Acoustic Echo Detection using Deep Learning

QURAT UL AIN

Department of Electronics

Quaid-i-Azam University, Islamabad
Pakistan

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of

Master of Philosophy
in
Electronics

May, 2021

DEPARTMENT OF ELECTRONICS
QUAID-I-AZAM UNIVERSITY
ISLAMABAD

A thesis entitled " *Acoustic Echo detection using deep learning* for degree of Master of Philosphy has been accepted by

____

Advisor
Dr. Muhammad Zia
Associate Professor
Department of Electronics
Quaid-i-Azam University, Islamabad
Pakistan

____

Chairman
Dr. Aqeel Bukhari
Professor
Department of Electronics
Quaid-i-Azam University, Islamabad
Pakistan

I dedicated my dissertation to my parents.

Acknowledgements

I would like to thanks Dr Muhammad Zia for the guidance, encouragement and advice he have provided through out one year research time. I have been extremely lucky to have a supervisor who cared so much about my work and responded to my questions and queries so promptly. I must express my gratitude to my parents and friends for their support and encouragement.

QURAT UL AIN
April 31, 2021

Abstract

In acoustic echo cancellation, adaptation of echo estimation filter relies on the detection of the state of acoustic echo canceller (AEC). Conventional method of detection of echo only state has high probability of false detection or miss detection, which results into divergence or slow convergence of the normalized least mean square (NLMS) algorithm. In this work, we focus on enhancing the accuracy of the echo only state detection using deep learning algorithms instead of conventional detector. We prepare data set from the speaker and microphone signals to train the deep learning algorithms. We use Alex Net, Deep convolution neural network (DCNN), Recurrent neural network (RNN) and K-nearest neighbor(KNN) for the echo detection. We prepared two training data sets each containing 2000 echo samples and 2000 samples without echo. Two testing data sets contains 60 echo samples and 40 no-echo samples. The aforementioned algorithms are trained on data sets, then tested on testing data sets achieves promising results. This trained echo detector detects echo only state and helps achieve better convergence of the NLMS adaptive filter.

Contents

List of Figures

List of Tables

Nomenclature

SYMBOLS

Symbol	Description
$x(n)$	Far-end signal
$h(n)$	Room impulse response
$e(n)$	Echoed signal
$s(n)$	Near-end signal
$y(n)$	Microphone signal
$\hat{e}(n)$	Replica of signal
$\hat{s}(n)$	Local talk estimation
TP	True positive
TN	True negative
FP	False positive
FN	False negative

Acronym	**Description**
ED	Echo Detector
AEC	Acoustic echo canceller
$NLMS$	Normalized Least mean squares
DL	Deep Learning
ML	Machine Learning
CNN	Convolution Neural Network
RNN	Recurrent Neural Network
$BLSTM$	Bidirectional long short term memory
CRN	Convolution recurrent Network
$LSTM$	Long short term memory
KNN	K-nearest neighbor
DTD	Double talk detector
CR	Cross-correlation
NCR	Normalized cross-correlation
CM	Confusion matrix

Chapter 1

Introduction

Acoustic echo is major impairment in the voice signal, which occurs due to the coupling between speaker and microphone at the end terminals of the voice communication system. As a result, far-end person hears his own delayed voice [1, 2]. Acoustic echo cancellation (AEC) is a technique which enhances voice quality by removing acoustic echo using post processing. An adaptive filter is the major block in AEC, which subtracts estimated acoustic echo from the microphone signal. In fact, adaptive filter mimics acoustic channel by estimating room impulse response (RIR) [3].

Acoustic echo cancellation has four pivotal modules. They are echo detector, echo estimator based on adaptive filter, echo sub-tractor, and echo suppressor [4]. Note that echo suppressor and adaptive room impulse estimator are active only, when echo detector detects echo only state. Plethora of works have done on echo detection [5]. The conventional echo state detection method rely on energy difference and correlation between speaker input and microphone signals. False echo state detection results in divergence of adaptive finite impulse response (FIR) filter, which estimates RIR. Conventional echo detection methods causes slower convergence rate due to conservative approach in echo detection in order to avoid false detection [5, 6]. Supervised machine learning is an alternate to the less reliable energy and correlation based echo detector.

Machine learning (ML) facilitates systems to learn automatically without complex algorithms [1]. ML algorithms take images as input. Data to be processed is converted into required image size. These algorithms are K-nearest neighbor (KNN), Support vector machine (SVM) [7]. Features for these algo-

[1] https://www.expert.ai/blog/machine-learning-definition/

rithms are extracted base on type of the problem. These are basically classifiers that are less complex than deep learning algorithms.

Deep learning (DL) is category of machine learning that consist of trained complex algorithms [2]. These algorithms takes input as image, automatically extract features from it and classify classes. Algorithms are Convolution neural network (CNN) and Recurrent neural network (RNN) [8]. CNN takes input as image while RNN takes audio input. CNN algorithms are AlexNet and Deep convolution neural network (DCNN) [9]. RNN algorithms are Long short-term memory.

1.1 Related work

Acoustic echo detection and cancellation has received considerable attention. A research on echo detection has done using pattern recognition and correlation. A similarity function was generated by feature extraction techniques and this function detect echo and estimate delay in double talk scenarios [5].

Research on echo cancellation in IP networks is done using NLMS algorithm. A echo path locator is used that is helpful in finding delay and echo path [10]. Researchers also worked on echo cancellation and double talk detection with estimation of impulse response.New approach named as loudspeaker-impulse-response (LIME) adapt existing version of acoustic echo cancellation and double talk detector algorithms. When there is no local talk loudspeaker contains echo only and adaptive filter adapts easily but in case of double talk filter diverge. Here double talk detection is essential for adaptive filter to adapt. This LIME approach compared with normalized cross-correlation (NCR) and cross-correlation (CR) [11]. This approach shows less calculation complexity with double talk detection (DTD) [12].

Acoustic echo cancellation and noise removal are also elaborated as speech separation using deep learning networks. This approach used convolution recurrent network (CRN) and recurrent network with LSTM. These algorithms are trained on estimated near-end speech spectrograms so that near-end speech can detected from microphone signal. This near-end detector further suppress echo and noise from speeches [1].

[2] https://searchenterpriseai.techtarget.com/definition/deep-learning-deep-neural-network

Researchers extended work on AEC as speech separation that separates local talk and loudspeaker signal so that clear speech signal transmitted to far-end. Recurrent neural network with bidirectional LSTM is used that is trained to estimate ideal ratio masking [13]. Features are extracted from near-end and far-end signals. These features are concatenated and transferred to BLSTM network for training. This trained estimated mask is further used for separation of far-end signal. This trained model also removes echo from double talk signal [2].

Echo delay estimation is one of the demanding issue in echo cancellation. Various devices are used for delay estimation but unfortunately these devices make adaptive filter slow in convergence. Researchers worked on multi-task network that estimate delay and cancel echo. Two convolution neural networks are able to estimate echo path and magnify signals [14].

Recently, dual-signal transformation LSTM network (DTLN) utilized for acoustic echo cancellation. DTLN performed short Fourier transform and feature extraction in this approach. DTLN approach showed its performance on clean and noisy echo conditions. Network comprised with two cores. Each core consist of two LSTM layers and fully-connected layer. This network can be applied echo cancellation in real-time [15].

1.2 Motivation

Acoustic echo distort speech signal and result to poor quality transmission. Acoustic echo cancellers along with adaptive filters are used to remove echo. AEC with adaptive filter for echo detection finds complexity to converge in local talk presence. As a result adaptive filter fails to adapt. Microphone signal contains echoed signal plus local talk. In this research, we used deep learning networks for echo detection. We used CNN,RNN and machine learning algorithm K-nearest neighbor(KNN). We trained these networks on echo and no echo data sets then test these networks on new separate test data. Networks achieve good detection accuracy. These network as a echo detector are placed in conventional acoustic echo cancellation.

The propose detector detects echo only signal with high probability of true and adaptive filter adapts the RIR. Adaptive filter (NLMS) adapts using echo signal as an error and speaker signal as reference. In case of single talk, double talk and silence, NLMS doesn't adapt and echo detector doesn't converge. In few cases, if detector detect double talk as echo signal, then adaptive filter adapts using incorrect error and adaptive filter diverges.

1.3 Contribution

This thesis focuses on enhancing detection echo-only state of the AEC using deep learning networks. The contributions of this work are as follows:

1. We prepared data sets for the training and testing of the deep learning algorithms using speaker and microphone samples. For ground truth, we manually extracted echo-only and no-echo speech segments and prepared spectrogram in image format to use as input to the machine learning algorithms.

2. We adopted pre-trained AlexNet, DCNN and RNN for the echo-only segment detection. Adaptive algorithms for AEC use these echo-only segment for learning RIR and estimate acoustic echo.

3. We train and evaluate deep learning algorithms and compare their performance. The simulation results reveals that the proposed deep learning approach for echo-only detection provides encouraging results.

1.4 Thesis outline

Chapter1 consists of introduction of this thesis with related work, motivation and contributions.
Chapter2 contains basics of echo cancellers, echo detection using adaptive filter NLMS, how to implement AEC in deep learning, difference between deep learning and machine learning, transfer learning and introduction to AlexNet, DCNN, RNN and KNN.
Chapter3 explains about data collection, echo, no echo samples preparations, datasets details and explain algorithms working and training on echo and no echo datasets.
Chapter4 contains overview on datasets, feature extraction and accuracy calcu-

lations using CM for AlexNet, DCNN, RNN and KNN.
Chapter5 contains conclusion and future work of this work.

Chapter 2

Acoustic Echo cancellation and Deep Learning

In this chapter, we explain Acoustic echo cancellation, role of different states of echo cancellation and echo detection

2.1 Acoustic Echo cancellation

Acoustic echo cancellation is a well-known application of adaptive filters. Basically, adaptive filter estimates impulse response between loudspeaker and microphone. There are many adaptive filter introduced such as normalized least mean square (NLMS), least mean square (LMS) and recursive least square (RLS) [16]. The most popular adaptive algorithm is NLMS due to good convergence rate, stability and low complexity as compare to LMS and RLS [17]. Good convergence and stability depends up on the step size of NLMS algorithm.

Echo is one of major obstacle for the communication for the users. Basic task in AEC is the estimation of the impulse response between microphone and loudspeaker. Adaptive filter such as NLMS is used to estimate impulse response for AEC and other system identification problems. NLMS produces replica of echo, which is subtracted from the microphone output signal. NLMS models impulse response of the system [18]. Model of the conventional AEC is shown in Fig 2.1, where remote speech $x(n)$ is input to the speaker and $s(n)$ is the local speech signal. Acoustic echo canceller comprises of the four states as mention below:

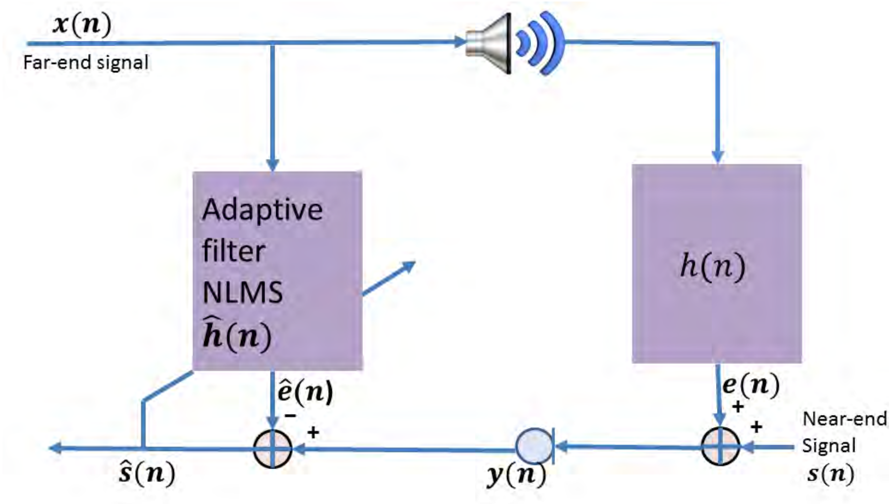


Figure 2.1: Conventional Acoustic echo cancellation model

1. $x(n) = 1, s(n) = 1$; **Double talk**
2. $x(n) = 0, s(n) = 0$; **Silence**
3. $x(n) = 0, s(n) = 1$; **Single talk**
4. $x(n) = 1, s(n) = 0$; **Echo only**

where $x(n)$ is far-end signal and $s(n)$ Local talk.

2.1.1 Principle of Acoustic echo canceler

Figure 2.1 describes the system model of acoustic echo canceler. Acoustic echo canceler takes input $x(n)$, which is far-end signal, and convolves with room impulse response(RIR) $\hat{h}(n)$ estimated by adaptive filter to estimate echo. Thus, far-en signal is fed to adaptive filter to estimate echo $\hat{y}(n)$. The estimated echo is then subtracted from the microphone signal and produces error signal $e(n)$. The error signal has local talk and residual echo, which adaptive filter can not estimate [19]. The error signal $e(n)$ contains echo speech when far-end is active and local end is silent. This state of AEC is echo only state. In this state, $e(n)$ contains only error in estimation of echo by adaptive filter. The error signal

$e(n)$ is used to training NLMS filter. Note that detection of echo only state is pivotal in the adaptation of the NLMS algorithm. Probability of false detection of echo only state is high when adaptive filter is not trained, which either causes divergence of AEC filter of slows down the convergence.

In this work, we propose deep learning approach to detect echo only state. Note that, frequency domain adaptive filter, time-domain NLMS method and filter banks are well-known approaches of RIR estimation in AEC [19]. Frequency-domain adaptive filter (FDAP) is quite useful in echo cancellers. It is applicable on adaptive filters with higher order. It achieves higher convergence rate with less complexity. This filter utilizes correlation and convolution methods for implementation of canceller in DFT domain. Wiener filter was introduced for echo and noise suppression [20]. Next, we explain NLMS algorithm.

2.1.2 Echo detection using NLMS

Adaptive filtering techniques have been extensively used in acoustic echo cancellation. Adaptive filters like normalized least mean square (NLMS) is good in terms of stability and convergence. Stability of NLMS can be achieved by factor such as step size μ. This factor controls stability and mean square error (MSE) [21, 22]. Conventional AEC is shown in Fig 2.1. Far-end signal $x(n)$ passes through the impulse response filter $h(n)$ and output $e(n)$ as shown in Figure 2.1. The output of RIR is

$$e(n) = x(n) * h(n) \tag{2.1}$$

This echo signal $e(n)$ mixes with near-end speech $s(n)$ an generate microphone signal $y(n)$.

$$y(n) = e(n) + s(n) \tag{2.2}$$

NLMS adaptive filter $\hat{h}(n)$ immitates RIR that generates estimated echo signal $\hat{e}(n)$. Estimate of local talk $\hat{s}(n)$ can be calculated by difference of microphone signal $y(n)$ and estimated local talk signal $\hat{s}(n)$.

$$\hat{s}(n) = y(n) - \hat{e}(n) \tag{2.3}$$

Room impulse response (RIR) $h(n)$ and estimated adaptive filter $\hat{h}(n)$ coefficients vector of length N are defined as:

$$h(n) = [(h_0(n)\ h_1(n)\ h_2(n) \dots h_{N-1}(n)]^T \tag{2.4}$$

$$\hat{h}(n) = [\hat{h}_0(n)\ \hat{h}_1(n)\ \hat{h}_2(n) \dots \hat{h}_{N-1}(n)]^T \tag{2.5}$$

where T express transpose of vector coefficients. Using (2.1) and (2.1), $y(n)$ becomes:

$$y(n) = [x_0(n)\ x_1(n) \dots x_{N-1}(n)]^T h(n) + s(n), \tag{2.6}$$

where $x(n) = [x_0(n)\ x_1(n) \dots x_{N-1}(n)]$.
So (2.6) is

$$y(n) = x^T(n)h(n) + s(n) \tag{2.7}$$

Using (2.3) we get

$$\hat{s}(n) = y(n) - \hat{e}(n) \tag{2.8}$$

$$\hat{s}(n) = y(n) - x^T(n)\hat{h}(n-1) \tag{2.9}$$

Now, we have

$$\hat{s}(n) = x^T(n)h(n) + s(n) - x^T(n)\hat{h}(n-1) \tag{2.10}$$

$$\hat{s}(n) = x^T(n)[h(n) - \hat{h}(n-1] + s(n) \tag{2.11}$$

(2.10) represent estimated local talk $\hat{s}(n)$ at $n-1$ time. Now estimated local talk at n time as:

$$\hat{s}_i(n) = x^T(n)[h(n) - \hat{h}(n)] + s(n) \tag{2.12}$$

Updated NLMS equation is:

$$\hat{h}(n) = \hat{h}(n-1) + \mu(n)x(n)\hat{s}(n) \tag{2.13}$$

where $\mu(n)$ represent stability of adaptive filter NLMS. Putting (2.14) in (2.13).

$$\hat{s}(n) = x^T(n)[h(n) - \hat{h}(n-1) + \mu(n)x(n)\hat{s}(n)] + s(n) \tag{2.14}$$

Rearranging (2.15), we have

$$\hat{s}(n) = x^T(n)[h(n) - \hat{h}(n-1)] + s(n) - x^T(n)\mu(n)x(n)\hat{s}(n) \tag{2.15}$$

Using (2.12) and taking $\hat{s}(n)$ common we get:

$$\hat{s}_i(n) = \hat{s}(n) - x^T(n)\mu(n)x(n)\hat{s}(n) \tag{2.16}$$

$$\hat{s}_i(n) = \hat{s}_i(n)[1 - \mu(n)x(n)x^T(n)] \tag{2.17}$$

Assume $\hat{s}_i(n)$ =0

$$1 - \mu(n)x(n)x^T(n) = 0 \tag{2.18}$$

$$\mu(n)x(n)x^T(n) = 1 \tag{2.19}$$

So,

$$\mu(n) = \frac{1}{x(n)x^T(n)} \tag{2.20}$$

Assuming a constant α called as normalized step size that multiply by (2.19) keeps balance in adjustment and rate of convergence. We assume δ as regularization constant [23]. Putting (2.14), we get:

$$\hat{h}(n) = \hat{h}(n-1) + \frac{\alpha x(n)\hat{s}(n)}{(x(n)x^T(n) + \delta} \tag{2.21}$$

NLMS performance depends on two factors α and δ [22]. (2.14) and (2.21) play important role in AEC echo detection. (2.21) controls stability and convergence of NLMS while (2.14) updates when echo sample detected.

2.2 Deep learning and Machine learning

Deep learning (DL) is subgroup of machine learning (ML) in artificial intelligence that concentrates on development of neural networks, train networks on labeled or unlabeled data set. The network learns from the data set and classifies objects with accuracy [24]. Accuracy of network can be enhanced using large training data set. Deep learning has many applications such as mobile face detection lock, echo detection, speaker identification and speech recognition [3].

Machine learning is defined as the branch of knowledge through which machines learn automatically from large data sets and produce results. In machine learning research, researchers focus on introducing new algorithms for the training of the classifiers. Basically, machine learning is branch where feature extraction is performed manually from data, trained on data set in well organized manner [25]. In contemporary innovations, machine learning plays key role in our daily routines. It is quite useful for the programmers to solve complex problems effectively.

Machine learning is further classified into four classes. They are, Supervised learning, Unsupervised learning, Semi-Supervised learning and Reinforcement learning [4]. Supervised learning refers to labeled data [26]. Training data comprised of input vector and label as output vector. This learning basically maps input over output.

Unsupervised learning utilizes unlabeled data set, having no process of cross validation [27]. Clustering in machine learning is example of unsupervised learning. In our research, supervised learning is used where we have labeled data set of echo and no-echo.

2.2.1 Deep learning vs Machine learning

Machine learning is subclass of artificial intelligence that enables systems to learn automatically. Arthur Samuel elaborates machine learning as ”discipline of study that provides potentials to computers to acquire knowledge without using complex programming”. Deep learning is new class in research of ML. Deep learning fabricates neural networks that trained like human brain. It has complex structure algorithms [5].

Machine learning algorithms are less complex as compare to deep learning algorithms. DL needs powerful hardware for training algorithms. GPU can be utilize in DL algorithm training as it contains more memory. ML algorithms trains quickly and DL algorithms takes much longer time for training due to complex structure. In our project, ML takes audio data as input whereas DL

[3] https://www.simplilearn.com/tutorials/deep-learning-tutorial/what-is-deep-learning

[4] https://towardsdatascience.com/types-of-machine-learning-algorithms-you-should-know-953a08248861

[5] https://flatironschool.com/blog/deep-learning-vs-machine-learning

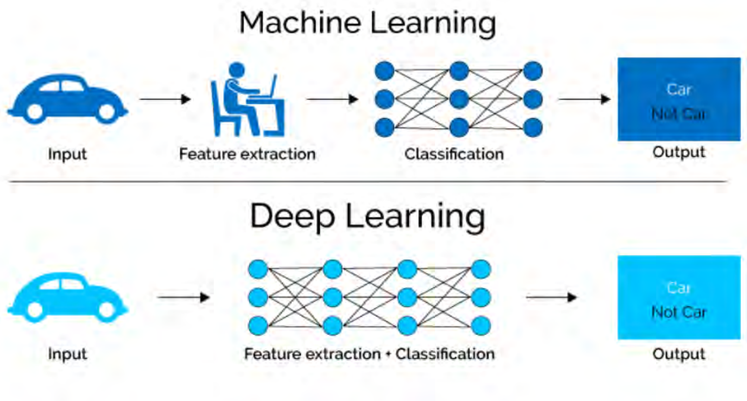


Figure 2.2: Deep learning vs Machine learning

algorithms take image as input. In ML, algorithms spectral features extraction are perform manually whereas in deep learning, network extract features automatically [26]. Difference is shown in Fig 2.2 [6]. Application of ML uses in email box and bank while DL applications are robots and self-driving cars[7].

2.3 AEC implementation in deep learning

In AEC, major issue is to estimate the impulse response between loudspeaker and microphone signals. Acoustic echo causes distortion in communication and audio systems [20]. Researchers worked on AEC systems that aims to remove the echo and maintain near-end speech [28].

Deep learning methods has been introduced to resolve AEC problem [29]. We utilize deep learning networks like CNN , RNN and machine learning algorithm like K-nearest neighbor (KNN). CNN algorithms are AlexNet and DCNN. We trained these algorithms on echo and no-echo (single talk and double talk) data set. After training, we tested these trained algorithms on new data set.

Fig 2.3 shows a echo detector block in conventional acoustic echo canceller (AEC) figure. Trained deep learning algorithms as an echo detector works in Fig 2.3.

This echo detector detects echo and adaptive NLMS filter adapt to the echo signal and converges. In case of no echo (single talk or double talk) echo de-

[6] https://ieeecs-media.computer.org/wp-media/2021/06/15234622/machinelearning1_-550x271.jpg

[7] https://flatironschool.com/blog/deep-learning-vs-machine-learning

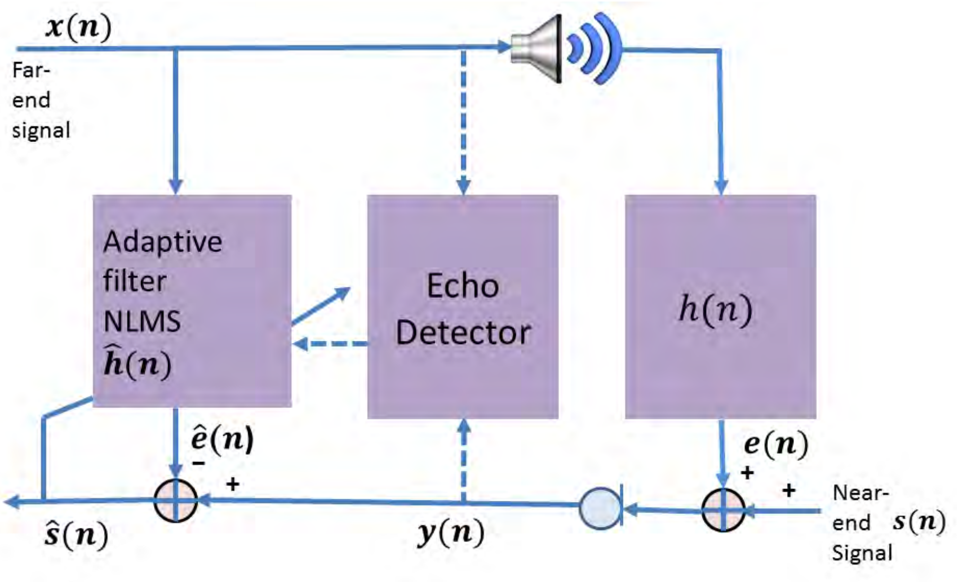


Figure 2.3: AEC with trained deep learning models

tector fails to detect signal and adaptive filter fails to adapt. In case of double talk, echo detector may be in few cases detect double talk as echo and adapts resulting in the NLMS to diverge. This trained detector helps adaptive filter from false detection and convergence.

CNN, RNN and KNN algorithms introduction is given below. AlexNet and DCNN are convolution neural networks [8].

2.4 Transfer Learning

Transfer learning is an approach used in machine learning and deep learning [30, 31]. This approach is formulated for one classification task and then can be again used as a scratch point in a new task. Pre-trained networks are utilized as beginning point in new research work. These networks assist us from complex computation and more time needed to design a scratch network. Transfer learning can be used in two ways. Firstly, feature extraction from pre-trained network and then train the network on it. Secondly, fine-tuning the pre-trained network and keep weights learned as initial parameter. Fine tuning basically uses previous trained network weights and avoid to adjust and prepare network from scratch [32].

2.5 Convolution neural network (CNN)

AlexNet is well-known pre-trained CNN [9]. It requires large set of images as input with size 227x227x3. The structure of AlexNet consists of input layer, multiple hidden layers and output layer as shown in Fig 2.4 [8] shows layers. Hidden layers comprise of convolution layer, pooling layer, rectified linear unit (ReLU) layer. Convolution layer performs convolution by extracting features from input with a convolution matrix 3x3. Extracted features are convolved features. After convolution layer, non-linear activation layer as ReLU layer maintain image feature pixel from negative value [33].

Pooling layer is placed after ReLU and performs down sampling, which lessens dimensions of the feature image. It reduces calculations and parameters in network [9].

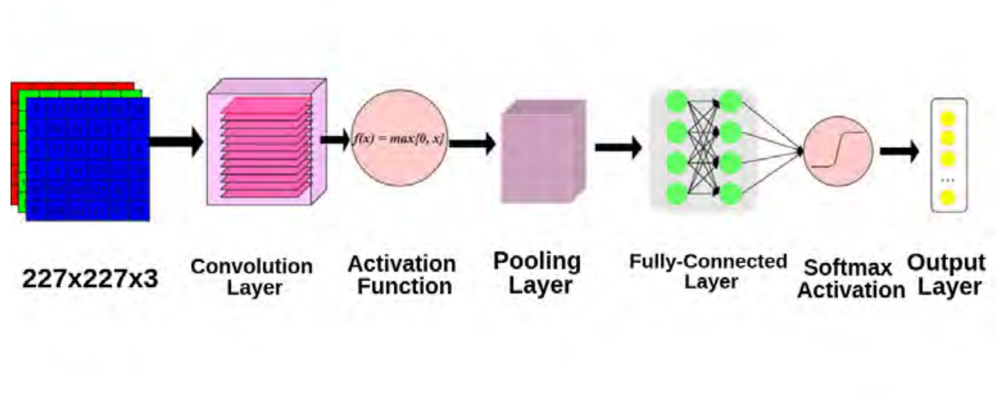


Figure 2.4: AlexNet model

AlexNet perform transfer learning [32]. It takes trained network and updates only last three layers according to new task and train [10]. Last three layers

[8] https://storage.googleapis.com/lds-media/images/cnn-architecture.width-1200.jpg

[9] https://medium.com/technologymadeeasy/the-best-explanation-of-convolutional-neural-networks-on-the-internet-fbb8b1ad5df8

[10] https://learnopencv.com/understanding-alexnet/

are fully-connected layer, softmax layer and output layer. Size of fully connected layer is same as number of classes for training. Deep convolution neural network (DCNN) is also one of CNN network. Architecture of DCNN is similar to AlexNet. It also takes image input but input size is 224x224x3.

DCNN also comprise of convolution layer, Pooling layer, Fully connected layer, ReLU and output layer [34]. These layers are deep as compared to Alex Net [11].

2.6 Recurrent neural network (RNN)

Recurrent neural network (RNN) consist of hidden layer long short-term memory (LSTM) that takes in sequence time-series data (audio) [35]. This network comprises cyclic connections that helps to deal with sequential data. RNN also deals with sequence labeling, language modelling and many more.

RNN-LSTM [36] network consists of memory blocks. These blocks consist of self connections that save temporal state of algorithm.

Memory block consists of input and output gate. Input gate saves input activation in memory cell. Output gate controls output activation in cell [37]. Spectral Features can be extracted in RNN manually.

2.6.1 Spectral features

Spectral features are the specific time-frequency features. CNN networks like AlexNet and DCNN extract features automatically [38] but RNN cannt extract features on their own so we have to extract features separately and then input to the network. Spectral features are pitch, MFCC, spectral centeriod, spectral flux, spectral roll-off etc [12].

2.7 Summary

In this chap, we explain about acoustic echo, acoustic echo detection and acoustic echo cancellation. We explain about adaptive filter NLMS and its advan-

[11] https://www.quora.com/Is-there-any-difference-between-CNN-and-Deep-CNN

[12] https://www.researchgate.net/post/What-are-the-Spectral-and-Temporal-Features-in-Speech-signal

tages. Deep learning techniques are applicable for echo detection. We explain difference between machine learning and deep learning. We have discussed deep learning algorithms like AlexNet with transfer learning, Deep convolution neural network (DCNN), recurrent neural network (RNN) and machine learning algorithm K-nearest neighbor (KNN).

Chapter 3

Methodology

In this chapter, we discussed the time series data collection and preparation. Preparation of echo and no-echo chunks from the audio samples and feature extraction for echo detection is presented. We also discussed deep learning models used for the experimentation and their performance for echo detection.

3.1 System Architecture

Here, we discussed different steps for echo detection. The steps in audio collection includes, pre-processing of the data, dividing data into training and testing sets, feature extraction, network training and classifying labels as shown in Fig 3.1.

First step involves the collection of the audios samples of the far-end and near-end speeches. In second step, played speech(far-end) and microphone signal(echoed signal+near-end signal) are compared. The captures samples files are divided in to equal segments and manually marked as echo, single talk and double talk and save in labeled files. In third step, features are extracted from the data set. In the fifth step, these features are passed to networks and data set get trained. Sixth step involves the testing of the trained network on new test data set and calculates the accuracy of network using confusion matrix.

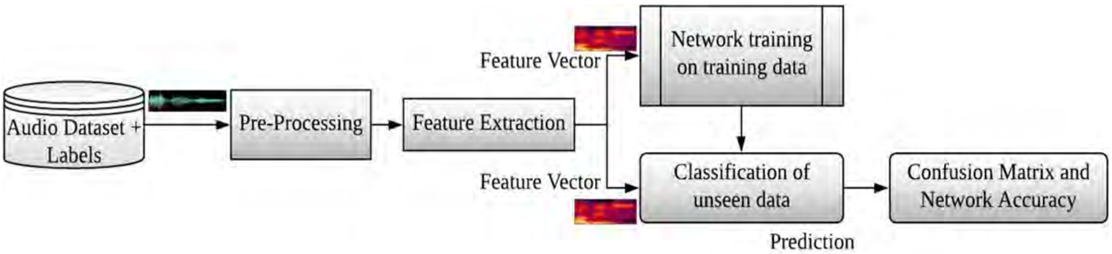

Audio Dataset + Labels
Pre-Processing
Feature Extraction
Feature Vector
Network training on training data
Feature Vector
Classification of unseen data
Confusion Matrix and Network Accuracy
Prediction

Figure 3.1: Block diagram of system architecture

3.2 Time series data preparation

In deep learning field, dataset preparation and feature extraction are basic components for deep learning algorithms. In this work, we have 30 recorded speeches sampled at 16kHz. Thus 30 Local talk or near-end speeches collected from various speakers. Two room impulse responses(RIRs) are used for the preparation of data set. We convolved captured samples with room impulse response(RIR) to produce echo signal. The output of RIR filter, which is echo signal, is added to different captured audio signal (local talk) to generate microphone signal. We examine both audios and marked points of echo and no-echo segments. We concatenate 2048 samples of input (played audio) to RIR filter with the corresponding 2048 microphone speech. Thus, each audio segment consist of 4096 samples for training and testing of the deep learning algorithms. We used AlexNet, DCNN, RNN deep learning models and KNN a machine learning algorithms. We wrote MATLAB script to generate samples in semi-automatic fashion and labeled as echo and no-echo samples for deep learning algorithms.

3.2.1 Echo samples preparation

Echo samples have 2048 samples of active speech of far-end signal (played speech) and 2048 samples of microphone sample with local speaker silent. All such segments of 4096 samples are labeled as echo data samples. We have used 15 reference voices for echo sample preparations. Each reference voice convolved with RIR room impulse response and convolved output of RIR are referred as echoed signal. As near end person is silent, microphone signal contains ”echo signal” only. Now, each reference signal and microphone signal are concatenated into equal segments of 2048 samples. Room impulse response introduces delay of 24 samples after convolution. We have collected 1000 equal size echo segments for training of deep learning algorithms. For data set 2, we have used same voices with different RIR delay of 30 samples and prepared 1000 equal size echo only samples.

3.2.2 No-echo samples preparation

No-echo samples contains single talk and double talk audio segments generated from played signal and microphone signal. We have 15 recorded voices and 15 near end speeches. Each far-end voice convolved with RIR to generate echo signal. RIR filter introduces delay of 24 samples. Each near-end speech (local talk) is added with echo signal and generate microphone signal. Reference signal and microphone signal are divided into 2048 samples each. When reference samples contain silence and microphone contains voice samples, such samples are denoted as ”single talk” of samples size 4096. Similarly, when reference segment contains voice samples and microphone also contains voice samples, such segments are referred as ”double talk”. We have prepared 1000 samples of no-echo samples for training. We have used same voices with second filter with delay of 30 samples and generated 1000 no-echo samples.

3.2.3 Feature extraction

Second step after data collection and preprocessing is feature extraction. In feature extraction, audio data reduces into different categories for processing [13]. Algorithms process feature vector and classify in to classes.

In CNN, feature extraction takes place automatically. We only convert audio data set to mel-spectrograms [39]. In this work, we have converted audios to spectrograms for CNN (AlexNet and DCNN) as these networks take image data as input. These networks extract large number of features from input image and takes only useful features for echo, no-echo classification.

RNN takes audio data input and acquires features manually [40]. In RNN, we extract spectral features from audios and provide to network. Spectral features are pitch, MFCC, spectral centroid, spectral flux etc [14]. Extracted features then provide to network for classification. In our work, we extract 17 spectral

[13] https://deepai.org/machine-learning-glossary-and-terms/feature-extraction

[14] https://www.researchgate.net/post/What-are-the-Spectral-and-Temporal-Features-in-Speech-signal

features. They are pitch, MFCC, spectral centroid. These features are used in RNN and machine learning algorithms. These features help in speech classification, voice activity detection in noise etc [15].

3.3 Learning networks

In this thesis, we use convolution neural network (CNN), which is a deep learning network and can be trained as bibary classifier to detect echo and no-echo speech segments. We use the following CNN network:

1. AlexNet with transfer learning.
2. Deep convolution neural network (DCNN)
3. Recurrent neural network (RNN)

For Machine learning classifier, we consider K-nearest neighbor (KNN) network.

3.3.1 AlexNet with transfer learning

AlexNet is one of pre-trained deep learning algorithm based on images [16]. In the recent years, image recognition have achieved remarkable achievement due to new advanced deep learning models [41].

Transfer learning is the key achievement of the pre-trained networks such as AlexNet [42]. Transfer learning is an approach used in machine learning and deep learning [30, 31]. This approach is formulated for one classification task and then can be used for fine tuning for a new classifier to solve new problem. Pre-trained networks are utilized as starting point in new deep learning application. This helps approach provides means of better trained network at the top of previous training. Transfer learning can be used in two ways. Fine tuning basically uses previous trained network weights instead of training from scratch. AlexNet architecture consists of five convolutional layer, three pooling layers, a fully connected layer, dropout layer, softmax layer and outputs labels

[15] https://www.mathworks.com/help/audio/ug/voice-activity-detection-in-noise-using-deep-learning.html

[16] https://www.mathworks.com/help/deeplearning/ug/transfer-learning-using-alexnet.html

according to number of outputs. Convolutional layers perform feature extraction from the inputs [43]. Pooling layers down-sample the feature vector and made internal calculations less complex [17]. Dropout layer prevents algorithm from the over-fitting [18]. Softmax layer performs as an activation function in output layer. It limits the outputs as 0 or 1 [19].

We use AlexNet as pre-trained network for the echo detection as shown in Fig 3.2. We use two data sets containing echo and no-echo audio samples in image format. Since AlexNet takes image data as input, we first convert these audio segments to melspectrograms and resized these images to AlexNet input size 227x227x3. Feature extraction takes place automatically by convolutional layers. For the pre-trained network, we use last three layers according to labels 2 echo and no-echo. In this way, we have trained AlexNet on echo and no-echo data set. We have tested trained AlexNet using testing data and achieved good accuracy.

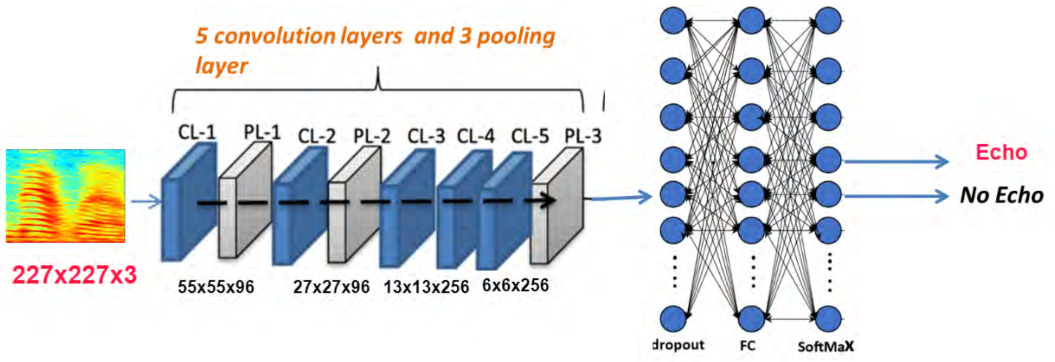


Figure 3.2: AlexNet architecture

Flo of MATLAB implementation for echo detection using AlexNet training is shown here in algorithm 1.

3.3.2 Deep convolutional neural network (DCNN)

Convolution neural network (CNN) [44] comprises of independent filter utilized for image classification and regression with deep structure also known as

[17] https://machinelearningmastery.com/pooling-layers-for-convolutional-neural-networks/

[18] https://towardsdatascience.com/machine-learning-part-20-dropout-keras-layers-explained-8c9f6dc4c9ab

[19] https://www.quora.com/What-is-Softmax-in-CNN

ALGORITHM 1: Echo detection using AlexNet

```
Data for training and testing
adsTrain=trainingDatasetfolder
adstest=testingDatasetfolder
creating labels
PTrain = adsTrain.Labels
QTest = adstest.Labels
training
trainAlexNet = trainNetwork(PTrain,QTrain,layers,options)
predict on test data
[AlexTest nProb]=classify(TrainAlexNet,Qtest
Accuracy
ALEXaccuracy=sum(AlexTest==QTest)/Qtest)*100
End
```

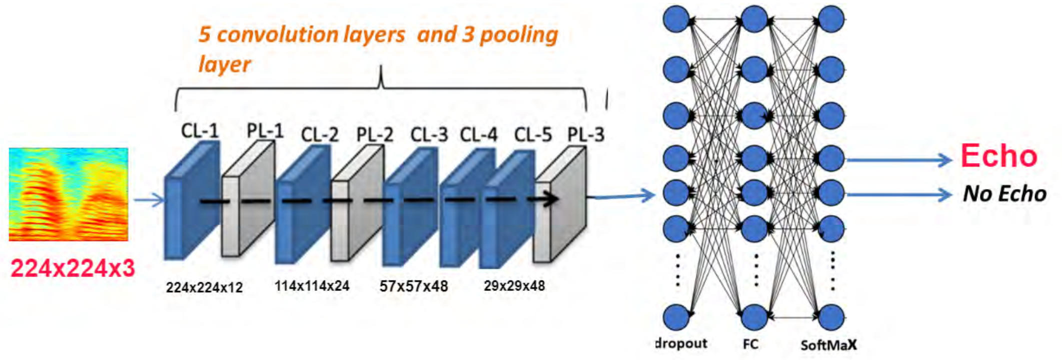


Figure 3.3: DCNN architecture

DCNN. It also take image as input. Each convolutional layer perform feature extraction from spectrograms. Remaining layers use feature extraction and reduce complex calculations in network. Input size of DCNN is 224x224x3. We have trained echo and no-echo on DCNN. We have set of parameter like segment duration, frame duration, hop duration and number of bands. These parameters are defined as:

1. Segment duration is the complete duration of a sample.
2. Frame duration contains duration of each sample spectrogram frame.
3. Hop duration is time shifting duration in each spectrogram of sample.

Using these parameters, we have converted audios to spectrograms and provided to DCNN. DCNN layers are shown in Fig 3.3. We have trained DCNN on two training data sets and then tested this algorithm on separate testing data sets. We have used MATLAB code for echo detection. Algorithm is shown as below:

ALGORITHM 2: Echo detection using DCNN

```
          Data for training and testing
          adsTrain=trainingDatasetfolder
          adstest=testingDatasetfolder
          for i = 1:numFiles
          Hop_Dr = Hop Duration
          Frame_Dr = Frame Duration
Algorithm
          Begin
          nBands, Seg_Dr, Hop_Dr, Frame_Dr ← Define Parameters
          melspectrogram(adsTrain[], Seg_Dr, Hop_Dr, Frame_D, nBands)→XTrain[]
          melspectrogram (adsTest[],Seg_Dr, Hop_Dr, Frame_D, nBands)→ XTest[]
          YTrain [] ← adsTrain.Labels[]
          YTest[] ← adsTest.Labels[]
          trainNetwork ← XTrain[], YTrain[], layers, options
          YPredicted[], Probability[] ← classify(trainedNetwork, XTest)
          Accuracy ← mean(YPredicted[] == YTest[])
          Confusion Matrix← YPredicted[], YTest[]
          End
```

3.3.3 Recurrent neural network (RNN)

RNN [45] is one of artificial neural network works with sequential data [20]. Long short term memory (LSTM) is hidden layer of RNN which helps RNN to store memory while training. RNNs achieves good performance in classification, prediction and speech recognition [46].

[20] https://en.wikipedia.org/wiki/Recurrent_neural_network

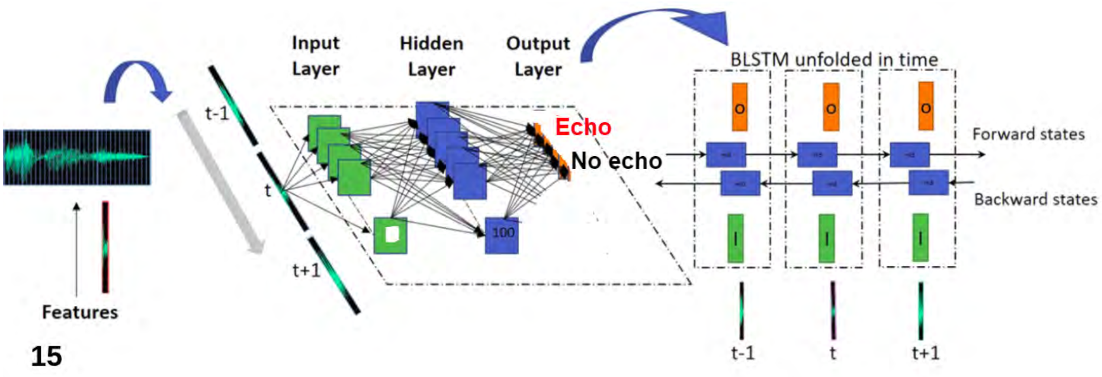


Figure 3.4: RNN architecture

RNN architecture is shown in Fig 3.4 [21]. RNN takes audio as input and feature extraction performs manually. RNN has input layer, hidden layer and output layer [13]. BLSTM [2] contains two LSTMs that has two direction left to right or right to left. We have used RNN for echo detection. We have used echo and no-echo data set, extracted 15 features like pitch and MFCC. We have provided feature vector to RNN and it classify echo and no-echo. We have used MATLAB code for echo detection algorithm 3 is shown:

[21] https://doi.org/10.3390/app11062508

ALGORITHM 3: Echo detection using RNN

```
          Data for training and testing
          adsTrain=trainingDatasetfolder
          adstest=testingDatasetfolder
Algorithm
          Begin
          Feature extraction
          trainfeatures[] = ExtractFeature(Traindata)
          testfeatures[] = ExtractFeature(testdata)
          trainNfeatures[]=trainfeatures.Labels
          testMfeatures[]=testfeatures.Labels
          define image size
          ImageSIZE = inputSize[]
          xtrain=trainfeatures
          ytrain=trainNfeatures
          xtest=testfeatures
          ytest=testMfeatures
          training
          RNNnetwork = trainNetwork(xtrain,ytrain,layers,options)
          predict on test data
          [RNNTest nProb]=classify(RNNnetwork,ytest
          Accuracy
          RNNaccuracy=sum(RNNTest==ytest)/ytest)*100
          YPredicted[], YTest[]
          End
```

3.3.4 K-Nearest neighbor (KNN)

KNN [47] is a supervised machine learning classifier. It takes audio labeled data set for training. For training, important parameter of KNN is number of nearest neighbors. For continuous data, KNN uses euclidean distance for calculations of nearest neighbor. It is known as lazy algorithm as it takes data set and memorize it then classify on new data [22]. We use KNN algorithm for echo and no-echo detection. We extracted 18 spectral features from the data set and provided to the network. After training, we tested KNN on different test data. We trained network on two data sets on KNN and then tested on two testing data sets.

[22] https://www.i2tutorials.com/why-knn-algorithm-is-called-as-lazy-learner/

3.4 Summary

In this chapter, we explained in details the data collection and preparation. We also explained AlexNet, DCNN, RNN and KNN and trained on echo detection. These echo detectors can be utilized in conventional echo cancellers for correct echo detection and controlling adaptation of NLMS convergence.

Chapter 4

Results And Performance Analysis

In this chapter, we present data sets and performance of the deep learning networks. The deep learning networks considered are AlexNet, DCNN, RNN, whereas machine learning algorithm we used is KNN for echo detection. Next, we discuss data set preparation deep learning and machine learning algorithms.

4.1 Data Set Preparation

We collected 30 far-end and 30 near-end speeches for data set from different speakers. Each speech is sampled at 16kHz with 16-bit PCM raw format. We used 2 room impulse responses (RIRs) with different delays. We generated echo signal by applying RIR on the speaker signal (far-end speech). In order to generate microphone signal, we added output of RIR to the near-end speech. By carefully and manually comparing reference signal (far-end) and microphone signal, we collected echo and no-echo samples of 2048 samples each. By concatenating 2048 samples of speaker signal and 2048 samples of microphone signal, we prepare one data sample of 4098 samples. Thus, one sample of data set is a sample segment of 4096 samples. By using 30 speeches from far-end and near-end speakers along with two RIRs, we prepared 4000 data samples.

In this way, we have prepared two data sets with RIRs delay of 30 and 20 samples. Fig 4.1 shows an echo sample containing 4096 total samples. First 2048 speech samples are played samples and next 2048 speech samples are recorded samples. Similarly Fig 4.2 shows a no-echo sample containing 4096 total samples. We have converted audio samples to melspectrograms as we need image data for AlexNet and DCNN training.

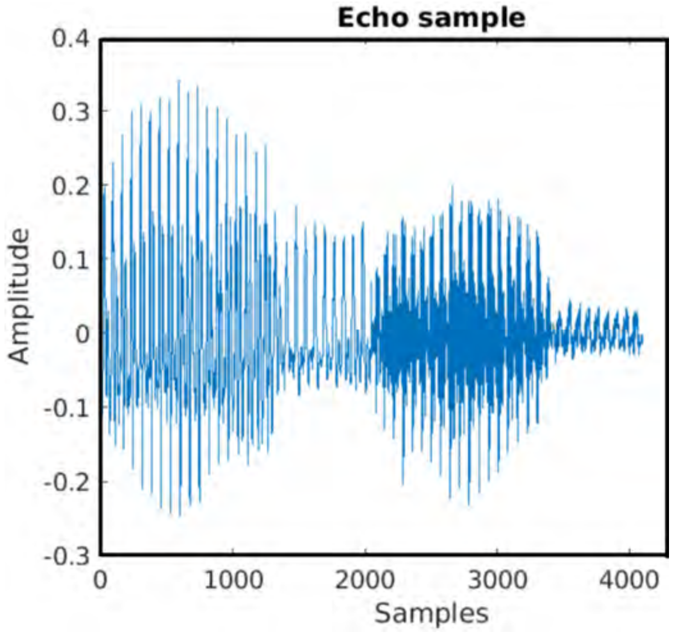


Figure 4.1: Echo sample waveform

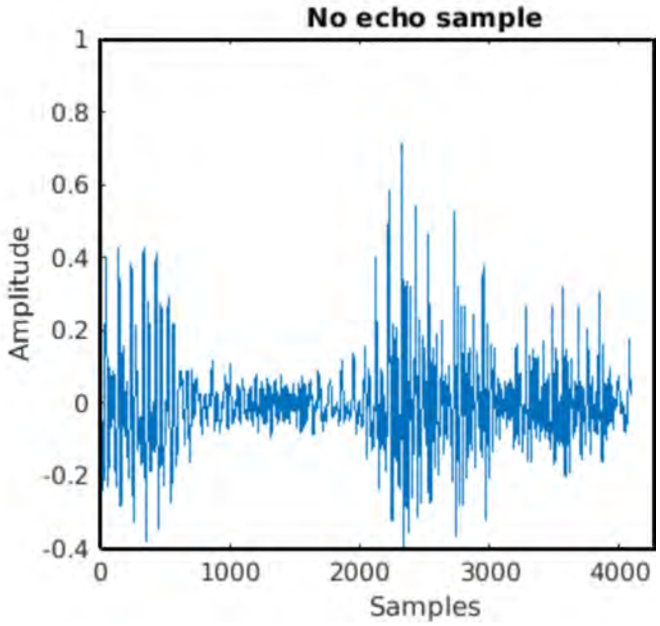


Figure 4.2: No-echo sample waveform

Fig 4.3 and Fig 4.4 shows echo spectrogram with 4096 total samples (2048 played and 2048 recorded) and no-echo spectrogram respectively.

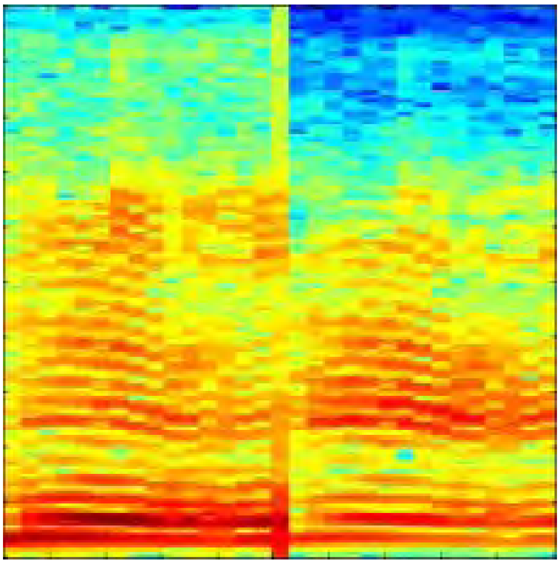


Figure 4.3: Echo sample spectrogram

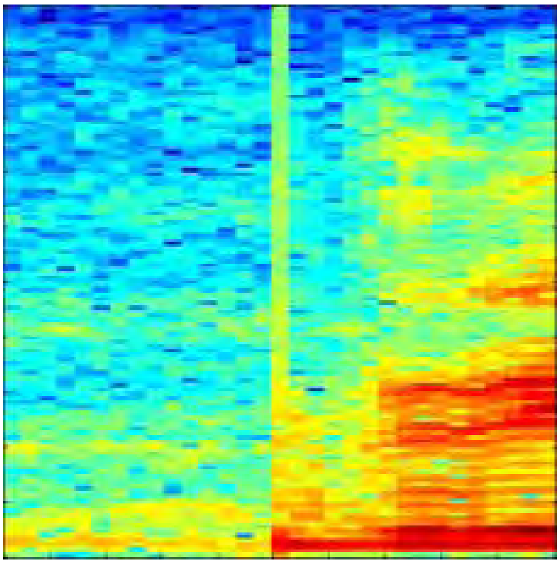


Figure 4.4: Spectrogram of no-echo sample

4.1.1 Spectral Features

AlexNet and DCNN extract features automatically from the input images [26]. In case of KNN and RNN, we extract spectral features like MFCC, Pitch, Spectral centroid, spectral flux etc. MFCC divides audio into windows, calculate discrete Fourier transform (DFT) and apply log- function on the magnitude and convert frequencies to mel scale [23].

Pitch feature calculate low frequency oscillations in voice segments [24]. Spectral centroid calculates spectrum location [25]. Spectral flux examines the behavior of power spectrum of audio signal and compare spectrum of each frame [26].

4.2 Training Parameters

AlexNet and deep convolution neural network (DCNN) are trained on two data sets of both classes. Before training both networks, parameters are tuned as shown in Table4.1:

Table 4.1: Training parameters of AlexNet and DCNN

Parameters	DCNN	AlexNet	RNN
Max Epochs	30	30	300
Learning rate	0.0001	0.0001	0.00001
Batch Size	50	50	50

Adaptive moment estimation (ADAM) optimizer is one of best optimizer and it optimize neural networks to train in short time, reduce losses and understand data more quickly. Adam optimizer has used in stochastic gradient descent approaches for training models. It has good properties which are helpful in noisy problems handling [27].

Epoch is defined as number of times network inspect data set. Learning rate supervises how efficiently model trains on data set. Learning rate is inversely

[23] https://en.wikipedia.org/wiki/Mel-frequency_cepstrum

[24] https://la.mathworks.com/help/audio/ug/speaker-identification-using-pitch-and-mfcc.html;jsessionid=47add25b38e551e0cd9a6f461f5d

[25] https://en.wikipedia.org/wiki/Spectral_centroid

[26] https://en.wikipedia.org/wiki/Spectral_flux

[27] https://towardsdatascience.com/optimizers-for-training-neural-network-59450d71caf6

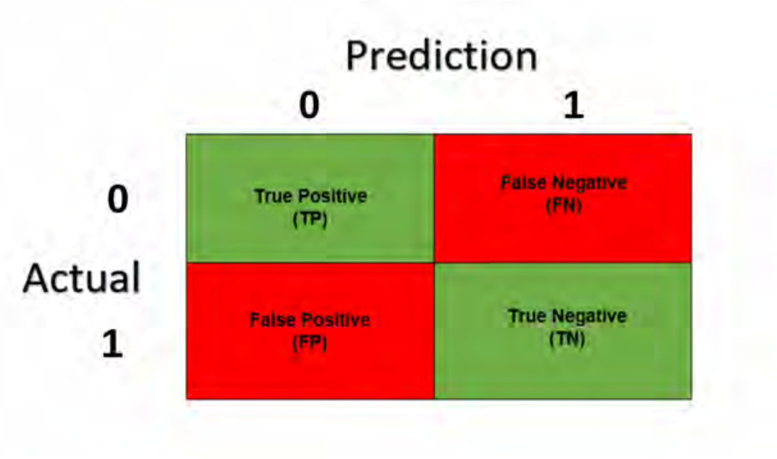


Figure 4.5: Confusion Matrix

proportional to the epochs [28]. Batch size shows number of samples handled before network get updated.

4.2.1 Confusion matrix

Confusion matrix is a presentation of the results of binary and multi-class classification problem [48]. It consist of four states as shown in Fig 4.5[29].

1. True positive.
2. True negative.
3. False positive.
4. False negative.

It describes performance of network that how accurate network detects or classifies. It also shows how many test files network detect correctly and incorrectly.

[28] https://machinelearningmastery.com/understand-the-dynamics-of-learning-rate-on-deep-learning-neural-networks/

[29] https://prasantmahato989.medium.com/cyber-crime-with-confusion-matrix-536719d1df1e

4.2.1.1 True Positive (TP)

True positive shows how many samples network detects correctly for 0 class. True positive is also called sensitivity. Rate of true positive can be calculated as:

$$TP\_rate = \frac{TP}{TP + FN} \tag{4.1}$$

4.2.1.2 True Negative (TN)

True negative shows how many samples for 1 class network detects correctly. True negative is also called as specificity. Rate of negative class can be calculated as:

$$TN\_rate = \frac{TN}{TN + FP} \tag{4.2}$$

4.2.1.3 False Positive (FP)

False positive comprises samples of class 1 that network detect as class 0.

$$FP\_rate = \frac{FP}{FP + TN} \tag{4.3}$$

4.2.1.4 False Negative (FN)

False negative shows samples of class 0 detect as class 1.

$$FN\_rate = \frac{FN}{FN + TP} \tag{4.4}$$

4.2.2 Accuracy

Accuracy describes the performance of network that how overall network predicts. Accuracy can be calculated by following formulae:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \tag{4.5}$$

4.3 Performance of AlexNet, DCNN, RNN and KNN

Now, we present performance of the trained networks for the detection of echo and non-echo states, which are binary classifiers. We present performance of the binary classifiers using confusion matrix. Two echo detection data sets used to train AlexNet, DCNN, RNN and KNN networks for binary classification. Confusion matrix shows the percentage of files (samples) detected accurately.

Confusion matrix for AlexNet with dataset 1 is shown Fig 4.6

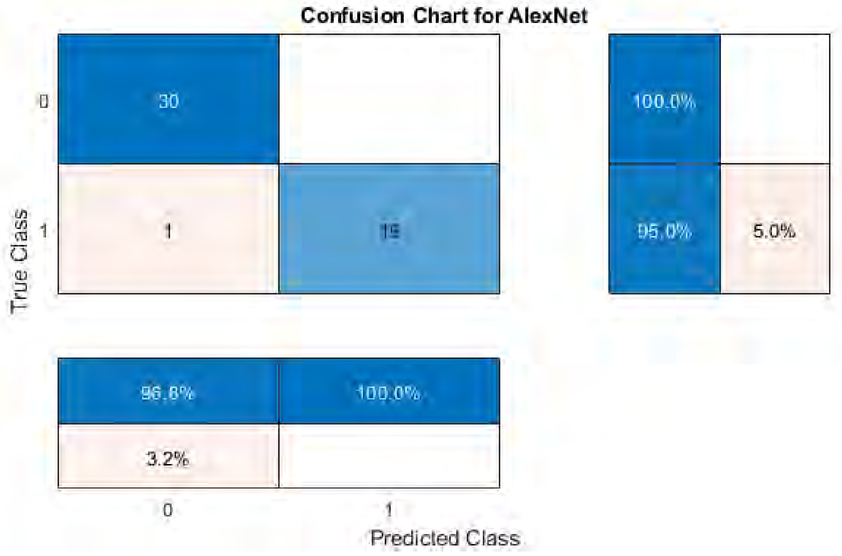


Figure 4.6: Confusion matrix of AlexNet with dataset 1

We tested AlexNet on 30 no-echo samples and 20 echo samples. In the confusion chart 0 label represent no-echo samples and 1 label represent echo samples. True positive shows that we have provided 30 no-echo samples files to the AlexNet network and it predicts all files correctly as no-echo. The 20 echo files are provided to AlexNet. It predicts 19 echo files correctly and 1 echo file as no-echo. The percentage of true class and predicted classes are shown in Fig 4.6.

Overall accuracy of the AlexNet network can be calculated as follows:

$$AlexNetAccuracy1 = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \tag{4.6}$$

$$AlexNetAccuracy1 = \frac{30 + 19}{30 + 19 + 1} \times 100 \tag{4.7}$$

$$AlexNetAccuracy1 = 98\% \tag{4.8}$$

Thus, AlexNet achieves 98% testing accuracy with data set 1. Error can be calculated as:

$$Error = \frac{FN + FP}{TP + TN + FP + FN} \tag{4.9}$$

$$Error = \frac{1 + 0}{1 + 19 + 30} \tag{4.10}$$

$$Error = 0.02 \tag{4.11}$$

Confusion matrix with data set 2 for the AlexNet is shown in Fig 4.7. Accuracy of AlexNet with data set 2 can be calculated as:

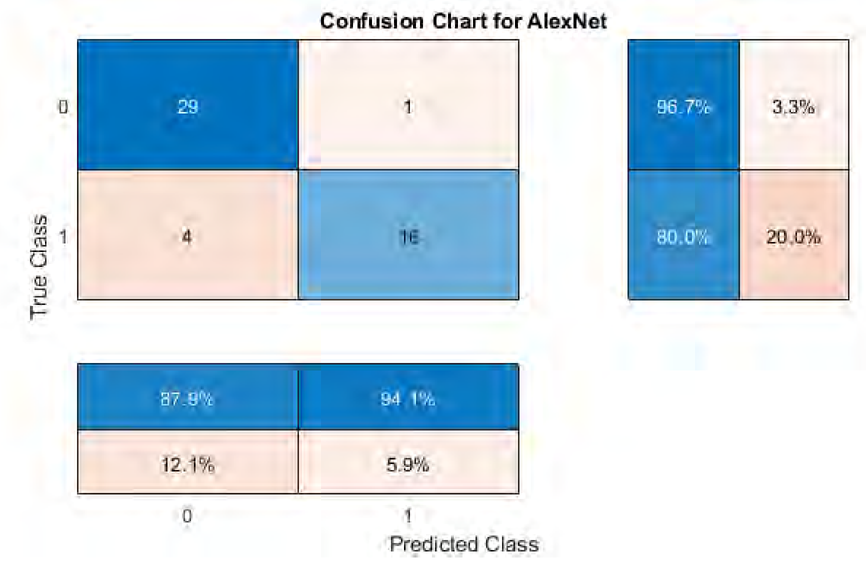


Figure 4.7: Confusion matrix of AlexNet with dataset 2

$$AlexNetAccuracy2 = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \tag{4.12}$$

$$AlexNetAccuracy2 = \frac{29 + 16}{29 + 16 + 4 + 1} \times 100 \tag{4.13}$$

$$AlexNetAccuracy2 = 90\% \tag{4.14}$$

True classes are those samples that we provide to network and predicted classes are those files that network detects correctly [30]. In Fig 4.7, accuracy with 96.7% is true positive (TP) rate which can be calculated by using (4.1), where error is 3.3%. True negative rate can be calculated using (4.2) and accuracy is 80% with error 20%.

False positive rate (FP) can be calculated using (4.3), which computes 87.9% accuracy with error of 12.1%. False negative (FN) can be calculated by (4.4) with 94.1% and error 5.9%.

Confusion matrix of recurrent neural network (RNN) with data set 1 is shown in Fig 4.8. Accuracy of RNN with dataset 1 can be calculated as:

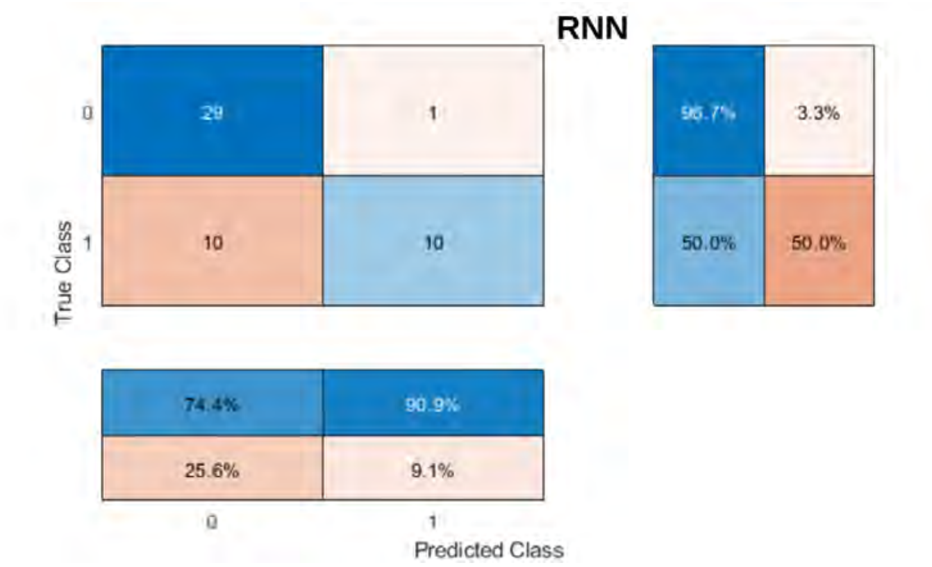


Figure 4.8: Confusion matrix of RNN with dataset 1

$$RNNAccuracy1 = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \tag{4.15}$$

$$RNNAccuracy1 = \frac{29 + 10}{29 + 10 + 1 + 10} \times 100 \tag{4.16}$$

$$RNNAccuracy1 = 78\% \tag{4.17}$$

Confusion matrix of RNN with dataset 2 is shown in Fig 4.9.

Fig 4.9 shows that all 30 no-echo files RNN detect correctly and 11 echo files RNN detect correctly as echo while 9 as no-echo. Accuracy can be calculated

[30] https://www.sciencedirect.com/topics/computer-science/predicted-class

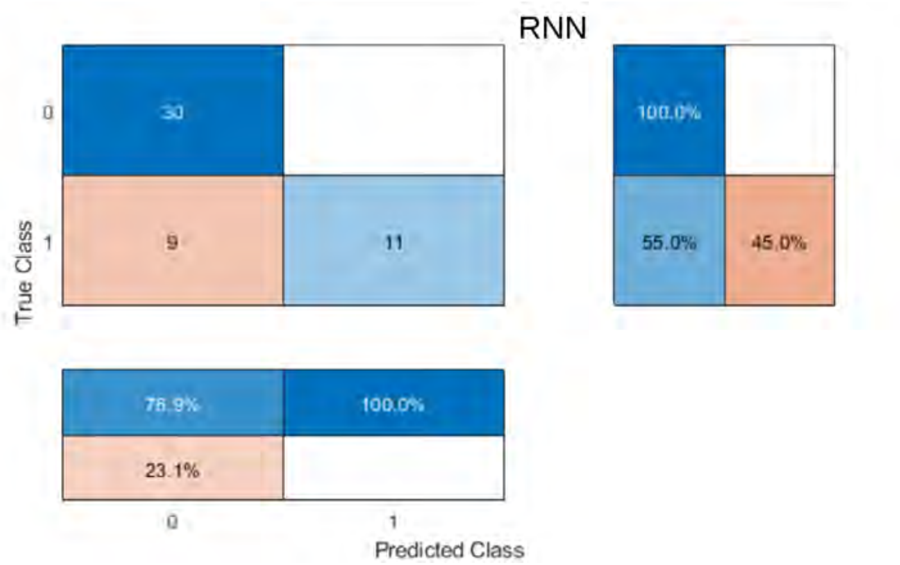


Figure 4.9: Confusion matrix of RNN with dataset 2

as:

$$RNNAccuracy2 = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \tag{4.18}$$

$$RNNAccuracy2 = \frac{30 + 11}{30 + 11 + 9} \times 100 \tag{4.19}$$

$$RNNAccuracy2 = 82\% \tag{4.20}$$

So, RNN achieves 82% accuracy with data set 2. Confusion matrix of DCNN with data set 1 is shown in Fig 4.10.

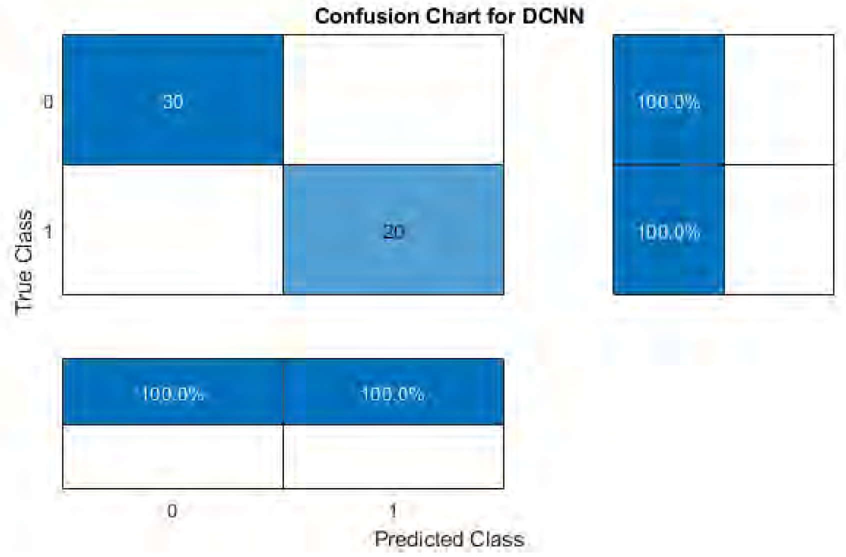


Figure 4.10: Confusion matrix of DCNN with dataset 1

DCNN is tested on 50 samples. It correctly detects all 30 files as no-echo and 20 as echo. Accuracy with data set 1 can be calculated as:

$$DCNNAccuracy1 = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \tag{4.21}$$

$$DCNNAccuracy1 = \frac{30 + 20}{30 + 20} \times 100 \tag{4.22}$$

$$DCNNAccuracy1 = 100\% \tag{4.23}$$

DCNN tested all files correctly,so percentage accuracy is 100%.

Confusion matrix of DCNN with data set 2 is shown in Fig 4.11.

Accuracy can be calculated as:

$$DCNNAccuracy1 = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \tag{4.24}$$

$$DCNNAccuracy1 = \frac{29 + 15}{29 + 15 + 5 + 1} \times 100 \tag{4.25}$$

$$DCNNAccuracy1 = 88\% \tag{4.26}$$

So, DCNN with data set 2 achieves 88% accuracy. Confusion matrix of KNN with data set 1 is shown in Fig 4.12. We tested KNN on 30 no-echo and 20 echo

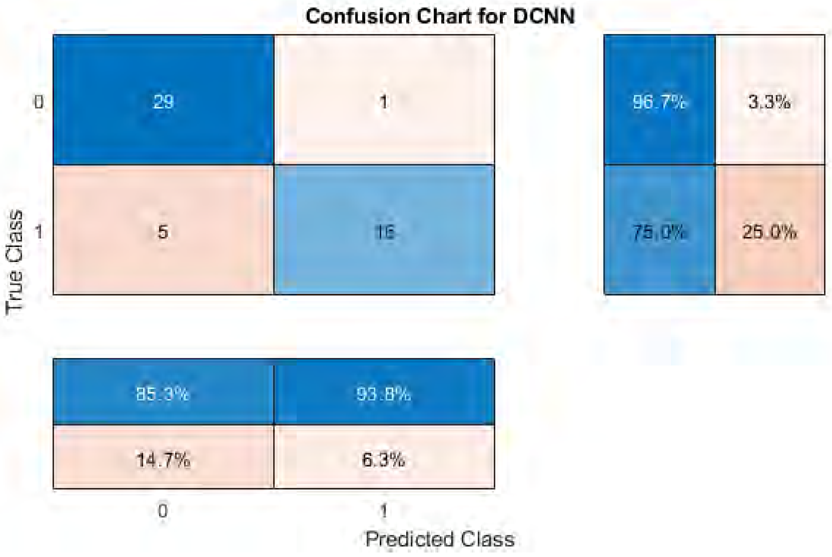


Figure 4.11: Confusion matrix of DCNN with dataset 2

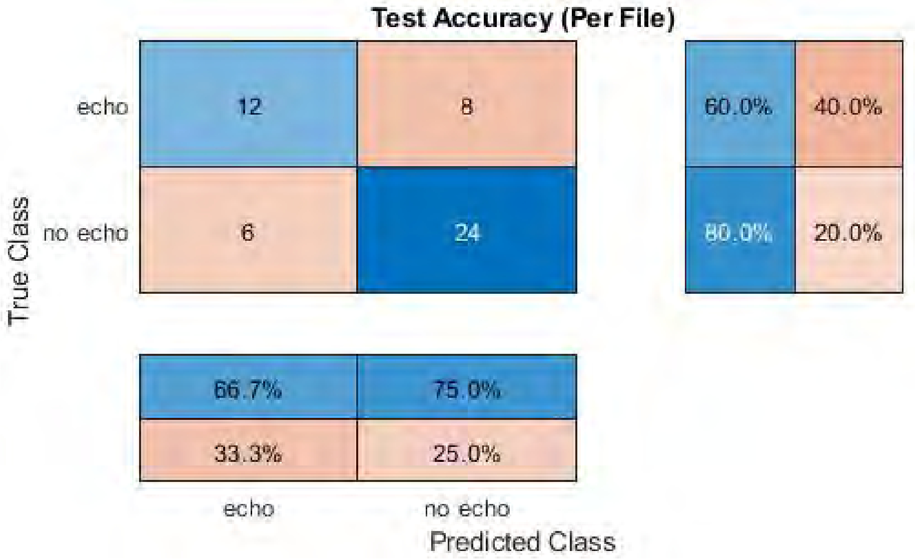


Figure 4.12: Confusion matrix of KNN with dataset 1

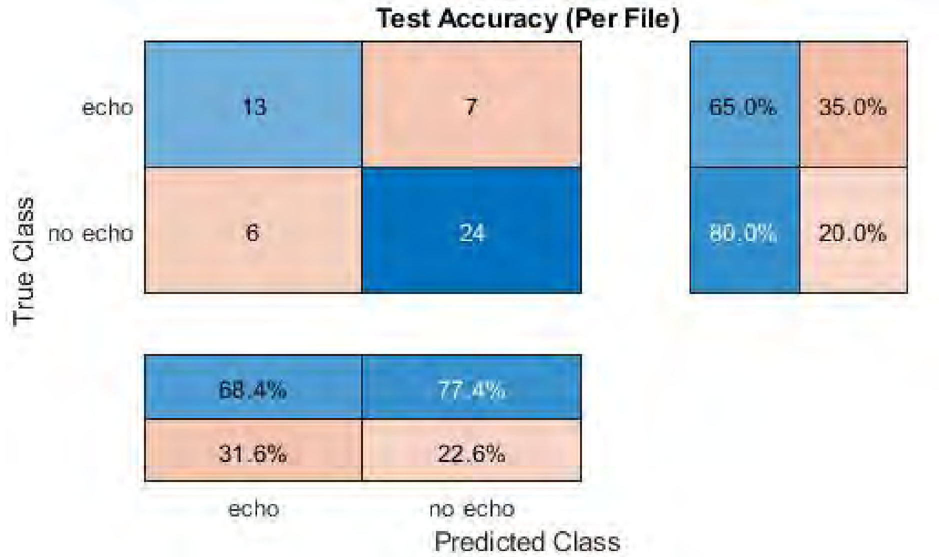
samples. Accuracy with data set 1 can be calculated as:

$$KNNAccuracy1 = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \tag{4.27}$$

$$KNNAccuracy1 = \frac{12 + 24}{12 + 24 + 8 + 6} \times 100 \tag{4.28}$$

$$KNNAccuracy1 = 72\% \tag{4.29}$$

KNN achieves 72% accuracy. We extract total 18 features and some of them are MFCC, pitch, spectral roll off point, spectral flux and spectral slope. Confusion matrix of KNN with data set 2 is shown in Fig 4.13. Accuracy of KNN with data set 2 can be calculated as:



	echo	no echo		
echo	13	7	65.0%	35.0%
no echo	6	24	80.0%	20.0%
	68.4%	77.4%		
	31.6%	22.6%		

Figure 4.13: Confusion matrix of KNN with dataset 2

$$KNNAccuracy2 = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \tag{4.30}$$

$$KNNAccuracy2 = \frac{13 + 24}{13 + 24 + 7 + 6} \times 100 \tag{4.31}$$

$$KNNAccuracy2 = 74\% \tag{4.32}$$

4.4 Comparison Between AlexNet, DCNN, RNN And KNN Results

In this thesis, we have prepared two data sets each with 1000 echo and 1000 no-echo samples with two different RIRs. We have prepared 30 no-echo and 20 echo samples each for two data sets. First RIR comprised of 25 samples delay and second RIR for data set 2 contains 30 samples delay. We have utilized in CNN networks AlexNet, DCNN, RNN and a machine learning algorithm KNN.

AlexNet and DCNN both accept image input of size 227x227x3 and 224x224x3, respectively. In these networks, feature extraction performs automatically after providing images to the network. We have trained these networks on two data sets and also tested on two separate data sets. In AlexNet with data set 1 and 2, we achieved accuracy 98% and 90% respectively. DCNN achieved accuracy with data set 1 and 2 as 100% and 88% respectively.

RNN and KNN take audio inputs and feature extraction performs manually. We have extracted features spectral centroid, spectral flux, pitch, MFCC manually from data set in MATLAB code. We also trained these networks on two data sets and tested on separate two data sets. RNN achieves 78% and 82% with data set 1 and 2 respectively. We have also trained KNN on two data sets. KNN achieves 72% and 74% respectively. AlexNet and DCNN detected echo more accurately as compared to KNN and RNN. AlexNet and DCNN extract features, which are suitable for the classification to detect echo accurately. In KNN and RNN, we extracted spectral features and achieve good accuracy. Comparison of the overall performance of AlexNet, DCNN, RNN and KNN is given in Table 4.2

Table 4.2: Results of KAlexNet, DCNN, RNN and KNN

Datasets	DCNN	AlexNet	RNN	KNN
Dataset 1	100%	98%	78%	72%
Dataset 2	88%	90%	82%	74%

Chapter 5

Conclusion and Future Work

In this thesis, we presented deep learning and machine learning based binary classifiers to detect echo state of acoustic echo canceller. Echo state detection is vital for adaptive normalized least mean-square (NLMS) algorithm. Energy and correlation based echo detection methods are not reliable and causes NLMS algorithm to diverge. We considered transfer learning based AlexNet, DCNN and RNN deep learning networks as binary classifier to detect echo and no-echo stated from the speaker and microphone signals. We also considered machine learning bases KNN network as binary classifier. We prepared two data set for the training of the deep learning and machine learning based classifiers.

We trained AlexNet, DCNN, KNN and RNN on two data sets and tested on separate two data set. AlexNet achieves 98% and 90% with data sets 1 and 2 respectively. DCNN achieves 100% and 88% with data sets 1 and 2 respectively. RNN achieves 78% and 82% accuracy and KNN achieves 72% and 74% accuracy. AlexNet and DCNN detect echo more accurately as compared to RNN and KNN.

5.1 Future Work

We can extend our work by adding more samples to data sets and then train these network. We can also prepared different data sets with different room impulse responses and compare accuracy. Impact of the proposed echo detector on AEC will also be investigated.

Bibliography

[1] Hao Zhang, Ke Tan, and DeLiang Wang. Deep learning for joint acoustic echo and noise cancellation with nonlinear distortions. In *INTERSPEECH*, pages 4255–4259, 2019.

[2] Hao Zhang and D Wang. Deep learning for acoustic echo cancellation in noisy and double-talk scenarios. *Training*, 161(2):322, 2018.

[3] E. Hari Krishna, M. Raghuram, K. Venu Madhav, and K. Ashoka Reddy. Acoustic echo cancellation using a computationally efficient transform domain lms adaptive filter. In *10th International Conference on Information Science, Signal Processing and their Applications (ISSPA 2010)*, pages 409–412, 2010.

[4] FK Becker and HR Rudin. Application of automatic transversal filters to the problem of echo suppression. *The Bell System Technical Journal*, 45(10):1847–1850, 1966.

[5] Rafid A. Sukkar. Echo detection and delay estimation using a pattern recogntion approach and cepstral correlation. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*, volume 4, pages IV–909–IV–912, 2007.

[6] Vinay Kothapally, Yong Xu, Meng Yu, Shi-Xiong Zhang, and Dong Yu. Joint aec and beamforming with double-talk detection using rnn-transformer. *arXiv preprint arXiv:2111.04904*, 2021.

[7] Yingjun Chen and Yongtao Hao. A feature weighted support vector machine and k-nearest neighbor algorithm for stock market indices prediction. *Expert Syst. Appl.*, 80(C):340–355, sep 2017.

[8] Saad Albawi, Tareq Abed Mohammed, and Saad Al-Zawi. Understanding of a convolutional neural network. In *2017 International Conference on Engineering and Technology (ICET)*, pages 1–6, 2017.

[9] Jing Sun, Xibiao Cai, Fuming Sun, and Jianguo Zhang. Scene image classification method based on alex-net model. In *2016 3rd International Conference on Informative and Cybernetics for Computational Social Systems (ICCSS)*, pages 363–367, 2016.

[10] J. Radecki, Z. Zilic, and K. Radecka. Echo cancellation in ip networks. In *The 2002 45th Midwest Symposium on Circuits and Systems, 2002. MWSCAS-2002.*, volume 2, pages II–II, 2002.

[11] J. Benesty, D.R. Morgan, and J.H. Cho. A new class of doubletalk detectors based on cross-correlation. *IEEE Transactions on Speech and Audio Processing*, 8(2):168–172, 2000.

[12] P. Ahgren. Acoustic echo cancellation and doubletalk detection using estimated loudspeaker impulse responses. *IEEE Transactions on Speech and Audio Processing*, 13(6):1231–1237, 2005.

[13] Heiga Zen and Haşim Sak. Unidirectional long short-term memory recurrent neural network with recurrent output layer for low-latency speech synthesis. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4470–4474, 2015.

[14] Yi Zhang, Chengyun Deng, Shiqian Ma, Yongtao Sha, and Hui Song. Deep multi-task network for delay estimation and echo cancellation. *arXiv preprint arXiv:2011.02109*, 2020.

[15] Nils L. Westhausen and Bernd T. Meyer. Acoustic echo cancellation with the dual-signal transformation lstm network. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7138–7142, 2021.

[16] F. Capman, J. Boudy, and P. Lockwood. Acoustic echo cancellation using a fast qr-rls algorithm and multirate schemes. In *1995 International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 969–972 vol.2, 1995.

[17] Constantin Paleologu, Silviu Ciochina, and Jacob Benesty. Variable step-size nlms algorithm for under-modeling acoustic echo cancellation. *IEEE Signal Processing Letters*, 15:5–8, 2008.

[18] Ahmed I. Sulyman and Azzedine Zerguine. Echo cancellation using a variable step-size nlms algorithm. In *2004 12th European Signal Processing Conference*, pages 401–404, 2004.

[19] Mhd Modar Halimeh, Thomas Haubner, Annika Briegleb, Alexander Schmidt, and Walter Kellermann. Combining adaptive filtering and complex-valued deep postfiltering for acoustic echo cancellation. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 121–125, 2021.

[20] A. A. M. Muzahid, K. M. R. Ingrid, S. I. M. M. Raton Mondol, and Y. Zhou. Advanced double-talk detection algorithm based on joint signal energy and cross-correlation estimation. In *2016 8th IEEE International Conference on Communication Software and Networks (ICCSN)*, pages 303–306, 2016.

[21] Sheng Zhang, Jiashu Zhang, and Hing Cheung So. Low-complexity decorrelation nlms algorithms: Performance analysis and aec application. *IEEE Transactions on Signal Processing*, 68:6621–6632, 2020.

[22] I. Kammoun and M. Jaidane. Exact performances analysis of a selective coefficient adaptive algorithm in acoustic echo cancellation. In *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221)*, volume 5, pages 3245–3248 vol.5, 2001.

[23] Shihab Jimaa. Convergence evaluation of a random step-size nlms adaptive algorithm in system identification and channel equalization. In Lino Garcia, editor, *Adaptive Filtering*, chapter 1. IntechOpen, Rijeka, 2011.

[24] Xuedan Du, Yinghao Cai, Shuo Wang, and Leijie Zhang. Overview of deep learning. In *2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, pages 159–164, 2016.

[25] Susmita Ray. A quick review of machine learning algorithms. In *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*, pages 35–39, 2019.

[26] Yang Xin, Lingshuang Kong, Zhi Liu, Yuling Chen, Yanmiao Li, Hongliang Zhu, Mingcheng Gao, Haixia Hou, and Chunhua Wang. Machine learning and deep learning methods for cybersecurity. *IEEE Access*, 6:35365–35381, 2018.

[27] Yogesh Kumar, Komalpreet Kaur, and Gurpreet Singh. Machine learning aspects and its applications towards different research areas. In *2020 International Conference on Computation, Automation and Knowledge Management (ICCAKM)*, pages 150–156, 2020.

[28] Amir Ivry, Israel Cohen, and Baruch Berdugo. Nonlinear acoustic echo cancellation with deep learning. *arXiv preprint arXiv:2106.13754*, 2021.

[29] Amin Fazel, Mostafa El-Khamy, and Jungwon Lee. Cad-aec: Context-aware deep acoustic echo cancellation. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6919–6923, 2020.

[30] Daniel López-Sánchez, Angélica González Arrieta, and Juan M Corchado. Deep neural networks and transfer learning applied to multimedia web mining. In *International Symposium on Distributed Computing and Artificial Intelligence*, pages 124–131. Springer, 2017.

[31] Javad Abbasi Aghamaleki and Sina Moayed Baharlou. Transfer learning approach for classification and noise reduction on noisy web data. *Expert Systems with Applications*, 105:221–232, 2018.

[32] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109(1):43–76, 2021.

[33] Raju Pal and Mukesh Saraswat. Enhanced bag of features using alexnet and improved biogeography-based optimization for histopathological image analysis. In *2018 Eleventh International Conference on Contemporary Computing (IC3)*, pages 1–6, 2018.

[34] Xingcheng Luo, Ruihan Shen, Jian Hu, Jianhua Deng, Linji Hu, and Qing Guan. A deep convolution neural network model for vehicle recognition

and face recognition. *Procedia Computer Science*, 107:715–720, 2017. Advances in Information and Communication Technology: Proceedings of 7th International Congress of Information and Communication Technology (ICICT2017).

[35] Alex Sherstinsky. Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network. *Physica D: Nonlinear Phenomena*, 404:132306, Mar 2020.

[36] Tong Liu, Tailin Wu, Meiling Wang, Mengyin Fu, Jiapeng Kang, and Haoyuan Zhang. Recurrent neural networks based on lstm for predicting geomagnetic field. In *2018 IEEE International Conference on Aerospace Electronics and Remote Sensing Technology (ICARES)*, pages 1–5, 2018.

[37] Haşim Sak, Andrew Senior, and Françoise Beaufays. Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition. *arXiv preprint arXiv:1402.1128*, 2014.

[38] Jun-Li Xu, Siewert Hugelier, Hongyan Zhu, and Aoife A. Gowen. Deep learning for classification of time series spectral images using combined multi-temporal and spectral features. *Analytica Chimica Acta*, 1143:9–20, 2021.

[39] Yu Su, Ke Zhang, Jingyu Wang, and Kurosh Madani. Environment sound classification using a two-stream cnn based on decision-level fusion. *Sensors*, 19(7):1733, 2019.

[40] Gil Keren and Björn Schuller. Convolutional rnn: An enhanced model for extracting features from sequential data. In *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 3412–3419, 2016.

[41] Md Zahangir Alom, Tarek M Taha, Christopher Yakopcic, Stefan Westberg, Paheding Sidike, Mst Shamima Nasrin, Brian C Van Esesn, Abdul A S Awwal, and Vijayan K Asari. The history began from alexnet: A comprehensive survey on deep learning approaches. *arXiv preprint arXiv:1803.01164*, 2018.

[42] Melissa N. Stolar, Margaret Lech, Robert S. Bolia, and Michael Skinner. Real time speech emotion recognition using rgb image classification and

transfer learning. In *2017 11th International Conference on Signal Processing and Communication Systems (ICSPCS)*, pages 1–8, 2017.

[43] Jing Sun, Xibiao Cai, Fuming Sun, and Jianguo Zhang. Scene image classification method based on alex-net model. In *2016 3rd International Conference on Informative and Cybernetics for Computational Social Systems (ICCSS)*, pages 363–367, 2016.

[44] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

[45] Nishmia Ziafat, Hafiz Farooq Ahmad, Iram Fatima, Muhammad Zia, Abdulaziz Alhumam, and Kashif Rajpoot. Correct pronunciation detection of the arabic alphabet using deep learning. *Applied Sciences*, 11(6), 2021.

[46] Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. Recurrent neural network regularization. *arXiv preprint arXiv:1409.2329*, 2014.

[47] Kashvi Taunk, Sanjukta De, Srishti Verma, and Aleena Swetapadma. A brief review of nearest neighbor algorithm for learning and classification. In *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*, pages 1255–1260, 2019.

[48] B. P. Salmon, W. Kleynhans, C. P. Schwegmann, and J. C. Olivier. Proper comparison among methods using a confusion matrix. In *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 3057–3060, 2015.