Speaker identification on multimodal environment using CNN

A thesis submitted in the partial fulfilment of the requirement for the

Degree of Master of Computer Science and Engineering

of

Jadavpur University

By

Tapas Chakraborty

Registration Number: **86947** of 2003-2004

Examination Roll Number: **M4CSE19013**

Under the guidance of

Dr. Nibaran Das

Department of Computer Science and Engineering

Jadavpur University, Kolkata-700032

India

2019

FACULTY OF ENGINEERING AND TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

JADAVPUR UNIVERSITY

CERTIFICATE OF RECOMMENDATION

This is to certify that the thesis entitled “Speaker identification on multimodal environment using CNN” has been satisfactorily completed by Tapas Chakraborty (University Registration No.: 86947 of 2003-04, Examination Roll No.:M4CSE19013). It is a bonafide piece of work carried out under my guidance and supervision and be accepted in partial fulfilment for degree of Master of Computer Science and Engineering, Department of Computer Science and Engineering, in the Faculty of Engineering and Technology, Jadavpur University.

____

(**Dr. Nibaran Das**)
Associate Professor
Department of Computer Science and Engineering

Countersigned:

____

Prof. Mahantapas Kundu

Head
Department of computer Sc & Engg
Jadavpur University, Kol-700032

____

Prof. Chiranjib Bhattacharjee

Dean
Faculty of Engineering and Technology
Jadavpur University, Kol-700032

DECLARATION OF ORIGINALITYAND COMPLIANCE OF ACADEMIC ETHICS

I hereby declare that this thesis entitled “Speaker identification on multimodal environment using CNN” contains literature survey and original research work by the undersigned candidate, as part of his Degree of Master of Computer Science and Engineering.

All information in this document has been obtained and presented in accordance with academic rules and ethical conduct.

I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

Name: Tapas Chakraborty
University Registration No.: 86947 of 2003-2004
Examination Roll No.: M4CSE19013

Thesis Title: Speaker identification on multimodal environment using CNN

Signature:

Date:

JADAVPUR UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE OF APPROVAL

This is to certify that the thesis entitled “Speaker identification on multimodal environment using CNN” is a bonafide record of work carried out by Tapas Chakraborty in partial fulfilment of the requirements for the award of the degree of Master of Computer Science and Engineering in the Department of Computer Science and Engineering, Jadavpur University during the period of July 2017 to May 2019. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed, or conclusion drawn there in but approve the thesis only for the purpose for which it has been submitted.

Signature of Examiner
Date:

Signature of Supervisor
Date:

ACKNOWLEGEMENT

First and foremost, I would like to start by thanking God Almighty for showering me with the strength, knowledge and potential to embark on this wonderful journey and to persevere and complete the embodied research work satisfactorily.

I am pleased to express my deepest gratitude to my thesis guide, **Dr. Nibaran Das**, Department of Computer Science and Engineering, Jadavpur University, Kolkata for his invaluable guidance, constant encouragement and inspiration during the period of my dissertation. I am extremely thankful to **Prof. Mahantapas Kundu,** for his continuous supervision and guidance.

I am highly indebted to **Jadavpur University** for providing me the opportunity and the required infrastructure to carry on my thesis. I am also grateful to the **Center for Microprocessor Applications for Training Education and Research** for giving me the proper laboratory facilities as and when required. I am thankful to all the teaching and non-teaching staff whose helping hands have smoothed my journey through the period of my research.

Last but not the least; I would like to thank my family members, classmates, seniors and friends for giving me constant encouragement and mental support throughout my work.

Tapas Chakraborty

University Registration No.: 86947 of 2003-2004
Examination Roll No.: M4CSE19013
Master of Computer Science and Engineering
Department of Computer Science and Engineering

Jadavpur University

TABLE OF CONTENTS

CHAPTER ONE

INTRODUCTION

1.1 SPEAKER RECOGNITION

Speaker Recognition (SR) is a branch of biometric recognition where the speaker specific psycho-physiological characteristics of speech waveform are analyzed to uniquely recognize individual speaker using speaker's voice signal [1, 2] and the system that is designed for this purpose is known as Automatic Speaker Recognition (ASR) system. These characteristics include both voice tract characteristics (spectral features) and voice source characteristics (supra-segmental features) of speech. Feature(s) is/are the attribute(s) (most of the time numerical) by which individual entities (speakers) are identified uniquely. A feature vector is usually an array of numbers. The process (or steps) of computing feature vector(s) is known as feature extraction. However, SR is an example of a typical Pattern Recognition (PR) problem but with the advancement of computer technology, other modern approaches like Machine Learning (ML), Artificial Neural Network (ANN), [15] Deep Learning (DL) are giving immense momentum to the SR research and have become the recent trends due to impressive improvement in recognition rate and computational time. Any conventional PR method consists of two necessary steps, Feature Extraction or Selection and Modelling or Classification [3, 4]. In SR speaker-specific features are extracted first from each of the voice signals available in the database, and then a model is built for each class (for SR each class represents a speaker) in the database. This process is known as Training/Enrolment. When the voice sample of the unknown speaker is available for SR, the same set of features are extracted in a similar manner. Then this set of feature vectors of the unknown speaker (test data) is compared with the model of known voice samples (for identification), and a statistical measure (score) for the voice samples of the unknown speaker is computed concerning all the known speaker's models. The maximum score (anyone measure or combination of measures) identify (classify) the unknown speaker as the speaker corresponding to that model. This process is known as Testing. In this step, we use all the speaker models for classification. For example, in GMM based SR using MFCC feature, we first compute MFCC feature vectors (13 MFC Coefficients) from the speech signals of all the speakers for training the GMMs of all speakers and next from testing speech signal the MFCCs are extracted similarly to compute scores (or similarity measures) with respect to every enrolled speaker (or trained speaker model). Each speaker provides a single score. The maximum score provides the classified speaker. Indeed, SR using GMM was introduced before 1992, and later many modifications are done in this approach. SR using Super Vector is an example of modification of GMM where a super vector is formed by concatenating the means of GMM. Here each speaker is represented by a high dimensional super vector.

1.2 CLASSIFICATION OF SR

The SR is classified into three groups, namely –

(a) Speaker Identification (SI) and Speaker Verification (SV),

(b) Text-dependent and Text-independent,

(c) Closed-set and Open-set.

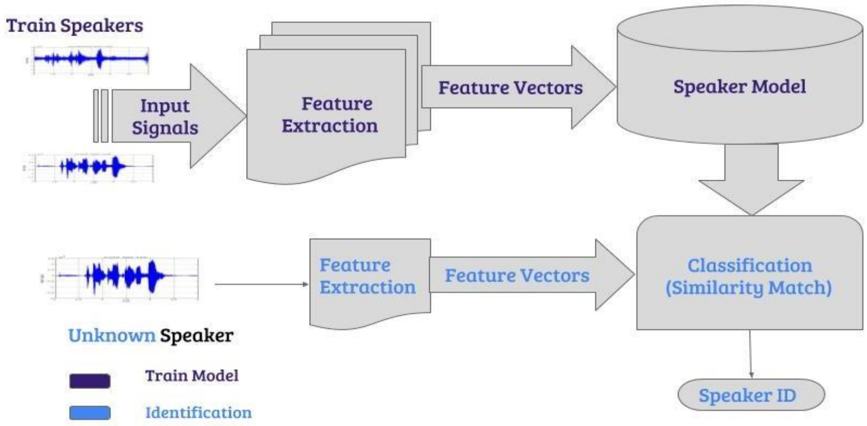

Figure 1: Block diagram of Speaker Identification

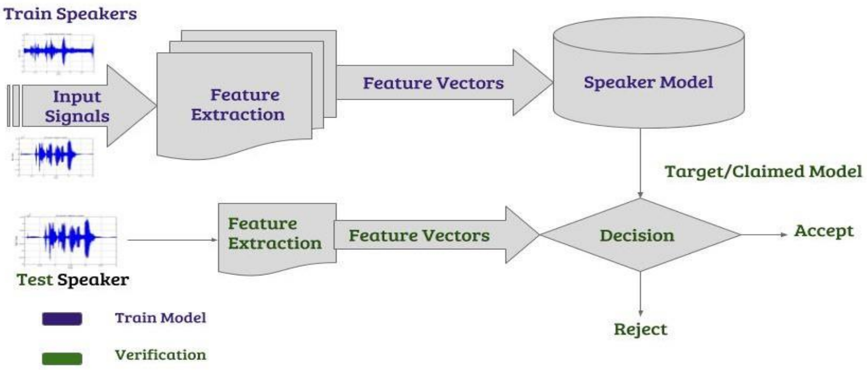

Figure 2: Block diagram of Speaker Verification

SI is the type of SR where we are required to determine the identity of an unknown speaker, i.e., which speaker among the enrolled speakers is speaking and in contrast to SI, SV is the task of authenticating an unknown speaker's identity, i.e., we are required to verify whether the claim of an unknown speaker will be accepted or rejected by the ASR system. Among these two types of SR, SV is the most popular one because of its application in access control and security. In text-dependent SR, the text (the content of speech) is mixed (or same) for training and testing speech data whereas in text-independent SR the text of training and the testing speaker is not fixed (or same). Finally, in closed-set SR, it is known that the unknown speaker is one of the enrolled speakers, but we do not know which one among them whereas in open-set SR the unknown speaker may or may not present among the enrolled speakers. It is known that among these types of SR open-set text-independent SI (OSTI-SI) is the most challenging class of SR. In OSTI-SI the score of unknown speakers is compared with the scores of all the enrolled speakers using a decision function to determine - 1) whether the unknown speaker is one of the enrolled speakers, 2) if yes, which one of the enrolled speakers it belongs to. Task 1) and 2) are accomplished simultaneously by adding a complementary model which is built by all speakers' speech data except the known speakers' data. This also means that an OSTI-SI SR is solved if we have a comparatively large number of speakers' data outside the known speakers. The nature of OSTI-SI SR problem is quite different from SV. Indeed, SV is always an open-set SR problem [5].

1.3 SR CHALLENGES

The SR is expanding day by day with a broad range of applications. Deploying an SR system of high accuracy for real-time applications is still challenging. The performance of the SR system degrades considerably due to the mismatch among the various factors. The factors that play a very crucial role in high-performance SR system are discussed as follows [2]:

1) **Noisy Environment:** The acquired speech waveforms for designing an SR system may contain various types of noise namely convolutional noise, additive noise [6], reverberations (speech containing echoes) as noise, random noise, impulse noise and so on [7, 8, 9].

2) **Environmental Mismatch:** It is very difficult to accumulate data in the same environment for training and testing. It is observed that the recognition accuracy is highly dependent upon the mismatch between training and testing environment.

3) **Recording Device Mismatch or Channel Mismatch:** The recognition accuracy of SR degrades drastically for recording device mismatch between training and testing data, which is observed in later sections of the presented paper.

4) **Language of Utterance Mismatch:** The recognition accuracy of SR also depends on language mismatch. Mismatch between training and testing data due to language does not affect as greatly as device and environment mismatch but the language has significant influence over the recognition accuracy.

5) **Short Utterance:** The acquisition of speech waveforms with enough duration for training and testing to design an SR system is very difficult. So, sometimes we are bound to design a system with limited data (3 ~ 6 seconds or less). The length of text of the speech or duration of utterance plays an important role in SR. Short utterance degrades the recognition accuracy considerably. Mandasari et al. [10] examine the effect of short utterance and propose a calibration strategy to model the calibration parameters using Quality Measure Functions

(QMFs) for SR to improve the recognition accuracy [11] .

6) **Long Utterance:** If the available data for design is very large then the data must be reduced using data reduction techniques which may leads to loss of significant data (information). The accumulation and annotation of large amount of data is also very difficult.

CHAPTER TWO

LITERATURE REVIEWS

Before the invention of neural networks, traditional classifiers like Gaussian Mixture Models (GMM) [2], Support Vector Machine (SVM) [16], Hidden Markov Models (HMM) [22] were used as classifiers. Those classifiers require prior knowledge and human effort in feature design. Mel Frequency Cepstral Coefficient (MFCC), Gammatone Frequency Cepstral Coefficients (GFCC) [2] features, extracted from audio signals, were used as input to traditional classifiers.

2.1 MFCC EXTRACTION

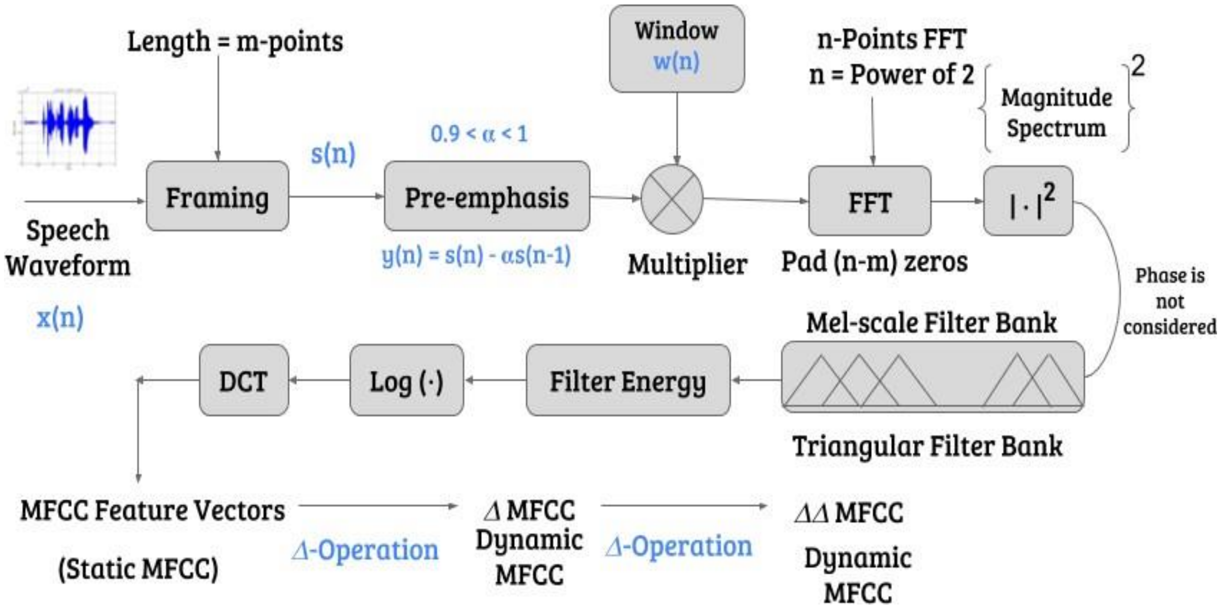

Figure 3: Complete Block Diagram of MFCC Computation

2.2 GMM BASED SR APPROACH

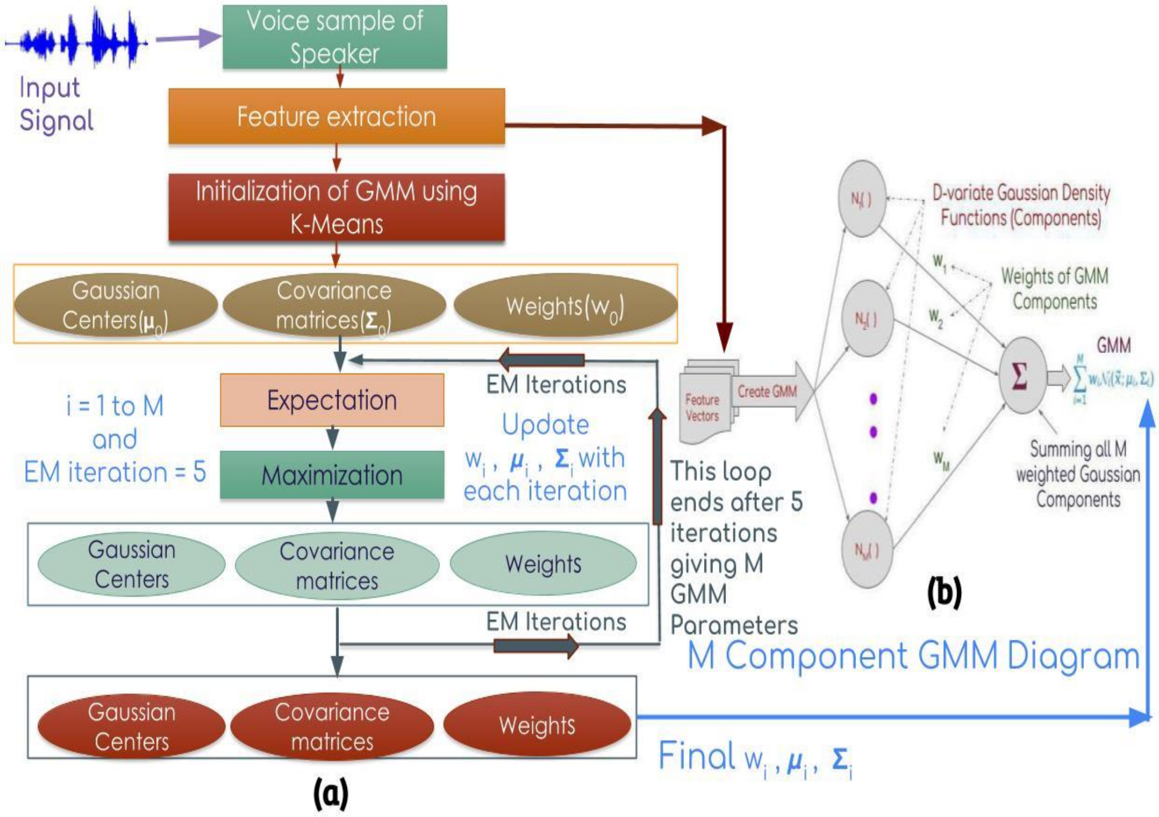
Input Signal
Voice sample of Speaker
Feature extraction
Initialization of GMM using K-Means
Gaussian Centers(μ_0)
Covariance matrices(Σ_0)
Weights(w_0)
i = 1 to M and EM iteration = 5
Expectation
Maximization
EM Iterations
Update w_i, μ_i, Σ_i with each iteration
Gaussian Centers
Covariance matrices
Weights
EM Iterations
Gaussian Centers
Covariance matrices
Weights
(a)
Final w_i, μ_i, Σ_i
This loop ends after 5 iterations giving M GMM Parameters
Feature Vectors
Create GMM
$N_1(\,)$
$N_2(\,)$
$N_M(\,)$
D-variate Gaussian Density Functions (Components)
w_1
w_2
w_M
Weights of GMM Components
GMM
$\sum_{i=1}^{M} w_i N_i(\vec{x}; \mu_i, \Sigma_i)$
Summing all M weighted Gaussian Components
(b)
M Component GMM Diagram

Figure 4: Complete Block Diagram of GMM based SR

2.3 SR USING RBM AND I-VECTOR

In recent years, supervector and i-Vector became a state-of-the-art technique, and RBM has been used for Speaker Recognition using i-Vector [5].

RBM has the power to extract Good features. Which mean RBM is used as dimension reduction stage. GMM supervectors are usually provided as input to RBM. Output of RBM will be reduced set of lower dimensional vectors. RBM tries to learn the entire session and speaker variability among these background super vectors. RBM trained in this way will then be used to transform unseen supervectors to lower dimensional vectors.

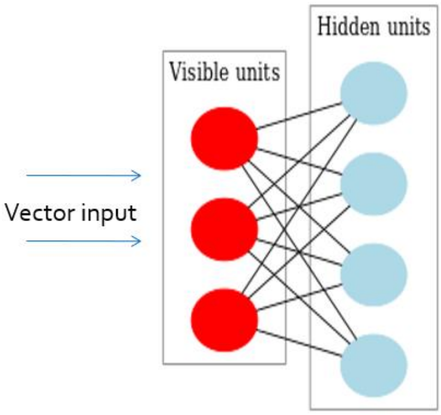

Figure 5: Block diagram of RBM which tries to learn session and speaker variability from input supervectors

With the advance of Neural Networks, Long Short-Term Memory (LSTM), Convolutional neural network (CNN) were applied in Speaker Recognition domain. Convolutional Neural Network [21] does not require any human effort in feature extraction as it automatically extracts features from the data.

CHAPTER THREE

PROPOSED WORK

3.1 CNN OVERVIEW

In this work, CNN model is used to extract useful features of the audio signal so that voice identification becomes more accurate. Initially, audio data was pre-processed to remove noise and silent frames. Then Log spectrograms are generated from those processed audio signals. Python signal analysis library Librosa [23] is used for this Log spectrogram generation task. Log spectrograms are then used as input to CNN. Initially, CNN has been trained using audio files of known speakers. Then it is tested with audio files of unknown speakers.

Below is the block diagram of the overall process.

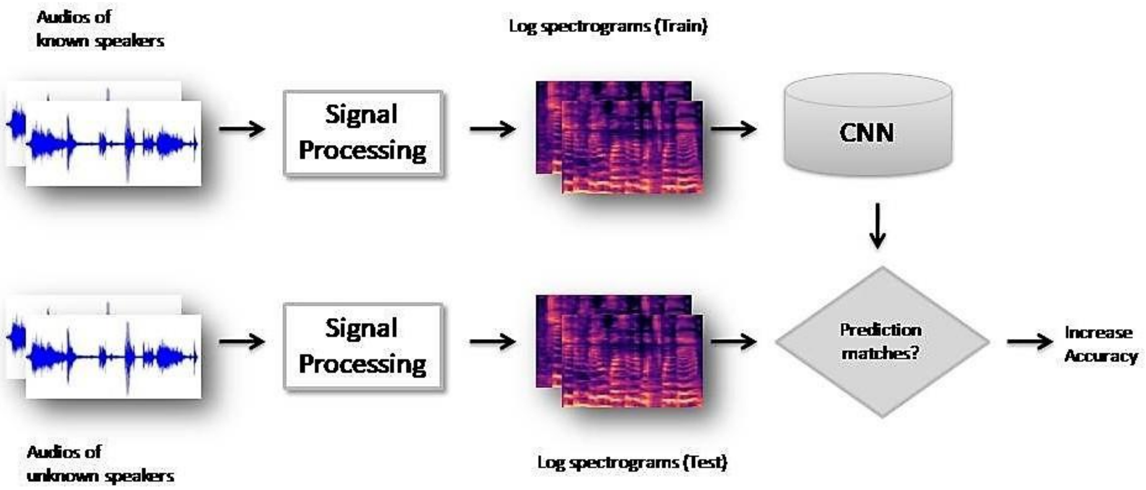

Figure 6: A block diagram of CNN based identification process

CNN is implemented on IITG-MV dataset [20]. Indian Institute of Technology Guwahati Multi-variability Speaker Recognition (IITG-MV SR) database has five different phases. In this work, only first phase data is used. It contains speech data in

‘.wav’ format containing hundred speakers. Each speaker spoke in different style (reading and conversation); in multiple languages (one in English and another language of their choice, i.e. favorite language); in various sessions and environments. Audio was recorded using five different devices (tablets, smartphones etc.) which makes this dataset exceptional.

Both device-matched and device-mismatched situation is tested, and CNN performed better than traditional GMM based approach in both cases.

3.2 AUDIO PRE- PROCESSING

First step of our experiment is to decide what should be the input for CNN. Processed data like MFCC should not be used as input to CNN, as CNN needs to extract features by itself. So raw audio signal should be used as input. However, Dieleman et al. [17] showed that CNN performs much better when Log Spectrogram is used as input rather than raw audio signals. Audio signals are then pre-processed to remove noise and silent parts.

3.3 PRE-EMPHASIS

The audio speech signal is passed through a high pass filter (HPF) to rise the amplitude of higher frequencies. It also removes low-frequency noises. If S (n) is representing the audio signal, pre-emphasis can be done using the below equation [2]

$$S(n) = S(n) - \alpha * S(n-1) \qquad (1)$$

Here α is a parameter whose value is experimentally chosen as 0.97.

3.4 SILENCE FRAME REMOVAL

To remove silence parts of audio signal, Short-Time Signal Analysis (STSA) is performed. The output of HPF, i.e. pre-emphasized signal, is first divided into several time frames of short duration (window of 20 ms with 10 ms overlap). Then energy of those frames is compared with the average frame energy of the audio signal and silence frames were identified. [2]

$$E_{avg}(\mathbf{x}) = \Sigma_i\left(\frac{|A(\mathbf{x})^2|}{N}\right) \qquad (2)$$

If $E_i > k$ Eavg for a * specific frame, then that frame is considered as voiced or active frame otherwise considered as silence frame. Here, k is a parameter whose value was experimentally chosen as 0.2.

3.5 LOG SPECTROGRAM

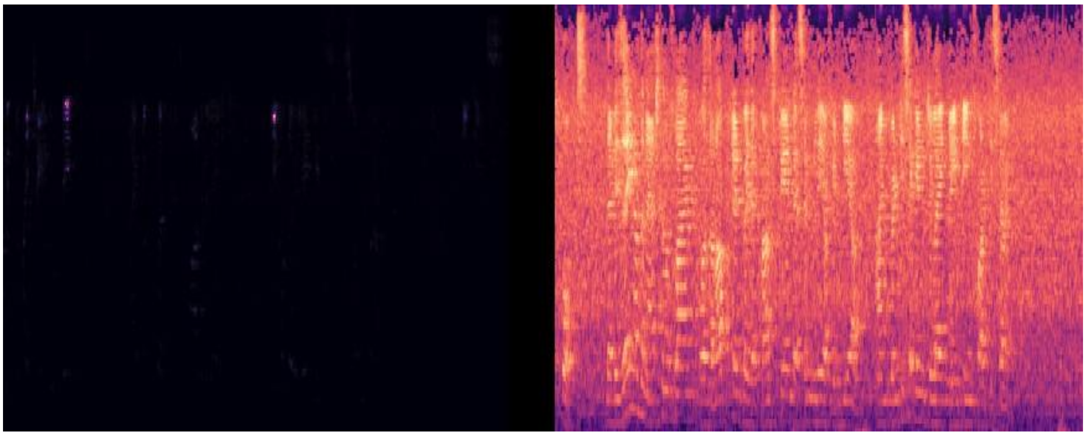

Figure 7: Diagram of Mel-Spectrogram (left) and

Log-spectrogram(right) generated from audio data

Mel-Spectrograms(left) are then generated on pre-processed audio data. Mel-Spectrogram clearly indicates that samples from the dataset are not substantial enough to be fed into CNN network without any processing. That is the reason why Log-spectrogram (right) is used to overcome this problem. Log Spectrogram indicates that it can efficiently extract the characteristics of each speaker from audio signals. If s is the audio signal, Log Spectrogram is computed using below formula.

$$L(s) = \log\left(1 + \beta * s\right) \qquad (3)$$

Parameter β was experimentally chosen as 10000.

3.6 CNN ARCHITECTURE

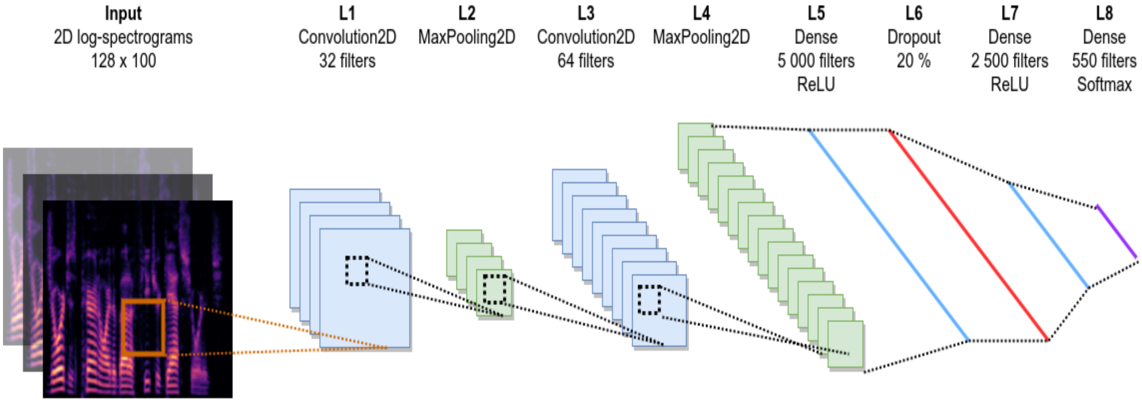

Figure 8: Block diagram of CNN Architecture

Convolutional neural network (CNN) is a class of deep neural networks which uses a variation of multilayer perceptron. Due to their shared-weights architecture and translation invariant characteristics, CNN is also known as shift - space invariant neural networks. CNN requires less preprocessing compared to other algorithms. This clearly indicates that CNN has the capability to learn key features automatically. Figure-3 shows the architecture of the custom CNN model that we have used in this paper.

Apart from the custom CNN model, standard CNN model (here Resnet) is also implemented. Resnet gave similar accuracy. Custom CNN model performs faster than Resnet, hence we choose custom CNN model for this experiment.

Parameter tuning is a vital step for any deep learning experiments. Several methods mentioned by Jiuxiang Gu et al. [19] has been taken into consideration. Kernel of size 13 gave the best performance, padding is half of kernel size and stride is set to 1. Max pooling is used for pooling. Dropout layer is also used in this experiment to prevent overfitting with a rate of n = 0.5

CHAPTER FOUR

EXPERIMENTAL METHODS

4.1 DATA PREPARATION

IITG-MV database is the source database for this experiment, which has a hundred speakers. Each speaker has two audio files, one spoken in conversation style and the other one spoken in reading style. CNN requires balanced training data so that each class has an equal influence on overall loss calculation. However, IITG-MV database [20] has unbalanced data. Some speaker has extremely short audio (around one-tenth of other speakers). Lowest duration is noted, and signals of only that duration are taken from all speakers, rest were ignored.

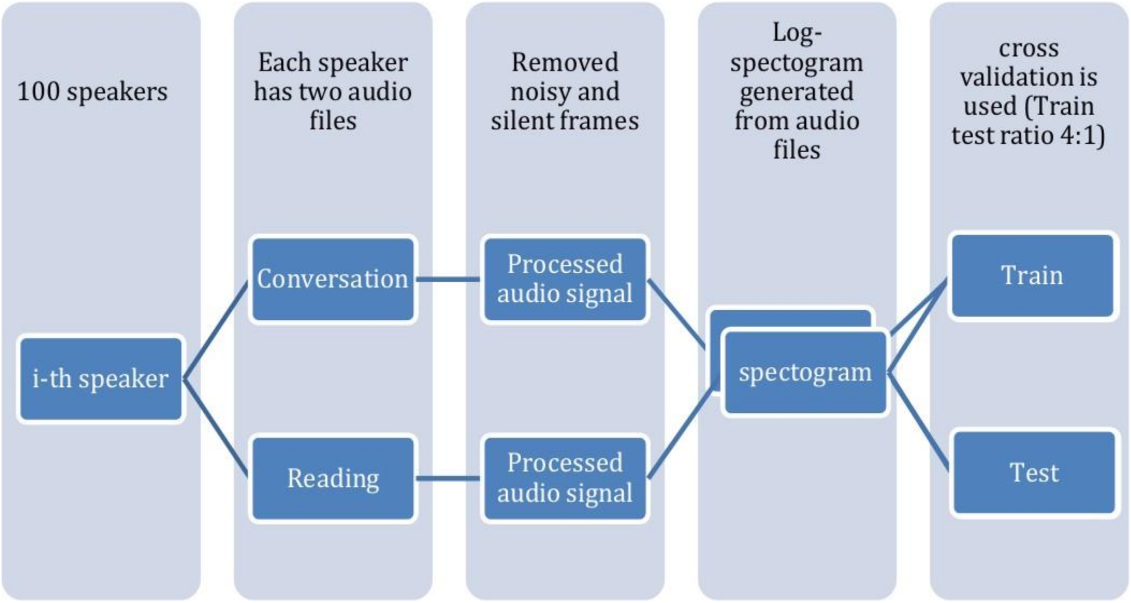
100 speakers
Each speaker has two audio files
Removed noisy and silent frames
Log-spectogram generated from audio files
cross validation is used (Train test ratio 4:1)
i-th speaker
Conversation
Reading
Processed audio signal
Processed audio signal
spectogram
Train
Test

Figure 9: A block diagram of data preparation

In earlier GMM based experiment, conversation style data was used as training while reading style data was used for testing. However, that approach cannot be followed in CNN, as conversation style data is significantly less compared to corresponding reading style data for some speakers. These two issues make the situation challenging as deep learning-based approaches always require a considerable amount of data for better performance. So, we have mixed conversation and reading style data, and then performed cross-validation on the entire dataset. Figure 4 shows detail steps and flow of this experiment.

4.2 CNN MODEL TRAINING

CNN is implemented using Pytorch. During training, data was picked randomly, and model was trained. Training data was of the form (X_i, Y_i) where X_i is input data for i_{th} speaker of shape 128*100*3 and Y_i is input label for i_{th} speaker. The goal of the training is to minimize overall training loss relating to all speakers.

4.3 IDENTIFICATION USING CNN

Let there are n speakers $S = \{1, 2, 3, ..., n\}$

The output layer of CNN has n nodes, one indicating each speaker. When a test speaker data is given into the CNN model, a vector having n scores will be returned as output. i^{th} scores indicate the probability of that unknown speaker to become to i^{th} speaker. The maximum score is considered in this case. Decision rule for speaker identification is given below

$$\hat{S} = \underset{k \in S}{\operatorname{argmax}}(p(\mathbf{x}_i)) \tag{4}$$

Here $\hat{S}$ is the identified speaker and i^{th} speaker's score is given by $p(\mathbf{x}_i)$, the formula given below. The identified speaker $\hat{S}$ has the maximum score.

$$p(\mathbf{x}_i) = \frac{e_i^x}{\sum_1^n e_k^x} \tag{5}$$

4.4 PERFORMANCE MEASURE

The accuracy is measured by the percentage of correct identification, the equation given below:

$$Accuracy\ (\%) = \Sigma_i \left(\frac{Number\ of\ speakers\ correctly\ classified}{Total\ number\ of\ Speakers}\right) * 100 \qquad (6)$$

Performance of this system can be measured by the confusion matrix. The confusion matrix is nothing but a table with two dimensions “actual,” and “predicted,” and similar sets of “classes” in both dimensions.

Two popular performance metrics, precision, and recall have also been used to measure the performance of this system. Precision is the fraction of events where speaker i was correctly identified out of all instances where the system declared. Conversely, recall is the fraction of events where speaker i was correctly identified out of all of the cases where the actual scenario is i.

CHAPTER FIVE

RESULTS AND ANALYSIS

5.1 DEVICE DEPENDENT CASE

Training and testing are done on the audio of same device. IITG-MV database has data for five different types of devices, DVR (D01), Headset (H01) Tablet PC (T01), Nokia 5130c (M01) and Sony EricssonW350i (M02) So five accuracy figures will be there, each one indicating each device.

Table 1 SI accuracy for Device dependent scenario for IITG-MV SR

Device	Num Speakers	GMM Accuracy	CNN Accuracy	Resnet Accuracy	CNN Precision	CNN Recall
D01	100	96	97	97	0.9813	0.9780
H01	100	93	98	97	0.9888	0.9860
T01	100	91	97	96	0.9792	0.9760
M01	100	95	95	95	0.9583	0.9555
M02	100	90	92	92	0.9484	0.9380

5.2 DEVICE INDEPENDENT CASE

In this case, audio data captured by all these five devices are mixed. The same experiment has been performed again on this combined dataset and accuracy has been noted. Maximum accuracy reported earlier was 64. CNN accuracy is significantly high compared with earlier reported accuracy [24].

Table 2 SI accuracy for Device independent scenario for IITG-MV SR

Device	Num Speakers	GMM Accuracy	CNN Accuracy	Resnet Accuracy	CNN Precision	CNN Recall
All	100	64	90	90	0.9187	0.9135

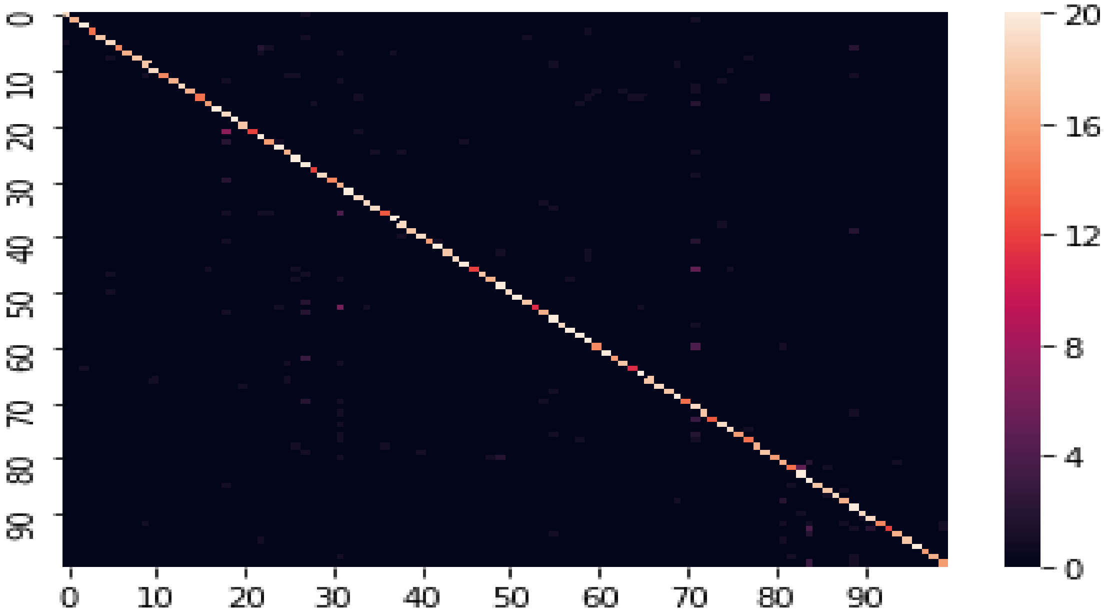

Figure 10: Confusion Matrix for device independent case

5.3 SESSION MISMATCH CASE

Voice of each speaker was recorded in two sessions. In this experiment, training has been performed using ‘session 1’ while validation and testing have been performed using ‘session 2’ data. IITG-MV database has data for five different devices. That is why we get five accuracy figures, one for each device.

Table 1 SI accuracy for session mismatch scenario for IITG-MV SR

Device	Num Speakers	Train	Test	CNN Accuracy	CNN Precision	CNN Recall
D01	100	Session 1	Session 2	63	0.6484	0.6484
H01	100	Session 1	Session 2	64	0.6588	0.6537
T01	100	Session 1	Session 2	63	0.6492	0.6484
M01	100	Session 1	Session 2	66	0.6782	0.6723
M02	100	Session 1	Session 2	65	0.6637	0.6594

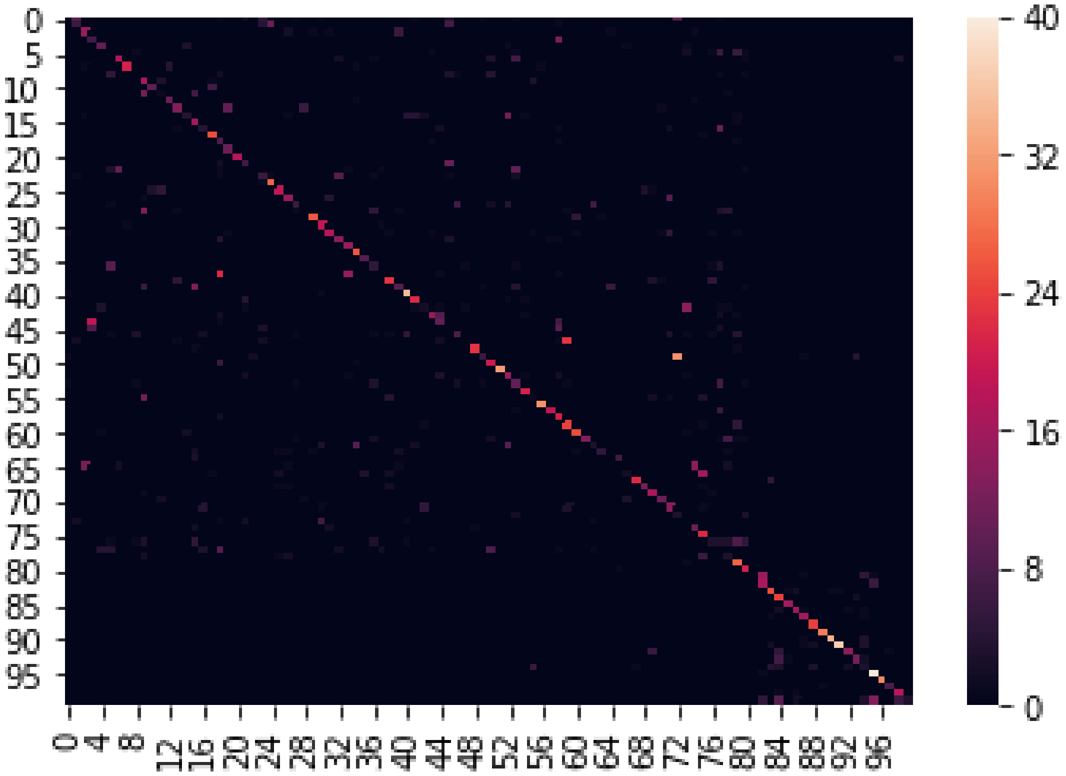

Figure 11: Confusion Matrix for session mismatch case

5.4 STYLE MISMATCH CASE

Voice of each speaker was recorded in two different styles – reading and conversation. In this experiment, training has been performed using 'reading' style while validation and testing have been performed using 'conversation' data. IITG-MV database has data for five different devices. That is why we get five accuracy figures, one for each device.

Table 1 SI accuracy for session mismatch scenario for IITG-MV SR

Device	Num Speakers	Train	Test	CNN Accuracy	CNN Precision	CNN Recall
D01	100	Reading	Conversation	65	0.6688	0.6692
H01	100	Reading	Conversation	66	0.6712	0.6737
T01	100	Reading	Conversation	64	0.6492	0.6488
M01	100	Reading	Conversation	63	0.6482	0.6423
M02	100	Reading	Conversation	62	0.6337	0.6294

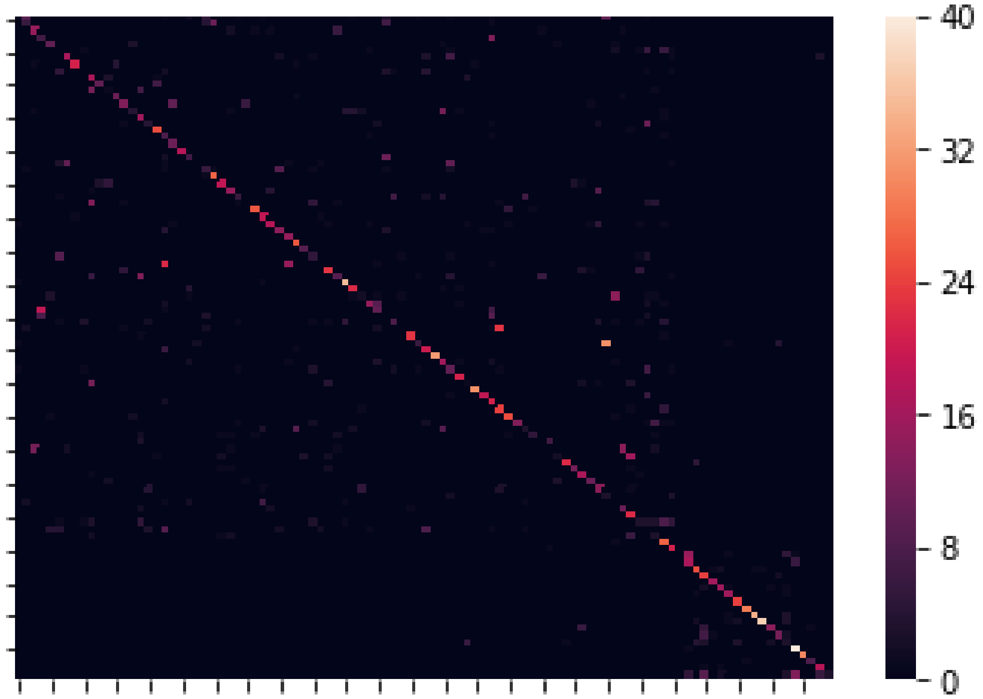

Figure 12: Confusion Matrix for style mismatch case

5.5 LANGUAGE MISMATCH CASE

Each speaker in IITG-MV spoke in two different languages. One was obviously English, and the other language was the speaker's favourite language. In this experiment, training has been performed using 'English language' while validation and testing have been performed using 'favourite language.' IITG-MV database has data for five different devices. That is why we get five accuracy figures, one for each device.

Table 1 SI accuracy for language mismatch scenario for IITG-MV SR

Device	Num Speakers	Train	Test	CNN Accuracy	CNN Precision	CNN Recall
D01	100	English	Favorite	85	0. 8613	0.8680
H01	100	English	Favorite	84	0.8490	0.8520
T01	100	English	Favorite	85	0.8624	0.8612
M01	100	English	Favorite	77	0.7812	0.7795
M02	100	English	Favorite	76	0.7734	0.7720

5.6 HOW MANY EPOCHS?

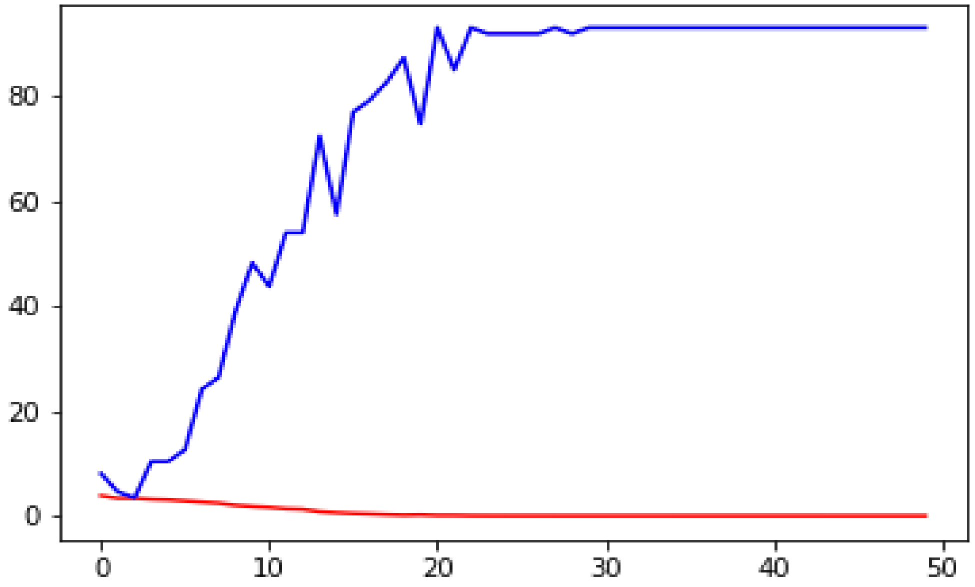

Figure 13: Train loss (red) and validation accuracy (blue) vs number of epochs

Training loss (red) is continuously decreasing with number of epochs. This is a clear indication that the model is learning patterns from training data. Also, validation accuracy (blue) is fluctuating till 25th epoch. However, no more fluctuations found after that, which indicates we can safely stop CNN training at 50th epoch.

5.7 KERNEL SIZE?

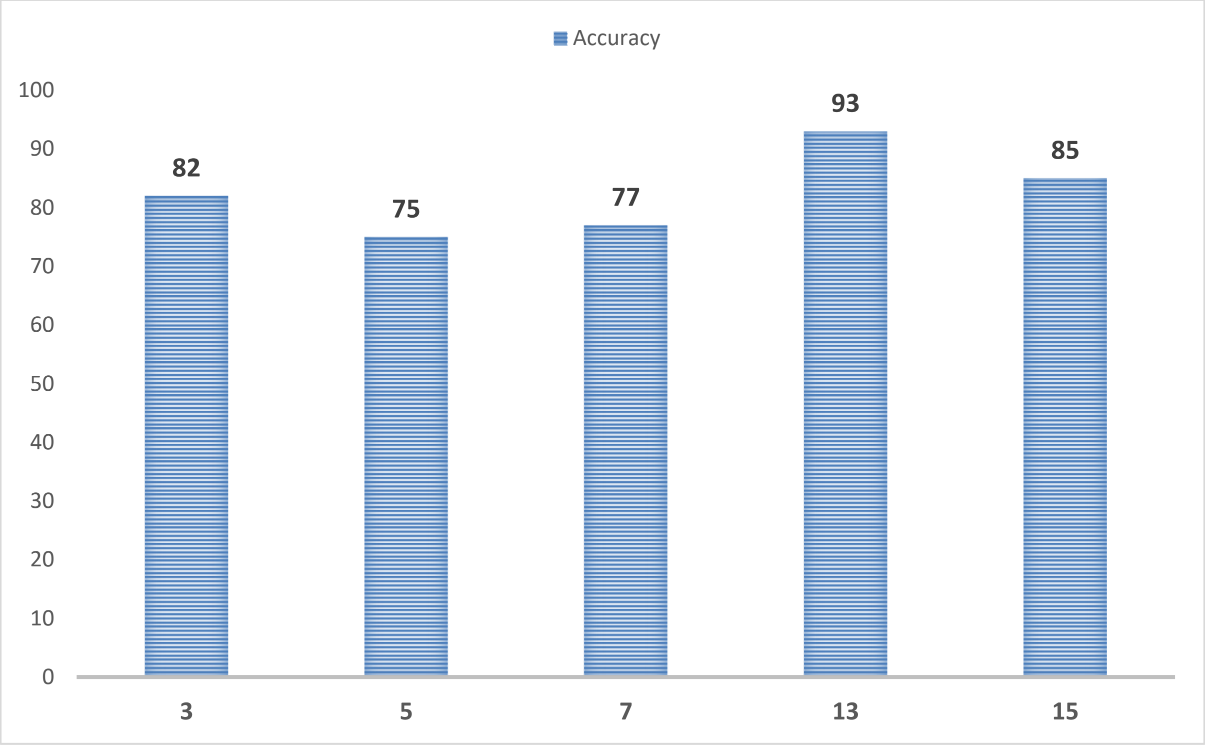

Figure 14: Variation of accuracy with kernel size

During CNN training, an important point is ‘What would be the kernel size’. We have tested square kernels for various size and found kernel with size 13*13 performs better than others. So, kernel of size 13 was used in this experiment.

CONCLUSION

CNN based approach gave significantly high accuracy in device independent speaker identification in the noisy dataset. The experimental setup is close to the real-life scenario, where speech data can be recorded in any environment or device or any language. We can conclude that CNN based approach can successfully be applied for real-world voice recognition. Though hyperparameter values can further be tuned to improve analysis on short frames. IITG-MV audio database has many real-life scenarios like environment mismatch, language mismatch, session mismatch, etc. The experiment was performed for all those mismatch scenarios. In the future, we can improve the accuracy of style and session mismatch. Also, we can work on who speaks when scenarios.

REFERENCES

[1] S. K. Pal, D. D. Majumder, Fuzzy sets and decision-making approaches in vowel and speaker recognition, IEEE Transactions on Systems, Man, and Cybernetics 7 (8) (1977) 625 629.

[2] B. Barai, D. Das, N. Das, S. Basu, M. Nasipuri, VQ/GMM Based Speaker Identification with Emphasis on Language Dependency, in: ACSS, Springer (In Press), Kolkata, 2018.

[3] Barai, B., Das, D., Das, N., Basu, S., Nasipuri, M.: An asr system using mfcc and vq/gmm with emphasis on environmental dependency (01 2018)

[4] Barai, B., Das, D., Das, N., Basu, S., Nasipuri, M.: Closed-set text-independent automatic speaker recognition system using vq/gmm. In: Intelligent Engineering Informatics, pp. 337–346. Springer (2018)

[5] J. Fortuna, P. Sivakumaran, A. Ariyaeeinia, A. Malegaonkar, Open-set speaker identification using adapted gaussian mixture models, in: Ninth European Conference on Speech Communication and Technology, 2005.

[6] D. Matrouf, W. Ben Kheder, P. M. Bousquet, M. Ajili, J. F. Bonastre, Dealing with additive noise in speaker recognition systems based on i-vector approach, in: 2015 23rd European Signal Processing Conference, EUSIPCO 2015, 2015, pp. 2092 2096. doi:10.1109/EUSIPCO.2015.7362753.

[7] N. Wang, P. Ching, N. Zheng, T. Lee, Robust speaker recognition using both vocal source and vocal tract features estimated from noisy input ut- terances, in: Signal Processing and Information Technology, 2007 IEEE International Symposium on, IEEE, 2007, pp. 772 777.

[8] K. S. Rao, S. Sarkar, Robust speaker recognition in noisy environments, Springer, 2014.

[9] H. Fujihara, T. Kitahara, M. Goto, K. Komatani, T. Ogata, H. G. Okuno, Speaker identi cation under noisy environments by using harmonic struc- ture extraction and reliable frame weighting, in: Ninth International Con- ference on Spoken Language Processing, 2006.

[10] M. I. Mandasari, R. Saeidi, M. McLaren, D. A. Van Leeuwen, Quality measure functions for calibration of speaker recognition systems in various duration conditions, IEEE Transactions on Audio, Speech and Language Processing 21 (11) (2013) 2425 2438. doi:10.1109/TASL.2013.2279332.

[11] M. I. Mandasari, R. Saeidi, D. A. Van Leeuwen, Quality measures based calibration with duration and noise dependency for speaker recognition, Speech Communication 72 (2015) 126 137. doi:10.1016/j.specom.2015. 05.009.

[12] P. Rose, Technical forensic speaker recognition: Evaluation, types and test- ing of evidence, Computer Speech & Language 20 (2-3) (2006) 159 191.

[13] N. Singh, R. Khan, R. Shree, Applications of speaker recognition, Procedia engineering 38 (2012) 3122 3126.

[14] E. Lleida, L. J. Rodriguez-Fuentes, Speaker and language recognition and characterization: Introduction to the csl special issue (2017).

[15] A. Reda, S. Panjwani, E. Cutrell, Hyke: a low-cost remote attendance tracking system for developing regions, in: Proceedings of the 5th ACM workshop on Networked systems for developing regions, ACM, New York, NY, USA, 2011, pp. 15 20.

[16] Campbell, W.M., Sturim, D.E., Reynolds, D.A.: Support vector machines using gmm super- vectors for speaker verification. IEEE signal processing letters 13(5), 308–311 (2006)

[17] Dieleman, S., Schrauwen, B.: End-to-end learning for music audio. In: 2014 IEEE Interna- tional Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 6964–6968. IEEE (2014)

[18] Ghahabi, O., Hernando, J.: Restricted boltzmann machines for vector representation of speech in speaker recognition. Computer Speech & Language 47, 16–29 (2018)

[19] Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., et al.: Recent advances in convolutional neural networks. Pattern Recognition 77, 354–377 (2018)

[20] Haris, B., Pradhan, G., Misra, A., Shukla, S., Sinha, R., Prasanna, S.: Multi-variability speech database for robust speaker recognition. In: Communications (NCC), 2011 National Confer- ence on. pp. 1–5. IEEE (2011)

[21] Jumelle, M., Sakmeche, T.: Speaker clustering with neural networks and audio processing. arXiv preprint arXiv:1803.08276 (2018)

[22] Madikeri, S., Bourlard, H.: Kl-hmm based speaker diarization system for meetings. In: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 4435–4439. IEEE (2015)

[23] McFee, B., Raffel, C., Liang, D., Ellis, D.P., McVicar, M., Battenberg, E., Nieto, O.: librosa: Audio and music signal analysis in python (2015)

[24] T. Chakraborty, B. Barai, B. Chatterjee, N. Das, S. Basu, M. Nasipuri : Closed-set Device-independent Speaker identification using CNN, in: International Conference On Intelligent Computing And Communication (ICICC - 2019), Springer, 2019.