

A Lightweight Speaker Recognition System Using Timbre Properties

Abu Quwsar Ohi¹, M. F. Mridha¹, Md. Abdul Hamid², Muhammad Mostafa Monowar², Dongsu Lee³, Jinsul Kim³

¹ Department of Computer Science & Engineering, Bangladesh University of Business & Technology, Dhaka, Bangladesh 1

quwsarohi@gmail.com, firoz@bubt.edu.bd 1

² Department of Information Technology, Faculty of Computing & Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia 2

mmonowar@kau.edu.sa, ferdousmridha@gmail.com 2

³ School of Electronics and Computer Engineering, Chonnam National University, Gwangju, South Korea 3

Abstract. Speaker recognition is an active research area that contains notable usage in biometric security and authentication system. Currently, there exist many well-performing models in the speaker recognition domain. However, most of the advanced models implement deep learning that requires GPU support for real-time speech recognition, and it is not suitable for low-end devices. In this paper, we propose a lightweight text-independent speaker recognition model based on random forest classifier. It also introduces new features that are used for both speaker verification and identification tasks. The proposed model uses human speech based timbral properties as features that are classified using random forest. Timbre refers to the very basic properties of sound that allow listeners to discriminate among them. The prototype uses seven most actively searched timbre properties, boominess, brightness, depth, hardness, roughness, sharpness, and warmth as features of our speaker recognition model. The experiment is carried out on speaker verification and speaker identification tasks and shows the achievements and drawbacks of the proposed model. In the speaker identification phase, it achieves a maximum accuracy of 78%. On the contrary, in the speaker verification phase, the model maintains an accuracy of 80% having an equal error rate (ERR) of 0.24.

Keywords: Timbre Analysis, Speech Processing, Random Forest, Embeddings



This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

 **JCC** © Journal of Contents Computing

Vol. 2, No. 1, pp. 139-152, June. 2020

Received 18 March 2020;

Revised 25 May 2020

Accepted 26 June 2020

***Corresponding Author;**

M. F. Mridha

Tel: 

E-mail:

quwsarohi@gmail.com

1 Introduction

Speaker recognition is the process of recognizing an individual by hearing a voice. Speaker recognition is an important perspective of biometric identification and verification. Commonly, speaker recognition is considered as a pattern recognition problem in which, the goal of the recognizer is to identify a speaker (previously known) by analyzing the vocal properties of a speech. Generally, humans recognize speakers based on the previously learned timbral properties of speech. Timbral properties refer to the basic properties of speech features such as hardness, softness, roughness, etc. Speaker recognition can be divided into two divisions based on the usage of the system, speaker identification [1], and speaker verification [2]. In terms of machine learning, the identification systems use multi-classification models, whereas the verification systems use binary-classification models. Concerning the utterance used for speaker recognition models, the model can be either text-independent or text-dependent. A text-dependent model only recognizes speakers based on the predefined keyword or passphrase that needs to be uttered by the speaker. This feature is preferred for unlocking devices or verification purposes. Microsoft implemented the text-dependent speaker verification on Windows 10 [3]. On the contrary, a text-independent model can recognize speakers based on any utterance of the speakers. At present, most state of the art speaker recognition model uses a text-independent recognition scheme. Speaker recognition has a wide variety of usage in the biometric authentication system, speaker diarization, forensics, and security [4, 5, 6]. Speaker recognition systems also have an estimable influence on business strategies. Speaker recognition systems can be implemented in bank customer-care services for identifying clients. Moreover, call-centers can be implemented with speaker recognition services to generate customer dependent services and agents. Furthermore, speaker recognition can be used to identify fraud callers. Speaker recognition systems have wide usage in the domain of speaker diarization. Speaker diarization is the process of labeling speech signals based on the identification of the speakers. Speaker diarization has an important role in dialogue generation. Although speaker recognition systems have greater industrial value, the challenge of speaker recognition systems is implementing an architecture that is suitable for real-time identification and verification. Currently, most state-of-the-art speaker recognition systems rely on deep neural networks (DNN). However, implementing these systems require heavy time-complexity feature extraction and pattern recognition procedure. In this paper, we introduce a speaker recognition procedure that is based on a statistical evaluation of speech timbral properties and does not require heavy feature extraction procedures. We propose a systematic approach of speaker recognition and verification system that extracts human timbral properties using regression. Further, the system implements a random forest classifier to the extracted timbral properties to identify speakers. The overall contributions of the paper can be concluded as follows:

- We introduce a speaker recognition system that identifies speakers based on the timbral properties of the speech.
- We report speech timbral properties can be extracted from mel-frequency cepstral coefficients (MFCC) using regression.
- We experiment with a famous dataset and evaluate the performance of our proposed architecture in speaker identification and verification scheme.

The paper is organized as follows. In Section 2 we analyze the architectures that are proposed in the speaker recognition domain. In Section 3, we describe the data set used to evaluate the proposed model. The overall architecture of the proposed model is derived in Section 4. The empirical results are reported in Section 5. Finally, Section 6 concludes the paper.

2 Related Works

Most of the models that are previously introduced use some common ideas, such as, Gaussian Mixture Model (GMM), Hidden Markov Model (HMM), Dynamic Time Wrapping (DTW), etc. However, the current strategy of speaker identification and verification relies on Deep Neural Network (DNN) architectures. The recent DNN architectures often rely on feature extraction through embeddings [7], which are also defined as feature vectors. These feature vectors are often termed as supervectors [8]. At present, most advanced models rely on supervectors. Currently, numerous versions of the supervectors are being implemented, among which, the most commonly practiced form is identity vectors, which is also described as ivectors [9, 10, 11]. I-vectors are extracted using GMM and performed better than most traditional methods. However, the present improvement of DNN architectures led to extract more robust identity vectors, termed as d-vectors [3]. Furthermore, more complex pre-processing of identity vectors are being formed using DNN that is named x-vectors [12]. Currently, x-vectors are performing better than the previous versions of identity vectors [13]. Although these voice identity vectors generating better results, the challenging task of implementing these vectors is the pre-training phase. Often these identity vectors require a large dataset to correctly generate an identity function that is suitable enough to generate discriminative identity vectors. Furthermore, if a system requires pre-training, then often it is considered to perform better if there exists a correlation between the pre-training data and testing data. Therefore, a greater correlation between pre-training and testing data causes better accuracy. On the contrary, a lesser correlation may result in achieving poor results. Therefore, identity vectors are not suitable for real-world speaker identification and verification tasks. Apart from using identity vectors, numerous speaker identification and verification models adapt to different schemes. Currently, a DNN architecture SincNet is introduced that directly processes raw waveform to identify speakers [14]. The architecture processes raw waveform via a learnable sinusoidal formula that generates dynamic time model properties to identify speakers. Furthermore,

various architectures extract speech features from MFCC [15, 16]. Moreover, a popular identification method named as triplet-loss is also implemented to identify speakers [17]. Although the state of the art models performs well, a tradeoff lies between choosing deep learning based models and non-deep learning based models. Models that do not implement neural networks, fall behind on gaining better estimations. On the contrary, the DNN or ANN-based models produce higher accuracy, yet they fall behind in recognizing speakers on the real-time continuous audio stream. Although the execution process of neural networks can be fastened up using GPUs, low-end devices are still vulnerable to implementing neural networks. Hence, they are not suitable to be used in most of the average-powered devices. To perform speaker recognition on IoT devices, and smartphones, these devices need to rely on powerful remote servers. To balance the accuracy of speaker recognition along with the computational complexities, we introduce a lightweight speaker recognition system. Instead of speech identification vectors, we implement a regression-based strategy using random forest, that extracts the timbral properties of human voices. As no prior datasets are available that can extract timbral from noise, we built a dataset that contains timbral scales based on the input speech. A total of seven timbral features are further passed to a random forest classifier. The classifier generates class labels based on the input speech frames.

3 Data Source

3.1 Librispeech Corpus

For training and evaluation, the LibriSpeech corpus is used [18]. It contains speech audios that are labeled based on the 40 speakers. The dataset contains silenced segments that were not stripped and our proposed architecture extracts speaker information by directly using the raw audio data.

3.2 Timbre Dataset Generation

The model performs regression to extract the timbre properties from speech audio. As there is almost no proper estimation and research done on vocal timbral properties, the dataset generation for timbral properties extraction was cumbersome. We found one tool developed by AudioCommons¹, which could extract all the seven features that are used in the model. Yet the tool produced erroneous outputs for some vocal speech. Therefore, we produced a small dataset that contains speech audios and the seven vocal timbral properties, boominess, brightness, depth, hardness, roughness, sharpness, and warmth for each speech audio. The dataset contains

¹<https://github.com/audiocommons/audiocommons>

400 samples of 0.3-seconds length audio speech with the seven timbral properties of each audio speech. The timbral features for each audio were firstly generated from the tool produced by AudioCommons and then filtered by human intuition. The 400 short audio speeches were randomly selected from LibriSpeech clean dataset. This dataset was used to train the seven individual feature extractor regressors.

4 Methodology

In this section, the methodology of the proposed model is presented. Moreover, Figure 1 presents the overall workflow of the architecture.

4.1 Input Processing

Inputs passed to the model are clean and noise-free audio streams, which may contain silence streams as well. Each of the audio streams is scaled using the following formula,

$$S(x) = \frac{x_i}{\max(\text{abs}(x_1, x_2, \dots, x_n))} \quad (1)$$

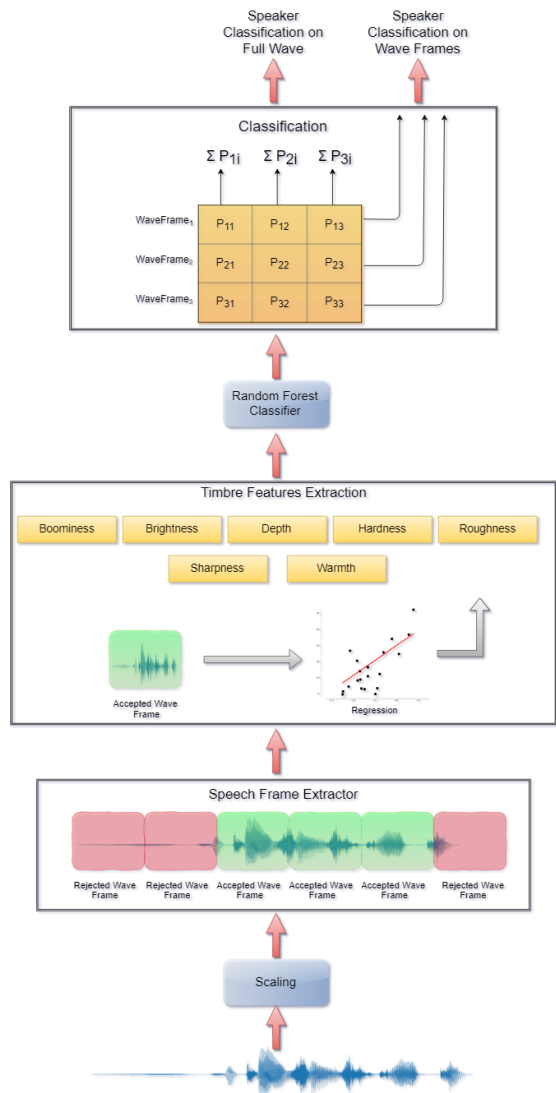


Fig. 1 The figure illustrates the workflow of the proposed architecture (from bottom to top). The continuous raw waves are first scaled and separated on multiple wave frames. The silence wave frames are filtered out, and the timbral features are extracted using a random forest regressor. The timbral features are further classified using a random forest classifier.

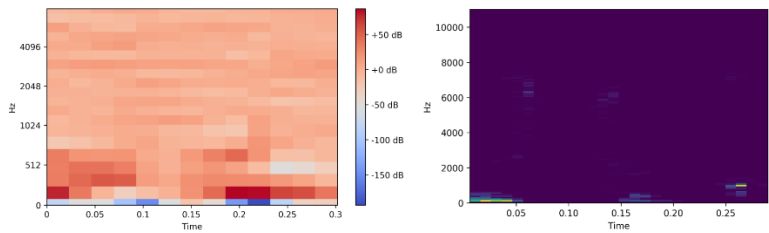


Fig. 2 An illustration of the MFCC-spectrogram and frequency spectrogram of a 0.3-second speech frame, having weighted sum values of 11818.91, and 0.61 respectively.

The scaled audio stream further helps to remove the silenced audio frames and the extracted features to be more accurate.

4.2 Speech Frame Extractor

The audio stream is further partitioned into audio segments. At first, this phase partitions every 0.3-second consecutive stream of the audio as frames. Each of the wave frames is further passed through the mean calculation function defined as follows,

$$AcceptedFrames = \{S|f(S) = \frac{\sum_{i=1}^n s_i}{n}\} \tag{2}$$

Here, a frame is rejected if the mean of the amplitudes of each wave frame is less than the threshold value that is set to 0.05. This threshold value helps to eliminate the silence parts of the audio streams, which are unnecessary.

4.3 Timbre Features Extraction

To extract the timbre properties of sound, the model uses random forest regression. As parameters for regression, a weighted sum of MFCC spectrogram and frequency spectrogram as features. The weighted sum is derived as follows,

$$Sum_{weighted} = \sum_{i=1}^{n,m} f(i) \times t(j) \times spec(i,j) \tag{3}$$

Where, f(i) = Frequency of the i'th index t(j) = Time of the j'th index spec(i,j) = Intensity of the sound on f(i) frequency, at time t(j)

The regressor is trained with the prepared dataset containing 400 wave frames and seven timbral properties. For each 0.3-second audio frame, the weighted sum is generated, and the seven timbral properties are trained individually with seven individual random forest regressors.

A short description of the seven extracted speech features is presented below. Boominess: Booming refers to the deep and resonant sounds. The boominess of sound also can be extracted using the implementation of Hatano and Hashimoto's boominess index [19].

- Brightness: Brightness refers to the higher frequency of sound.
- Depth: The term depth can be related to the spreading feel of sound concerning the loudness of sound.
- Hardness: This refers to the unbalanced and noisy tone of the sound.
- Roughness: This refers to the rapid modulation of sound.
- Sharpness: This refers to the amount of high-frequency sound concerning the amount of low frequency of sound. The sharpness of a sound also can be found refers to the amount of high-frequency sound concerning the amount of low frequency of sound. The sharpness of a sound also can be found using Fastl sharpness algorithm [20].
- Warmth: Warmth is the opposite of the brightness of the sound.

4.4 Speaker Classification

Each of the features is fed to the Random Forest classifier. To measure the quality of a split, the Gini impurity measure is used, which can be stated as,

$$G = \sum_{i=1}^c p(i) \times (1 - p(i)) \quad (4)$$

The features of each accepted wave frame processed separately in train and test sessions. In the test session, the classifier outputs the probabilities of each speech wave frame uttered from a particular person. The classification of this model can be for each wave frame or of the full audio stream. To classify each wave frame, the probability vector passed that is the output of the random forest classifier, is passed through the arguments of maxima that can be stated as,

$$\operatorname{argmax}_x f(x) = \{x | f(x) = \max_{x'} f(x')\} \quad (5)$$

To classify the speaker of the full input audio stream, the probability vectors of the individual wave frames are gathered and produced as a probability matrix. The matrix is then converted to a probability vector defined as,

$$P_i = \sum_j^n p_{ij} \quad (6)$$

The generated probability vector is passed through the arguments of maxima function stated in equation 5 to calculate the final classification for the full audio stream.

5 Empirical Results

5.1 Evaluation Setup

Relative and sharable performance measures are required to estimate how superior an algorithm or approach is. The major problem for evaluating any method is the adoption of training and testing sets, which can introduce an inconsistency in model performance. Most of the performance metrics are based upon the confusion matrix, which consists of true positive (TP), true negative (TN), false positive (FP), and false negative (FN) values [21]. The significance of these elements can vary on how the performance evaluation is done.

The term 'recognition' can be classified into two separate operations, identification, and verification. The identification system seeks the identity of persons, whereas the verification systems only check if the person is the one whom it is expected to be. The proposed system is tested both of the scenarios and evaluation data are presented in this section.

The accuracy of an identification system can be defined by how many correct guesses the model estimates, from the total estimations made by the model. The accuracy is measured as,

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}(7)$$

To evaluate the verification system, the Receiver Operating Characteristics Curve (ROC) and Equal Error Rate (EER) is calculated. The ROC curve is a well-known non-parametric estimation method in the field of biometric authentication and verification systems [22]. The ROC curve generates a visual of the probability of true detection (True Positive Rate or, TPR) versus the probability of false alarm (False Positive Rate or, FPR). The area generated by the ROC curve is known as the area under the curve (AUC). A higher value of AUC ensures the robustness of the verification system. EER can be evaluated from the ROC curve, by pointing the position where TPR is higher than FPR and $TPR + FPR = 1$. Lower EER value confirms the robustness of a verification system.

5.2 Experimental Setup

The experimental reports were generated by running the model on a 2.7Ghz Intel i3 processor with 4 gigabytes of ram. All the mentioned steps of the prototype are implemented using Python [23]. The random forest classifier and regressor models are implemented using scikit-learn [24]. Also, for additional calculation, implementation, and support, Numpy [25] and librosa [26] are used. The visual evaluation reports are generated using Matplotlib. The dataset used to test the architecture is directly inserted, and no variations or selections were made while testing the architecture.

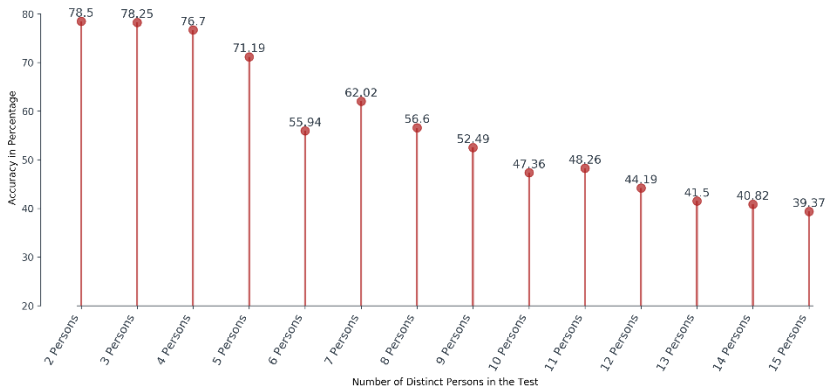


Fig. 3 The graph illustrates the accuracy score of the speaker identification phase of the proposed architecture. The vertical axis represents the accuracy scale, whereas the horizontal scale represents the number of unique persons introduced in the identification phase.

5.3 Experimental Results

5.3.1 Speaker Identification

Speaker identification is the process of targeting a speaker by hearing the voice. In terms of machine learning, speaker identification is a multiclass classification problem. Figure 3 represents the identification accuracy of the proposed architecture while presenting a different number of persons. The prototype’s performance degrades concerning the increasing number of individual persons. The degradation points to the characteristics of the features. The features which are extracted and used in our model are densely associated with each other. Therefore, the classifier fails to fit on training data appropriately. This degradation points out that the model can only be used for a small group of individuals for identification purposes.

5.3.2 Speaker Verification

Speaker verification is the method of confirming if the voice is of a specific person. Aside from the unbalanced accuracy of the identification score of the model, it presents better performance in speaker verification. In terms of machine learning, speaker verification is stated as a binary classification problem. Figure 4 illustrates the accuracy scores of the model including a different number of individuals in the verification phase. The proposed model generates a satisfactory score in the speaker

verification phase. It shows accuracy above 80% in most of the tested environments. The model continuously provided a stable accuracy, while the number of unique speakers was increased.

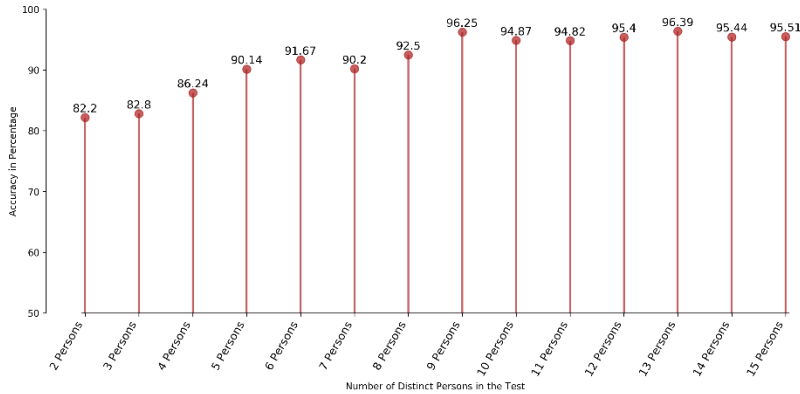


Fig. 4 An illustration of the MFCC-spectrogram and frequency spectrogram of a 0.3-second speech frame, having weighted sum values of 11818.91, and 0.61 respectively.

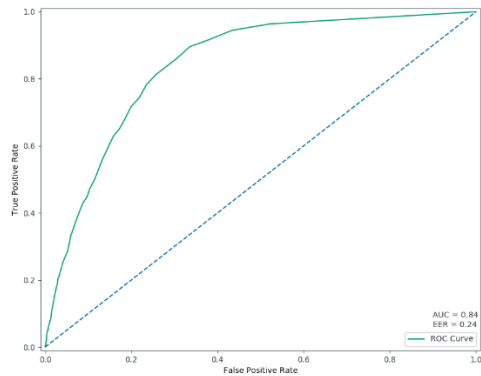


Fig. 5 The figure represents a ROC curve of the model. The curve is generated based on identifying a random individual from the dataset. The model generates an EER of 0.24, while the AUC is 0.84.

Figure 5 represents the ROC curve of the proposed model that is tested on a random individual. The proposed model gives an equal error rate (EER) of 0.24, while the area under the curve (AUC) being 0.84. The equal error rate represents that the model generates its best result in verifying an individual from a continuous stream of audio.

6 Conclusion

In this paper, we proposed a model that uses the timbral properties of voice, that is hardly used in any other research endeavors. The model is tested against a real-world continuous stream of audio, without any modification. Although the model almost fails in the speaker identification phase, it achieves a marginal score in the speaker verification phase. The model's accuracy can be improved if the scaling of the features is estimated more accurately. As the paper introduces new speech properties, further studying these features that are illustrated in this paper, the researchers of the speaker recognition system will be motivated to try out the vocal sound properties rather than only using sound waves or identity vectors as features. Therefore, we believe this research effort will influence the researches to explore new speech properties that may result in inventing more robust and lightweight architectures.

Acknowledgements

This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2020-2016-0-00314) supervised by the IITP(Institute for Information & communications Technology Planning & Evaluation) and X-mind Corps program of National Research Foundation of KREA(NRF) funded by the Ministry of Science, ICT (grant number).

References

- [1] Reynolds DA (1995) Speaker identification and verification using Gaussian mixture speaker models *Speech communication* 17(1-2):91-108
- [2] Reynolds DA, Quatieri TF, Dunn RB (2000) Speaker verification using adapted Gaussian mixture models *Digital signal processing* 10(1-3):19-41
- [3] Zhang SX, Chen Z, Zhao Y, Li J, Gong Y (2016) End-to-end attention based text-dependent speaker verification. In: 2016 IEEE Spoken Language Technology Workshop SLT, San Diego, p 171-178.
- [4] Beigi H (2011) Speaker recognition. In: *Fundamentals of Speaker Recognition*, Boston, p 543-559
- [5] Furui S (1992) Speaker-independent and speaker-adaptive recognition techniques. *Advances in Speech signal processing* 597-622
- [6] Sadaoki F (1997) Recent advances in speaker recognition *Pattern recognition letters*. *Pattern recognition letters* 18(9):859-872
- [7] Ohi AQ, Mridha MF, Safir FB, Hamid MA, Monowar MM (2020) Auto Embedder: A semi-supervised DNN embedding system for clustering. *Knowledge-Based Systems* 106190. doi: 10.1016/j.knosys.2020.106190

- [8] Kinnunen T, Li H (2010) An overview of text-independent speaker recognition: From features to super vectors. *Speech communication* 52(1): 12-40.
- [9] Garcia-Romero D, Espy-Wilson CY (2011) Analysis of i-vector length normalization in speaker recognition systems. In *Twelfth annual conference of the international speech communication association*
- [10] Mitchell, David VL (2011) Source-normalized-and-weighted lda for robust speaker recognition using i-vectors. In: *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Florence, p 5456-5459
- [11] Glembek O, Burget L, Brümmer N, Plchot O, Matejka P (2011) discriminatively trained i-vector extractor for speaker verification. In: *Twelfth Annual Conference of the International Speech Communication Association*
- [12] Snyder D et al (2018) X-vectors: Robust dnn embeddings for speaker recognition. In: *IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP)*, IEEE, Calgary, p 5329-5333
- [13] Villalba J, Nanxin C, Snyder D, Garcia-Romero D, McCree A, Gregory S, Jonas B et al (2019) State-of-the-Art Speaker Recognition for Telephone and Video Speech: The JHU-MIT Submission for NIST SRE18, *Interspeech*, p 1488-1492
- [14] Ravanelli M, Bengio Y (2018) Speaker recognition from raw waveform with sincnet. In: *2018 IEEE Spoken Language Technology Workshop*, p 1021-1028
- [15] Nakagawa S, Asakawa K, and Wang L (2007) Speaker recognition by combining mfcc and phase information. In: *Eighth annual conference of the international speech communication association*, Belgium, 2007
- [16] Murty, KSR & Yegnanarayana B (2005) combining evidence from residual phase and MFCC features for speaker recognition *IEEE signal processing letters* 13(1) 52:55
- [17] Zhang C & Koishida K (2017) End-to-end text-independent speaker verification with triplet loss on short utterances In: *Interspeech*, pp 1487-1491
- [18] Panayotov V, Chen G, Povey D & Khudanpur, S (2015). Librispeech: an asr corpus based on public domain audio books ,n *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* p. 5206-5210
- [19] Shin SH, Ih JG, Hashimoto T, & Hatano S (2009) Sound quality evaluation of the booming sensation for passenger cars. *Applied acoustics* 70(2):309-320
- [20] Fastl H (2005) Psycho-acoustics and sound quality. In: *Communication acoustics*, Springer, Heidelberg, p 139-162.
- [21] Peres DJ, Cancelliere A (2014) Derivation and evaluation of landslide-triggering thresholds by a Monte Carlo approach. *Hydrology and Earth System Sciences* 18(12):4913
- [22] Oliphant TE (2007) Python for scientific computing. *Computing in Science & Engineering* 9(3):10-20
- [23] Pedregosa F et al (2011) Scikit-learn: Machine learning in Python, *The Journal of Machine Learning Research* 12:2825-2830
- [24] Walt SVD, Colbert SC, Varoquaux G (2011) The NumPy array: a structure for efficient numerical computation. *Computing in science & engineering* 13(2):22-30
- [25] McFee B, Raffel C, Liang D, Ellis DP, McVicar M, Battenberg E, Nieto O (2015) librosa: Audio and music signal analysis in python. In: *Proceedings of the 14th python in science conference*, vol 8. TX, Austin, p 18-25
- [26] Hunter JD (2007) Matplotlib: A 2D graphics environment. *Computing in science & engineering* 9(3):90-95