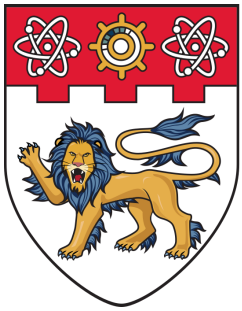


NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

NOISE-ROBUST AUTOMATIC SPEAKER VERIFICATION

Qualifying Examination Report

Submitted to the College of Computing and Data Science

of the Nanyang Technological University

by

Truong Duc Tuan

Supervisor: Assoc. Prof. Chng Eng Siong

July, 2024

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research, is free of plagiarised materials, and has not been submitted for a higher degree to any other University or Institution.

.....05.July.2024.....

Date

Truong Duc Tuan

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it is free of plagiarism and of sufficient grammatical clarity to be examined. To the best of my knowledge, the research and writing are those of the candidate. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

.....05.July.2024.....

Date

Assoc Prof Chng Eng Siong

Authorship Attribution Statement

This thesis contains material from 1 paper accepted at conferences in which I am listed as an author.

Chapter 3 is published as **D. -T. Truong**, R. Tao, J. Q. Yip, K. Aik Lee and E. S. Chng, "Emphasized Non-Target Speaker Knowledge in Knowledge Distillation for Automatic Speaker Verification," ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2024, pp. 10336-10340, doi: 10.1109/ICASSP48485.2024.10447160.

The contributions of the co-authors are as follows:

- I conducted the literature review and brainstormed the new idea.
- I prepared the manuscript drafts. The manuscript was revised by Prof Chng and Prof Lee.
- I co-designed the experiments with Ruijie and Jiq Qi. I performed all the experimental work at the College of Computing and Data Science. I also analyzed the data.

05 July 2024

Date

Truong Duc Tuan

List of Figures

List of Tables

Contents

List of Abbreviations

Automatic Speaker Verification	ASV
Deep Neural Networks	DNNs
Self-supervised Learning	SSL
Knowledge Distillation	KD
Equal Error Rate	EER
Voice Activity Detection	VAD
Mel Filter-bank	Fbanks
Mel-frequency Cepstral Coefficients	MFCCs
Convolutional Neural Network	CNN
Gaussian Mixture Model	GMM
Universal Background Model	UBM
Joint Factor Analysis	JFA
Time-Delay Neural Network	TDNN
Recurrent Neural Network	RNN
Long Short-Term Memory	LSTM
Fully Connected	FC
Probabilistic Linear Discriminant Analysis	PLDA
Linear Discriminant Analysis	LDA
Room Impulse Responses	RIRs
False Rejection	FR
False Acceptance	FA
False Reject Rate	FRR
False Acceptance Rate	FAR
Detection Cost Function	DCF
Mean Square Error	MSE
Denoising Autoencoders	DAE
Maximum A Posteriori	MAP
Generative Adversarial Network	GAN
Diffusion Probabilistic Model	DPM
Signal-to-noise Ratio	SRN
Decoupled Knowledge Distillation	DKD
Floating-point Operations	FLOPs
Text-to-speech	TTS
Voice conversion	VC
Deepfake Speech Detection	DSD

Abstract

Automatic Speaker Verification (ASV) is a process of identifying speakers by their unique voice characteristics. Since speech is a fundamental means of human interaction, ASV systems have the potential to impact a wide range of applications in daily human activities. Deep neural networks (DNNs) with their powerful capacity of extracting data representation, have revolutionized the field of ASV, outperforming conventional methods. However, most DNN-based ASV systems are trained based on the supervised learning scheme, which is prone to overfitting the training speech. Therefore, it limits their effectiveness in real-world deployment with the presence of diverse unseen environmental noise. Without the need for labeled data, self-supervised learning (SSL) models are trained on a large volume of unlabeled data. Consequently, they can extract robust speech representations, which can be a crucial element in building a robust ASV system. Therefore, this study aims to improve the robustness of ASV systems against input noisy speech by leveraging SSL models.

To facilitate the ASV model to obtain robust speaker representation, we propose an effective knowledge distillation (KD) method to transfer the knowledge from large robust SSL models to smaller student models. During the distillation process, we modify the conventional label-level KD method by emphasizing the classification probabilities of non-target speakers. This modification is motivated by our empirical study, which demonstrates that the classification probabilities of non-target speakers are important for training ASV models. In experiments on three different student model architectures, our method achieves an average improvement of 13.67% in Equal Error Rate (EER) on the 'in-the-wild' VoxCeleb corpus compared to other KD methods.

Chapter 1

Introduction

1.1 Background

Speech is a primary form of human communication, relying on the unique characteristics of the speaker including the vocal tract shape, speaking accent, and rhythm [1]. This enables us to distinguish person to person based on their voices. The process of identifying a speaker by their voice can be automatically performed by a computer system. This system is called Automatic Speaker Verification (ASV) [2], which is the core research area of this thesis. The real-world application of ASV systems can be found in the two main tasks: authentication and identification [3]. Specifically, in the banking sector, users can be verified by the system for making transactions via phone calls or other related telephone banking services. On the other hand, during criminal investigations, voice recordings—such as a terrorist attack threat or a phone ransom demand—can serve as crucial evidence. In such scenarios, determining the identity of the speakers through audio evidence can accelerate the investigation process by limiting the number of potential suspects.

Since deep neural networks (DNNs) can be expanded in size and depth to be trained on vast datasets, they can extract sophisticated representations from input data and discriminative capability. Hence, DNNs have been extensively adopted in computer vision, natural language processing, and speech processing. In modern ASV systems, deep neural networks play a crucial role and have significantly outperformed traditional approaches [3–5]. However, a major challenge faced by deep neural networks is their vulnerability to the performance gap between the training and test datasets due to the problem of overfitting. In deep-learning-based ASV models, a popular example of this mismatch is the variety of background noises encountered in real-world environments. Studies by [6] and [7] have demonstrated that ASV systems leveraging DNNs, trained on clean speech, exhibit good performance in noise-free conditions, but struggle when testing with noise-

distorted speech data. Hence, tackling this issue is crucial to facilitate the practical deployment of ASV systems in real-world scenarios.

1.2 Motivation

With the ever-increasing amount of audio in real-world scenarios, improving the noise-robustness of ASV systems remains an open issue. Therefore, this study focuses on developing new techniques to tackle this problem.

To improve the robustness of speaker verification models in noisy environments, several techniques have been introduced, which can be broadly categorized into two approaches: feature-based and model-based compensation. Feature-domain compensation methods are frequently used as a front-end process to transform noisy features into clean ones. Various feature compensation techniques are initially proposed to focus on acoustic feature adaptation where noisy speech is normalized to the new one with less noise distortion [8, 9] or learn to map noisy regions of speech spectrum to clean ones [10, 11]. Another solution is data augmentation [12, 13], which adds new noisy speech to the training set for multi-condition data, hence improving the generalization of ASV models. In contrast, instead of estimating the clean feature of input speech, model-based compensation approaches adjust the parameters of ASV models to adapt to noisy speech. In this approach, one simple method is to train a speech enhancement module to recover the clean speech from the corresponding noisy one for further ASV process [14, 15]. Additionally, model-based adaptation is a widely used approach to align the model with the differences in input speech conditions between the train and test sets [16, 17].

The majority of current approaches for robust speaker verification are the supervise-learning scheme that replies to labeled data. However, a sufficient amount of training data may not always be available, and building training data for each new speech condition can be labor-intensive. If a large scale of labeled training data is not available, the trained model can still encounter the same problem of overfitting to the specific domain of training data. In order to achieve universal feature representations in diverse speech conditions for robust speaker verification, large-scale training data is needed. Addressing the challenge of costly data labeling, self-supervised learning (SSL) has become a feasible solution. By deriving target labels from corresponding unlabeled inputs, SSL eliminates the need for manual annotation. With the abundance of unlabeled data available today, SSL presents a promising paradigm for learning high-quality data representations. In the field of speech, researchers have adapted SSL, resulting in several SSL models such as wav2vec 2.0 [18], HuBERT [19], and WavLM [20]. These existing models have been explored and demonstrated the effectiveness of SSL models for various speech tasks,

including Automatic Speech Recognition [21, 22], Keyword Spotting [23, 24], Speaker Age Estimation [25, 26]. Several studies utilize large SSL models for robust ASV and achieve new state-of-the-art results [27, 28]. However, these studies simply concatenate the SSL model and a task-specific downstream model, and train the whole system with the corresponding task-specific loss. This makes the system costly as well as may be unable to take full use of the rich representation of SSL models for downstream tasks. This motivates our study to develop novel methods to better leverage the representation from self-supervised learning to improve the noise robustness of speaker verification.

1.3 Contribution

In this study, we address the problem of improving the noise-robustness of speaker verification mentioned in Section 1.2. In detail, we introduce a knowledge distillation method based on the output posterior of each speaker for speaker verification. Although SSL models have revolutionized various speech processing tasks [21–26] including ASV [29], these models are computationally expensive. To better utilize SSL models, knowledge distillation can be employed to transfer the robust speech representation to smaller student models. In speaker verification, one commonly used knowledge distillation method is label-level KD, which focuses on minimizing the Kullback–Leibler divergence between the output probabilities of the teacher and student networks. However, the conventional label-level KD overlooks the significant knowledge from non-target speakers, particularly their classification probabilities, which can be crucial for automatic speaker verification. Therefore, our first method is to modify the conventional label-level knowledge distillation to emphasize the classification probabilities of non-target speakers, which involves splitting and amplifying the non-target speaker's probabilities during the knowledge distillation process. The proposed method is applied to three different student model architectures and achieves an average of 13.67% improvement in EER on the VoxCeleb dataset compared to embedding-level and conventional label-level KD methods

1.4 Thesis Organization

The remaining parts of this thesis is structured as follows:

1. Chapter 2 provides an overview of the literature covering automatic speaker verification systems, including aspects such as input features, deep neural network architectures, and evaluation benchmarks. Noise-robust methods for ASV over the years are then discussed, with an analysis of the strengths and limitations present in each approach.

2. Chapter 3 proposes the knowledge distill method for automatic speaker verification that emphasizes non-target speaker knowledge.

3. Chapter 4 concludes this study by summarizing the contributions and proposing several potential directions for future research.

Chapter 2

Literature Review

In this chapter, we first introduce an overview information of automatic speaker verification including the system architecture, common datasets, and evaluation metrics in Section 2.1. In detail, each module of the ASV system including pre-processing, model architecture, scoring, and evaluation benchmark is discussed with a review of the existing methods. Then, in Section 2.2, a literature survey of current noise-robust speaker verification methods is presented with the listed methods grouped into two categories. The first category is feature compensation, which covers a range of techniques for enhancing the robustness of speaker embedding against noisy speech. The second category is model compensation approaches, which modify the acoustic model for a better adaptation to unseen speech conditions. In general, this chapter provides general information of this study and background knowledge for the contributions in Chapter 3.

2.1 Overview of Automatic Speaker Verification

Automatic Speaker Verification is a process of identifying whether two voices are from the person. ASV systems can be categorized into two types based on their design: stage-wise and end-to-end [3]. The stage-wise system involves two distinct stages: training speaker feature extraction and verifying speakers [2]. Conversely, end-to-end systems are trained to directly generate the similarity score between a pair of utterances in a single stage [7]. Fig. 2.1 depicts the pipeline of both stage-wise and end-to-end ASV systems.

In the stage-wise ASV system, the model first learns to extract speaker-specific feature representation [3]. The training phase can be conceptualized as a classification task, where the ASV model is designed with two main components: a feature extractor and a classification head. The feature extractor is responsible for transforming the raw audio input into a set of meaningful features that capture the unique characteristics of the speaker's voice. Following this, the classification head takes these extracted features

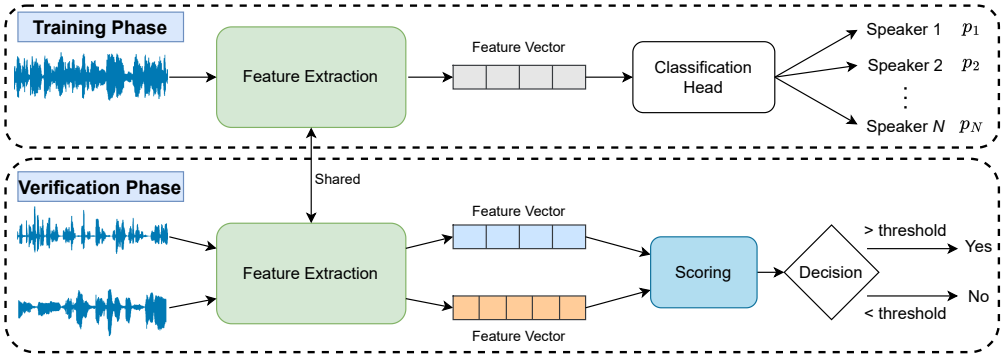
and predicts the posterior probability for each of the N speaker identities present in the training dataset. The classification head essentially acts as a softmax layer that outputs a probability distribution over the N speakers, indicating the likelihood p of the input audio belonging to each speaker. The model is then trained by optimizing the cross-entropy loss function, which measures the discrepancy between the predicted probabilities and the true speaker labels. In the verification phase, the learned feature extractor will compute the speaker embeddings of the registered (enrollment) and test audio inputs. Then, a similarity score is calculated between the two embeddings. The system then determines the verification result by comparing this similarity score to a predefined threshold. On the other hand, the end-to-end ASV system takes a pair of enrolment and test utterances as input and directly computes the similarity score. The feature extraction and scoring modules of end-to-end systems are jointly trained using pairwise loss functions such as Binary Cross-Entropy or Contrastive loss [7].

ASV is an open-set task since it can perform verification on new unseen speakers (different identities from the training speakers) by comparing two input utterances in the verification step. The concept of speaker verification may be easily confused with speaker identification. Different from speaker verification, speaker identification is a close-set problem since the system is trained to predict the speaker identification in a fixed given speaker set and it will be unable to predict new identity unless the model is re-trained [30]. Based on the dependence of the content in the input utterances, ASV is further divided into two approaches: text-dependent and text-independent systems. In the former approach, the registered and test utterances need to be spoken under the same sentences/phrases [3]. Text-dependent ASV systems usually have good performance, however, users are constrained to speak predefined words which can limit their application in real-world scenarios. On the other hand, in the text-independent ASV system, there are no restrictions based on the speech content of input utterances. Therefore, the text-independent approach is more generalized and can have broader applications in real-world systems. Therefore, this thesis will focus on text-independent ASV systems.

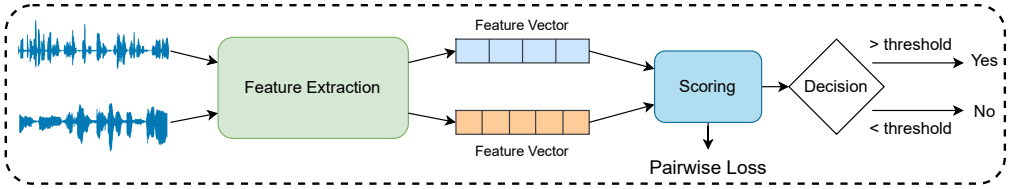
Whether using stage-wise or end-to-end approaches, the ASV system generally consists of two main components: a feature extractor and a decision-making module, as depicted in Fig. 2.2. The feature extraction component encompasses the pre-processing and core model architecture modules. Subsequent sections will provide a detailed explanation of each component.

2.1.1 Pre-processing

In ASV systems, the pre-processing stage is the beginning step which enhances the quality of the input speech and reliability of the system. This stage involves various methods



(a) Stage-wise system



(b) End-to-end system

Figure 2.1: The pipeline of stage-wise and end-to-end ASV systems.

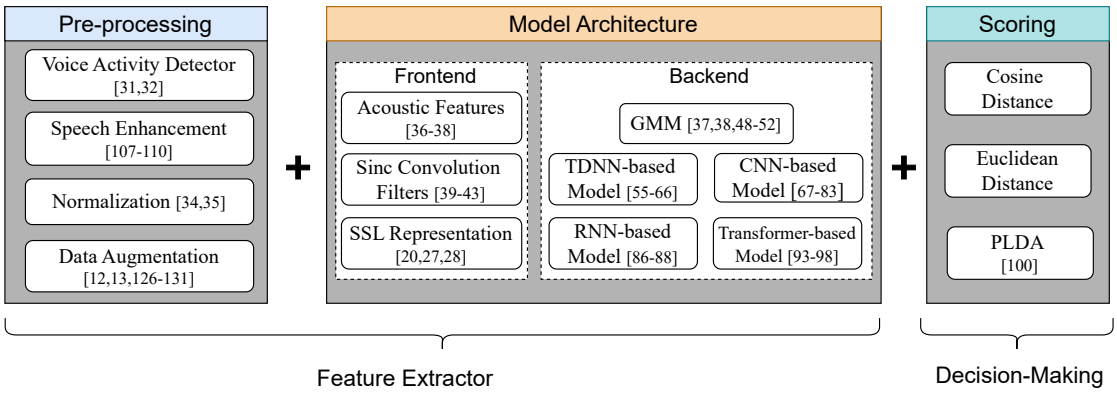


Figure 2.2: A diagram of components in ASV system

to refine the raw audio signal, ensuring that the subsequent feature extraction and classification processes are based on the most accurate representation of the speaker's voice. These methods include:

- Voice Activity Detection (VAD): This method distinguishes between speech and non-speech segments within an audio input. By removing non-speech portions, VAD effectively minimizes the computational load and improves the system's focus on relevant speech containing vocal or speaker information [31]. Commonly employed techniques such as energy-based VAD [32] determine the presence of speech by analyzing the energy levels within frames obtained during speech parameterization. Frames exhibiting energy levels below a predefined threshold are identified as non-

speech and subsequently excluded from further processing.

- Speech Enhancement: Given that speech in real-world environments is frequently distorted by noise and reverberation, speech enhancement methods can be utilized to preserve the integrity of the speaker's voice by removing background noise and other acoustic inferences. Speech enhancement methods can be categorized into traditional approaches including spectral subtraction and filtering methods or DNN-based speech enhancement methods [33]. The details of these methods will be further discussed in Section 2.2.2.

- Normalization: This critical method in the pre-processing stage aims to standardize the signal levels on the same scale by performing linear transformations across different audio samples [5]. This process ensures the stability in the amplitude or energy of the speech signals, reducing the variability in feature extraction [34, 35].

- Data Augmentation: Techniques can be applied to diversify the training speech, enriching it with variations in pitch, speed, and environmental conditions [2]. This enhances the system's ability to generalize across different scenarios. Further elaboration on data augmentation methods can be found in Section 2.2.1.

In summary, these four methods - VAD, speech enhancement, normalization, and data augmentation - form the pre-processing stage and contribute to the overall performance and reliability of ASV systems.

2.1.2 Model Architecture

The architecture of ASV systems consists of two primary components: the front-end and the back-end modules [3]. The front-end module is responsible for converting raw input waveforms into acoustic features. The back-end module then takes these acoustic features and transforms them into speaker embeddings, which are used for the final speaker verification process.

Front-end

Rather than employing the raw waveform directly, most ASV systems first transform the temporal speech signal into a spectro-temporal representation. This transformation simulates the cochlear processing found in the human auditory system. This spectral representation, known as acoustic features, includes Mel Filter-bank (Fbanks), and Mel-frequency Cepstral Coefficients (MFCCs) [36], etc. Among these, MFCCs are widely used features for speaker verification [31]. The process of extracting MFCCs begins by applying

the short-time discrete Fourier transform to the speech signal, yielding the representation of the signal's power or magnitude spectrum. This spectrum is then passed through a set of filters based on the Mel scale to obtain the Mel filter-bank output. After taking the logarithm of this output, the MFCC features are generated by using the discrete cosine transform. In numerous prior studies on speaker verification using clean speech, MFCCs have consistently demonstrated superior performance compared to other acoustic features [37, 38].

Nevertheless, MFCCs and other spectral features may not be optimal for speaker verifications, so researchers have recently focused on alternative ways of replacing these features in ASV systems with learnable feature representation. [39] was the first study utilizing learnable Convolutional Neural Network (CNN) layers as a set of filters to transform raw waveforms into intermediate representations. Further analysis of these CNN filters revealed that the filters capture information present in low-frequency regions. In contrast to the freely learned filters in vanilla CNNs, [40] proposed SincNet, which transforms the raw input waveform using a constrained set of parametrized sinc functions of band-pass filters. Experiments demonstrate that SincNet filters can accurately extract key speaker characteristics, such as pitch and formants. Furthermore, [41] introduced an approach that replaces the traditional ASV frontend components—MFCC, VAD, and CMVN—with learnable convolutional layers featuring large strides and kernel sizes, a temporal gating unit, and an instance normalization layer, respectively. Following this, the multi-scale waveform encoder Y-vector [42] leveraged three parallel convolutional branches operating at distinct temporal resolutions to extract speech features directly from the raw input signal. Experimental evaluations on the challenging VoxCeleb1-H and VoxCeleb1-E datasets[1] [43] demonstrate that the Y-vector outperforms other systems using traditional acoustic features.

Recently, self-supervised learning models have demonstrated their ability to extract sophisticated representations from raw audio. SSL enables the generation of labels directly from input data, thereby facilitating training on extensive unlabeled speech datasets to obtain robust representations of human speech. These learned representations have been used to substantially enhance the performance of speech downstream tasks. In the context of ASV, various studies successfully fully adapted SSL models as front-end feature representation for the speaker verification task and outperformed other front-end modules [27, 28]. Notably, by finetuning on the speaker verification loss, the system combining the latest SSL model WavLM and ECAPA-TDNN models outperforms other existing state-of-the-art ASV models by a significant gap [20].

[1]The description of this dataset is presented in Section 2.1.4.

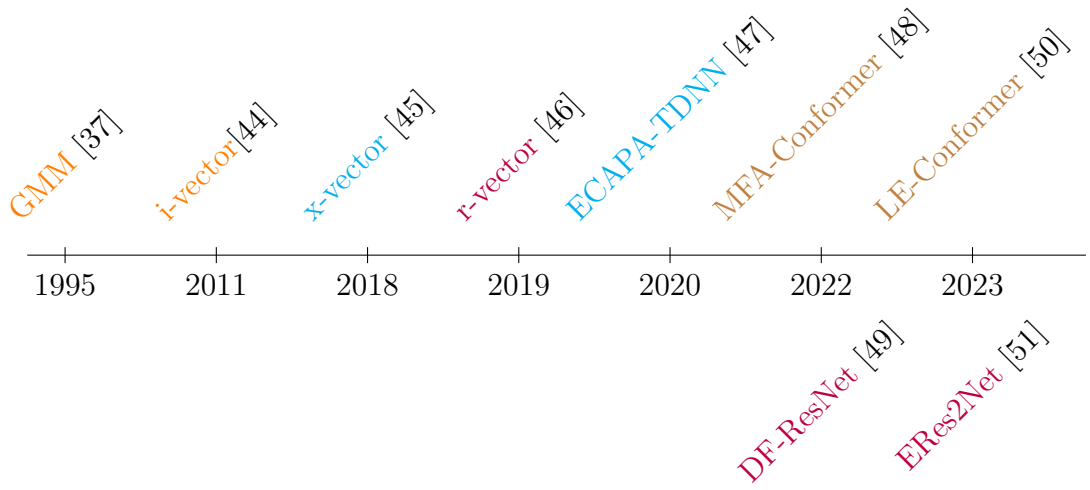


Figure 2.3: The development timeline of highlighted ASV back-end model architectures. The orange, blue, purple, and blue colors denote the statistical, TDNN-based, CNN-based, and transformer-based systems, respectively.

Back-end

The back-end model architecture or speaker extractor of the ASV system plays an important role in converting the acoustic feature in the front-end model into speaker embedding. Figure 2.3 shows the development progress of the ASV speaker extractor with different approaches over time.

Back to the year of 1995, the statistical approach based on the Gaussian Mixture Model (GMM) was first used for ASV [37]. GMM including various Gaussian density functions models speaker-dependent distributions of input speech signals represented by acoustic features, i.e. MFCCs. However, limited training data for each target speaker can hinder the performance of GMM systems. To overcome this, [38] proposed to utilize a GMM-based universal background model (UBM) to train on a larger amount of data with all available speakers, then the parameters of GMM-UBM are adapted to specific target speakers using maximum a posterior adaptation [52]. Based on GMM-UBM, [53, 54] further developed a fixed-length super-vector for speaker verification to address the issue of evaluating variable-length input utterances. The super-vector is created by concatenating the mean values of all GMM components. The existence of super-vector opens the new direction of using back-end classification modules including support vector machines (SVMs) [55] for speaker verification [56, 57]. However, the GMM-UBM super-vector is vulnerable to unseen domain speech (channel, environment) and has a large dimension making it hard to train with the back-end classifier [56].

To address the first problem, [58] proposed joint factor analysis (JFA) [59] to discard channel variabilities and preserve speaker-dependent information in the super-vector.

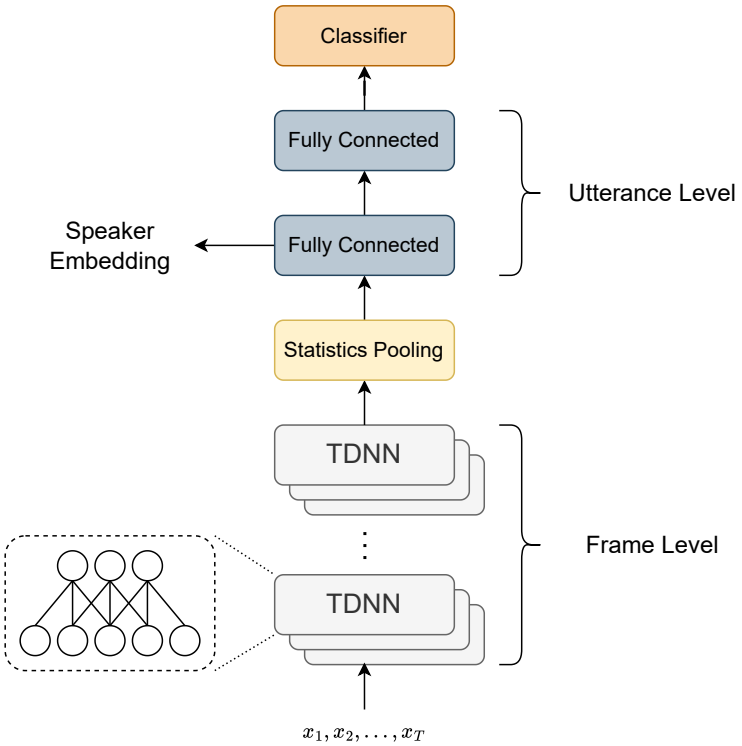


Figure 2.4: The architecture of x-vector model [45].

However, the channel factor also carries speaker-dependent information [44]. A refinement to the JFA method is introduced by [44], combining the variability of the speaker, session, and channel subspaces into a unified subspace, within which a factor known as the i-vector is obtained. Since the i-vector has smaller dimensions and can be used as a feature representation for the back-end classifier, it solves the second problem of large dimension GMM super-vector.

With the powerful ability to extract discriminative representation, the emergence of deep neural networks in ASV starts with d-vector speaker embedding in 2014 [60]. The d-vector model is comprised of several fully connected layers and a pooling layer, which convert variable-length inputs into a fixed-size embedding vector. During training, an input speech is used to train the combination of the d-vector and a classifier module in a supervised manner, enabling the classification of the target speaker among a set of training speakers. In the field of computer vision, Convolutional Neural Networks (CNNs) have been shown highly effective in capturing local dependencies. In speech processing, a variant of CNNs that applies 1-D convolution over time, known as a time-delay neural network (TDNN), excels at capturing local temporal context information along the temporal axis of speech data. To avoid information loss of frame-by-frame speaker modeling with the average aggregation process in d-vector, [45] proposed an x-vector that utilizes a TDNN architecture and statistical pooling layer for speaker verification. Fig.

2.4 illustrates the architecture of the x-vector model. While fully connected layers learn a transformation across the entire input context, the TDNN layer models the input utterance using a narrow temporal window that captures neighboring frames. However, a deeper layer of the multi-layer TDNN can have a wider context, and this helps the model capture information with varying temporal resolution. The output of the last TDNN layer is aggregated by the statistic pooling layer, and fed to fully connected layers to get the final x-vector. x-vector has been the first DNN-based ASV that outperforms the statistic-based i-vector system.

Following the success of the x-vector, ASV systems based on TDNN have gained popularity, with several new systems emerging. For instance, [61] introduced an expanded TDNN structure with an extended temporal context and denser TDNN layers, resulting in increased parameters. In contrast, [62] developed a factorized TDNN to reduce the model size by decomposing the TDNN layers' weight matrix into the product of two low-dimensional matrices. Similarly, [63–65] suggested densely connected TDNN models with both short-term and long-term contexts, using fewer parameters compared to existing TDNN-based models. Additionally, the ECAPA-TDNN model [47] and its variants [66–71] applied an attention mechanism to the TDNN layer, enhancing temporal and channel information capture and subsequently elevating the performance of TDNN-based ASV systems to achieve new state-of-the-art results.

Unlike 1-dimensional CNNs, which slide solely along the temporal dimension in the TDNN layer, 2-dimensional CNNs incorporate convolutions across both time and frequency domains. Inspired by the effectiveness of residual network architecture in image recognition, 2-D CNN ResNet models are commonly employed as the backbone speaker feature [46, 72–74]. There are several attempts to enhance 2-D CNN ASV systems. Similar to 1-DNN ASV literature, attention modules including Squeeze-and-Excitation and convolutional block attention [75] are integrated into the 2-D CNN module to enhance the discrimination of the model [76–80]. Besides, [51, 81–83] shows that multi-scale embedding fusion from different layers of the ResNet model could benefit the ASV performance. Furthermore, [49] showed that by trading off the width of the ResNet model with the deeper model, the ASV performance increases while the number of parameters remains the same. Interestingly, recent work [84] illustrates an insight into the ratio of strides between the temporal and frequency in the 2-D CNN ASV model that can strongly improve the performance.

Given that speech data is a form of sequence data, another direction for ASV involves employing sequence modeling architectures that are capable of capturing long dependencies within the input signal. Recurrent Neural Networks (RNNs) [85] and Long Short-Term Memory (LSTM) models [86] were among the first to be utilized in ASV [87–89].

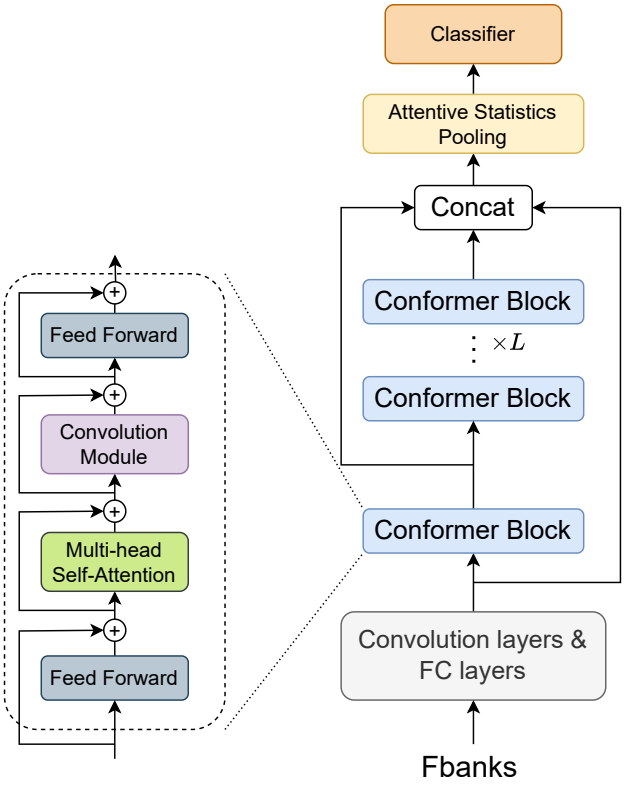


Figure 2.5: The architecture of MFA-Conformer model [48].

However, the training time for RNNs and LSTMs tends to be high due to the sequential nature of the learning algorithm. To overcome this, the transformer model [90] proposed self-attention mechanisms that model input sequences in parallel faster and more efficiently. Transformer models are first adopted as a pooling layer in the ASV system to retain important information when performing a mapping from frame-level to utterance-level embedding [91–93]. Furthermore, various studies that integrate the CNN extractor with the Transformer architecture have been introduced, such as using the stacking Transformer encoders [94–96], or combining Transformers with convolutional layers [48, 50, 97]. Notably, the MFA-Conformer model by [48] is the first to show that Transformer-based ASV systems can outperform CNN-based methods, such as ECAPA-TDNN and ResNet, on the VoxCeleb dataset [43]. This is achieved using the Conformer architecture [98], a convolution-augmented Transformer design. As illustrated in Figure 2.5, MFA-Conformer first projects Fbanks features by convolution and fully connected (FC) layers to form an input for the Conformer model. The Conformer model consists of L Conformer blocks, each Conformer block includes the MHSA, feed-forward module, and the additional Convolutional layer to capture local dependencies within the speech representation. Finally, embeddings from each Conformer block are concatenated and passed through a pooling layer to generate the speaker embedding for the classifier.

2.1.3 Scoring

In ASV systems, the scoring stage is to measure the similarity score between the pair of enrollment and test speaker vectors, enabling the final decision-making [2]. This decision is typically determined by comparing the similarity score to a predefined threshold value. Vector similarity and Probabilistic Linear Discriminant Analysis (PLDA) [99] are the two main scoring methods utilized in modern ASV systems.

Vector Similarity

Vector similarity between two n-length speaker vectors $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{y} \in \mathbb{R}^n$ can be measured by Euclidean Distance and Cosine Similarity [44]. The Euclidean Distance formula is presented below:

$$\|\mathbf{x} - \mathbf{y}\|_2 = \sqrt{\sum_{i=1}^{n} (\mathbf{x}_i - \mathbf{y}_i)^2} \tag{2.1}$$

The similarity of the two vectors is high when the Euclidean Distance is low and vice versa. On the other hand, Cosine Similarity calculates the cosine of the angle formed between two vectors within a multi-dimensional space. This score is derived from the dot product of two speaker embedding vectors that have been normalized to an L2 length, as illustrated in the following equation:

$$\cos(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x} \cdot \mathbf{y}}{\|\mathbf{x}\| \times \|\mathbf{y}\|} = \frac{\sum_{i=1}^{n} \mathbf{x}_i \mathbf{y}_i}{\sqrt{\sum_{i=1}^{n} (\mathbf{x}_i)^2} \sqrt{\sum_{i=1}^{n} (\mathbf{y}_i)^2}} \tag{2.2}$$

The advantage of using vector similarity scoring is its computational efficiency. Notably, the scores obtained through this method remain symmetrical and are not affected by the potential swapping of enrollment and test speaker vectors. Additionally, this approach does not require any trainable parameters in the scoring stage, as the vector similarity score serves as the final output.

Probabilistic Linear Discriminant Analysis

While vector similarity scoring is effective for matched training-test conditions, it may struggle to handle new conditions, often necessitating fine-tuning of the embedding extractor on the new data domain to adapt. An alternative scoring approach involves using trainable PLDA-based scoring. PLDA training typically requires less data and time to fine-tune compared to the entire embedding extractor, making it a more cost-effective option for adapting a speaker ASV system to unseen conditions.

PLDA is a probabilistic adaptation of Linear Discriminant Analysis (LDA). LDA projects the data representation into a lower-dimensional subspace, aiming to increase the separation between different classes (maximizing interclass covariance) while minimizing the dispersion within each class (minimizing intraclass covariance). While LDA is effective in classification tasks where test data originates from known classes, it faces limitations in speaker verification tasks. The aim of speaker verification is to identify if two provided utterances are from the same speaker, even when the model has not previously been exposed to utterances from that specific speaker. Probabilistic LDA addresses this challenge by formulating a speaker embedding w in a typical PLDA configuration as follows:

$$w = m + Vy + z \tag{2.3}$$

where m is the mean of i-vectors, and y denotes the speaker embedding variable with standard typical prior and the residual. z is a residual term following the Gaussian distribution $\mathcal{N}(0, \mathbf{\Sigma})$. PLDA uses the expectation-maximization algorithm to estimate the model parameters (V, Σ). During the verification stage, the verification score between the enrollment vector w_1 and the test vector w_2 is computed by the log-likelihood ratio of hypothesis H_s, assuming both speaker embeddings are accurate representations of the same speaker, and hypothesis H_d, indicating they belong to two distinct speakers. This computation is mathematically represented as:

$$\begin{aligned} PLDA_{score} &= \log \frac{p\left(\mathbf{x}_1, \mathbf{x}_2 \mid H_s\right)}{p\left(\mathbf{x}_1, \mathbf{x}_2 \mid H_d\right)} \\ &= \log \left(\begin{bmatrix} w_1 \\ w_2 \end{bmatrix}; \begin{bmatrix} m \\ m \end{bmatrix}, \begin{bmatrix} \Sigma + VV^T & VV \\ VV & \Sigma + VV^T \end{bmatrix} \right) \\ &- \log \left(\begin{bmatrix} w_1 \\ w_2 \end{bmatrix}; \begin{bmatrix} m \\ m \end{bmatrix}, \begin{bmatrix} \Sigma + VV^T & 0 \\ 0 & \Sigma + VV^T \end{bmatrix} \right) \end{aligned} \tag{2.4}$$

2.1.4 Corpus and Evaluation Benchmark

This section introduces the datasets and metrics used to train and benchmark the proposed ASV systems in this dissertation.

Datasets

Modern ASV systems based on deep learning models typically require extensive training data that includes a large number of speakers and their corresponding labels. Additionally, a diverse test set is essential to more accurately and fairly evaluate the ASV system,

Dataset	Voxceleb1	Voxceleb2
No. utterances	153,516	1,128,246
Total length (hours)	352	2442
Avg length per utterances (s)	8.2	7.8
No. speakers	1251	6112
No. male speakers	690	3761
Avg No. utterances per speaker	116	185

Table 2.1: Dataset statistics of VoxCeleb1 & 2 datasets.

Evaluation subset	No. speakers	No. utterances	No. trials	Description
VoxCeleb1-O	40	4708	37,611	The original evaluation partition of VoxCeleb1 dataset
VoxCeleb1-E	1251	145,160	579,818	Cover the entire VoxCeleb1 dataset, addressing the limited number of speakers in VoxCeleb1-O
VoxCeleb1-H	1190	137,924	550,894	A hard evaluation set, including trial pairs with matching nationality and gender. There are 18 nationality-gender combinations

Table 2.2: VoxCeleb evaluation set [43] includes 3 subset: Voxceleb1-O, Voxceleb1-E, and Voxceleb1-H.

as deep learning models are prone to overfitting.

VoxCeleb [43], addressed these needs, providing a large-scale audio-visual corpus derived from YouTube videos. The VoxCeleb dataset comprises 1,245,525 utterances from 7,245 speakers, representing a multilingual collection with speech from 145 different nationalities. This dataset covers a broad range of age groups, nationalities, languages, and accents. The VoxCeleb audio samples are drawn from diverse sources, such as red-carpet interviews, recordings in both outdoor and indoor settings, public speeches, multimedia content produced by professionals, and amateur recordings captured on handheld devices. Notably, all recordings are exposed to real-world environment conditions, such as ambient conversation, giggling, music, concurrent speech, and room acoustics, in addition to variations in recording devices and channel interference. Consequently, VoxCeleb serves as an "in-the-wild" corpus featuring realistic, noisy, and unconstrained conditions. VoxCeleb has two partitions including VoxCeleb1 and VoxCeleb2 which were released in 2017 and 2018, respectively. VoxCeleb2 is used for the training set and VoxCeleb1 is used for evaluation. The detailed statistic of VoxCeleb1 & 2 is shown in Table. 2.1. When the VoxCeleb2 corpus is employed as the training data, the VoxCeleb1 dataset can serve as an evaluation set and be further partitioned into three distinct subsets as illustrated in Table 2.2.

To improve both the variety and quantity of training data, data augmentation methods are employed, incorporating reverberation and additive noise. Reverberation involves adding realistic echoes and reflections to audio by convolving it with room impulse responses (RIRs). [100] proposed a dataset of RIRs simulated under various room conditions, which includes a mix of 325 real and simulated RIRs. Real RIRs are typically recorded in actual rooms using specialized equipment to capture acoustic characteristics such as reflections, reverberation, and absorption by different surfaces. Conversely, simulated RIRs are generated using mathematical models that approximate sound behavior based on room dimensions, absorption coefficients, and other properties. For each of the three room sizes including small, medium, and large, 200 rooms are sampled, with 100 RIRs sampled per room based on different speaker and receiver positions.

For additive noise, the MUSAN dataset [101] is renowned for its wide variety of noise types and extensive data collection. The MUSAN dataset comprises 933 types of ambient sounds, music spanning various genres, and 60 hours of human speech in 12 languages. It is widely used for assessing the robustness of speaker verification models against noisy speech [102, 103]. The dataset consists of three types of noisy signals from real-world recordings:

- General noise: spanning 6 hours, includes technical sounds like dual-tone multi-frequency signals, dial tones, and machine noises, along with environmental background sounds such as wind, footsteps, rain, and animal sounds.
- Music: features 42 hours of popular Western music genres.
- Babble: includes 20 hours of reading speech and 40 hours of recordings from US government hearings, panels, and discussions.

Evaluation Metrics

To evaluate ASV systems, two primary error types are considered: false rejection (FR) and false acceptance (FA). False rejection occurs when legitimate speakers are incorrectly rejected, while false acceptance happens when an impostor is mistakenly granted access. These errors are often known as miss errors and false alarms, respectively. Given a list of trials in the evaluation stage, the false reject rate (FRR) is calculated based on the proportion of legitimate attempts that result in FR errors, and the false accept rate (FAR) is determined by the proportion of impostor attempts that result in FA errors. An effective automatic speaker verification system aims to minimize both FAR and FRR. However, since the output of an ASV system is a scalar score representing the similarity between two utterances, the decision threshold becomes crucial. If the threshold is set

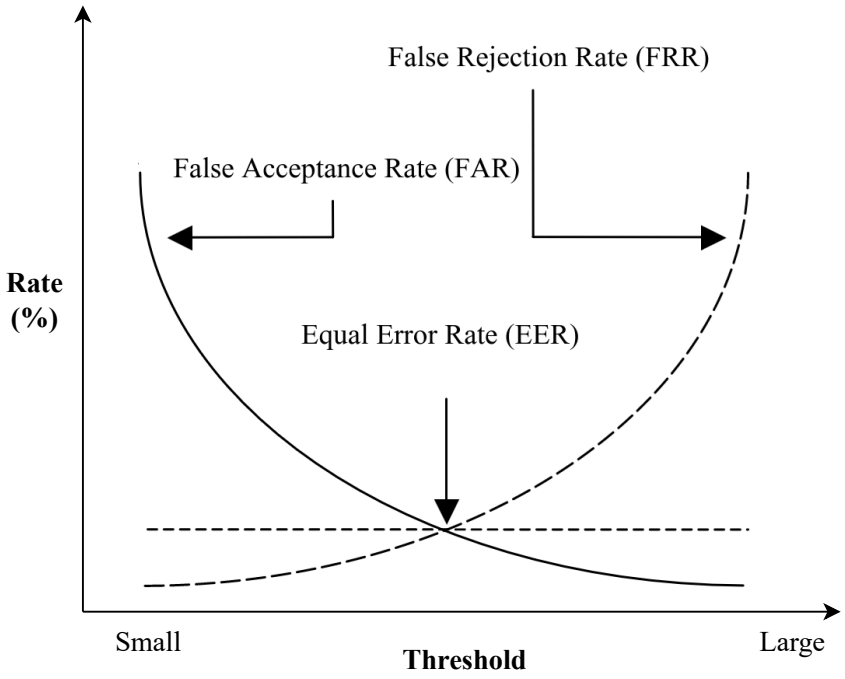


Figure 2.6: Relationship between FAR, FRR, and Equal Error Rate (EER) (adapted from [104]).

too high, it increases FR errors; if it is set too low, it increases FA errors. Consequently, there is always a tradeoff between the two error rates when attempting to reduce one. By adjusting the threshold, a point can be reached where FAR equals FRR, known as the equal error rate (EER). The EER is a key metric in ASV systems as it reflects both the error rate and its threshold independence. The illustration of EER and the trade-off between FAR and FRR are shown in Figure 2.6.

While the EER is simple to compute, it may not be the best metric for practical scenarios because it does not differentiate between false acceptance and false rejection errors. To address this, the Detection Cost Function (DCF) is introduced, which quantifies the numerical costs and penalties associated with both error types [105]. It is defined as a weighted sum of the FRR and FAR at a specific decision threshold τ, as shown below:

$$\text{DCF} = C_{FR} P_{FR} P_{\text{Target}} + C_{FA} P_{FA} \left(1 - P_{\text{Target}}\right)$$

where C_{FR} and C_{FA} are the costs of the detection errors, P_{FR} is the probability of (FR|Target, Threshold = τ), P_{FA} is the probability of (FA|Nontarget, Threshold = τ), and P_{Target} is the prior probability of target speakers. The values of C_{FR}, C_{FA}, and P_{Target} are predefined and can be adjusted based on the importance of FA and FR er-

Approaches	Techniques	References	Advantages	Disadvantages
Model Compensation	Speech Enhancement	[14, 15], [17], [106–115]	- straightforward and obtain the clean speech as output - learn the relationship between noisy and clean speech	- could lose speaker characteristic - overfitting to the training noise
	Model-domain Adaptation	[16, 116–123]	- higher performance - flexible to different noisy environments	- additional computational cost - require a considerable quantity of adaptation data
Feature Compensation	Data Augmentation	[12, 13], [124–129]	- no requirement to modify the model structure - easy to be implemented	- require much data, otherwise easy overfitted
	Feature-Domain Adaptation	[8–11], [130–132]	- no requirement to modify the model structure - flexible to different noisy environments	- overfitting to the training noise

Table 2.3: Summarize noise-robust ASV methods

rors in different contexts. For instance, in a banking voice authentication system, it's crucial to tightly regulate access to bank accounts, prioritizing minimal false acceptance rates. Therefore, the value of C_{FA} can be increased accordingly. In practice, performance metrics such as the minimum Detection Cost Function (minDCF), determined through threshold variation, and the actual DCF (actDCF), established using application-specific thresholds, are commonly utilized.

2.2 Current Approaches for Automatic Robust Speaker Verification

In real-world applications, ASV systems encounter several challenges. One notable issue is that, in environments with noise and reverberation, the ASV system's performance can degrade significantly [2, 6, 7]. This section presents a broad survey of recent research methodologies aimed at improving the robustness of speaker verification systems across diverse environmental conditions. Since the degradation of the ASV models comes from the mismatch in speech condition between the training data and the testing data, the robust ASV methods can be broadly classified into two categories: (i) feature compensation, which involves adapting features extracted in the testing environment to align with those in the training environment, or vice versa, and (ii) model compensation, where speaker model parameters are adjusted to accommodate variations in the noisy input speech [31]. Before diving into the details of each category in the following sections, Table 2.3 summarizes the strengths and weaknesses of both categories.

2.2.1 Feature Compensation

Feature compensation approaches typically avoid altering the number of parameters in the acoustic model. Instead, they adjust speaker features in the test set to align with speaker features in the training one. Due to their independence from the acoustic model, these methods typically do not necessitate additional parameters in the model size [133]. Feature-space techniques can be further divided into two subtypes: data augmentation and feature-domain adaptation.

Data augmentation

To mitigate the overfitting issue and enhance the noise resilience of deep neural network ASV systems, employing large-scale multi-condition training data proves to be an effective and straightforward approach. However, collecting large-scale speaker verification data can be manually extensive. Hence, one approach for preparing large multi-condition training data is to utilize data augmentation techniques on existing datasets.

By training on augmented data consisting of additive noise and reverberation, the x-vector is able to lower EER the corresponding baselines [124]. Another useful augmentation method is Mixup [125] which constructs additional training samples by linearly interpolating random pairs of samples from the original training data and their labels. [12] adapted the mixup strategy to the training procedure of x-vector and obtained significant improvement without using any extra data sources. Instead of augmenting on signal level, SpecAugment [126] created occluded spectrum distorts of the input speech by randomly masking both the temporal and frequency dimensions. [13] show that the SpecAugment method can help the generalization ability of ASV systems by outperforming the baseline models on several datasets. To reduce the information loss by masking in SpecAugment, GuidedMix [127] proposed replacing the masked area with a patch from a different spectrum. This ensures that the essential discriminative regions are preserved and that the patches remain meaningful after data augmentation. Experiments performed on VoxCeleb reveal that GuidedMix enhances performance across diverse testing scenarios including clean, noisy, and short-duration conditions. Since the augmentation parameters of data augmentation methods are mostly pre-defined, [128] recently adopted a population-based augmentation technique [129] to optimize augmentation parameters. This method outperforms manual parameter setting by learning to adjust the augmentation hyperparameters.

Feature-Domain Adaptation

Instead of having an additional speech enhancement module to obtain clean speech from noisy speech, the feature-domain adaptation approach directly adapts the representation of the speaker extractor to the distorted speech input. Feature-domain adaptation can be done by aligning the distribution shift between the clean and noisy speech input during the training.

Several studies have employed statistical techniques to align the distribution between the source and target domains, utilizing methods such as correlation alignment [8, 9]. Correlation alignment shifts covariance matrices of out-of-domain features with those of in-domain features in an unsupervised manner, without necessitating class labels. Subsequently, [10] developed a speaker extractor trained to simultaneously minimize the cross-entropy of the speaker classifier and the mean square error (MSE) between the speaker embedding and an optimal clean x-vector extracted using a pre-trained system. Similarly, [11] introduced the Barlow Twin loss function, which enhances the similarity of speaker embeddings between clean and noisy versions of the same input signal to achieve invariant representation. [130] also introduces a within-sample variability-invariant loss to constrain embeddings extracted from clean utterances and their noisy counterparts. This loss encourages consistency in embeddings across clean utterances and their noisy versions, preventing the network from incorporating unwanted noises or variations into the speaker embedding.

Instead of aligning the embedding levels of clean and noisy inputs, [131] proposed a novel gradient regularization method to reduce speaker-irrelevant noise. This method aligns the gradients of noisy utterances with those of their clean counterparts and ensures that gradients across different types of noise point in a similar direction, hence it prevents the speaker model from encoding irrelevant noisy information. Furthermore, [132] employed a knowledge distillation technique to transfer knowledge from robust teacher models to student ones, minimizing the feature-level and instance-level distance between them.

2.2.2 Model Compensation

Unlike feature-domain methods, model-domain approaches involve modifying the parameters of the acoustic model to account for noise effects [134]. While these methods generally achieve higher accuracy, they typically come with higher computational costs [135]. Model compensation methods for noise-robust ASV can consist of two main approaches: model-domain adaptation and speech enhancement.

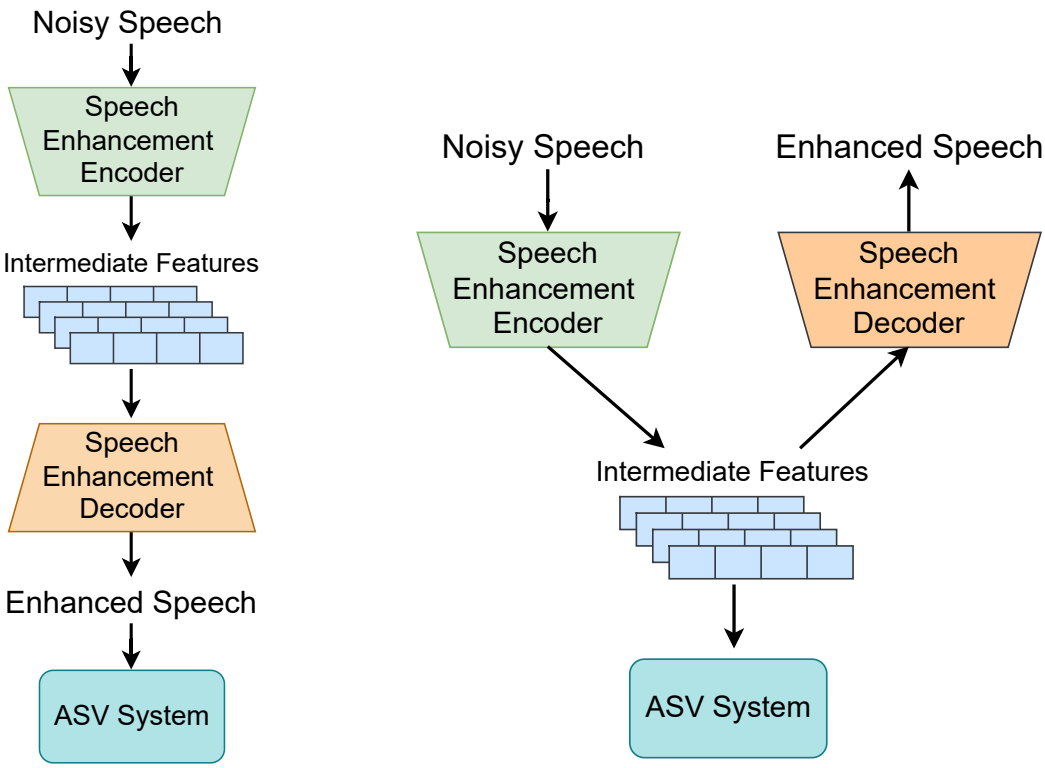
Model-domain Adaptation

Model-domain adaptation approach is to add parameters or additional layers to the acoustic model for a better adaptation of unseen speech conditions. Statistical adapters are widely used in the early age of robust speaker verification. For example, [116] and [117] employed supervised and unsupervised versions of Bayesian adaptation to transfer the model parameter to the target domain. Later, [118] proposed an additional hybrid layer combining denoising autoencoders (DAE) and maximum a posteriori (MAP) on x-vector feature space. The noisy x-vector undergoes denoising through DAE, followed by additional denoising using MAP on the DAE's output. Recently, [119] proposed two types of learnable domain adapters: the Block Domain Adapter and the Embedding Domain Adapter, which attach to different blocks of the model's architecture and final speaker embedding, respectively. These adapters are lightweight, highly adaptable at various levels, and with different backbone architectures. Experiments show that both adapters are effectively generalized to previously unseen domains.

Another strategy for model-domain adaptation involves adversarial learning, where the speaker extractor is trained to extract an intermediate representation that remains invariant to shifts across various speech conditions. This can be achieved by employing a dual-objective training process where the model learns to focus on speaker-specific features while being adversarially trained to disregard noise-related variations, thereby improving performance in noisy environments. Initially, [120–122] proposed an adversarial speaker verification scheme where an encoder for noise-robust speaker embeddings, a classifier for speaker identification, and a discriminator for noise type classification are jointly optimized. This method reduces speaker classification loss while simultaneously increasing noise classification loss, thus learning condition-invariant embeddings. In contrast to previous methods, [16] introduce an unsupervised adversarial invariance framework. This architecture separates speaker-related features from irrelevant factors without requiring supervision, resulting in significant enhancements in speaker verification performance under noisy environments. Instead of directly applying adversarial training to the feature representation that can contain domain information, [123] has separated encoders for the feature representation of source and target domains. Subsequently, adversarial training is conducted within a shared feature space, aiming to promote condition invariance of the speaker embedding.

Speech Enhancement

To tackle distortions in input speech, a logical solution is to incorporate a speech enhancement module into the ASV system. This module can either be put in front of the



(a) Cascade system.

(b) Joint intermediate embedding.

Figure 2.7: The two types of integration of speech enhancement and automatic speaker verification systems.

ASV system to enhance the quality and clarity of the speech signal (cascade system) or its intermediate representation is utilized for the speaker verification task (joint intermediate representation), as depicted in Figure 2.7.

In terms of signal-level speech enhancement, traditional methods often rely on signal analysis techniques such as computational auditory scene analysis [106] and cepstral analysis [107]. Additionally, various robust speech features are employed alongside normalization techniques like cepstral mean and variance normalization [108] or short-time gaussianization [109]. Another common signal-level speech enhancement technique is pre-emphasis, which boosts high-frequency components over low-frequency ones to improve signal quality. This method increases the energy of high-frequency signals, making them more robust compared to high-frequency noise components, thereby improving the signal-to-noise ratio.

More recently, deep neural networks have shown promise in speech enhancement for robust ASV [7]. These networks are trained to learn non-linear mapping functions that transform noisy speech to clean speech by optimizing model parameters to minimize discrepancies between clean and corrupted features or signals [15, 110]. A similar approach is to train a DNN to learn how to mask distorted segments across both the temporal and

frequency domains of the input speech signal [14, 111, 112]. This typically involves using either hard binary-based masks or soft ratio-based masks. Similar to mapping-based speech enhancement, models for mask-based speech enhancement can be trained using mean square error loss to minimize differences between the estimated clean spectrum and the actual clean spectrum. Apart from model architecture, [113] proposed a novel VoiceID loss for joint training speech enhancement and ASV system. The VoiceID loss is to learn a ratio mask that enables the ASV model to predict the same speaker between the noisy and masked speech. In addition to the reconstruction loss method, generative adversarial networks (GANs) have been utilized to learn a speech enhancement mapping for speaker verification [17, 114]. In this setup, the GAN comprises a generator and a discriminator, trained adversarially. The generator learns to create the cleaned spectrum from the noisy spectrum input, while the discriminator's objective is to distinguish the cleaned spectrum from its ground-truth clean counterpart. This enhances the generator's ability to generate high-quality cleaned speech.

While [14, 111, 112] proposed masking-based speech enhancement models that are trained independently with the ASV system, ExU-Net [103] employed joint training of speech enhancement and speaker verification. This approach aims to avoid the issue of losing speaker information during the speech enhancement process, as mentioned in [14]. Similar to Fig. 2.7b, instead of using SE as a front end to clean signal level input, ExU-Net is proposed to use the intermediate feature of the SE module to simultaneously optimize speaker verification and feature enhancement losses. Similarly, [136] decouples the speech extractor into a speaker encoder and reconstruction module, where the speaker encoder is responsible for extracting speaker-related information, while the reconstruction module enhances the model's capacity to control noise elements. Recently, diffusion probabilistic models (DPMs) have emerged as powerful tools for mitigating noise in speech enhancement. This advancement has resulted in Diff-SV [137], an extension of the ExU-Net model that utilizes diffusion probabilistic models. As shown in Table 2.4, it demonstrates state-of-the-art performance in noise robustness, outperforming existing methods on the VoxCeleb 1 dataset in multiple signal-to-noise ratios (SRN) of adding MUSAN noise [101] to the testing utterances.

Noise type	SNR	VoiceID [113]	Wu et al. [112]	Sun et al. [136]	Cai et al. [130]	ExU-Net [103]	Diff-SV [137]
Clean		6.79	7.6	2.90	3.12	2.76	**2.35**
Babble	0	37.96	20.11	10.96	11.78	9.57	**8.74**
	5	27.12	12.02	6.13	5.97	5.52	**4.51**
	10	16.66	9.63	4.28	4.44	4.06	**3.33**
	15	11.25	8.48	3.52	3.73	3.28	**2.82**
	20	8.99	7.99	3.21	3.36	2.99	**2.61**
Music	0	16.24	12.92	10.84	7.79	7.35	**6.04**
	5	11.44	10.1	6.52	5.23	4.9	**3.96**
	10	9.13	8.95	4.66	4.11	3.69	**3.10**
	15	8.10	8.35	3.67	3.63	3.14	**2.75**
	20	7.48	7.95	3.21	3.30	2.93	**2.60**
Noise	0	16.56	13.12	10.24	7.34	6.8	**6.01**
	5	12.26	10.57	6.96	5.65	5.23	**4.52**
	10	9.86	9.28	5.02	4.35	4.07	**3.49**
	15	8.69	8.59	3.91	3.85	3.39	**2.93**
	20	7.83	8.1	3.40	3.44	3.1	**2.64**
Average EER		13.52	10.24	5.59	5.07	4.55	**3.90**

Table 2.4: Current state-of-the-art results of noise-robust ASV models on VoxCeleb 1 test set. Different SNR levels of MUSAN noisy audio are added to the original utterance to assess the robustness of ASV models across varying noise levels.

Chapter 3

Improving the Robustness of Automatic Speaker Verification via Knowledge Distillation

3.1 Introduction

Automatic speaker verification (ASV) is the process of authenticating an individual's claimed identity based on voice characteristics. By leveraging large-scale neural networks trained on abundant unlabelled speech data, self-supervised learning (SSL) models have revolutionized various speech processing tasks [21–26], including ASV [29]. Specifically, the recent SSL WavLM model [20] is designed to train on noisy input using target pseudo-labels generated from clean speech. This approach enables WavLM to capture robust representations despite noisy input. When fine-tuned for speaker verification, the combination of WavLM and ECAPA-TDNN models has achieved state-of-the-art results on the VoxCeleb dataset. However, this model is computationally expensive. To better utilize SSL models, knowledge distillation can be employed to transfer the robust speech representation to smaller student models. In ASV, KD encompasses two common approaches: one is embedding-level method [138–141], which attempts to make student models mimic the teacher's intermediate feature embedding by reducing the distance between representation spaces; the other is label-level method [138, 142], which focuses on minimizing the Kullback–Leibler divergence between the output probabilities of the teacher and student networks.

In the training step of an ASV model, the objective is to classify input speech into target speaker (the ground-truth speaker) and avoid assigning it to non-target speakers (incorrect speakers). While the importance of the target speaker is evident, non-target speakers can also enhance the model's discriminability since there would be nu-

merous non-target speakers sharing similar voice characteristics with the target speaker. In past studies, [143] compared ASV models performance trained on two training sets with different numbers of speakers but the same number of utterances, and found that a larger number of speakers improved the performance. Similarly, in face recognition, [144] also observed that an increasing number of training non-target classes improved model performance within a fixed-size training set. Building on these observations, we hypothesize that integrating knowledge from non-target speakers can enhance ASV model performance. However, the conventional label-level KD considers correlations among the teacher's output probabilities of all speakers, the importance of non-target speakers' probabilities can be overshadowed by the target speaker with high classification confidence in the teacher model. Based on this hypothesis, the conventional label-level KD approach for ASV can be improved by emphasizing the knowledge of non-target speakers.

To validate the assumptions above, this paper initially shows an experiment illustrating the importance of non-target speakers in ASV. When the number of training utterances remains the same, we observe that an increasing number of non-target training speakers leads to better results. Based on this observation, we investigate the significance of non-target speakers in the conventional label-level KD for ASV models. Following Decoupled Knowledge Distillation (DKD) [145], we segregate the output classification probabilities of the teacher and student models into two distinct probabilities of target and non-target speakers. Subsequently, the probabilities of non-target speakers are emphasized during KD using a specific weight. We utilize the large-scale SSL model WavLM-TDNN [20] as our teacher model and employ three different network architectures: x-vector [45], ResNet34 [46], and CAM++ [65] as student models. Our experiments show that DKD with an emphasis on the non-target speakers' output probabilities, outperforms both embedding-level and conventional label-level KD methods across student models.

3.2 Methodology

3.2.1 The impact of non-target speakers for ASV

To validate the hypothesis that a larger set of non-target speakers benefits ASV models, we conducted a toy experiment. We trained the x-vector model using a fixed 100,000 training utterances of the VoxCeleb 2 dev set [43]. These training utterances are evenly distributed among each training speaker, hence increasing the number of speakers will lead to fewer training utterances per speaker. As depicted in Figure 3.1, the performance of the x-vector model consistently improves with an increasing number of speakers in the

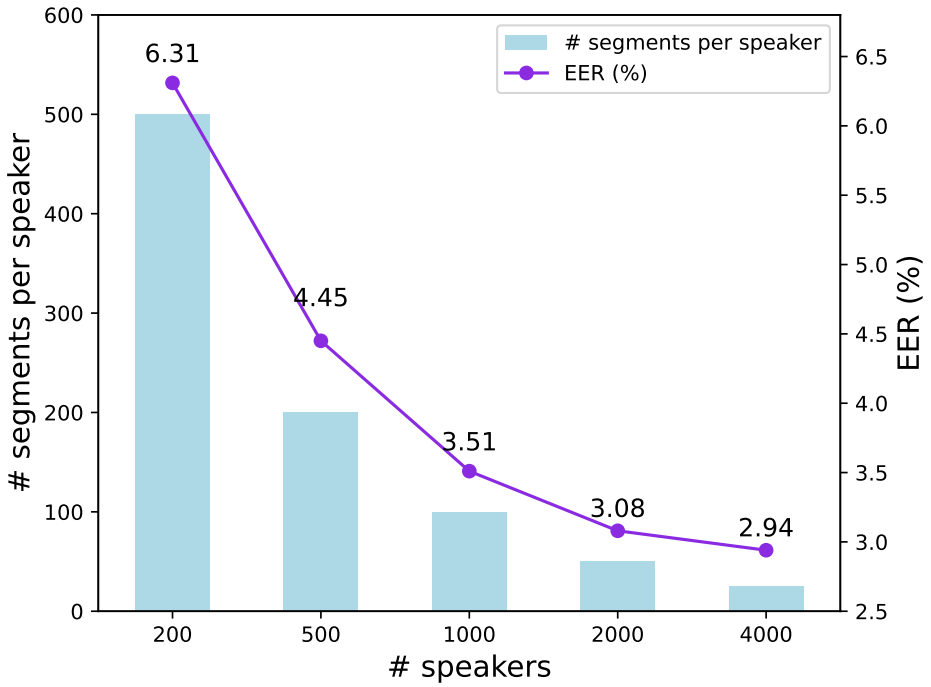


Figure 3.1: Vox1-O results (EER %) of x-vector model trained on a fixed number of utterances but varying numbers of speakers.

training set. This indicates that involving more non-target speakers enhances the model's ability to distinguish the target speaker from others. Inspired by this finding, we further extract and emphasize non-target speaker knowledge during the knowledge distillation process.

3.2.2 Rethinking conventional label-level KD

Following the reformulation of the conventional label-level KD for the computer vision task in [145], we interpret the conventional label-level KD loss for automatic speaker verification. In the training phase of the ASV model, the model's output is classification probabilities $\mathbf{p}$ over the set $\mathcal{K}$ of K training speakers, in which the probability p_i of the i-th speaker is computed using the softmax function to transform the logits vector z_i into a probability distribution as follows:

$$p_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{3.1}$$

In the conventional label-level KD, the student model tries to mimic the teacher model by minimizing the Kullback-Leibler Divergence D_{KL} between the student ($\mathcal{S}$) and teacher

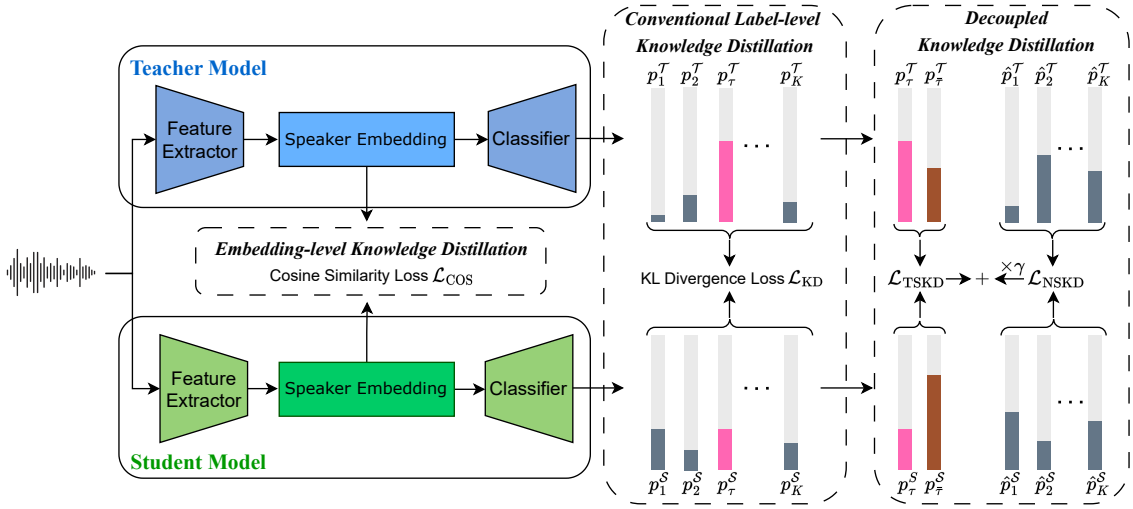


Figure 3.2: Our Decoupled Knowledge Distillation (DKD) with an emphasis on non-target speaker knowledge in comparison with the embedding-level knowledge distillation (using cosine distance loss $\mathcal{L}_{\mathrm{COS}}$) and the conventional label-level knowledge distillation (using Kullback–Leibler divergence loss $\mathcal{L}_{\mathrm{KD}}$). $\mathcal{T}$, $\mathcal{S}$, K, and τ denote the teacher model, the student model, the number of training speakers, and the target speaker, respectively. p_i, $p_{\bar{\tau}}$, and $\hat{p}_i$ are respectively defined as Eq.(3.1) and Eq.(3.4). $\mathcal{L}_{\mathrm{TSKD}}$, $\mathcal{L}_{\mathrm{NSKD}}$ and γ are defined as Eq.(3.6) and Eq.(3.8), respectively.

($\mathcal{T}$) output probability distributions. The D_{KL} loss is defined as:

$$\mathcal{L}_{\mathrm{KD}} = D_{\mathrm{KL}}(\mathbf{p}^{\mathcal{T}} \| \mathbf{p}^{\mathcal{S}}) = \sum_{i \in K} p_i^{\mathcal{T}} \log(\frac{p_i^{\mathcal{T}}}{p_i^{\mathcal{S}}}) \tag{3.2}$$

where $\mathbf{p}^{\mathcal{T}}, \mathbf{p}^{\mathcal{S}}$ denote the output probabilities of the teacher and student networks, respectively. We further split the set $\mathcal{K}$ of indexes $\{i = 1 \dots K\}$ in $\mathcal{L}_{\mathrm{KD}}$ into the target speaker τ and a set of non-target speakers $\mathcal{K} \backslash \{\tau\}$ as:

$$\mathcal{L}_{\mathrm{KD}} = p_\tau^{\mathcal{T}} \log(\frac{p_\tau^{\mathcal{T}}}{p_\tau^{\mathcal{S}}}) + \sum_{i \in \mathcal{K} \backslash \{\tau\}} p_i^{\mathcal{T}} \log(\frac{p_i^{\mathcal{T}}}{p_i^{\mathcal{S}}}) \tag{3.3}$$

We define the probability of classifying a speaker belonging to $\mathcal{K} \backslash \{\tau\}$ as $p_{\bar{\tau}}$, and the probability of predicting a specific non-target speaker $i \neq \tau$ over all non-target speakers as $\hat{p}_i$:

$$p_{\bar{\tau}} = \frac{\sum_{i \in \mathcal{K} \backslash \{\tau\}} e^{z_i}}{\sum_{j \in K} e^{z_j}}, \quad \hat{p}_i = \frac{e^{z_i}}{\sum_{j \in \mathcal{K} \backslash \{\tau\}} e^{z_j}} \tag{3.4}$$

From (3.1) and (3.4), we replace $p_i = p_{\bar{\tau}}\hat{p}_i$ in (3.3):

$$\begin{aligned}\mathcal{L}_{\text{KD}} &= p_\tau^{\mathcal{T}} \log(\frac{p_\tau^{\mathcal{T}}}{p_\tau^{\mathcal{S}}}) + p_{\bar{\tau}}^{\mathcal{T}} \sum_{i \in \mathcal{K}\setminus\{\tau\}} \hat{p}_i^{\mathcal{T}} \log(\frac{p_{\bar{\tau}}^{\mathcal{T}} \hat{p}_i^{\mathcal{T}}}{p_{\bar{\tau}}^{\mathcal{S}} \hat{p}_i^{\mathcal{S}}}) \\ &= p_\tau^{\mathcal{T}} \log(\frac{p_\tau^{\mathcal{T}}}{p_\tau^{\mathcal{S}}}) + p_{\bar{\tau}}^{\mathcal{T}} \sum_{i \in \mathcal{K}\setminus\{\tau\}} \hat{p}_i^{\mathcal{T}} \log(\frac{p_{\bar{\tau}}^{\mathcal{T}}}{p_{\bar{\tau}}^{\mathcal{S}}}) \\ &+ p_{\bar{\tau}}^{\mathcal{T}} \sum_{i \in \mathcal{K}\setminus\{\tau\}} \hat{p}_i^{\mathcal{T}} \log(\frac{\hat{p}_i^{\mathcal{T}}}{\hat{p}_i^{\mathcal{S}}}) \end{aligned} \tag{3.5}$$

Since $p_{\bar{\tau}}^{\mathcal{T}}$, $p_{\bar{\tau}}^{\mathcal{S}}$ are independent to the class index i and $\sum_{i \in \mathcal{K}\setminus\{\tau\}} \hat{p}_i^{\mathcal{T}} = 1$, we can simplify (3.5) to:

$$\mathcal{L}_{\text{KD}} = \underbrace{p_\tau^{\mathcal{T}} \log(\frac{p_\tau^{\mathcal{T}}}{p_\tau^{\mathcal{S}}}) + p_{\bar{\tau}}^{\mathcal{T}} \log(\frac{p_{\bar{\tau}}^{\mathcal{T}}}{p_{\bar{\tau}}^{\mathcal{S}}})}_{D_{\text{KL}}(\mathbf{b}^{\mathcal{T}} \| \mathbf{b}^{\mathcal{S}})} + p_{\bar{\tau}}^{\mathcal{T}} \underbrace{\sum_{i \in \mathcal{K}\setminus\{\tau\}} \hat{p}_i^{\mathcal{T}} \log(\frac{\hat{p}_i^{\mathcal{T}}}{\hat{p}_i^{\mathcal{S}}})}_{D_{\text{KL}}(\hat{\mathbf{p}}^{\mathcal{T}} \| \hat{\mathbf{p}}^{\mathcal{S}})} \tag{3.6}$$

From (3.6), the conventional label-level KD can be re-formulated into the sum of two terms: 1) Target Speaker Knowledge Distillation (TSKD) loss $\mathcal{L}_{\text{TSKD}}$: the D_{KL} over the binary classification probability $\mathbf{b} \in \mathbb{R}^2$ of the target speaker and all non-target speakers, and 2) Non-Target Speaker Knowledge Distillation (NSKD) loss $\mathcal{L}_{\text{NSKD}}$: the D_{KL} of the multi-class classification probability $\hat{\mathbf{p}} \in \mathbb{R}^{K-1}$ between $K-1$ non-target speakers as shown in (3.7) and Fig.3.2.

$$D_{\text{KL}}(\mathbf{p}^{\mathcal{T}} \| \mathbf{p}^{\mathcal{S}}) = \underbrace{D_{\text{KL}}(\mathbf{b}^{\mathcal{T}} \| \mathbf{b}^{\mathcal{S}})}_{\mathcal{L}_{\text{TSKD}}} + (1 - p_\tau^{\mathcal{T}}) \underbrace{D_{\text{KL}}(\hat{\mathbf{p}}^{\mathcal{T}} \| \hat{\mathbf{p}}^{\mathcal{S}})}_{\mathcal{L}_{\text{NSKD}}} \tag{3.7}$$

From the above equation, when the teacher model predicts the target speaker accurately, a large value of $p_\tau^{\mathcal{T}}$ results in a smaller $(1 - p_\tau^{\mathcal{T}})$, which leads to the suppression of $\mathcal{L}_{\text{NSKD}}$. This could potentially hinder the distillation of knowledge from non-target speakers in the label-level KD method.

3.2.3 Decoupled Knowledge Distillation with an emphasis on non-target speaker knowledge

In Section 3.2.1, it was demonstrated that leveraging more non-target speaker knowledge can enhance the performance of ASV models. In other words, $\mathcal{L}_{\text{NSKD}}$ may play a crucial role in the knowledge transfer from the teacher to student models. Decoupled Knowledge Distillation (DKD) [145] proposed a modification to remove the dependency factor $(1-p_\tau^{\mathcal{T}})$ in (3.7) by introducing hyperparameters to balance the $\mathcal{L}_{\text{TSKD}}$ and $\mathcal{L}_{\text{NSKD}}$. However, to

place a greater emphasis on $\mathcal{L}_{\text{NSKD}}$, we adjusted the original DKD method by simply replacing $(1 - p_\tau^{\mathcal{T}})$ with the hyperparameter γ in the following manner:

$$\mathcal{L}_{\text{DKD}} = \mathcal{L}_{\text{TSKD}} + \gamma \mathcal{L}_{\text{NSKD}} \tag{3.8}$$

Finally, the DKD loss $\mathcal{L}_{\text{DKD}}$ is combined with the classification loss to optimize the student model. Fig. 3.2 illustrates the comparison between DKD with an emphasis on non-target speaker knowledge, embedding-level, and conventional label-level knowledge distillation.

3.3 Experiment settings and results

3.3.1 Experiment settings

Dataset

We utilized the VoxCeleb2 dev dataset [43] for training and evaluated the performance on three test trials, Vox1-O, Vox1-E, and Vox1-H. During training, we applied data augmentation using the MUSAN noise corpus [101] and RIRs reverberation [100], with a probability of 0.6.

Model

The teacher model is the SSL-based ASV system [20] combining WavLM Large and ECAPA-TDNN [47]. On the other hand, we utilized various network architectures for our student models including TDNN-based x-vector [45], CNN-based ResNet-34 [46], and D-TDNN-based CAM++ [65]. The WavLM Large part is first pre-trained on a large scale of unlabelled data in an SSL manner and the entire system is then fine-tuned with the VoxCeleb2 dev set. The feature dimension of embeddings in each model remained unchanged, except in the embedding-level KD experiment where it was specifically set to 256 to match the corresponding embeddings of the teacher model.

Training and Evaluation

During the training, each audio sample was randomly cropped to a 2-second segment, then 80-dimensional Fbank features were extracted using a frame length of 25 ms and a frameshift of 10 ms. For the classification loss function, we employed the AAM-softmax [146] with a scale of 32 and a margin scheduler. In the proposed KD method, from the ablation study in Section 3.3.2, the value γ in Equation. (3.8) is set to 2.0 in all the remaining experiments. For evaluation, speaker embeddings were scored using cosine

Table 3.1: Results on the VoxCeleb1 test sets. *COS* and *KLD* denote embedding-level and conventional label-level KD

System	Params (M)	FLOPs (G)	Distillation Method	EER (%) / minDCF Vox1-O	Vox1-E	Vox1-H
Teacher model WavLM-TDNN [20]	316.62	∼26	-	0.383 / -	0.480 / -	0.986 / -
TDNN-based Student model x-vector [45]	4.61	0.53	-	1.835 / -	1.822 / -	3.110 / -
			COS	1.760 / 0.189	1.742 / 0.185	2.879 / 0.255
			KLD	1.585 / 0.171	1.589 / 0.171	2.704 / **0.244**
			Ours	**1.319** / **0.160**	**1.388** / **0.155**	**2.440** / 0.226
CNN-based Student model ResNet34 [46]	6.64	4.55	-	0.862 / 0.089	1.035 / 0.112	1.827 / 0.176
			COS	0.829 / 0.088	0.943 / 0.107	1.694 / 0.164
			KLD	0.771 / **0.086**	0.939 / 0.103	1.728 / 0.166
			Ours	**0.766** / 0.101	**0.850** / **0.096**	**1.615** / **0.161**
D-TDNN-based Student model CAM++[65]	7.18	1.72	-	0.718 / -	0.879 / -	1.735 / -
			COS	0.713 / 0.118	0.901 / 0.108	1.768 / 0.182
			KLD	0.633 / **0.101**	0.790 / 0.093	1.572 / 0.159
			Ours	**0.590** / 0.118	**0.735** / **0.085**	**1.494** / **0.148**

similarity and score normalization. Performance is reported on two metrics: Equal Error Rate (EER) and the minimum of the normalized detection cost function (MinDCF) with $P_{target} = 0.01$ and $C_{fa} = C_{miss} = 1$. All experiments are conducted using *Wespeaker* toolkit [147]. Stochastic gradient descent optimization with a cosine annealing scheduler and linear warm-up scheduler is utilized. The learning rate ranged from 0.1 to 1e-4, while the momentum was set to 0.9 and weight decay was applied at a rate of 1e-4

3.3.2 Results and Analysis

Results of the proposed method

Table 3.1 presents a comparison of the performance of teacher and student models trained solely with classification loss, along with the results of different knowledge distillation methods. Although both embedding-level and conventional label-level KD methods outperform student networks trained solely with classification loss, the improvement remains limited. Under the limited number of parameters and floating-point operations (FLOPs), the smallest student model x-vector using DKD emphasizing non-target speakers exhibits the largest improvement of 28.12% in Vox1-O EER, compared to its baseline trained with classification loss only. Moreover, our proposed method enables the state-of-the-art CAM++ model to further boost its performance with an EER of 0.590%, while the model's size and FLOPs are respectively 97.73% and 93.39% smaller than the teacher model. Lastly, all three student networks trained using our proposed method have a better result than the embedding-level and conventional label-level KD methods, especially in challenging sets like Vox1-E and Vox1-H. This indicates that DKD emphasizing non-target speaker probabilities effectively improves the performance of student models.

Table 3.2: Results of x-vector using different γ values in Eq.(3.8)

γ	EER (%) / minDCF		
	Vox1-O	**Vox1-E**	**Vox1-H**
$1 - p_\tau^\mathcal{T}$	1.585 / 0.171	1.589 / 0.171	2.704 / 0.244
0.0	1.622 / 0.152	1.646 / 0.175	2.786 / 0.252
1.0	1.463 / 0.166	1.452 / 0.155	2.520 / **0.225**
2.0	**1.319** / 0.160	**1.388** / **0.155**	**2.440** / 0.226
4.0	1.361 / **0.143**	1.415 / 0.156	2.511 / 0.229

Ablation Study: The impact of $\mathcal{L}_{\text{NSKD}}$

We conducted an ablation study on the hyperparameters γ in the DKD formula to show how the robustness of the proposed method varies. To save computational cost, we solely present the results of the student model x-vector, which are summarized in Table 3.2. When emphasizing the $\mathcal{L}_{\text{NSKD}}$ with non-zero values of γ, all the results exceed the performance of $\gamma = 1 - p_\tau^\mathcal{T}$, which is equivalent to the result of the conventional label-level knowledge distillation method. It is observed that removing $NSKD$ by assigning $\gamma = 0$ obtains a worse result than the conventional label-level knowledge distillation. In alignment with the findings from Section 3.2.1, an increasing value of γ leads to better performance, implying the increased significance of $\mathcal{L}_{\text{NSKD}}$. Notably, the best hyperparameter configuration of $\gamma = 2$ achieved an average of 13% improvements in EER compared to the conventional label-level knowledge distillation.

3.4 Summary

This paper has shown the benefit of leveraging non-target speakers for training automatic speaker verification models. Based on this finding, we modified the conventional label-level KD to emphasize the classification probabilities of non-target speakers, which involves splitting and amplifying the non-target speaker's probabilities during the knowledge distillation process. Experimental results on the VoxCeleb test sets show an average of 13.67% improvement in EER of the proposed method compared to other knowledge distillation methods across three different architecture student models.

Chapter 4

Conclusion and Future Work

4.1 Conclusions

In this report, we first reviewed the overview architecture of automatic speaker verification systems, in which the details of each component are introduced and discussed. We then discussed the issue of speaker verification performance degradation in noisy acoustic environments and reviewed existing methods to address this challenge. Although the current approaches can improve the robustness of ASV systems under noisy conditions, the majority of current approaches are under the supervised learning scheme that replies to labeled training data. However, the process of collecting and annotating data for each new domain is both costly and labor-intensive, and sufficient training data may not always be readily available for training a robust ASV model.

To address the problem of expensive data labeling, this study has proposed to leverage the representation from self-supervised learning models to improve the noise-robustness of ASV systems. Self-supervised learning generates target labels from unlabeled input data, eliminating the need for manual annotation. Given the abundance of unlabeled data, SSL models can learn robust representations and improve performance in speech tasks including speaker verification. Nevertheless, robust SSL models are typically large-scale, making them expensive to operate. Therefore, our proposed method utilized a novel knowledge distillation method to transfer the knowledge from the robust large SSL model to smaller ASV models. Based on our initial finding about the importance of non-target speakers in training ASV models, the proposed KD method places an emphasis on the knowledge of these speakers. The results show that the proposed method improves the performance of all student models in the 'in-the-wild' VoxCeleb dataset compared to baseline KD methods.

The potential of the self-supervised representation for robust speaker verification is not fully exploited yet. We believe that more fruitful results can be obtained from further

possible research directions, which are presented in the following section.

4.2 Future Works

The objective of this thesis is to develop algorithms to improve the robustness of speaker verification models against noisy input speech. This led to the development of our method, which addresses this problem by leveraging SSL representation. However, the proposed technique has several constraints. Firstly, the SSL models are trained on prefix tasks like acoustic modeling which may be not directly fit to the speaker verification task. Secondly, SSL models are computationally extensive due to their large number of parameters, hence it can limit the deployment ability of SSL-based ASV systems in real-world scenarios. In addition, with the demonstrated vulnerability of speaker verification to deepfake spoofing attacks [148], research in robust speaker verification can be expanded to strengthen the security of speaker verification systems against these attacks.

4.2.1 Self-supervised learning for speaker verification

In order to obtain better performance on the ASV task, recent SSL models are first pre-trained on the SSL pretext task and then finetuned to perform speaker verification loss. In the pre-trained stage, SSL methods either use contrastive learning to distinguish positive samples from negative ones [18] or predict discrete pseudo-labels of masked input tokens conditioned on the rest of the input sequence [20]. Both approaches aim to implicitly acquire acoustic features that represent phoneme-based information, prioritizing self-supervised learning in the context of the automatic speech recognition task over capturing speaker-specific characteristics. Therefore, the pre-trained setting of SSL models may be not entirely suitable for speaker verification tasks, thereby limiting ASV performance. Since both global information and local details convey speaker information, a more effective method is to consider a pre-training strategy combining utterance-level loss and frame-level loss to better capture speaker information.

4.2.2 Speaker-aware speech enhancement for speaker verification

Although self-supervised learning models for ASV can achieve outstanding performance and robust to noisy speech, they are large in size and computationally intensive to train. A promising direction of noise-robust ASV is to replace the large SSL model with a lightweight speech enhancement module that aims to purify the noisy input speech. However, several studies show that the purification step of the speech enhancement module

can cause the loss of speaker information in the enhanced speech, hence harming the performance of the ASV module in the later step [102]. Joint training of the speech enhancement module and the ASV model can prevent the distortion of speaker information caused by the enhancement process. Additionally, by conditioning the speech enhancement module on the speaker embedding extracted from the ASV model as prior information, it can more effectively suppress noise interference and preserve speaker information.

4.2.3 Anti-spoofing

Another significant challenge for ASV systems arises from the imitation of the speaker's voice. Recent text-to-speech (TTS) and voice conversion (VC) systems, powered by advanced deep generative neural networks, can generate highly realistic synthetic human voices. These developments pose a threat to ASV systems since attackers can exploit TTS or VC systems to produce deepfake speech, effectively impersonating a target speaker. Additionally, fraudsters can utilize deepfake speech to replicate individuals' voices and request fraudulent transfers of money, resulting in millions of dollars being lost to such scams in recent years [149].

To mitigate the risks posed by deepfake speech to social trust and safety, this study can extend to develop Deepfake Speech Detection (DSD) methods as a critical countermeasure. The current state-of-the-art deepfake speech detection models primarily rely on DNNs [150, 151]. As a result, they are susceptible to overfitting to the available training data. However, the rapid evolution of deepfake speech introduces novel techniques that can potentially bypass the defense of existing deepfake speech detection systems by presenting unseen characteristics of deepfake speech. Similar to the use of SSL representation in the field of ASV, the recent DSD model [98], which combines the rich sequence representation of a self-supervised learning model XLSR and the transformer-based Conformer architecture, yields the state-of-the-art result in the ASVspoof 2021 corpus [152]. It is investigated that the dependency information from the temporal and spectral domain of speech can reveal artifact details of synthesis speech [153, 154]. However, the multi-head self-attention module in Conformer blocks focuses on computing cross-covariance of the input tokens along the temporal dimension, hence it may overlook the dependencies between the temporal and channel dimensions of input sequences, which are crucial for the DSD task. To better leverage the SSL representation for the DSD task, the correlation between temporal and channel dependencies can be facilitated.

Bibliography

[1] T. Kinnunen and H. Li, "An overview of text-independent speaker recognition: From features to supervectors," *Speech Communication*, vol. 52, no. 1, pp. 12–40, 2010.

[2] Y. Tu, W. wei Lin, and M. wai Mak, "A survey on text-dependent and text-independent speaker verification," *IEEE Access*, vol. 10, pp. 99 038–99 049, 2022.

[3] A. Q. Ohi, M. F. Mridha, M. A. Hamid, and M. M. Monowar, "Deep speaker recognition: Process, progress, and challenges," *IEEE Access*, vol. 9, pp. 89 619–89 643, 2021.

[4] M. Jakubec, R. Jarina, E. Lieskovska, and P. Kasak, "Deep speaker embeddings for speaker verification: Review and experimental comparison," *Eng. Appl. Artif. Intell.*, vol. 127, no. PA, 2024.

[5] N. Shome, A. Sarkar, A. Ghosh, R. Laskar, and R. Kashyap, "Speaker recognition through deep learning techniques: A comprehensive review and research challenges," *Periodica Polytechnica Electrical Engineering and Computer Science*, vol. 67, Mar. 2023.

[6] X. Zhao, Y. Wang, and D. Wang, "Robust speaker identification in noisy and reverberant conditions," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 4, pp. 836–845, 2014.

[7] Z. Bai and X.-L. Zhang, "Speaker recognition based on deep learning: An overview," *Neural Networks*, vol. 140, pp. 65–99, 2021.

[8] M. J. Alam, G. Bhattacharya, and P. Kenny, "Speaker verification in mismatched conditions with frustratingly easy domain adaptation," in *The Speaker and Language Recognition Workshop*, 2018.

[9] K. A. Lee, Q. Wang, and T. Koshinaka, "The coral+ algorithm for unsupervised domain adaptation of plda," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 5821–5825.

[10] M. MohammadAmini, D. Matrouf, J.-F. Bonastre, S. Dowerah, R. Serizel, and D. Jouvet, "Learning Noise Robust ResNet-Based Speaker Embedding for Speaker Recognition," in *Proc. The Speaker and Language Recognition Workshop (Odyssey 2022)*, 2022, pp. 41–46.

[11] M. Mohammadamini, D. Matrouf, J.-F. Bonastre, S. Dowerah, R. Serizel, and D. Jouvet, "Barlow Twins self-supervised learning for robust speaker recognition," in *Proc. Interspeech 2022*, 2022, pp. 4033–4037.

[12] Y. Zhu, T. Ko, and B. Mak, "Mixup Learning Strategies for Text-Independent Speaker Verification," in *Proc. Interspeech 2019*, 2019, pp. 4345–4349.

[13] S. Wang, J. Rohdin, O. Plchot, L. Burget, K. Yu, and J. H. ernocký, "Investigation of specaugment for deep speaker embedding learning," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7139–7143, 2020.

[14] S. E. Eskimez, P. Soufleris, Z. Duan, and W. Heinzelman, "Front-end speech enhancement for commercial speaker verification systems," *Speech Communication*, vol. 99, pp. 101–113, 2018.

[15] O. Plchot, L. Burget, H. Aronowitz, and P. Matëjka, "Audio enhancing with dnn autoencoder for speaker recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 5090–5094.

[16] R. Peri, M. Pal, A. Jati, K. Somandepalli, and S. S. Narayanan, "Robust speaker recognition using unsupervised adversarial invariance," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6614–6618, 2019.

[17] P. S. Nidadavolu, S. Kataria, J. Villalba, L. P. García-Perera, and N. Dehak, "Unsupervised feature enhancement for speaker verification," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7599–7603, 2019.

[18] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "Wav2vec 2.0: A framework for self-supervised learning of speech representations," in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, 2020.

[19] Z. Chen *et al.*, "Large-scale self-supervised speech representation learning for automatic speaker verification," in *IEEE ICASSP*, 2022, pp. 6147–6151.

[20] S. Chen *et al.*, "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, pp. 1505–1518, 2021.

[21] M. Ravanelli *et al.*, "Multi-task self-supervised learning for robust speech recognition," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 6989–6993.

[22] C. Wang *et al.*, "Improving self-supervised learning for speech recognition with intermediate layer supervision," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 7092–7096.

[23] D. Seo, H.-S. Oh, and Y. Jung, "Wav2kws: Transfer learning from speech representations for keyword spotting," *IEEE Access*, vol. PP, pp. 1–1, May 2021.

[24] W.-T. Kao, Y. Wu, C.-P. Chen, Z.-S. Chen, Y.-P. Tsai, and H. yi Lee, "On the efficiency of integrating self-supervised learning and meta-learning for user-defined few-shot keyword spotting," *2022 IEEE Spoken Language Technology Workshop (SLT)*, pp. 414–421, 2022.

[25] T. Gupta, T. D. Truong, T. T. Anh, and E. S. Chng, "Estimation of speaker age and height from speech signal using bi-encoder transformer mixture model," in *Proc. Interspeech 2022*, 2022, pp. 1978–1982.

[26] D.-T. Truong, T. T. Anh, and C. E. Siong, "Exploring speaker age estimation on different self-supervised learning models," in *2022 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2022, pp. 1950–1955.

[27] Z. Fan, M. Li, S. Zhou, and B. Xu, "Exploring wav2vec 2.0 on Speaker Verification and Language Identification," in *Proc. Interspeech 2021*, 2021, pp. 1509–1513.

[28] N. Vaessen and D. A. van Leeuwen, "Fine-tuning wav2vec2 for speaker recognition," *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7967–7971, 2021.

[29] S. Wang *et al.*, "Leveraging in-the-wild data for effective self-supervised pretraining in speaker recognition," in *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024, pp. 10 901–10 905.

[30] D. A. Reynolds, "An overview of automatic speaker recognition technology," *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, pp. IV–4072–IV–4075, 2002.

[31] K. Rao and S. Sarkar, *Robust Speaker Recognition in Noisy Environments*. Jan. 2014.

[32] T. Kinnunen and P. Rajan, "A practical, self-adaptive voice activity detector for speaker verification with noisy telephone and microphone data," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 7229–7233.

[33] S. P. Gaddamedi, A. Patel, S. Chandra, P. Bharati, N. Ghosh, and S. K. D. Mandal, "Speech enhancement: Traditional and deep learning techniques," in *Proceedings of 27th International Symposium on Frontiers of Research in Speech and Music*, K. Hirose, D. Joshi, and S. Sanyal, Eds., Singapore: Springer Nature Singapore, 2024, pp. 75–86.

[34] W. Cai, J. Chen, and M. Li, "Analysis of Length Normalization in End-to-End Speaker Verification System," in *Proc. Interspeech 2018*, 2018, pp. 3618–3622.

[35] C. Kim and R. M. Stern, "Power-normalized cepstral coefficients (pncc) for robust speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, pp. 1315–1329, 2016.

[36] A. das, M. Ranjan Jena, and K. Kumar Barik, "Mel-frequency cepstral coefficient (mfcc) - a novel method for speaker recognition," *Digital Technologies*, vol. 1, no. 1, pp. 1–3, 2015.

[37] D. Reynolds and R. Rose, "Robust text-independent speaker identification using gaussian mixture speaker models," *Speech and Audio Processing, IEEE Transactions on*, vol. 3, pp. 72 –83, Feb. 1995.

[38] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models," *Digit. Signal Process.*, vol. 10, pp. 19–41, 2000.

[39] H. Muckenhirn, M. Magimai.-Doss, and S. Marcell, "Towards directly modeling raw speech signal for speaker verification using cnns," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4884–4888.

[40] M. Ravanelli and Y. Bengio, "Speaker recognition from raw waveform with sincnet," *2018 IEEE Spoken Language Technology Workshop (SLT)*, pp. 1021–1028, 2018.

[41] W. wei Lin and M. wai Mak, "Wav2spk: A simple dnn architecture for learning speaker embeddings from waveforms," in *Interspeech*, 2020.

[42] G. Zhu, F. Jiang, and Z. Duan, "Y-Vector: Multiscale Waveform Encoder for Speaker Embedding," in *Proc. Interspeech 2021*, 2021, pp. 96–100.

[43] A. Nagrani, J. S. Chung, W. Xie, and A. Zisserman, "Voxceleb: Large-scale speaker verification in the wild," *Computer Speech Language*, vol. 60, p. 101 027, 2020.

[44] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[45] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5329–5333.

[46] H. Zeinali, S. Wang, A. Silnova, P. Matějka, and O. Plchot, *But system description to voxceleb speaker recognition challenge 2019*, 2019. arXiv: 1910.12592.

[47] B. Desplanques, J. Thienpondt, and K. Demuynck, "ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN Based Speaker Verification," in *Proc. Interspeech 2020*, 2020, pp. 3830–3834.

[48] Y. Zhang *et al.*, "MFA-Conformer: Multi-scale Feature Aggregation Conformer for Automatic Speaker Verification," in *Proc. Interspeech 2022*, 2022, pp. 306–310.

[49] B. Liu, Z. Chen, S. Wang, H. Wang, B. Han, and Y. Qian, "DF-ResNet: Boosting Speaker Verification Performance with Depth-First Design," in *Proc. Interspeech 2022*, 2022, pp. 296–300. DOI: 10.21437/Interspeech.2022-484.

[50] M. Sang, Y. Zhao, G. Liu, J. H. Hansen, and J. Wu, "Improving transformer-based networks with locality for automatic speaker verification," in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5. DOI: 10.1109/ICASSP49357.2023.10096333.

[51] Y. Chen, S. Zheng, H. Wang, L. Cheng, Q. Chen, and J. Qi, "An Enhanced Res2Net with Local and Global Feature Fusion for Speaker Verification," in *Proc. INTERSPEECH 2023*, 2023, pp. 2228–2232.

[52] D. Greig, B. Porteous, and A. H. Seheult, "Exact maximum a posteriori estimation for binary images," *Journal of the royal statistical society series b-methodological*, vol. 51, pp. 271–279, 1989.

[53] R. Kuhn *et al.*, "Eigenvoices for speaker adaptation," *5th International Conference on Spoken Language Processing (ICSLP 1998)*, 1998.

[54] P. Kenny, M. Mihoubi, and P. Dumouchel, "New MAP estimators for speaker recognition," in *Proc. 8th European Conference on Speech Communication and Technology (Eurospeech 2003)*, 2003, pp. 2961–2964.

[55] C. Cortes and V. N. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, pp. 273–297, 1995.

[56] W. M. Campbell, D. E. Sturim, and D. A. Reynolds, "Support vector machines using gmm supervectors for speaker verification," *IEEE Signal Processing Letters*, vol. 13, pp. 308–311, 2006.

[57] W. M. Campbell, D. E. Sturim, D. A. Reynolds, and A. Solomonoff, "Svm based speaker verification using a gmm supervector kernel and nap variability compensation," *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, vol. 1, 2006.

[58] P. Kenny, G. Boulianne, P. Ouellet, and P. Dumouchel, "Joint factor analysis versus eigenchannels in speaker recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 4, pp. 1435–1447, 2007.

[59] P. Kenny, "Joint factor analysis of speaker and session variability: Theory and algorithms," Jan. 2006.

[60] E. Variani, X. Lei, E. McDermott, I. L. Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 4052–4056.

[61] Y.-Q. Yu and W.-J. Li, "Densely Connected Time Delay Neural Network for Speaker Verification," in *Proc. Interspeech 2020*, 2020, pp. 921–925.

[62] D. Povey *et al.*, "Semi-Orthogonal Low-Rank Matrix Factorization for Deep Neural Networks," in *Proc. Interspeech 2018*, 2018, pp. 3743–3747.

[63] Y.-Q. Yu and W.-J. Li, "Densely Connected Time Delay Neural Network for Speaker Verification," in *Proc. Interspeech 2020*, 2020, pp. 921–925.

[64] Y.-Q. Yu, S. Zheng, H. Suo, Y. Lei, and W.-J. Li, "Cam: Context-aware masking for robust speaker verification," *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6703–6707, 2021.

[65] H. Wang, S. Zheng, Y. Chen, L. Cheng, and Q. Chen, "CAM++: A Fast and Efficient Network for Speaker Verification Using Context-Aware Masking," in *Proc. INTERSPEECH 2023*, 2023, pp. 5301–5305.

[66] N. R. Koluguri, T. Park, and B. Ginsburg, "Titanet: Neural model for speaker representation with 1d depth-wise separable convolutions and global context," *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 8102–8106, 2021.

[67] S. Han, J. Byun, and J. W. Shin, "Time-domain speaker verification using temporal convolutional networks," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 6688–6692.

[68] T. Liu, R. K. Das, K. Aik Lee, and H. Li, "Mfa: Tdnn with multi-scale frequency-channel attention for text-independent speaker verification with short utterances," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 7517–7521.

[69] B. Han, Z. Chen, and Y. Qian, "Local information modeling with self-attention for speaker verification," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 6727–6731.

[70] Z. Zhao, Z. Li, W. Wang, and P. Zhang, "Pcf: Ecapa-tdnn with progressive channel fusion for speaker verification," in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

[71] S. Han, Y. Ahn, K. Kang, and J. W. Shin, "Short-segment speaker verification using ecapa-tdnn with multi-resolution encoder," in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

[72] C. Li *et al.*, *Deep speaker: An end-to-end neural speaker embedding system*, 2017. arXiv: 1705.02304.

[73] M. Hajibabaei and D. Dai, "Unified hypersphere embedding for speaker recognition," *arXiv preprint arXiv:1807.08312*, 2018.

[74] W. Cai, J. Chen, and M. Li, "Exploring the Encoding Layer and Loss Function in End-to-End Speaker and Language Recognition System," in *Proc. The Speaker and Language Recognition Workshop (Odyssey 2018)*, 2018, pp. 74–81.

[75] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Computer Vision – ECCV 2018: 15th European Conference, Munich, Germany, September 8–14, 2018, Proceedings, Part VII*, Munich, Germany: Springer-Verlag, 2018, 3–19.

[76] S. Yadav and A. K. Rai, "Frequency and temporal convolutional attention for text-independent speaker recognition," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6794–6798, 2019.

[77] W. Xia and J. H. Hansen, "Speaker Representation Learning Using Global Context Guided Channel and Time-Frequency Transformations," in *Proc. Interspeech 2020*, 2020, pp. 3226–3230.

[78] L. Zhang, Q. Wang, and L. Xie, "Duality temporal-channel-frequency attention enhanced speaker representation learning," *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 206–213, 2021.

[79] B. Liu, Z. Chen, and Y. Qian, "Attentive Feature Fusion for Robust Speaker Verification," in *Proc. Interspeech 2022*, 2022, pp. 286–290.

[80] J. Li, Y. Tian, and T. Lee, "Convolution-based channel-frequency attention for text-independent speaker verification," in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

[81] A. Hajavi and A. Etemad, "A deep neural network for short-segment speaker recognition," in *Interspeech*, 2019.

[82] Y. Tang, G.-H. Ding, J. Huang, X. He, and B. Zhou, "Deep speaker embedding learning with multi-level pooling for text-independent speaker verification," *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6116–6120, 2019.

[83] Y. Jung, S. M. Kye, Y. Choi, M. Jung, and H. Kim, "Improving Multi-Scale Aggregation Using Feature Pyramid Module for Robust Speaker Verification of Variable-Duration Utterances," in *Proc. Interspeech 2020*, 2020, pp. 1501–1505.

[84] T. Liu, K. A. Lee, Q. Wang, and H. Li, "Golden gemini is all you need: Finding the sweet spots for speaker verification," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 32, pp. 2324–2337, 2024.

[85] K.-L. Du and M. Swamy, "Recurrent neural networks," in Dec. 2014, pp. 337–353, ISBN: 978-1-4471-5570-6. DOI: 10.1007/978-1-4471-5571-3_11.

[86] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, pp. 1735–80, Dec. 1997.

[87] S. El-Moneim, M. Nassar, M. Dessouky, N. Ismail, A. El-Fishawy, and F. Abd El-Samie, "Text-independent speaker recognition using lstm-rnn and speech enhancement," *Multimedia Tools and Applications*, vol. 79, Sep. 2020.

[88] J. Kharibam and K. Thongam, "Automatic speaker recognition from speech signal using bidirectional long-short-term memory recurrent neural network," *Computational Intelligence*, vol. 39, Jan. 2020.

[89] Z. Zhao *et al.*, "A lighten cnn-lstm model for speaker verification on embedded devices," *Future Generation Computer Systems*, vol. 100, pp. 751–758, 2019.

[90] A. Vaswani *et al.*, "Attention is all you need," in *Neural Information Processing Systems*, 2017.

[91] M. India, P. Safari, and J. Hernando, "Self Multi-Head Attention for Speaker Recognition," in *Proc. Interspeech 2019*, 2019, pp. 4305–4309.

[92] M. India, P. Safari, and J. Hernando, "Double multi-head attention for speaker verification," *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6144–6148, 2020.

[93] H. Zhu, K. A. Lee, and H. Li, "Serialized Multi-Layer Multi-Head Attention for Neural Speaker Embedding," in *Proc. Interspeech 2021*, 2021, pp. 106–110.

[94] Y. Shi, M. Chen, Q. Huang, and T. Hain, *T-vectors: Weakly supervised speaker identification using hierarchical transformer model*, 2020. arXiv: 2010.16071.

[95] R. Wang *et al.*, "Multi-view self-attention based transformer for speaker recognition," *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6732–6736, 2021.

[96] N. Zhang, J. Wang, Z. Hong, C. Zhao, X. Qu, and J. Xiao, "Dt-sv: A transformer-based time-domain approach for speaker verification," *2022 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–7, 2022.

[97] B. Han, Z. Chen, and Y. Qian, "Local information modeling with self-attention for speaker verification," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 6727–6731.

[98] A. Gulati *et al.*, "Conformer: Convolution-augmented Transformer for Speech Recognition," in *Proc. Interspeech 2020*, 2020, pp. 5036–5040.

[99] S. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," *2007 IEEE 11th International Conference on Computer Vision*, pp. 1–8, 2007.

[100] T. Ko, V. Peddinti, D. Povey, M. L. Seltzer, and S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 5220–5224.

[101] D. Snyder, G. Chen, and D. Povey, *MUSAN: A Music, Speech, and Noise Corpus*, arXiv:1510.08484v1, 2015.

[102] S. Shon, H. Tang, and J. R. Glass, "Voiceid loss: Speech enhancement for speaker verification," in *Interspeech*, 2019.

[103] J. ho Kim, J.-S. Heo, H. jin Shim, and H. jin Yu, "Extended u-net for speaker verification in noisy environments," in *Interspeech*, 2022.

[104] N. Clarke, S. Furnell, and P. Reynolds, "Biometric authentication for mobile devices," *Proceeding of the 3rd Australian Information Warfare and Security Conference*, Jan. 2002.

[105] S. Sadjadi *et al.*, "The 2016 nist speaker recognition evaluation," Aug. 2017, pp. 1353–1357.

[106] X. Zhao, Y. Shao, and D. Wang, "Casa-based robust speaker identification," *IEEE Transactions on Audio, Speech Language Processing - TASLP*, vol. 20, pp. 1608–1616, Jul. 2012.

[107] S. Furui, "Cepstral analysis technique for automatic speaker verification," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 29, pp. 254 –272, May 1981.

[108] O. Plchot *et al.*, "Developing a speaker identification system for the darpa rats project," Oct. 2013, pp. 6768–6772.

[109] J. Pelecanos and S. Sridharan, "Feature warping for robust speaker verification," in *Proceedings of 2001 A Speaker Odyssey: The Speaker Recognition Workshop*, Crete, Greece: European Speech Communication Association, 2001, pp. 213–218.

[110] O. Novotný, O. Plchot, O. Glembek, J. Černocký, and L. Burget, "Analysis of dnn speech signal enhancement for robust speaker recognition," *Computer Speech Language*, vol. 58, pp. 403–421, 2019.

[111] X. Zhao, Y. Wang, and D. Wang, "Robust speaker identification in noisy and reverberant conditions," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 4, pp. 836–845, 2014.

[112] Y. Wu, L. Wang, K. A. Lee, M. Liu, and J. Dang, "Joint Feature Enhancement and Speaker Recognition with Multi-Objective Task-Oriented Network," in *Proc. Interspeech 2021*, 2021, pp. 1089–1093.

[113] S. Shon, H. Tang, and J. R. Glass, "Voiceid loss: Speech enhancement for speaker verification," in *Interspeech*, 2019.

[114] D. Michelsanti and Z. Tan, "Conditional generative adversarial networks for speech enhancement and noise-robust speaker verification," in *Interspeech*, 2017.

[115] F. Zhao, H. Li, and X. Zhang, "A robust text-independent speaker verification method based on speech separation and deep speaker," May 2019, pp. 6101–6105.

[116] J. Villalba and E. Lleida, "Bayesian adaptation of PLDA based speaker recognition to domains with scarce development data," in *Proc. The Speaker and Language Recognition Workshop (Odyssey 2012)*, 2012, pp. 47–54.

[117] B. J. Borgström, E. Singer, D. Reynolds, and S. O. Sadjadi, "Improving the Effectiveness of Speaker Verification Domain Adaptation with Inadequate In-Domain Data," in *Proc. Interspeech 2017*, 2017, pp. 1557–1561.

[118] A. Emînî, P.-G. Noé, and M. Driss, "Denoising x-vectors for robust speaker recognition," May 2020.

[119] W. Huang, B. Han, S. Wang, Z. Chen, and Y. Qian, "Robust cross-domain speaker verification with multi-level domain adapters," in *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024, pp. 11 781–11 785.

[120] H. Yu, Z. Tan, Z. Ma, and J. Guo, "Adversarial network bottleneck features for noise robust speaker verification," in *Interspeech 2017, 18th Annual Conference of the International Speech Communication Association, Stockholm, Sweden, August 20-24, 2017*, F. Lacerda, Ed., ISCA, 2017, pp. 1492–1496.

[121] Z. Meng, Y. Zhao, J. Li, and Y. Gong, "Adversarial speaker verification," *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6216–6220, 2019.

[122] J. Zhou, T. Jiang, L. Li, Q. Hong, Z. Wang, and B. Xia, "Training multi-task adversarial network for extracting noise-robust speaker embedding," *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6196–6200, 2018.

[123] M. Sang, W. Xia, and J. H. Hansen, "Deaan: Disentangled embedding and adversarial adaptation network for robust speaker representation learning," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 6169–6173.

[124] O. Novotný, O. Plchot, P. Matějka, L. Mošner, and O. Glembek, "On the use of X-vectors for Robust Speaker Recognition," in *Proc. The Speaker and Language Recognition Workshop (Odyssey 2018)*, 2018, pp. 168–175. DOI: 10.21437/Odyssey.2018-24.

[125] H. Zhang, M. Cissé, Y. N. Dauphin, and D. Lopez-Paz, "Mixup: Beyond empirical risk minimization," in *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, 2018.

[126] D. S. Park *et al.*, "Specaugment: A simple data augmentation method for automatic speech recognition," in *Interspeech*, 2019.

[127] R. Xiao, Z. Li, X. Miao, W. Wang, and P. Zhang, "Guidedmix: An on-the-fly data augmentation approach for robust speaker recognition system," *Electronics Letters*, vol. 58, Nov. 2021.

[128] W. Lin and M.-W. Mak, "Robust speaker verification using population-based data augmentation," May 2022, pp. 7642–7646.

[129] D. Ho, E. Liang, X. Chen, I. Stoica, and P. Abbeel, "Population based augmentation: Efficient learning of augmentation policy schedules," in *Proceedings of the 36th International Conference on Machine Learning*, K. Chaudhuri and R. Salakhutdinov, Eds., ser. Proceedings of Machine Learning Research, vol. 97, PMLR, 2019, pp. 2731–2741.

[130] D. Cai, W. Cai, and M. Li, "Within-sample variability-invariant loss for robust speaker recognition under noisy environments," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6469–6473, 2020.

[131] J. Li, J. Han, and H. Song, "Gradient Regularization for Noise-Robust Speaker Verification," in *Proc. Interspeech 2021*, 2021, pp. 1074–1078.

[132] L. Zhang, Q. Wang, K. A. Lee, L. Xie, and H. Li, "Multi-Level Transfer Learning from Near-Field to Far-Field Speaker Verification," in *Proc. Interspeech 2021*, 2021, pp. 1094–1098.

[133] S. Sarkar and K. Sreenivasa Rao, "Stochastic feature compensation methods for speaker verification in noisy environments," *Applied Soft Computing*, vol. 19, pp. 198–214, 2014.

[134] K.-K. Yiu, M.-W. Mak, and S.-Y. Kung, "Environment adaptation for robust speaker verification," in *Proc. 8th European Conference on Speech Communication and Technology (Eurospeech 2003)*, 2003, pp. 2973–2976.

[135] J. Li, L. Deng, R. Haeb-Umbach, and Y. Gong, "Chapter 4 - processing in the feature and model domains," in *Robust Automatic Speech Recognition*, Oxford: Academic Press, 2016, pp. 65–106.

[136] Y. Sun, H. Zhang, L. Wang, K. A. Lee, M. Liu, and J. Dang, "Noise-disentanglement metric learning for robust speaker verification," in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

[137] J.-H. Kim, J. Heo, H.-S. Shin, C.-Y. Lim, and H.-J. Yu, "Diff-sv: A unified hierarchical framework for noise-robust speaker verification using score-based diffusion probabilistic models," in *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024, pp. 10 341–10 345.

[138] S. Wang, Y. Yang, T. Wang, Y. Qian, and K. Yu, "Knowledge distillation for small foot-print deep speaker embedding," in *IEEE ICASSP*, 2019, pp. 6021–6025.

[139] Z. Peng, X. He, K. Ding, T. Lee, and G. Wan, "Label-free knowledge distillation with contrastive loss for light-weight speaker recognition," in *ISCSLP*, 2022, pp. 324–328.

[140] X. Liu, M. Sahidullah, and T. Kinnunen, "Distilling multi-level x-vector knowledge for small-footprint speaker verification," in *arXiv preprint arXiv:2303.01125*, 2023.

[141] J. Heo, C. yeong Lim, J. ho Kim, H. seo Shin, and H.-J. Yu, "One-Step Knowledge Distillation and Fine-Tuning in Using Large Pre-Trained Self-Supervised Learning Models for Speaker Verification," in *Proc. INTERSPEECH*, 2023, pp. 5271–5275.

[142] L. Zhang, Z. Chen, and Y. Qian, "Knowledge distillation from multi-modality to single-modality for person verification," in *Proc. INTERSPEECH*, 2021, pp. 1897–1901.

[143] N. Vaessen and D. van Leeuwen, "Training speaker recognition systems with limited data," in *Proc. Interspeech 2022*, 2022, pp. 4760–4764.

[144] Y. Zhang and W. Deng, "Class-balanced training for deep face recognition," in *IEEE/CVF CVPRW*, 2020, pp. 3594–3603.

[145] B. Zhao, Q. Cui, R. Song, Y. Qiu, and J. Liang, "Decoupled knowledge distillation," in *IEEE/CVF CVPR*, 2022, pp. 11 943–11 952.

[146] J. D. et al, "ArcFace: Additive angular margin loss for deep face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 10, pp. 5962–5979, 2022.

[147] H. Wang *et al.*, "Wespeaker: A research and production oriented speaker embedding learning toolkit," in *IEEE ICASSP*, 2023, pp. 1–5. DOI: 10.1109/ICASSP49357.2023.10096626.

[148] P. Gupta, H. A. Patil, and R. C. Guido, "Vulnerability issues in automatic speaker verification (asv) systems," *EURASIP J. Audio Speech Music Process.*, vol. 2024, no. 1, 2024.

[149] H. Chen and K. Magramo, "Finance worker pays out $25 million after video call with deepfake 'chief financial officer'," *CNN*, Feb. 2024. [Online]. Available: `https://edition.cnn.com/2024/02/04/asia/deepfake-cfo-scam-hong-kong-intl-hnk/index.html`.

[150] Z. Almutairi and H. Elgibreen, "A review of modern audio deepfake detection methods: Challenges and future directions," *Algorithms*, vol. 15, no. 5, 2022.

[151] J. Yi, C. Wang, J. Tao, X. Zhang, C. Y. Zhang, and Y. Zhao, *Audio deepfake detection: A survey*, 2023. arXiv: `2308.14970`.

[152] X. Liu *et al.*, "Asvspoof 2021: Towards spoofed and deepfake speech detection in the wild," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 2507–2522, 2023.

[153] J. Yang, R. K. Das, and H. Li, "Significance of subband features for synthetic speech detection," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 2160–2170, 2020.

[154] K. Sriskandaraja, V. Sethu, P. N. Le, and E. Ambikairajah, "Investigation of sub-band discriminative information between spoofed and genuine speech," in *Interspeech*, 2016.