognition methods, the sound signals are transformed into a spectrogram by using the splitting technique and the signal is divided into time frames to generate the spectrogram. Then after using FFT, each frame is splitting into the frequency components. Hence, a vector of amplitudes at each frequency is used to describe each time frame [4]. The training set of the speech signals is constructed by using these frame vectors. The next step in this process is to perform the classification task. There were different methods proposed in the literature for the pre-processing step & classification.

For decade, the Gaussian Mixture Model (GMM) and Hidden Markov basedmodels (HMM) ruled the voice recognition due to many advantages including their mathematical elegance and capability to model time varying sequences [5]. The limitation ofHMM is the requirement of a large amount of training data. The GMM can successfully separatethe noise from the speech in noisy speech utterances but it increases the computational complexity [6]. The traditional approach of speech recognition focuses on representing eachword by its feature vector and pattern matching with the statistically available vectors using neural networks [7]. On the opposite of the earlier method of HMM, neural networks do not requireprior knowledge of the speech process and do not need statistics of speech data [8]. In recentdevelopment, it has been seen that the Artificial Neural Network (ANN) performance was not adequate for the classification of large size data. It showed the poor generalization for the largedataset. Beside this, there are two major problems occurs in classification of sound signals. The first problem is related to the availability of noise in the environment, and the second problem is related to the cutting of leading and trailing edges, thus, there is a time delay between when people make sounds and speech detection. Therefore, it is necessary to explore some effective methods for the feature extraction and sampling for the formation of the pattern vector. Various methods for feature extraction such as MFCC, linear prediction coefficient (LPC), Linear Prediction Cepstral Coefficients (LPCC), Line spectral Frequencies (LSF) and many more are generally used [9]. Hence, with the development of neural network processing for the pattern classification, the deep learning emerged as a new attraction area of machine learning for the pattern recognition.

The deep neural network models consist with multiple layers. These models are usually neural networks consisting of different levels of non-linear operations. The deep neural network architecture learns with deep learning algorithm by extracting specific features and information [10]. Thus, it learns from successive by organized layers from extracted features from each layer at a time. Therefore, the learning is achieved by training each layer in supervise mode. Hence, due to these capabilities of deep learning algorithms, various applications of sound recognition system are explored. It has been found that deep neural networks yield better results than the classical or conventional methods of speech recognition [11]. Mostly, the convolutional neural network architecture of deep learning gained the popularity for classification of images due to its ability of deeper layer architecture with different methods of normalization and pooling schemes [12]. The pre-trained models of CNNs used in various applications related to image classification labelling tasks. Later on, different approaches have been reported in the area of deep neural networks in speech related applications [13]. Initially, pre-trained deep learning models were used for sound-signal classification, treating spectrograms of sound samples as images [14]. Subsequently, the various capabilities and limitations of deep learning in speech recognition were explored [15]. Furthermore, state-of-the-art solutions for automatic spoken language recognition were presented from both computational and phonological perspectives [16]. A huge progress has been reported in the area of spoken language recognition.

There are several aspects of language recognition such as language characterization and modeling methods. Deep learning can make full use of the correlation between features and merge speech features from consecutive rates substantially [17].Further, another attempt has been considered to incorporate deep learning models into existing components such as HMM to implement the hybrid approach. This hybrid approach uses deep learning models with the language decoding component. Thus, the deep learning models are found more effective and efficient in yielding the automatic sound recognition system as well as for the recognition of spoken language or words [18]. The CNN structure allows us to acquire frequency invariant features when applied along the frequency axis [19]. Thus, the convolution neural network is considered as suitable choice for performing the task of spoken digit classification. Therefore, in spoken digit classification task, CNN models of deep learning are used to learn discriminative features directly from the audio signals. These models use spectrogram or raw waveform as input and yield good performance in classification. The convolutional neural network models of deep learning are capable to accommodate the spoken digits as the spectrogram like inputs and by using filters, the model is able to learn the spectro-temporal patterns of sound signal of spoken digit and differentiate them for the classification purpose [20]. Thus, convolutional neural networks (CNNs) have exhibited the superior

performance for the classification of spoken digits over the conventional or earlier proposed techniques due to its ability for deeper feature extraction. Beside this, the CNN requires less pre-processing and it is independent from the prior knowledge. In this present paper, we are optimizing the convolutional neural network architecture and its number of parameters during the training beside to improve the accuracy for classification of spoken digits. In this proposed technique, we considered the MFCC features from the datasets of spoken digits to construct the training set. This training set considered with feature map of input data and presented for the training to the CNN. The two different CNNs are considered for the feature extraction and classification. In first architecture of CNN, the three convolutional blocks with max-pooling layers are used followed by a fully connected network with classification layers. In second model the global pooling is used after the convolutional blocks and passed the feature map from the pooling directly to the classification layers. The experimental results from both the network models are obtained and analysed on the dataset of spoken digits. The analysis of the obtained results predicts that the taxonomic accuracy of the proposed optimized architectures surpasses the existing pre-trained CNN methods with MFCC feature extraction for the classification of spoken digits.

This paper is organized as follows: Section 2 of the paper discusses the different literature review on classification of spoken digits. Section 3 describes the pre-processing and feature extraction techniques of the proposed models. Section 4describes the architecture of proposed CNN models. Section 5 of the paper consist the implementation and experiment details of proposed models of convolution neural network.Section 6includes the results and discussion. Finally, the conclusion is presented in the last section followed by the references.

2. RELATED WORK

With the development of ANNs, deep learning dominates the field of pattern recognition, particularly in image recognition and image labeling. It is essentially a method of training deep structural models and also an algorithm for modelling complex relationship between data through multiple layers [21], which is currently used in speech and image recognition, machine translation and social filtering. In the last decade, it has been observed that the deep neural networks obtained better accuracy than Gaussian Mixture Model (GMM),Hidden neural models for classification of spoken words or digits [22]. The convolutional neural networks and recurrent neural networks have been used widely in the development of automatic speech recognition systems [23]. An overview is presented on the deep neural network that incorporate many number of hidden layers which are trained using some new techniques with optimization of learning parameters [24]. The results showed that the deep neural networks that incorporate many hidden layers and are trained by new techniques outperform from all the existing convolutional methods of machine learning on different speech recognition problems. Further, Peng Y. [25] focused on the different work that can be utilised to improve the learning performance. It included the enhancement in network architectures, activation functions, optimization methods and learning. It has been observed that these enhancements improve the performance of deep neural network in other signal processing applications beside the speech recognition. Convolutionalneural networks have been explored in many sound classification tasks [26].

A composite method has been introduced [27] with a non-homogeneous classification CNN and support vector machine (SVM). In this, the softmax layer of CNN is substituted by SVM. In a further development, Yong Xuebin et al. designed a speech recognition system for a group of words and used these methods of classification including CNNs [28]. The recognition of isolated alphabets or digits of any language involves ambiguity due to their short acoustic duration and some alphabets or digits are acoustically identical to each other [29]. Digit recognition has been considered one of the challenging tasks in the sound recognition domain which includes the spoken digits of English, Portuguese, Arabic and Bangla [30]. The CNN and RNN are used to obtain the lattice-free MMI training and i-vector modelling explored the remarkable achievement for all model's architecture [31].Further, it is obtained that LSTM performed significantly in a Spiking Neural Network (SNN). In this approach, the success rate for recognizing any digit has been obtained up to 88% and it will increase upto 99.4% with more conventional pre-processing techniques [32]. Various other hybrid techniques were used for the recognition of spoken digits in different languages [33]. In some of the research work, the TIDIGITS dataset is used [34] which contain 2,412 training utterances and 1,144 test utterances.

In other work, the OGI Multilingual corpus is utilized with 8126 samples for training and 454 samples for testing [35]. In all these methods Mel-Frequency Cepstral Coefficients (MFCC) is used for feature extraction from the spoken digits sound samples and further the principal component analysis (PCA) is used for dimension reduction from the features followed by the SVM [36] for the classification. An intelligent-based System is proposed which used the Deep Feed Forward Neural Network (DFNN) with hyper-parameter optimization techniques for the classification of Spoken English digit data [37]. Short-

Term Fourier Transform (STFT) was used for extracting the features from the input samples before performing one-hot encoding to make labels into compatible format for training a classification model.
Further, the classification approach has been extended to use Random Forest (RF), Gradient boosting (GB) and DFNN for the classification of the Spoken English digit data [38]. Different approaches on feature extraction were also used with different methods of classification for the Spoken English digits [39].The vector quantization is explored for the feature matching using Euclidian distance matric aiming to recognise a spoken digit [40]. In this approach, the digits (0-9) dataset is created in a single file with 10-second length with microphone as 15 persons in 3 sessions with time-gap of one week duration; the training set consist of two sessions while the third session is used for the test set.The accuracy up to 93.3% is achieved. In [41] Dynamic Time Wrapping (DTW) is used to detect speech digit recognition. In this approach, first zero crossing and energy parameters were detected for the digit boundary detection, thereafter, MFCCs are used to provide an estimate of the vocal tracts filters which are fed into a DTW classification. Here, isolated English digits (0-9) were recorded and accuracy of 95% has been achieved for 100 samples. Further, a mechanism to speaker-independent connected digit recognizer for Malayalam language using Perceptual Linear Predictive (PLP) ceptral coefficient is proposed for speech parameterization and continuous density HMM to the recognition process. This proposed approach considered the data recording with Samsung galaxy smartphones in noiseless condition with 10 speakers and obtained the accuracy up to 98.6% in recognition and 64.8% for speaker dependent and speaker independent respectively [42].
Ali et al., [43] have studied the Urdu language and used the MFCC, delta and delta-delta features and classified by support vector machine, a random forest and linear discriminant analysis classifiers. In this study, a comparison among SVM, RF, and LDA has been conducted, and it was reported that the performance of SVM is better than the other methods. Further, Chapaneri[44] extracted WMFCC features and used improved features for DTW algorithms for speaker independent spoken English digits. In this method, the data has been considered from TIDIGITS dataset which consists of 10 male, 20 female speakers and 400 utterances that uses 240 utterances for training and 160 utterances for testing and 98.13% accuracy is obtained. Various feature extraction techniques with different classifiers were proposed for the spoken digit classification and speaker identification. Different models of deep learning with other classifiers were also proposed in hybrid manner to improve the accuracy of spoken English digits but all these approaches utilized small datasets with conventional feature extraction and different classification methods. It is observed that the small datasets can be prone to overfitting and exhibits the poor generalization. Beside this, rest of the feature extraction methods and classification techniques have been superseded by deep learning techniques in recent times. One of the important works that could be found is on the audio MNIST dataset where deep learning techniques are utilised on a dataset of 30,000 audio digit samples but this dataset does not contain non-digit audio samples [45]. In further development, the HMM is used with MFCCs for the audio digit samples of Bengali language and obtained the accuracy more than 95% for the digits from 0 to 5 and 90% for the digits from 6 to 9. These results indicated the variation in accuracy for some digits due to confusion of different dialects of two pairs of digits [46]. Different approaches for feature extraction are used as the pre-processing steps prior to training and convolutional neural networks used as the classifier. Since, CNN's basic instinct is to work as the image classifier. So that, being an image classifier, challenge is to find an appropriate image-like representation of the spoken digit signals. Different time-frequency representation of audio-signals is considered with CNN. The time-frequency representation i.e., Mel-Spectrogram is probably the most common and has been used in many applications of spoken digit recognition [47].
Another common approach is reported with the use of frequency domain filter banks. In this approach, two commonly used filters were moving i.e., average filters and mel-filters. Hence, the resulting time-frequency representations are referred as smooth-spectrogram and mel-spectrogram respectively. It has been observed that these representations of sound samples shown to be useful in speech and acoustic event classification. Further, the spectrogram-smoothed-spectrogram and mel-spectrogram forms the baseline method is proposed [48]. Beside this, the proposed work investigates the formation of time-frequency representation using wavelet transform due to the capability of wavelet transform to offers better frequency localization in the lower frequency range to make it more suitable for speech classification tasks compared to conventional techniques.
Furthermore, different time-frequency representation reveals spectral information at different frequencies. The CNN is employed as classifier with these techniques to improve the classification performance [49]. Here, in our proposed work the existing methods of CNNs with MFCC feature extraction are used for the classification of spoken digits. The main contribution of the proposed work is to optimize the number of parameters of convolutional neural network. In this optimization process for the first CNN model, convolutional block were used. In each convolution block the convolution layer and

local max pooling used. The feature map obtained from the last convolution block is directly applied to the dense network for classification. In the second model again the convolution blocks are used in each block, convolution layer and local max-pooling is used. The feature map obtained from the last convolution block is presented to the global pooling layer. The output of global pooling layer is directly passed to the dense network without any flatten layer. The proposed architecture replaces the flatten layer with global pooling layer. Thus, the CNN architecture is optimized with flatten layer of conventional CNNs. The batch normalization is used with every convolutional block for the parameter optimization in both the proposed models. Apart from this, the regularization has been applied after each pooling layer in both the architectures. The proposed methods for optimization of the CNNs models not only reduce the number of parameters but it also improves the classification accuracy and considered the less number of iteration for the convergence during the training.

Hence, to consider the recent development in spoken digit recognition with deep learning techniques, the proposed work of this paper presents the hybrid feature extraction techniques on the existing dataset of English-spoken digits. The removal of noise and background sound from each isolated digit is performed with feature extraction. The optimized convolution neural network is selected for the classification. Different combinations of pooling and convolution layers are evaluated to select the optimized CNN architecture for the classification. The different optimized CNN models were proposed and their performance has been evaluated on the selected dataset.

3. Pre-Processing and Feature Extraction

In our proposed work, we considered the sound samples of English digits spoken by different people. The dataset Speech Commands v1 (SCV1) is used to collect the sound samples. The Speech Commands V1 (SCV1) [50] dataset is composed of single spoken English words. It consists of64,727 one-second .wav audio file of 30 common speech commands. The audio files are arranged into the folders based on the word they contain. So we have used only the folders that contain the digits. We used the total 13,323 samples in which 9,992 for training and 3,331 samples for testing the models and total ten (10) classes are used for the classification of input samples.The digit class distribution of speech_commands_v1 (SCV1) dataset can be shown in Fig. 2.

	File Names	Number of Sample
0	one	1276
1	five	1092
2	four	2400
3	six	1485
4	seven	1411
5	eight	1113
6	three	1188
7	nine	1144
8	two	908
9	zero	1306

Figure 2. Class distribution of Speech_Commands_v1 (SCV1) dataset

Different methods are used in combination for feature extraction as a pre-processing step to construct the training and test sets of sound samples for convenient and effective classification with CNN. One of the very common methods i.e., MFCC or mel-frequency cepstral coefficient, and its derivatives is used to extract the features from the audio files using the 'Libros' method of Python [51], and append into Python 'NumPy' array with the careful labeling of data as 0, 1, 2, 3, 4, 5, 6, 7, 8, and 9. The collected waveform is partitioned into frames of fixed duration. These frames are further passed to the windowing of waveform. The DFT is used to construct the time-frequency representation of the audio waveform. The absolute value of the DFT transform is computed and time-frequency representation is resized to a dimension of 64 x 64. Here, we consider the image-processing technique ofinterpolation. Various interpolation kernels are available for this purpose but we consider the bicubic interpolation in time-frequency image resizing [52] as:

$$k(x, y) = \sum_{i=0}^{3} \sum_{j=0}^{3} a_{ij} x^i y^j \quad (1)$$

The interpolation can be computed by applying the convolution using the following kernel in both dimensions. `

$$k(x) = \begin{cases} \frac{3}{2}|x|^3 - \frac{5}{2}|x|^2 + 1 & |x| \le 1 \text{ ` } \\ -\frac{1}{2}|x|^3 - \frac{5}{2}|x|^2 - 4|x| + 2 & 1 < |x| \le 2 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

The plot of time-domain signal for the spoken-digit zero and its spectrogram and scalogram representation can be presented in Fig. 3.

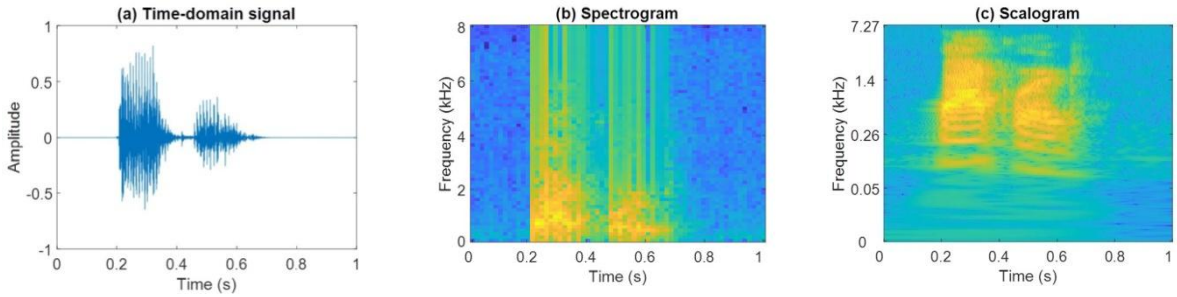

Figure 3. (a) time-frequency representation (b) spectrogram representation (c) Scalogram representation for English spoken word "zero"

Therefore, the melfilterbank, log & discrete cosine transformation (DCT) applied to construct MFCC extracted feature vector for the training to convolutional neural network. The whole process of the pre-processing can be shown in Fig. 4.

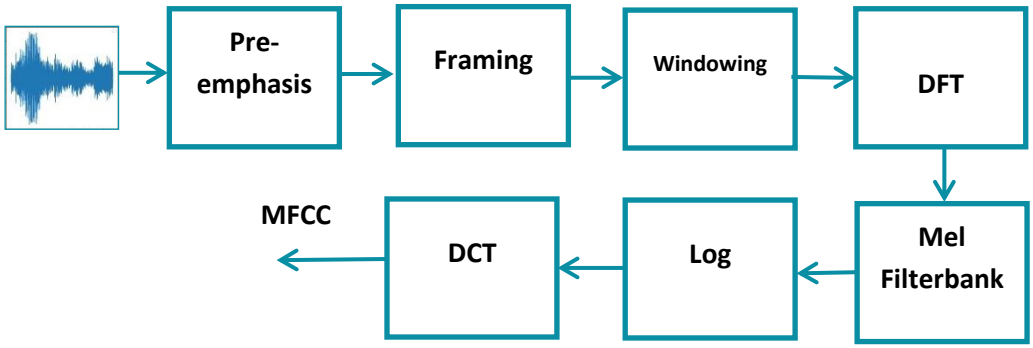

Figure 4. MFCC Feature extraction process

In this process, pre-emphasis step is applied with a high-pass filter to the audio signal to emphasize the higher frequencies and balance the spectrum. This step helps to improve the stability and quality of the subsequent processing. The relationship between the inputand output signal can be shown as.

$$S[m] = X[m] - \alpha X[m-1] \quad (3)$$

Where, $S[m]$ is the output pre-emphasis signal and $X[m]$ is the input signal. We consider default value of α (i.e. 0.97). Second step is framing, framing divide the pre-emphasized signal into short frames of typically 20-40 milliseconds, with a certain overlap between adjacent frames. This framing allows us to analyse the signal's spectral content over short-time windows. After framing, windowing is done for minimizing the disruptions at the starting and at the end of the frame. Multiply each frame by a window function (e.g., Hamming window) to reduce spectral leakage caused by abrupt signal endings. The output after windowing the signal can be presented as:

$$S(m) = X(m) * W_n(m) \quad (4)$$

Where, $0 \le m \le N_m - 1$ and $S(m)$ represents the output signal, $X(m)$and $W_n(m)$ represents the input signal and Hamming window [53] which can be presented as:

$$W_n(m) = \alpha + \beta cos\frac{2\pi m}{N_m - 1} \quad (5)$$

Where,$\alpha = 0.54\ and\ \beta = 0.46$

The Discrete Fourier Transform (DFT) is further applied to each windowed frame to obtain the frequency spectrum. The DFT represents the signal in the frequency domain.Each windowed frame havingN_m samples are converted into frequency domain as:

$$D_i = \sum_{m=0}^{N_m - 1} D_m e^{\frac{-j2\pi km}{N_m}} \quad (6)$$

Where,$i = 0, 1, 2 \dots \dots .. N_m - 1$

A set of Mel filters are applied to the magnitude spectrum obtained from the DFT. These filters approximate the human auditory system's frequency response and emphasize certain frequency bands.The mapping between real frequency (Hz) and Mel frequency can be given as:

$$f_{mel} = 2595 * log\left(1 + \frac{f}{700}\right) \quad (7)$$

After that the logarithm of the filter bank energies is considered to compress the dynamic range and mimic the logarithmic perception of sound intensity by humans then after to get the final MFCC features. Thus, DCT convert the log Mel spectrum intotime domain so that, only the lower-frequency components are retained.

We also considered another hybrid method for the feature extraction. In this approach three auditory i.e., MFCC, delta and delta-deltas [54] are used to extract the feature map. These extracted features are arranged in 2-D feature maps, each of which represents MFCC, delta and delta-delta features distributed along both frequency using the frequency band index and time using the frame number within each context window. The cepstralcoefficients are usually referred to as static features, so that they only contain information from a given frame. Hence, The extra information about the temporal dynamics of the signal is obtained by computing the first and second derivatives of the cepstral coefficients. The delta coefficient can be obtained for frame in term of static coefficient as:

$$d_t = \frac{\sum_{n=1}^{N} n(c_{t+n} - c_{t-n})}{2\sum_{n-1}^{N} n^2} \quad (8)$$

Where,$d_t$is a delta coefficient from frame $t$computed in terms of the static coefficients $c_{t-n}$to c_{t+n}. The delta-delta features can be extracted by taking the derivation of d_t to represent the change between frames in corresponding delta features. The feature extraction process with derivative for the sound signal input is shown in Fig 5.

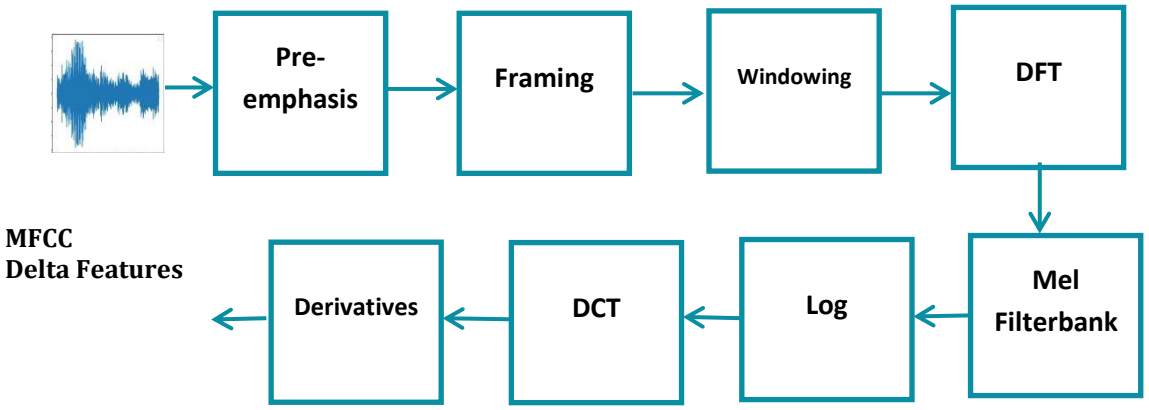

Figure 5. Process to extract MFCC Delta Features

4. CNN Architecture

Convolution neural networks have been considered an important architecture in deep learning. It includes input feature maps, convolutional kernels, pooling, and output layers [55]. Mostly, the stochastic gradient descent (SGD) algorithm is used to incorporate learning in CNN. The CNN is a multilayer neural network; each layer consists of several independent neurons. The complexity of a multilayer feed-forward neural network is optimized by CNN with filters and pooling. The architecture of CNN can be shown in the Fig. 6.

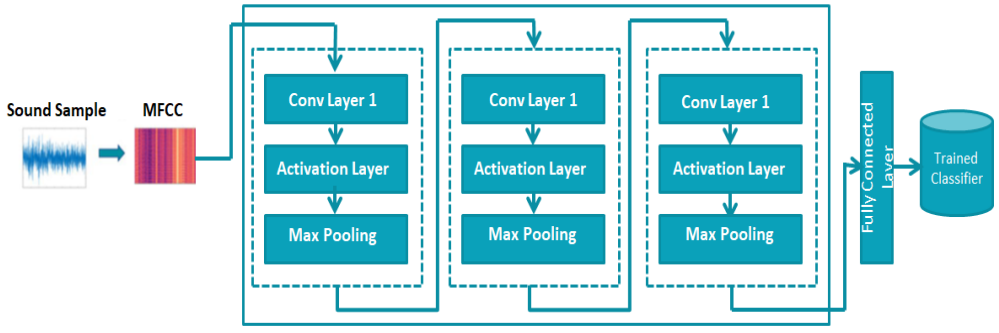

Figure 6. Proposed Convolutional Neural Network architecture for digits classification

In most cases, a sound spectrogram is considered as an image. The CNN considers the filters which slide over these images to produce feature maps at convolution layers. In the convolution layers, each plane is connected to one or more feature maps of the preceding layer and an activation function is applied to obtain the results as plane's output. The plane output is a 2D matrix i.e., feature map. The convolution layer produces one or more feature maps. Each map is then connected to exactly one plane in the next sub-sampling i.e., pooling layer. Since, sharing of weights and locations are grouped together and represented by a single value in order to minimize differences in the extracted features along the frequency dimension when the input patterns are shifted. This property is essential and important for the speech signals [56]. Let the speech input to convolution neural network is I that is divided into N feature vectors as:

$$I = [I_1, I_2, \dots\dots\dots\dots I_N] \quad (9)$$

Now, the activation is computed by multiplying a small local input with weight vector w. The activations of convolution layer are considered with k filters. The activation vector for the k filters can be computed as:

$$\square_{k,N} = f\left[\sum_{i=1}^{s-1} w_{i,j} I_{l+k} + u_j\right] \quad (10)$$

Here, f is the activation function, s is the size of the filter in the number of inputs, $w_{i,j}$ is the weight vector representing the $l^{t\square}$ component of the $k^{t\square}$ filter.

Further, the max pooling is used to remove variability in the convolution layer's units, that exists due to speaking styles, channel distortion etc. The activations of the $m^{t\square}$ channel of the max-pooling are denoted as:

$$P_m = \left[P_{m,1}, P_{m,2}, \dots\dots\dots\dots\dots\dots P_{n,k}\right]^T \quad (11)$$

Here, each activation is computed as:

$$P_{m,k} = max_{j=1}^{r}(\square_{m\times n+j,k}) \quad (12)$$

The processed information passes from one convolution block (convolution layer + pooling layer) to another in feed forward manner. The output map is obtained to present it for the fully connected network (dense network). The feature map represented to fully connected network can be expressed as:

$$X = [P_1^T, P_2^T, \dots\dots\dots\dots\dots\dots P_K^T] \quad (13)$$

Here, each P represents the matrix of size $(r \times s) \times 1$ so that, the size of X feature map is of $X_{(r\times s)\times\times k}$ *or* $X_{l\times k}$ where,$l = r \times s$

The dense network layer contains l numbers of units followed by v units in the hidden layer and c units in classification layer. The topology of dense network can be shown in Fig. 7.

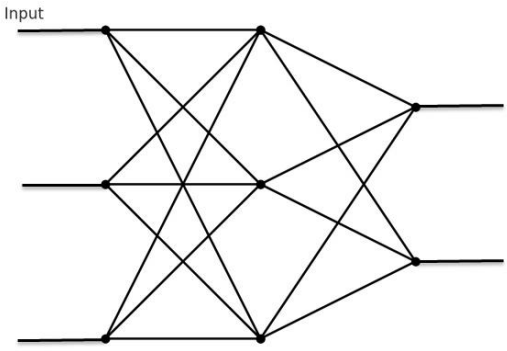

Figure 7. Topology of Dense Network

Convolutional neural network is a feed forward neural network which generally uses the back propagation learning to adjust the unknown parameters of the network. In which the input signal passes through the input layer, the hidden layer, and finally to the output layer. The mean square error is computed as the objective function and the weights are adjusted to minimize the objective function. Hence, for the output layer of the network the residual error is computed as:

$$\delta_o^l(m) = -\left(y - a^l(m).f'\left(z_o^l(m)\right)\right) \quad (14)$$

Similarly, the residual error for the hidden layer can be calculated as:

$$\delta_{\square}^l(m) = -\left((w^l)^T.\delta^{(l+1)}.f'\left(z_{\square}^l(m)\right)\right) \quad (15)$$

The partial derivative can be calculated as:

$$\nabla_w l\, J(w,b;x,y) = \delta^{l+1}\left(a^{(l)}\right)^T \quad (16)$$

$$\nabla_w l\, J(w,b;x,y) = \delta^{l+1} \quad (17)$$

The objective function of neural network is computed as:

$$J(w,b;x,y) = \frac{1}{m}\max_{i=1}^{m}\left(\frac{1}{2}\left\|h_{w,b}\left(x^{(i)}\right) - y^{(i)}\right\|^2\right) \quad (18)$$

The objective function uses the gradient descent method to determine the parameters w and b as:

$$w_j^{(l)} = w_j^{(l)} - \eta\frac{\partial}{\partial w_j^{(l)}} J(w,b) \quad (19)$$

$$\text{and} b_i^{(l)} = w_i^{(l)} - \eta\frac{\partial}{\partial b_i^{(l)}} J(w,b) \quad (20)$$

The essence of the process of neural network learning is to make objective function have a smaller value by optimization of the parameters. There are lots of modifications were also proposed to improve the effectiveness and performance of back propagation learning algorithm [57]. The mini-batch stochastic gradient descent is mostly employed with regularization & normalization for the training of convolutional neural network to determine the unknown parameters for desired classification.

5. Implementation and Experiment Details

CNNs offer a promising approach for speech recognition for many languages such as Bengali, Urdu, English, Hindi, &Pastho,and allowing for automatic feature extraction and achieving high accuracy on a variety of sound recognition tasks.In this present work, the audio signals of English spoken digit sounds are considered for classification. The existing datasetSpeech Commands v1 (SCV1) of digit sound samples is used to provide the training of the proposed two architectures of convolutional neural networks with different feature extraction methods. There are many methods those are used to extract the features from sound samples i.e. Log-Mel Scale Spectrogram (LM),Mel frequency cepstral coefficient (MFCC), Gammatone Frequency Cepstral Coefficients (GFCC) and Spectrogram etc. MFCCis the most widely used feature extraction scheme for speech recognition and audio classification due to its better adaptability of network when noise is taken into consideration.

We have considered the spectrogram method for the feature extraction and to represent the audio data into the time-frequency patches. In this process of feature extraction, audio data pre-processing is performed with sampling, quantization, pre-emphasisprocessing, and windowing to convert the analog audio signal into a sequence of audio frames.Further, log-scale Mel-spectrogram is used to represent the pre-processed audio data into the time-frequency patches. The acoustic variants that aren't important for speech recognitionare reduced by log.It generates an N-by-M matrix of features, where N isthe number of partitioned analysis frames of the speech signal and M is the number of coefficientsreturned per frame. In this study, MFCC produces 32 coefficients for each frame. The MFCC features are shown in Fig 8.

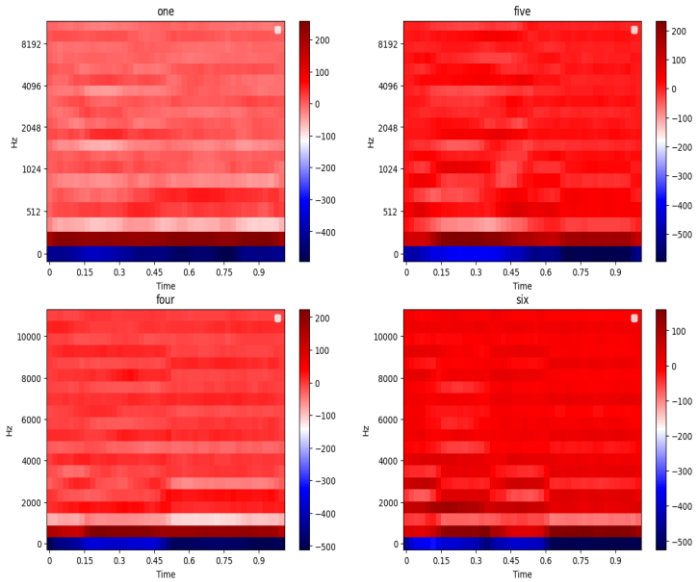

Figure 8. MFCC features for digits sound samples

We have also considered three auditory features: MFCC, delta, and delta-deltas. These features are extracted and then combined into a single feature set for both Convolutional Neural Network architectures. These combined features are shown Fig 9.

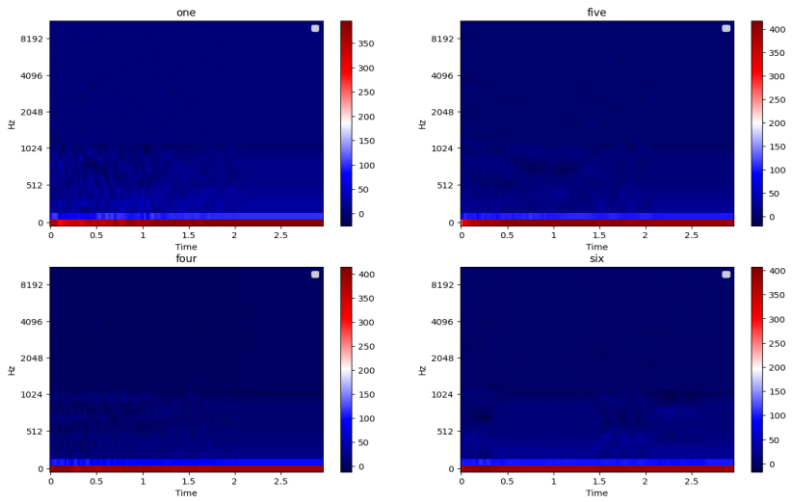

Figure 9. MFCC, Delta MFCC and Delta Deltas MFCC features of spoken digits sound samples.

In our first experiment, we considered the three layer convolutional neural network architecture. Each Convolutional Neural Network layer consists with convolution kernel with non-linear activation function followed by max-pooling layer. Prior to training, the sound samples of spoken English digits are pre-processed and converted into images. In this pre-processing step, we considered Mel Frequency Cepstral Coefficients (MFCC), as well as combined features including MFCC, delta MFCC (ΔMFCC), and delta-delta MFCC (ΔΔMFCC) for the training of CNN. In this proposed architecture, we considered the filters for three convolution layers as: $3 \times {}^{I}/_{2}, 2 \times {}^{I}/_{2}, {}^{I}/_{2}$; here, I is the number of rows of a two-dimensional matrix that represents the number of features and sound samplings. We used MFCC feature vectors with a constant size 32, because CNN cannot process vectors of varying size although MFCC vectors might vary in size for different audio input so that, we considered the MFCC feature vector uniform in size. Hence, in the pre-processing task, if the MFCC feature extraction method is required more than 32 elements in the extracted feature vector than the extra features are removed, whereas if the less than 32 features are obtained then those features are padded by filling with the zero. Therefore, we have a two-dimensional feature vector map that represents the number of features and sound sampling. Thus, we considered the size of input Time-Frequency patch as input feature map I of size 32 x 44 to the CNN. Hence, the proposed CNN architecture considered the 48, 32 and 16 filters in successive layers respectively. The size of the filters is considered 2x2 for all layers with stride of one without any padding. Thus, the training data is passed through the first 2D convolutional layer with 48 filters and a ReLU activation function. Then the output of the first convolution layer is passed through the batch normalization and then to the max pooling to reduce the parameters. This process is performed further with the second convolution layer of 32 filters and third convolution layer with 16 filters. After the last pooling layer, the output feature map is presented to the dropout layer with 0.25 dropout rate. The dropout layer drops some values to reduce the chances of over fitting. In the first experiment, the feature map obtained from the last convolution block is directly applied to the dense network for classification and in second experiment one fully connected layer with 128 hidden units are used on the flat output, and rectified linear activation function followed by the classification layer with 10 units and softmax activation function.

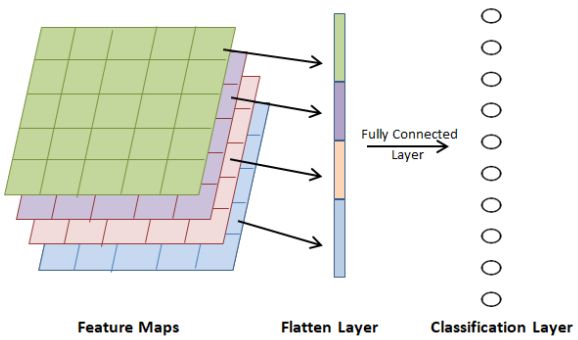

Figure 10. Convolutional Neural Network with flatten layer

The objective function and optimizer used in the model are least mean square error and mini batch stochastic gradient descent method. In the proposed experimental setup, the parameters are computed as:

Number of parameters $= (m \times n \times p + b) \times O'$ (21)

Here,$m \times n$ is the size of the individual channel or filter, p is the number of input channels or feature map, O' is the output feature maps or channels and brepresents the bias for the current filter. Therefore, with the use of equation 21, we obtained the number of parameters for the first convolution layer (C1) as:

number of parameters for C1 $= (2 \times 2 \times 1 + 1) \times 48 = 240$

Similarly, the number of parameters for C2, and C3 convolution layers are number of parameters for C2 $= (2 \times 2 \times 48 + 1) \times 32 = 6176$ and, number of parameters for C3 $= (2 \times 2 \times 32 + 1) \times 16 = 2064$

The max pooling layer is used with each convolution layer. The output from the last max pooling layer of C3 is of size $3 \times 4 \times 16$ is flatten and represented as (192×1) column vector as input to the dense network. The weight matrix (w1) between flatten layer and first hidden layer of dense network D of size 128×192 is initialize with random numbers and the weight vector of size 10×128 between hidden layer and classification layer is also initialized with random numbers. The summary of number of parameters for first proposed CNN architecture can be presented in Table 1.

Table 1. Number of parameters in each layer of the proposed CNN (Including Flatten Layer)

Layer (type)	**Output Shape**	**Param #**
Conv2D 1	(None, 31, 43, 48)	240
MaxPooling2D 1	(None, 15, 21, 48)	0
Conv2D 2	(None, 14, 20, 32)	6176
MaxPooling2D 2	(None, 7, 10, 32)	0
Conv2D 3	(None, 6, 9, 16)	2064
MaxPooling2D 3	(None, 3, 4, 16)	0
Flatten	(None, 192)	0
Dense 1	(None, 128)	24704
Dense 2 (Output layer)	(None, 10)	1290
Total params: 34,474 Trainable params: 34,474 Non-trainable params: 0		

In the second experiment, we again considered same CNN architecture with the small correction after the third convolution layer (C3) to optimize the number of learnable parameters of network. In the first experiment, the number of parameters is increasing after the third convolution layer so the global max-pooling is applied after the third convolution layer (C3). The global max pooling optimizes the number of parameters and it justify the removal of flatten layer before the dense network. Therefore, the two hidden layers are used after the global max pooling layer followed by the classification layer. Thus, the global max pooling is used to reduce the number of parameters in CNN architecture without degrading the performance as shown in Fig. 11.

The global max-pooling layer takes the maximum of each feature map and presents it directly to the activation layer in a fully connected layer. It applies max pooling on the spatial dimensions until each spatial dimension is one, and leaves other dimensions unchanged. The output of global-max-pooling is directly presented to the first layer of dense network. Global max pooling is an extreme of max pooling that can reduce a tensor with size of $w \times w \times d$ to $1 \times 1 \times d$ i.e. $1 \times 1 \times 1$After summation of all the parameters together, we get the total number of learnable parameters within the second CNN architecture i.e., 11,946 which are less than first architecture with flatten layer i.e. 34,474.

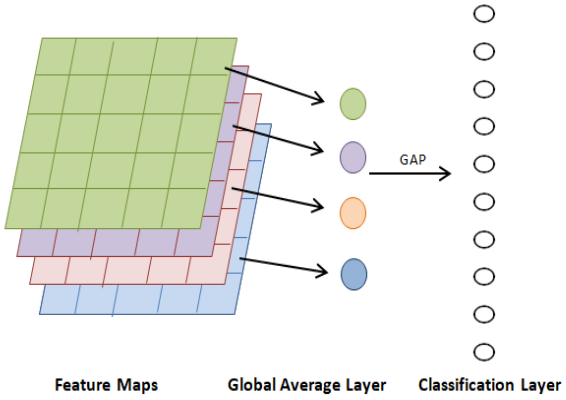

Figure 11. Global Average Pooling Architecture

Both the CNN networksare trained for the spoken English digits input samples and the parameters of the network areoptimized using Adam stochastic gradient learning and sparse categorical crossentropy methods to minimize the error. Thus, the architecture of second experiment with number of parameters can be presented in Table 2.

Table 2. Number of parameters in each layer of the proposed CNN (Including Global Max Pooling Layer)

Layer (type)	**Output Shape**	**Param #**
Conv2D 1	(None, 31, 43, 48)	240
MaxPooling2D 1	(None, 15, 21, 48)	0
Conv2D 2	(None, 14, 20, 32)	6176
MaxPooling2D 2	(None, 7, 10, 32)	0
Conv2D 3	(None, 6, 9, 16)	2064
Max Pooling2D 3	(None, 3, 4, 16)	0
Global Max Pooling2D	(None, 16)*	0
Dense 1	(None, 128)	2176
Dense 2	(None, 10)	1290
Total params: 11,946 (466 KB) Trainable params: 11,946 (466 KB) Non-trainable params: 0 (0.00 Byte)		

Hence, both the proposed CNN architectures are presented with pre-processed sound signals in the form of time-frequency patch of 2D vector as input. These networks have been trained for unknown learnable parameters using Adam stochastic gradient optimizer.

6. RESULT AND DISCUSSION

The spoken English digits sound classification task is performed with two proposed convolution neural network architectures. We considered the Speech Commands v1 (SCV1)dataset for the training. The classes are identified as zero, one, two, three, four, five, six, seven, eight, and nine [58]. The dataset consists of 13323 audio clips of spoken digits in English language. The MFCC and combined MFCC, ΔMFCC, & ΔΔMFCCare used for the feature extraction and represented the sound samples in the form feature maps of images. Each image feature map is considered of size 32×44 and 32×132 respectively. Hence, after constructing the feature maps as input vectors for training, we presented them to both the proposed CNN models. The 9992 sound samples were considered for the training and rest of the 3321 samples were used as the test pattern set.Both the architectures were trained with Adam stochastic gradient learning and sparse categorical crossentropy error has been used as the performance index or objective function. The simulation results of training were not obtained on fixed epochs to avoid overfitting. Hence, we considered the mechanism of early stopping of the training based on saturation index on the performance matrices. We have used early stopping of the training when a monitored metric has stopped improving on both the CNN models. The comparative study & performance of both the CNNs models are analysed on the basis of number of parameters, number of epochs, training time, and accuracy

for the test pattern vectors. We did various experiments on proposed CNN architectures. Architectures of different experiments are shown in Table 3.

Table 3. Description of various experiments on proposed CNN Models

Proposed CNN Architecture	Feature Extraction Method	Global Pooling Layer	Flatten Layer	Dense Layer
OptNet11	MFCC (32 x 44)	No	Yes	Yes
OptNet12	MFCC (32 x 44)	No	Yes	No
OptNet13	Combined MFCC,Δ, ΔΔ (32 x 132)	No	Yes	Yes
OptNet14	Combined MFCC,Δ, ΔΔ (32 x 132)	No	Yes	No
OptNet21	MFCC (32 x 44)	Global Average	No	Yes
OptNet22	MFCC (32 x 44)	Global Average	No	No
OptNet23	Combined MFCC, Δ, ΔΔ (32 x 132)	Global Average	No	Yes
OptNet24	Combined MFCC, Δ, ΔΔ (32 x 132)	Global Average	No	No
OptNet25	MFCC (32 x 44)	Global Max Pooling	No	Yes
OptNet26	MFCC (32 x 44)	Global Max Pooling	No	No
OptNet27	Combined MFCC, Δ, ΔΔ (32 x 132)	Global Max Pooling	No	Yes
OptNet28	Combined MFCC, Δ, ΔΔ (32 x 132)	Global Max Pooling	No	No

The model training started from acquiring the corresponding leaning patterns in the input speech spectrum. The learning features go through our suggested network layers and training parameters have been updated through learning process. Table 4 shows simulated results between both the CNN models.

Table 4. Simulated results between both the CNN models

Model Name	Number of Parameters	Training time (Sec)	Epochs	Accuracy (%)
OptNet11	34,474	38.88	21	99.48
OptNet12	10,410	369	22	99.34
OptNet13	1,02,058	19.93	09	99.33
OptNet14	15,690	33.55	16	99.36
OptNet21	11,946	41.95	21	99.30
OptNet22	8,650	40.05	20	98.56
OptNet23	11,946	27.37	12	97.29
OptNet24	8,650	71.10	34	98.24
OptNet25	11,946	21.92	11	99.06
OptNet26	8,650	38.57	21	99.31
OptNet27	11,946	45.45	20	99.24
OptNet28	8,650	42.85	19	98.98

It can be seen from the table 4 that the first model with three convolution blocks and flatten layer followed by dense network (OptNet11) considered the 34,474 parameters, 21 epochs & 38.88 seconds in the convergence and shows highest accuracy 99.48% which indicates that this modelcan effectively capture features without using pooling layers. Whereas, models (OptNet25, OptNet26, OptNet27, OptNet28) using Global Average Pooling have lower accuracy as compared to those with flatten layers or Global Max Pooling, particularly for the combined MFCC features. The model with three convolution layers followed by global max pooling or global average pooling considers only 8650 parameters in convergence. It can be analysis between both the models that Global Max Pooling models also achieved high accuracy but not as high as those with Flatten layers. The comparative analysis of different experiments is shown in Fig 12.

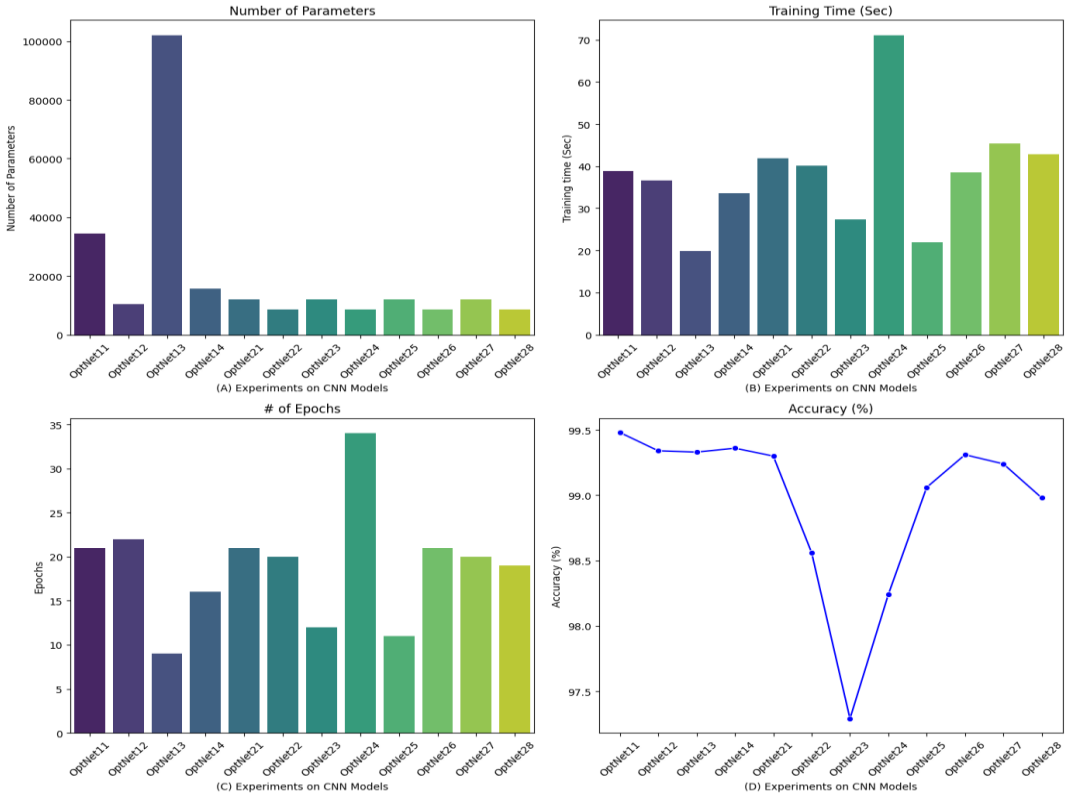

Figure 12. Comparative analysis on different experiments in terms of (a) Number of Parameters, (B) Training Time, (C) Epochs and (D) Accuracy

It can be seen from Fig. 12 that the combined features MFCC, Δ, ΔΔ generally provide good accuracy but also show varied results based on the model architecture. i.e., OptNet13 achieved 99.33% with a Flatten layer, while OptNet23 with Global Average Pooling dropped to 97.29%. Comparative analysis between number of parameters and accuracy is shows in Fig 13.

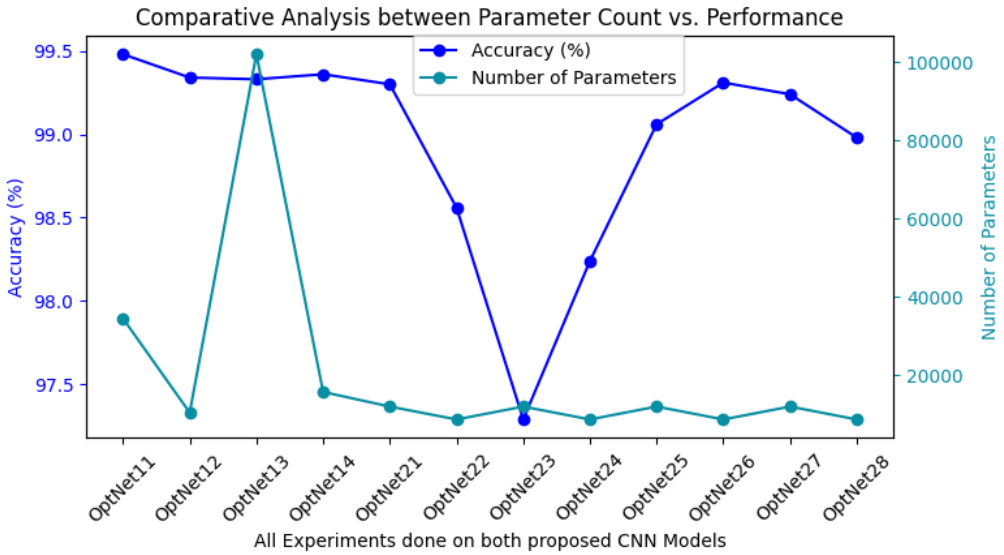

Figure 13. Comparative Analysis between Number of Parameters and Performance

Simulated results shows that models with a higher number of parameters like OptNet13 with 102,058 parameters does not exhibits the best accuracy so thatthere is no correlation is showing in between number of parameters and accuracy for classification. Thus, the models are considered as complexity does not always leads for better performance.

The other parameters for analysis the performance of both the networks are epochs and training time. Model (OptNet13) with 1,02,058 parameter counts, 19.93 sec training time, 09 epochs with performance accuracy 99.33% and other model (OptNet25) with 11,946 parameter counts, 21.92 sec training time , 11 epochs with 99.06% accuracy are identified as the best CNN model for the desired classification task. Therefore, the confusion matrices for best two models are obtained and represent in Fig 14. Fig 15 and Fig 16 are explaining CNN training and validation stages with all training parameters of both the models.

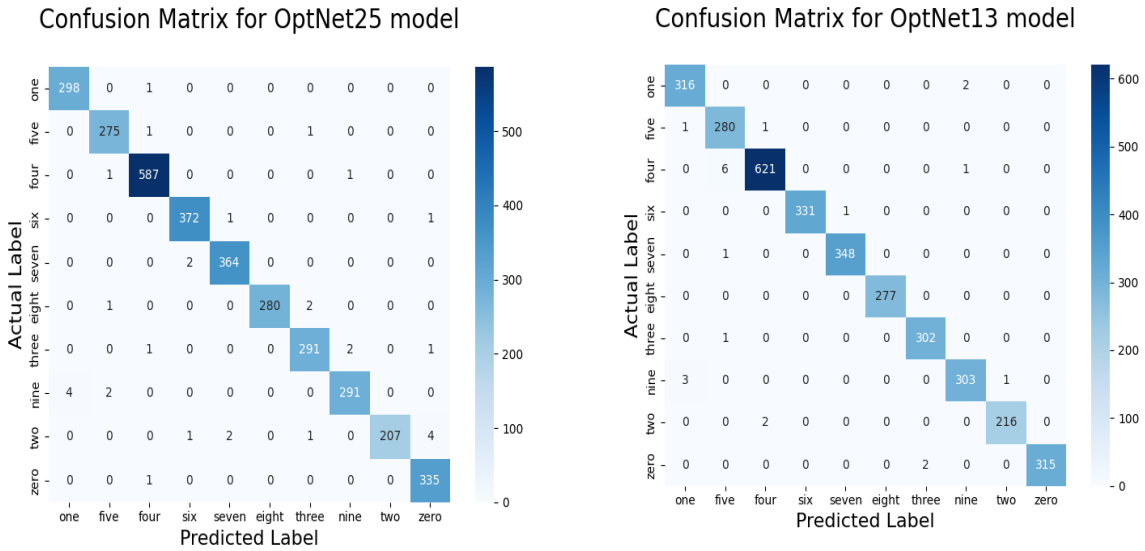

Figure 14. Confusion Matrix for OptNet25 and OptNet13 models

Table 5 shows the class-wise accuracy, precision, recall, specificity and F1-score.

Table 5. Results of OptNet13 and OptNet25 models used in Speech_Command_V1 dataset

Model	Class Label	Class Name	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)	F1-Score (%)
OptNet13	0	One	99.91	99.68	99.37	99.97	99.53
	1	Five	99.73	97.56	99.29	99.77	98.42
	2	Four	99.70	99.52	98.89	99.89	99.20
	3	Six	99.97	100.00	99.70	100.00	99.85
	4	Seven	99.97	99.71	100.00	99.97	99.86
	5	Eight	99.94	99.28	100.00	99.93	99.64
	6	Three	99.97	100.00	99.67	100.00	99.83
	7	Nine	99.97	100.00	99.67	100.00	99.84
	8	Two	99.82	98.18	99.08	99.87	98.63
	9	Zero	99.94	100.00	99.37	100.00	99.68
OptNet25	0	One	98.40	99.00	100.00	100.00	99.00
	1	Five	98.20	99.00	99.00	100.00	99.00
	2	Four	98.60	99.00	100.00	100.00	99.00
	3	Six	99.70	99.00	99.00	100.00	99.00
	4	Seven	99.40	99.00	99.00	100.00	99.00
	5	Eight	100.00	100.00	99.00	100.00	99.00
	6	Three	99.70	99.00	99.00	100.00	99.00
	7	Nine	98.70	99.00	98.00	100.00	98.00
	8	Two	99.10	100.00	900	100.00	98.00
	9	Zero	99.40	98.00	100.00	100.00	99.00

Table 6. TP, TN, FP, and FN parameters for OptNet13 and OptNet25 models used in CSV1 digits audio dataset

Model Name	Class Label	Class Name	TP	FP	TN	FN
OptNet13	**0**	One	316	1	3008	2
	1	Five	280	7	3038	2
	2	Four	621	3	2696	7
	3	Six	331	0	2995	1
	4	Seven	348	1	2978	0
	5	Eight	277	2	3048	0
	6	Three	302	0	3024	1
	7	Nine	303	0	3023	1
	8	Two	216	4	3105	2
	9	Zero	315	0	3010	2
OptNet25	**0**	One	298	4	2072	1
	1	Five	275	3	2095	2
	2	Four	587	4	1783	1
	3	Six	372	1	2000	2
	4	Seven	364	1	2008	2
	5	Eight	280	3	2090	2
	6	Three	291	3	2077	4
	7	Nine	291	1	2077	6
	8	Two	207	3	2158	7
	9	Zero	335	6	2033	1

Loss and accuracy graph for OptNet25 and OptNet13 models can be shown in Fig. 15 and 16.

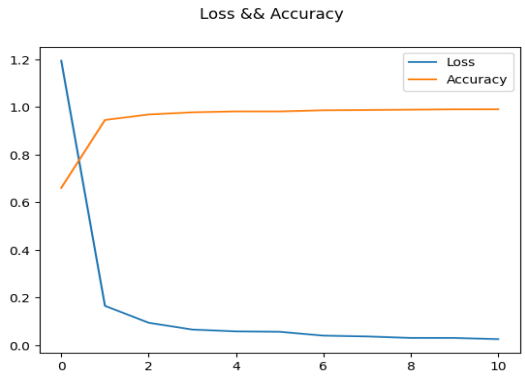

Figure 15. Loss &Accuracy for optCNN25 Model

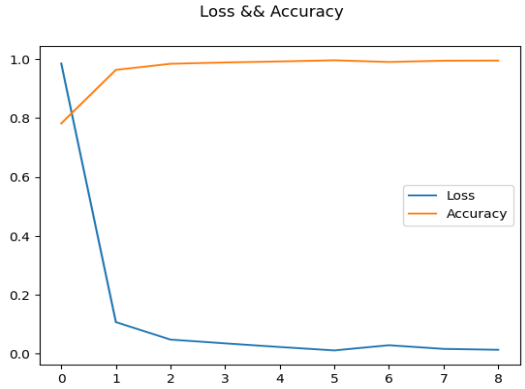

Figure 16. Loss &Accuracy for optCNN13 Model

The Fig. 17 presents the detailed analysis for performance of both the networks during the training with respect to all the parameters.

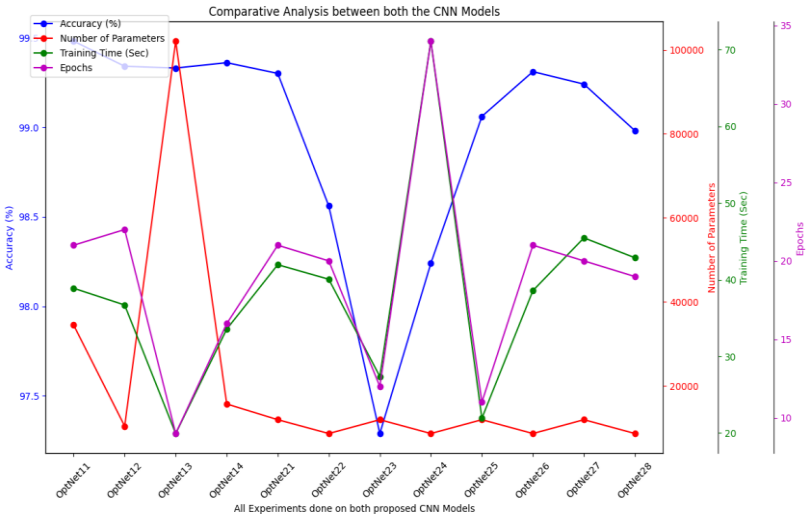

Figure 17. Detailed analysis for performance of both the networks during the training

Therefore, proposed convolutional neural network models with pre-processing techniques of feature extraction perform effectively for the classification of spoken English digits. The proposed models are trained & validated with given dataset of large samples. The simulated results are exhibiting the best performance for the convolutional neural network architecture which contains global max pooling without flatten layers & dense layer. It exhibits 99.31% validation accuracy with 8650 parameters and 21 epochs for convergence. Fig. 12 represents the comparative analysis of proposed convolution models for the classification, accuracy. The confusion matrices of both the proposed models i.e., with and without global pooling approaches are presented in Fig. 14. It is clearly indicating from the confusion matrices that the rate of misclassification is less in second type of model in comparison of first model. Thus, the convolutional neural networks with global max pooling and without flatten & dense layers generate less cases of misclassification so that the classification accuracy improves for this model.

7. CONCLUSION

Two different convolutional neural network models with hybrid pre-processing techniques of feature extraction are considered to analyse the performances of classification for spoken English digits. Different experiments are conducted to obtain the optimize models with effective classification accuracy. Convolutional neural network with convolution layers of filters followed by global max pooling without flatten layer and dense network is identified as the most optimized architecture with 99.31% classification accuracy for the test samples. It is observed from the simulated results that the rate of misclassification is less in comparison to other models. It has been also observed that the CNN model followed by global max pooling without flatten layers explores less number of parameters and maintain the accuracy level up to 98% to 99% for test samples. The feature extraction techniques also play the crucial role to improve the classification accuracy. It has been observed that the MFCC feature extraction methods improving the classification accuracy in comparison of hybrid techniques of feature extraction. The first type of proposed architecture maintains the classification accuracy in the range of 99% but lots of variations in number of parameters whereas the second architecture exhibiting the reduction in parameters to maintain the classification accuracy in the range of 98% to 99%. It has also been observed that more feature extraction in pre-processing step increases the number of parameters but it is not reflecting that the large number of parameters improves the classification accuracy. Therefore, the convolutional neural network models with MFCC feature extraction techniques is more optimal and perform with 99% accuracy for the classification of spoken English digits. There are lots of scope of work with different analysis methods to develop more optimal CNN models with improve classification rate. It is also important that the performance of proposed model can be analysed for the spoken English alphabets or words classification.

REFERENCES

[1] Reddy, B. R., &Mahender, E. (2013). Speech to text conversion using android platform. International Journal of Engineering Research and Applications (IJERA), 3(1), 253-258.

[2] Luce, P. A., &Pisoni, D. B. (1998). Recognizing spoken words: The neighborhood activation model. Ear and hearing, 19(1), 1-3

[3] Alsobhani, A., ALabboodi, H. M., & Mahdi, H. (2021, August). Speech recognition using convolution deep neural networks. In Journal of Physics: Conference Series (Vol. 1973, No. 1, p. 012166). IOP Publishing.

[4] Barkhuijsen, H., De Beer, R., Bovee, W. M. M. J., & Van Ormondt, D. (1985). Retrieval of frequencies, amplitudes, damping factors, and phases from time-domain signals using a linear least-squares procedure. Journal of Magnetic Resonance (1969), 61(3), 465-481.

[5] Gales M., Young S., "The Application of Hidden Markov Models in Speech Recognition," Foundations and Trends in Signal Processing, vol. 1, no. 3, pp. 195–304, 2008.

[6] Nassif A. B., Shahin I., Attili I., Azzeh M., Shaalan K., "Speech Recognition Using Deep Neural Networks: A Systematic Review", IEEE, vol. 7, pp. 19143-19165, 2019.

[7] Zou J., Han Y., So S. S., "Overview of Artificial Neural Networks. In: Livingstone" D.J. (Eds) Artificial Neural Networks. Methods in Molecular Biology™, vol. 458, 2008.

[8] Zhang XL, Luo, ZG. , Li, M. J, "Journal of Computer Science and Technology", Springer, vol. 29, no. 6, pp. 1072–1082, 2014.

[9] Salau, A. O., & Jain, S. (2019, March). Feature extraction: a survey of the types, techniques, applications. In 2019 international conference on signal processing and communication (ICSC) (pp. 158-164). IEEE.

[10] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, Deep learning, Vol. 1, MIT press, , Cambridge, MA, USA, 201

[11] Yu, D., & Deng, L. (2016). Automatic speech recognition (Vol. 1). Berlin: Springer.

[12] Alzubaidi, L., Zhang, J., Humaidi, A.J. et al. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. J Big Data 8, 53 (2021).

[13] A. B. Nassif, I. Shahin, I. Attili, M. Azzeh and K. Shaalan, "Speech Recognition Using Deep Neural Networks: A Systematic Review," in IEEE Access, vol. 7, pp. 19143-19165, 2019, doi: 10.1109/ACCESS.2019.2896880.

[14] Mushtaq, Z., Su, S. F., & Tran, Q. V. (2021). Spectral images based environmental sound classification using CNN with meaningful data augmentation. Applied Acoustics, 172, 107581.

[15] L. Deng et al., ``Recent advances in deep learning for speech research at Microsoft,'' in Proc. IEEE Int. Conf. Acoust., Speech Signal Process., ay 2013, pp. 86048608.

[16] H. Li, B. Ma, and K. A. Lee, ``Spoken language recognition: From fundamentals to practice,'' Proc. IEEE, vol. 101, no. 5, pp. 11361159, May 2013.

[17] Zeng, T. (2022, December). Deep Learning in Automatic Speech Recognition (ASR): A Review. In 2022 7th International Conference on Modern Management and Education Technology (MMET 2022) (pp. 173-179). Atlantis Press.

[18] Zhou, W., Schlüter, R., & Ney, H. (2020, May). Full-sum decoding for hybrid hmm based speech recognition using LSTM language model. In ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 7834-7838). IEEE.

[19] Rippel, O., Snoek, J., & Adams, R. P. (2015). Spectral representations for convolutional neural networks. Advances in neural information processing systems, 28.

[20] Cotton CV, Ellis DPW. Spectral vs. spectro-temporal features for acoustic event detection. In: IEEE workshop on applications of signal processing to audio and acoustics. p. 69–72.

[21] Sarker, I. H. (2021). Deep learning: a comprehensive overview on techniques, taxonomy, applications and research directions. SN computer science, 2(6), 420.

[22] A. Revathi and Y. Venkataramani, "Speaker independent continuous speech and isolated digit recognition using VQ and HMM," 2011 International Conference on Communications and Signal Processing, Kerala, India, 2011, pp. 198-202, doi: 10.1109/ICCSP.2011.5739300.

[23] Saon, G., &Picheny, M. (2017). Recent advances in conversational speech recognition using convolutional and recurrent neural networks. IBM Journal of Research and Development, 61(4/5), 1-1.

[24] Deng, L., Hinton, G., & Kingsbury, B. (2013, May). New types of deep neural network learning for speech recognition and related applications: An overview. In 2013 IEEE international conference on acoustics, speech and signal processing (pp. 8599-8603). IEEE.

[25] Peng, Y. (2022). Digital Recognition Methods Based on Deep Learning. Scientific Programming, 2022(1), 9691331.

[26] Sinha, H., Awasthi, V., &Ajmera, P. K. (2020). Audio classification using braided convolutional neural networks. IET Signal Processing, 14(7), 448-454.
[27] Meister, S., Wermes, M., Stueve, J., & Groves, R. M. (2021). Cross-evaluation of a parallel operating SVM–CNN classifier for reliable internal decision-making processes in composite inspection. Journal of Manufacturing Systems, 60, 620-639.
[28] Yang, X., Yu, H., &Jia, L. (2020, April). Speech recognition of command words based on convolutional neural network. In 2020 International Conference on Computer Information and Big Data Applications (CIBDA) (pp. 465-469). IEEE.
[29] D. F. Silva, V. M. A. de Souza, G. E. A. P. A. Batista, and R. Giusti, "Spoken digit recognition in Portuguese using line spectral frequencies," in Ibero-American Conference on Artificial IntelligenceSpringer, New York, NY, USA, 2012.
[30] D. F. Silva, V. M. A. de Souza, and G. E. A. P. A. Batista, "A comparative study between MFCC and LSF coe^{3}cients in automatic recognition of isolated digits pronounced in Portuguese and English," ActaScientiarum. Technology, vol. 35,pp. 621–628, 2013.
[31] Hu, W., Cai, M., Chen, K., Ding, H., Sun, L., Liang, S., ...&Huo, Q. (2017, November). Sequence discriminative training for offline handwriting recognition by an interpolated CTC and lattice-free MMI objective function. In 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR) (Vol. 1, pp. 61-66). IEEE.
[32] Lotfi Rezaabad, A., &Vishwanath, S. (2020, July). Long short-term memory spiking networks and their applications. In International Conference on Neuromorphic Systems 2020 (pp. 1-9).
[33] S. Becker, M. Ackermann, S. Lapuschkin, K.-R. Müller, and W. Samek, "Interpreting and explaining deep neural networks for classification of audio signals," arXiv preprint arXiv:1807.03418, 2018.
[34] J. v. Doremalen and L. Boves, "Spoken digit recognition using a hierarchical temporal memory," in 9th Annual Conference of the International Speech Communication Association (INTERSPEECH-2008), Brisbane, Australia, 2008, pp. 2566-2569.
[35] I. Bazzi and D. Katabi, "Using support vector machines for spoken digit recognition," in Sixth International Conference on Spoken Language Processing (ICSLP 2000), Beijing, China, 2000, pp. 433-43
[36] Bazzi, I., &Katabi, D. (2000, October). Using support vector machines for spoken digit recognition. In INTERSPEECH (pp. 433-436).
[37] K. Tyagi and K. Tyagi, "A comparative analysis of optimization techniques," International Journal of Computer Application, vol. 131, no. 10, pp. 6–12, 2015.
[38] Oruh, J., &Viriri, S. (2022). Deep Learning-Based Classification of Spoken English Digits. Computational Intelligence and Neuroscience, 2022(1), 3364141.
[39] Nisar, S., Shahzad, I., Khan, M. A., & Tariq, M. (2017, February). Pashto spoken digits recognition using spectral and prosodic based feature extraction. In 2017 Ninth International Conference on Advanced Computational Intelligence (ICACI) (pp. 74-78). IEEE.
[40] Al-Haddad, S. A. R., Samad, S. A., Hussain, A., Ishak, K. A., &Mirvaziri, H. (2007, December). Decision fusion for isolated Malay digit recognition using dynamic time warping (DTW) and hidden Markov model (HMM). In 2007 5th Student Conference on Research and Development (pp. 1-6). IEEE.
[41] Dhingra, SD, Nijhawan, G, and Pandit, P (2013). Isolated speech recognition using MFCC and DTW. International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering. 2, 4085-4092.
[42] Kalith, IM, Ashirvatham, D, and Thelijjagoda, S (2016). Isolated to connected Tamil digit speech recognition system based on hidden Markov model. International Journal of New Technologies in Science and Engineering. 3, 1-11.
[43] Ali, H, Jianwei, A, and Iqbal, K (2015). Automatic speech recognition of Urdu digits with optimal classification approach. International Journal of Computer Applications. 118, 1-5.
[44] Chapaneri, SV (2012). Spoken digits recognition using weighted MFCC and improved features for dynamic time warping. International Journal of Computer Applications. 40, 6-12.
[45] S. Becker, M. Ackermann, S. Lapuschkin, K.-R. Müller, and W. Samek, "Interpreting and explaining deep neural networks for classification of audio signals," arXiv preprint arXiv:1807.03418, 2018.
[46] Sharmin, R., Rahut, S. K., &Huq, M. R. (2020). Bengali spoken digit classification: A deep learning approach using convolutional neural network. Procedia Computer Science, 171, 1381-1388.
[47] R. V. Sharan and T. J. Moir, "Acoustic event recognition using cochleagram image and convolutional neural networks," Applied Acoustics, vol. 148, pp. 62-66, 2019.
[48] R. V. Sharan, S. Berkovsky, and S. Liu, "Voice command recognition using biologically inspired time-frequency representation and convolutional neural networks," in 42nd Annual International

Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Montreal, QC, Canada, 2020, pp. 998-1001.

[49] Sharan, R. V. (2020, December). Spoken digit recognition using wavelet scalogram and convolutional neural networks. In 2020 IEEE Recent Advances in Intelligent Computational Systems (RAICS) (pp. 101-105). IEEE.

[50] Warden P. Speech Commands: A public dataset for single-word speech recognition, 2017.

[51] McFee, B., Raffel, C., Liang, D., Ellis, D. P., McVicar, M., Battenberg, E., & Nieto, O. (2015). librosa: Audio and music signal analysis in python. In Proceedings of the 14th python in science conference (Vol. 8).

[52] R. V. Sharan and T. J. Moir, "Time-frequency image resizing using interpolation for acoustic event recognition with convolutional neural networks," in IEEE International Conference on Signals and Systems (ICSigSys), Bandung, Indonesia, 2019, pp. 8-11.

[53] Podder, P., Khan, T. Z., Khan, M. H., &Rahman, M. M. (2014). Comparative performance analysis of hamming, hanning and blackman window. International Journal of Computer Applications, 96(18), 1-7.

[54] Rashmi, P., & Singh, M. P. (2023). Convolution neural networks with hybrid feature extraction methods for classification of voice sound signals. World Journal of Advanced Engineering Technology and Sciences, 8(2), 110-125.

[55] O'shea, K., & Nash, R. (2015). An introduction to convolutional neural networks. arXiv preprint arXiv:1511.08458.

[56] Abdel-Hamid, O., Mohamed, A. R., Jiang, H., Deng, L., Penn, G., & Yu, D. (2014). Convolutional neural networks for speech recognition. IEEE/ACM Transactions on audio, speech, and language processing, 22(10), 1533-1545.

[57] Rojas, R., & Rojas, R. (1996). The backpropagation algorithm. Neural networks: a systematic introduction, 149-182.

[58] Warden P. Speech Commands: A public dataset for single-word speech recognition, 2017.