PUBLICATIONS OF
THE UNIVERSITY OF EASTERN FINLAND

Dissertations in Forestry and Natural Sciences

ALEKSANDR SIZOV

SECURE AND ROBUST SPEECH REPRESENTATIONS FOR SPEAKER AND LANGUAGE RECOGNITION

UNIVERSITY OF
EASTERN FINLAND

PUBLICATIONS OF THE UNIVERSITY OF EASTERN FINLAND
DISSERTATIONS IN FORESTRY AND NATURAL SCIENCES

N:o 296

Aleksandr Sizov

SECURE AND ROBUST SPEECH REPRESENTATIONS FOR SPEAKER AND LANGUAGE RECOGNITION

ACADEMIC DISSERTATION

To be presented by the permission of the Faculty of Science and Forestry for public examination in the Louhela auditorium, Joensuu Science Park, Länsikatu 15, University of Eastern Finland, Joensuu, on December 18th, 2017, at 12 o'clock.

University of Eastern Finland
School of Computing
Joensuu 2017

Grano Oy
Jyväskylä, 2017
Editors: Pertti Pasanen, Matti Tedre,
Jukka Tuomela, and Matti Vornanen

Distribution:
University of Eastern Finland Library / Sales of publications
julkaisumyynti@uef.fi
http://www.uef.fi/kirjasto

ISBN: 978-952-61-2687-6 (Print)
ISSNL: 1798-5668
ISSN: 1798-5668
ISBN: 978-952-61-2688-3 (PDF)
ISSNL: 1798-5668
ISSN: 1798-5676

Author's address:
University of Eastern Finland
School of Computing
P.O.Box 111
80101 JOENSUU
FINLAND
email: sizov@cs.uef.fi

Supervisors:
Associate Professor Tomi Kinnunen, Ph.D.
University of Eastern Finland
School of Computing
P.O.Box 111
80101 JOENSUU
FINLAND
email: tkinnu@cs.uef.fi

Scientist Kong Aik Lee, Ph.D.
Agency for Science, Technology and Research (A*STAR)
Institute for Infocomm Research (I^2R)
1 Fusionopolis Way
21-01 Connexis
138632 SINGAPORE
SINGAPORE
email: kalee@i2r.a-star.edu.sg

Reviewers:
Senior Technical Staff Douglas A. Reynolds, Ph.D.
Lincoln Laboratory
Massachusetts Institute of Technology
Human Language Technology Group
244 Wood Street, Lexington
MA 02420-9108
UNITED STATES
dar@ll.mit.edu

Associate Professor Claude Barras, Ph.D.
LIMSI, CNRS, Université Paris–Saclay
Spoken Language Processing Group
Campus universitaire bât 508
Rue John von Neumann
91403 ORSAY CEDEX
FRANCE
claude.barras@limsi.fr

Opponent:
Professor Tuomas Virtanen
Tampere University of Technology
Laboratory of Signal Processing
P.O. Box 553
33101 TAMPERE
FINLAND
tuomas.virtanen@tut.fi

Aleksandr Sizov
Secure and robust speech representations for speaker and language recognition
Joensuu: University of Eastern Finland, 2017
Publications of the University of Eastern Finland
Dissertations in Forestry and Natural Sciences

ABSTRACT

Despite decades of research on the topic, recognizing speakers and languages from unconstrained speech data remains a challenging problem. A number of factors adversely affect the recognition accuracy of these applications. First, each recording device and a potential coding algorithm induce artifacts, called channel differences, that have to be compensated for. For example, channel difference happens when a person is enrolled in the system with a regular cellular phone recording but is tested with a close-talking microphone. Second, the voice of a human being is not constant over time: it changes with age or sickness. Third, short utterance duration and varied lengths across enrollment and test utterances cause problems.

This work contains three case studies that developed and advanced methods for speaker verification and language identification, with a special emphasis on channel robustness and the ability to withstand attempts of unauthorized access to systems protected by speaker verification, called *spoofing*. The probabilistic linear discriminant analysis (PLDA) model was extensively used for all three cases. This model is a back-end factor analysis model with wide usage both in speaker and language recognition and it is a good starting point for this research because (1) it allows one to explicitly model different kinds of variabilities present in a speech signal, (2) it is perfectly suited for a state-of-the-art feature representation in the field of speaker and language recognition ,called *i-vector*, and (3) it has fast training and scoring algorithms. The classic PLDA model, as it is used in modern speaker and language recognition systems, is difficult to use directly for joint speaker verification and anti-spoofing, though, because it does not allow to model spoofing variability. This motivated the author to develop a new factor analysis model for the task of secure and robust speaker verification, *two-stage* PLDA. It is a modified version of the PLDA model, equipped with an additional subspace for modeling artificial speech and a special training routine that allows for this subspace to learn only complementary information to the one already preserved in the model. Moreover, two-stage PLDA allows the seamless integration of speaker verification and anti-spoofing components that does not require any fusion. The proposed method for robust language identification is also a derivative of PLDA, but in this case, its novelty is in the discriminative fine-tuning of PLDA language models, as well as in the new objective function.

The National Institute of Standards and Technology (NIST) corpora was used to evaluate both models. The NIST Language Recognition Evaluation (LRE) 2015 corpus was used to evaluate the latter. The improvement in terms of the average detection cost, the standard objective metric used in NIST LRE campaigns, was about 10% relative to the best generative baseline and 6% relative to the best discriminative baseline. Secure speaker verification models were evaluated on a modified NIST Speaker Recognition Evaluation (SRE) 2006 corpus that has been appended via artificial speech trials, namely two voice conversion methods: the popular joint-

density Gaussian mixture model (JD-GMM) based method, and a simplified unit selection (US) method. The proposed joint approach of modeling additional artificial speech subspace outperforms other considered methods by a large margin and shows promising generalization abilities to unseen attack types.

Universal Decimal Classification: *004.934, 159.946.3, 534.78*
Library of Congress Subject Headings: *Speech; Language and languages; Identification; Biometric identification; Authentication; Factor analysis; Vector spaces; Mathematical optimization; Corpora (Linguistics)*
Yleinen suomalainen asiasanasto: *puhe; puheääni; kieli; tunnistaminen; puhujantunnistus; biotunnistus; verifiointi; todentaminen; faktorianalyysi; vektorit (matematiikka); optimointi; korpukset*
Keywords: *speaker verification, anti-spoofing, language identification, factor analysis, probabilistic linear discriminant analysis*

ACKNOWLEDGEMENTS

Research part of this work was carried out in two pieces. First, at the School of Computing, University of Eastern Finland, Finland, during 2013–2015 under the supervison of Tomi Kinnunen, where I hold an early-stage researcher position provided by the University of Easter Finalnd. Second, at the Institute for Infocomm Research, A*STAR, Singapore, during 2015–2017 under the supervision of Kong Aik Lee, where I was funded by the A*STAR Research Attachment Programme. I am very thankful to both of my supervisors for the support and guidance that they provided to me.

Also, I'd like to thank my numerous colleagues who worked with me during these years on one project or another, most notably: Elie Khoury, Trung Hieu Nguyen, Hanwu Sun, Guangsen Wang, Md Sahidullah, Zhizheng Wu, and Ville Hautamäki. Last but not least, I am sincerely thankful to my pre-examination reviewers, Douglas A. Reynolds and Claude Barras, for the time they spent to read this dissertation and to provide very helpful feedback.

Joensuu, November 22 2017

Aleksandr Sizov

LIST OF ABBREVIATIONS

ASV	Automatic speaker verification
BTAS	Biometrics: theory, applications, and systems
CLDNN	Convolutional, long short-term memory and deep neural network
CQCC	Constant Q cepstral coefficient
CQT	Constant Q transform
DCF	Detection cost function
DCT	Discrete cosine transform
DNN	Deep neural network
EER	Equal error rate
EM	Expectation-maximization
FAR	False acceptance rate
FT	Fourier transform
FRR	False rejection rate
GMM	Gaussian mixture model
HMM	Hidden Markov model
IMFCC	Inverted mel-frequency cepstral coefficient
JD-GMM	Joint density Gaussian mixture model
LLR	Log-likelihood ratio
LP	Linear predictive
LRE	Language recognition evaluation
MCEP	Mel-cepstral
MFCC	Mel-frequency cepstral coefficient
MMI	Maximum mutual information
NIST	National Institute of Standards and Technology
PLDA	Probabilistic linear discriminant analysis
ROCCH-EER	Receiver operating characteristic convex hull for computation of EER
RNN	Recurrent neural network
SDCC	Shifted delta cepstral coefficient
SRE	Speaker recognition evaluation
STDFT	Short-time discrete Fourier transform
SVM	Support vector machine
TTS	Text-to-speech
UBM	Universal background model
US	Unit selection
VC	Voice conversion

LIST OF PUBLICATIONS

This thesis consists of the present review of the author's work in the field of speaker and language recognition and the following selection of the author's publications:

I A. Sizov, K.A. Lee and T. Kinnunen, "Unifying probabilistic linear discriminant analysis variants in biometric authentication," *Proceedings of the Joint IAPR International Workshop on Structural, Syntactic, and Statistical Pattern Recognition*, Joensuu, Finland, pp. 464–475, August 2014

II A. Sizov, E. Khoury, T. Kinnunen, Z. Wu and S. Marcel, "Joint Speaker Verification and Anti-Spoofing in the i-Vector Space," *IEEE Transactions on Information Forensics and Security*, 10(4): 821–832, April 2015

III A. Sizov, K.A. Lee and T. Kinnunen, "Direct Optimization of the Detection Cost for I-vector based Spoken Language Recognition", *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(3): 588–597, March 2017.

Throughout the overview, these papers will be referred to by Roman numerals. In addition to the above-listed publications, the author has collaborated on a number of other co-authored publications [1–9], all dealing with the topics of speaker and language recognition and probabilistic modeling.

AUTHOR'S CONTRIBUTION

In **I**, the author summarized and linked three *probabilistic linear discriminant analysis* (PLDA) variations, derived necessary equations that were absent in previous publications on this topic and performed the experiments. In **II**, the author developed a novel mathematical formulation—in terms of the probabilistic latent subspace methods—of the initial idea that Dr. Khoury and Dr. Kinnunen suggested, to conduct anti-spoofing detection in the i-vector space. Dr. Wu provided the artificial speech corpus. The author and Dr. Khoury jointly carried out experiments and wrote the article. In **III**, the author extended his preliminary conference publication [6] results, proposing a new objective function for discriminative training that provides for a better approximation of the cost function for *language recognition evaluation* 2015 (LRE'15). Together with Dr. Lee, the author suggested applying discriminative training in a low-dimensional PLDA latent space, which led to the increased robustness to channel and language effects.

TABLE OF CONTENTS

1 INTRODUCTION

Speech-based services, including speech recognition and speech synthesis, have become ubiquitous in our daily lives. Not only do they provide an increasingly convenient way of interacting with a smartphone but also a new trend in *smart home* virtual assistants, such as Amazon Echo/Alexa and Google Home[1], enables users to interact naturally with the digital environment in their homes and on the internet. The widespread presence of speech technology makes people rely on it increasingly. This, in turn, requires certain auxiliary services to increase security and user convenience. One of them is the ability to reliably identify who interacts with a device, a problem known as *speaker recognition* [10]. Besides consumer applications, speaker recognition also finds use in telephone banking systems, border control, and police surveillance. Another design requirement is the ability to use speech-based services in environments with high degrees of background noise or varied channel conditions, called *robustness*. In publication **I**, the author studied a model designed to cope with variable channel conditions.

Another related application is the ability to recognize a language of spoken utterance. A seemingly straightforward solution for such a task would be to use a speech recognition engine that supports multiple languages to detect the language that is most likely to have generated a given utterance. A drawback of such a solution is that the creation of such an engine requires vast amounts of labeled data, which can be prohibitively expensive or difficult to collect or transcribe. A more viable solution is to recognize a language without recognizing its speech contents. This can be achieved by extracting and directly modeling certain language cues present in spoken utterances [11]. In publication **III**, the author proposed new robust methods to model such acoustic language cues. More precisely, he investigated two research questions in this publication. First, would it be beneficial to substitute a default objective function for discriminative training with the one that is directly related to the performance metric? Second, what would be the effects if discriminative training were conducted in a low-dimensional language-oriented subspace instead of the original space?

Although the goal of speaker recognition as a biometric system, called *automatic speaker verification* (ASV), is to prevent unauthorized access to a system it protects, speaker verification systems have been found to be highly prone to various *spoofing attacks* intended to mimic a given target speaker's voice. *Text-dependent* speaker verification systems—systems that constrict a test phrase to be in a predefined set of enrollment phrases—can mainly be spoofed by using speech synthesis [12], voice conversion [13], and impersonation attacks [14], whereas replay attacks have limited applicability because an intruder has to record exactly the same phrases as were used during the enrollment [15] (for details about different types of spoofing attacks, refer to Section 4.1). *Text-independent* speaker verification systems—systems that do not put any lexical constraints neither on the enrollment (training) nor test phrases—are prone to all of the above-mentioned attacks [16–19] plus replay attacks [20]. Many of these attacks have become increasingly achievable for people

[1] `https://madeby.google.com/home/`

without technical expertise. For example, a replay attack requires a person to record and play back a target speaker recording to a verification system, whereas state-of-the-art speech synthesis software is supposedly able to create a target speaker's voice from as little as one minute of that person's speech [2].

To address this issue, speaker recognition researchers, jointly with speech synthesis and voice conversion researchers, launched a series of challenges intended to help with detecting different types of spoofing attacks: the *Automatic Speaker Verification Spoofing and Countermeasures* (ASVspoof) challenge with its first edition[3] in 2015 and its second edition[4] in 2017, and the BTAS (*Biometrics: Theory, Applications, and Systems*) 2016 Speaker Anti-spoofing Competition[5].

This dissertation work, in the fields of machine learning and engineering, advances the state-of-the-art in speaker and language recognition in a number of ways. Specifically, the author developed new computational methods for anti-spoofing and language identification in **II** and **III** respectively. In **II**, the author proposed the largely unexplored approach of a common feature space for joint speaker verification and anti-spoofing, with all of the modeling performed via a single specially trained classifier. The hypothesis behind this model was that if an anti-spoofing component is built upon a speaker verification component, it will learn only the necessary residual directions of variability that can reliably tell apart spoofing and natural speech. The author also contributed to the design of the first edition of the ASVspoof challenge [8], which the research community has now widely adopted. To promote reproducible research and potentially benefit the research community, the author has made selected codes and data publicly available. Specifically, publication **I** comes with Python and Matlab source codes[6], and publication **II** comes the with source code[7] and corpus of feature vectors (specifically, i-vectors [21]) used in the publication[8].

The rest of this article-based dissertation is organized as follows. Chapter 2 introduces the different classes and types of features that were used in **I–III**. In Chapter 3, the author first explains the requirements that speaker and language recognition tasks set up for classifiers, and then describes both the generative and the discriminative classifiers used in the rest of the dissertation research. Chapter 4 focuses on the problem of speaker anti-spoofing: first, it defines spoofing attacks and their classification, then it introduces the first challenge in speaker anti-spoofing and elaborates on the development of stand-alone countermeasures for the challenge and for the period after it. The Chapter concludes with the most important aspect of any countermeasure: its integration into a biometric system and their joint work. Chapter 5 presents the different evaluation metrics that were used in **I–III**. The final Chapter concludes the dissertation and outlines possible future work direction. Publications **I**, **II** and **III** are attached as appendices.

[2] https://lyrebird.ai/

[3] http://www.asvspoof.org/index2015.html

[4] http://www.asvspoof.org/

[5] http://www.biometrics-center.ch/testing/btas-2016-speaker-anti-spoofing

[6] https://sites.google.com/site/fastplda/

[7] https://pypi.python.org/pypi/xspear.fast_plda

[8] http://www.idiap.ch/resource/biometric/data/TIFS2015.zip

2 FEATURE EXTRACTION

By feature extraction, we understand the process that (1) compresses a raw speech signal and (2) emphasizes certain traits in a speech signal that lead to the better performance of machine learning algorithms. We consider two different classes of features, *frame-level* and *utterance-level* features. Both of them were used in the case studies of this dissertation work. Specifically, *mel-frequency cepstral coefficient* (MFCC) features were used in publications **I** and **II**, *shifted delta cepstral coefficient* (SDCC) features are used in publication **III**, and i-vectors were used in all three publications.

2.1 FRAME-LEVEL FEATURES

The majority of the frame-level features use the *short-time discrete Fourier transform* (STDFT) under the assumption of the local stationarity of a speech signal, resulting in frame-level analysis with a typical frame duration of 15 to 30 ms. This procedure is usually referred to as *short-term spectral analysis*. In essence, the classifier backends will treat the short-term features as statistically independent observations (this will be discussed in Chapter 3).

2.1.1 Mel-frequency cepstral coefficients

The most commonly used short-term feature set for speech-related tasks are MFCCs [22]. The idea behind MFCC processing is to mimic the way in which humans perceive sounds. Because the human articulatory system has co-evolved together with the human auditory system, many useful characteristics of speech and speaker can be computed by mimicking how humans process speech. Originally, MFCC features were developed for speech recognition [22]. Later, they were gradually established as baseline features for speaker and language recognition problems as well [23].

The extraction of MFCC starts with splitting the input speech waveform into (overlapping) frames with a typical length, N, ranging from 120 to 240 samples per frame, which corresponds from 15 to 30 ms, given a sampling rate of 8kHz. These samples typically go through a *pre-emphasis* filter to balance the high spectral roll-off effect typical of voiced speech. The frame is then multiplied point-wise with a *window function* to suppress the discontinuity at the frame boundaries.

Then, the frame samples $\{x[n]\}_{n=0}^{N-1}$ are processed by *discrete Fourier transform* (DFT) to obtain N-dimensional complex vector $\{X[k]\}_{k=0}^{N-1}$, also known as the *spectrum*:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-2\pi i k n / N} . \tag{2.1}$$

To better mimic human auditory perception, power spectrum $|X[k]|^2$ is then weighted by using a mel scale to place more emphasis on the lower-frequency components of the spectrum, and it is subsequently fed to the logarithmic function. As the

final step, the *discrete cosine transform* (DCT) is applied on top of this weighted log-spectrum with the optional removal of higher coefficients for compression purposes. The resulting MFCC vector, $\mathbf{c}$, can be summarized as follows:

$$\mathbf{c} = \mathbf{D} \log(\mathbf{H}|X|^2), \tag{2.2}$$

where $\mathbf{D}$ is a DCT matrix, and $\mathbf{H}$ is an MFCC filter bank matrix with the first dimension equal to the number of filters.

Recent research in the field of voice conversion and speech synthesis anti-spoofing [24] suggested that synthetic speech artifacts might be more concentrated and more easily detectable in the regions less important for human auditory perception, specifically, in the higher-frequency range. The so-called *inverted* MFCC (IMFCC) [25] scale—which has a reversed filter bank structure compared with the MFCC—might be a better choice for such a task. Its only difference from the MFCC scale is in the reversed filter bank structure.

2.1.2 Constant Q Cepstral Coefficients

Constant Q transform [26,27] (CQT) is an alternative to Fourier transform (FT). The main difference between two transforms is that during the conversion from time domain to frequency domain a ratio between a center of a filter frequency and its bandwidth is kept constant for CQT, while FT has a constant bandwidth. This means that CQT has a higher frequency resolution for lower frequencies and vice versa. This is so-called *equal temperament scale* of the western music [28]. Interestingly, such approach turned out to work surprisingly well for anti-spoofing [29,30] producing state-of-the-art results for detection of all machine-generated attacks.

2.1.3 Linear predictive coding features

Linear predictive (LP) coding features [31] are derived from LP coding model that predicts the next time-domain sample $x[n]$ based on a linear combination of the past time-domian samples $x[n-1], x[n-2], \ldots, x[n-K]$ as follows:

$$\hat{x}[n] = -\sum_{i=1}^{K} a_i x[n-i]. \tag{2.3}$$

Coefficients $\{a_i\}_{i=1}^{K}$ are denouted as LP coding features. In **II**, these features are used to reconstruct a signal, resulting in a signal replica that has passed through synthesis-channel and, thus, can be used to train an anti-spoofing detector.

2.1.4 Delta and double-delta features

The process described above for the computation of MFCCs processes each one of the short-term frames independently without using any data outside of it. However, human speech production is a time-varying system that never remains static. Certain characteristics of the acoustic speech signal, such as transitions in diphthongs, stretch across longer periods of time than the typical frame duration. One of the commonly used and simple ways of accounting for this is to use *numerical differentiation* and approximate derivatives of the frame-level features, known as *delta*

coefficients. The simplest way in which to compute the delta coefficients at time frame t, $\mathbf{d}_t$, is to compute the symmetric difference of the following form:

$$\mathbf{d}_t = \frac{\mathbf{c}_{t+\Theta} - \mathbf{c}_{t-\Theta}}{2\Theta}, \tag{2.4}$$

where Θ is a half-window size, and $\mathbf{c}_t$ is the feature vector of the static MFCC coefficients (or other features) at frame t. A more robust way of computation is to use a regression of the following form [32]:

$$\mathbf{d}_t = \frac{\sum_{\theta=1}^{\Theta} \theta(\mathbf{c}_{t+\theta} - \mathbf{c}_{t-\theta})}{2\sum_{\theta=1}^{\Theta} \theta^2}. \tag{2.5}$$

If numerical differentiation is applied again on top of the delta coefficients this produces *double-delta* coefficients, denoted by $\mathbf{d}_t^2$. The most commonly used feature configuration is a stack of base features with deltas and double-deltas: $\mathbf{o}_t^\mathsf{T} = \left[\mathbf{c}_t, \mathbf{d}_t, \mathbf{d}_t^2\right]^\mathsf{T}$, with dimensionality being triple the number of static MFCCs.

2.1.5 Shifted delta cepstral coefficients

A more flexible way of incorporating additional temporal information that often leads to an increased accuracy especially in language recognition tasks [33, 34] are SDCCs. If MFCCs have a widespread across almost all speech-related tasks, SDCCs are used mostly for language recognition. SDCC features are built on top of MFCC features or, in principle, any other similar short-term spectral features and are parametrized via four integer control parameters: the number of cepstral coefficients—K, time advance and delay for the delta computation—Θ, number of blocks to concatenate—k, and the time shift between consecutive blocks—P. The SDCC vector at time frame t, $\mathbf{s}_t$, is given as follows:

$$\mathbf{s}_t = \begin{bmatrix} \mathbf{c}(t+\Theta) - \mathbf{c}(t-\Theta) \\ \mathbf{c}(t+P+\Theta) - \mathbf{c}(t+P-\Theta) \\ \vdots \\ \mathbf{c}(t+(k-1)P+\Theta) - \mathbf{c}(t+(k-1)P-\Theta) \end{bmatrix}. \tag{2.6}$$

2.2 UTTERANCE-LEVEL FEATURES

Because speech utterances in a typical corpus vary in length, an extraction of frame-level features from each utterance in a corpus results in feature arrays of different length. This is a problem for many back-end modeling techniques because they require fixed-length inputs. One solution—that over the years gradually changed and evolved to provide state-of-the-art pefromance—is to train a Gaussian mixture model (GMM) on all feature vectors in a corpus (GMM trained for such purpose is called a *universal background model* (UBM) [10]), and concatenate all mean vectors of the resultant model into a single high-dimensional UBM *supervector* [35]. A robust way to produce such supervector for each new utterance is simply to adapt the UBM supervector to the new data. A typical supervector might have from a few thousand to tens of thousands dimensions. Although this is not a problem for some discriminative classifiers like SVM [35], generative classifiers do not have sufficient amount of training data for such high-dimensional inputs.

A clever build-up on top of the supervector framework to extract a low-dimensional fixed-length representation, so-called *i-vector*, was suggested in 2011 [21]. An i-vector representation has become one of the most successful and versatile tools for a broad range of speech-related tasks, including speaker recognition [21], language recognition [36], speaker anti-spoofing [37], and acoustic modelling [38]. Its versatility is explained by the fact that an i-vector is a compact representation of many sources of variability at once, and back-end modeling techniques are selected for a particular task at hand. Here, we briefly summarize the i-vector extraction process. Sections 3.1.3 and 4.4 describe our modeling approaches for different tasks.

Formally, i-vector is a *maximum a posteriori* (MAP) estimate of a latent variable in a multi-Gaussian factor analysis model based on a special *Gaussian mixture model* (GMM), known as the UBM. I-vector $\boldsymbol{\varphi}$ is inferred as follows:

$$\boldsymbol{\varphi} = \arg\max_{\mathbf{x}} \left[\prod_{j=1}^{J} \prod_{h=1}^{H_j} \mathcal{N}\left(\mathbf{o}_h \middle| \boldsymbol{\mu}_j + \mathbf{T}_j\mathbf{x}, \Sigma_j\right) \right] \mathcal{N}\left(\mathbf{x}|\mathbf{0}, \mathbf{I}\right) , \tag{2.7}$$

where $\{\boldsymbol{\mu}_j, \Sigma_j\}_{j=1}^{J}$ are the mean vectors and covariance matrices of the UBM with J components, $\{\mathbf{T}_j\}_{j=1}^{J}$ are blocks of the *total variability* matrix $\mathbf{T}$, and $\{\mathbf{o}_h\}_{h=1}^{H_j}$ is a set of acoustic feature vectors for a given utterance aligned to the j-th mixture component. Because this alignment is not given to us, we estimate it by using Baum-Welch statistics, where the *responsibilities* of h-th feature vector being generated by j-th UBM component, γ_{hj} are usually computed either via the GMM [21] or *deep neural network* (DNN) [39, 40]). Utterance-dependent zero-th and first order Baum-Welch statistics, $\mathbf{N}$ and $\mathbf{F}$, are then used to compute an i-vector as follows:

$$\boldsymbol{\varphi} = \mathbf{L}^{-1}\mathbf{T}^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}\mathbf{F}, \tag{2.8}$$

where $\mathbf{L}^{-1} = (\mathbf{I} + \mathbf{T}^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}\mathbf{N}\mathbf{T})^{-1}$ is the posterior covariance of the latent variable $\mathbf{x}$. Given the set of acoustic observations, the zero-order and first-order statistics are computed as

$$N_j = \sum_{h=1}^{H} \gamma_{hj}, \tag{2.9}$$

and

$$\mathbf{F}_j = \sum_{h=1}^{H} \gamma_{hj}(\mathbf{o}_h - \boldsymbol{\mu}_j), \tag{2.10}$$

respectively. Note that we have used the centralized first order statistics [21] in (2.10). Diagonal matrix $\mathbf{N}$ is produced by stacking $N_j\mathbf{I}$ along a diagonal, whereas to produce $\mathbf{F}$ we concatenate all $\mathbf{F}_j$ matrices for a given utterance.

3 ROBUST SPEAKER AND LANGUAGE RECOGNITION TECHNIQUES

The topics covered in this dissertation require us to make decisions about a given utterance, whether it belongs to a particular class: the *same* or *different* speaker in the case of speaker verification, one of the few languages for language identification, and *genuine* or *spoof* speech for anti-spoofing detection. To make such decisions, we constructed a statistical model for each class in question and work with the likelihood function of those models given a speech sample. As a rule of thumb, generative models are better suited for open-set problems, whereas discriminative models are preferred for closed-set problems, as the latter optimize the model parameters using the data from both target and competing classes. Speaker verification is an example of an open-set problem, designed to recognize new speakers who were not present for the system during its development stages. Language identification, in turn, involves both open-set and closed-set tasks. The former occur when unknown languages can be present in the data. In this work, we consider only the closed-set task and demonstrate that discriminative methods outperform generative ones. Counter-intuitively, anti-spoofing, having only two possible decision classes and being a closed-set task, does not follow the general trend: different spoofing methods could be so diverse from one another that generative methods are competitive with the discriminative ones [41, 42].

3.1 GENERATIVE MODELING

Generative modeling refers to a class of methods that first estimate class-conditional probabilities $p(\mathbf{x}|\mathcal{C}_k)$, and then combine them with prior $p(\mathcal{C}_k)$ to obtain required probabilities $p(\mathcal{C}_k|\mathbf{x})$ as

$$p(\mathcal{C}_k|\mathbf{x}) \propto p(\mathbf{x}|\mathcal{C}_k)p(\mathcal{C}_k)\,, \tag{3.1}$$

where $\propto$ means "proportional to." This gives us additional flexibility to use the same class-conditional models, that were trained once, for various use-cases that differ by prior probabilities of the classes. For example, this allows to compensating for a data imbalance in language recognition, as was the case in **III**, where the amount of data for certain languages differed by almost three orders of magnitude. Regarding speaker verification, the modification of prior probabilities for target and impostor classes enables the tuning of the system for various security levels suitable for particular applications.

3.1.1 Gaussian mixture model

The *gaussian mixture model* [43] (GMM) is one of the most essential tools for many speech-related tasks, including speaker and language recognition. The flexibility of the model for estimating any smooth probability distribution, given a sufficient number of Gaussian components, together with the ease of use have reserved it a place among state-of-the-art systems for decades. In fact, it is so versatile that it is

currently successfully used both as a classifier and as a front-end for the extraction of high-level features, such as i-vectors (see Section 2.2). The generative assumption behind this model is that each data point is generated via one of the Gaussian components with the probability of its corresponding weight.

Set of mixture weights $\{w_j\}_{j=1}^{J}$, a set of mean vectors $\{\boldsymbol{\mu}_j\}_{j=1}^{J}$, and a set of covariance matrices $\{\boldsymbol{\Sigma}_j\}_{j=1}^{J}$ describe multivariate GMM with J components. Unlike the case of a single Gaussian distribution that has a unique closed-form maximum likelihood solution provided via a sample mean and a sample covariance matrix, the same does not hold for a GMM and we resort to the *expectation-maximization* (EM) algorithm [44] to maximize a log-likelihood function of data $\mathbf{O} = \{\mathbf{o}_n\}_{n=1}^{N}$ given all model parameters denoted as $\boldsymbol{\Theta}$:

$$\log p(\mathbf{O}|\boldsymbol{\Theta}) = \sum_{n=1}^{N} \log \sum_{j=1}^{J} w_j \mathcal{N}\left(\mathbf{o}_n \middle| \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\right) . \tag{3.2}$$

The EM algorithm is an iterative process that consists of two alternating stages: at the *expectation* stage for each data sample, we estimate the posterior probabilities of belonging to each mixture component, called *responsibilities*, while keeping the model parameters fixed, and at the *maximization* stage, we maximize the model parameters while keeping the responsibilities fixed.

In practice, short-term spectral features are usually used as an input to GMM (see Section 2.1) and the number of Gaussian components, J, is selected based on the size of a speech corpus used for training the model. Another characteristic of the model that is dependent on the amount of training data is a type of covariance matrix: isotropic, diagonal, or full. For each covariance matrix, the number of parameters is set to one in an isotropic case, grows linearly with the dimensionality of the input space for a diagonal case, and grows quadratically for a full covariance matrix case. A more flexible approach of controlling the complexity of the model, as well as the data generation process, is called *factor analysis*.

3.1.2 Classic factor analysis model

Factor analysis defines *linear-Gaussian models* [45] with full covariance matrices but with a restricted number of free parameters in them that follow more complicated data-generation assumptions. The classic factor analysis formulation

$$\mathbf{x} = \boldsymbol{\mu} + \mathbf{W}\mathbf{z} + \boldsymbol{\varepsilon} , \tag{3.3}$$

$$\mathbf{z} \sim \mathcal{N}(\mathbf{z}|\mathbf{0}, \mathbf{I}) , \tag{3.4}$$

$$\boldsymbol{\varepsilon} \sim \mathcal{N}\left(\boldsymbol{\varepsilon} \middle| \mathbf{0}, \sigma^2\mathbf{I}\right) , \tag{3.5}$$

describes observable variable $\mathbf{x}$ as a combination of mean vector $\boldsymbol{\mu}$, noise latent variable $\boldsymbol{\varepsilon}$, and low-dimensional latent variable $\mathbf{z}$ multiplied by factor-loading matrix $\mathbf{W}$.

3.1.3 Probabilistic linear discriminant analysis

A more advanced variant of factor analysis that better suits speech-related tasks and is extensively used in this dissertation together with its modifications is the so-called *probabilistic linear discriminant analysis* (PLDA) model that was first introduced

in [46]. It assumes that the j-th i-vector of speaker i, $\boldsymbol{\phi}_{ij}$, is generated as

$$\boldsymbol{\phi}_{ij} = \boldsymbol{\mu} + \mathbf{V}\mathbf{y}_i + \mathbf{U}\mathbf{x}_{ij} + \boldsymbol{\varepsilon}_{ij}\,, \tag{3.6}$$

$$\mathbf{y}_i \sim \mathcal{N}\left(\mathbf{y}_i|\mathbf{0},\mathbf{I}\right)\,, \tag{3.7}$$

$$\mathbf{x}_{ij} \sim \mathcal{N}\left(\mathbf{x}_{ij}|\mathbf{0},\mathbf{I}\right)\,, \tag{3.8}$$

$$\boldsymbol{\varepsilon}_{ij} \sim \mathcal{N}\left(\boldsymbol{\varepsilon}_{ij}|\mathbf{0},\Sigma\right)\,, \tag{3.9}$$

where $\boldsymbol{\mu}$ is mean vector, columns of matrices $\mathbf{V} \in \mathbb{R}^{D\times P}$ and $\mathbf{U} \in \mathbb{R}^{D\times M}$ span the between- and within-speaker subspaces, $\mathbf{y}_i$ and $\mathbf{x}_{ij}$ are their corresponding latent variables, and $\boldsymbol{\varepsilon}_{ij}$ is a latent residual noise variable distributed with diagonal covariance matrix Σ.

After PLDA was proposed for speaker recognition in 2010 [47] and for language recognition in 2011 [48], it remains a state-of-the-art method for back-end modeling. PLDA success can be attributed to both its easy and flexible formulation as well as to its excellent synergy with a compact yet powerful utterance representation in a form of an i-vector [21], which is also based on factor analysis.

3.2 DISCRIMINATIVE MODELING

Discriminative modeling refers to a class of statistical methods that either (1) estimate a *discriminant function* that directly assigns each data sample to a particular class or (2) estimate the posterior probability of a class given data sample $\mathbf{x}$: $p(\mathcal{C}_k|\mathbf{x})$. In this dissertation, observation $\mathbf{x}$ is a speech utterance that is represented either by a sequence of frame-level acoustic feature vectors (Section 2.1), or in the form of an utterance-level *i-vector* (Section 2.2). Discriminative methods differ how they model the data: this representation can be in the form of a set of support vectors [35], one or several Gaussian distributions [49] or a neural network. Another important design consideration is a *cost function*, used to optimize a model. A typical choice for a Gaussian model would be a cost function based on the *maximum mutual information* (MMI) training criterion [49], whereas a typical choice for a neural network predicting a data class would be a *cross-entropy* function between the model distribution and the empirical distribution defined by the training set.

3.2.1 Support vector machine

The *support vector machine* (SVM) [50] was originally proposed as a binary classifier that uses the so-called *maximum margin* hyperplane between two classes of points to separate them. In this dissertation, we consider only a linear SVM applied to an utterance-level i-vectors for the task of anti-spoofing detection. We choose linear SVM, that is, an SVM with a linear kernel, instead of SVM models with non-linear kernels because these non-linear kernels can be viewed as tools for implicitly increasing data dimensionality [45]. In our case, however, i-vectors already have sufficiently large dimensionality (200 to 1000) to provide comparable levels of performance and robustness with remarkable speed-up benefits compared with their non-linear counterparts (for more details about i-vectors please refer to Section 2.2). Let us for each i-vector $\mathbf{x}_n$ define its class label as either $t_n = +1$ if it belongs to natural speech or $t_n = -1$ if it belongs to artificial speech. The goal of the linear SVM approach is to achieve a solution of the following form:

$$y(\mathbf{x}) = \mathbf{w}^{\mathsf{T}}\mathbf{x} + b\,, \tag{3.10}$$

where $y(\mathbf{x})$ is the desired maximum margin hyperplane, $\mathbf{w}$ is a vector of weights and b is a bias. The resulting objective function to be minimized is as follows:

$$\mathcal{Q}_{\text{SVM}} = \frac{1}{2}\|\mathbf{w}\| - \sum_n a_n\{t_n(\mathbf{w}^\mathsf{T}\mathbf{x}_n + b) - 1\}\,, \tag{3.11}$$

where a_n are Lagrange multipliers. For the derivations, see [45].

3.2.2 Gaussian models

Each class is modeled by a multivariate Gaussian distribution [1]:

$$p(\mathbf{x}|\mathcal{C}_k) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \frac{1}{(2\pi)^{D/2}}\frac{1}{|\boldsymbol{\Sigma}_k|^{1/2}}\exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^\mathsf{T}\boldsymbol{\Sigma}_k^{-1}(\mathbf{x}-\boldsymbol{\mu}_k)\right\}, \tag{3.12}$$

where k-th class, $\mathcal{C}_k$, is represented by D-dimensional mean vector $\boldsymbol{\mu}_k$ and $D \times D$ covariance matrix $\boldsymbol{\Sigma}_k$. To speed up the model, to reduce memory requirements, and in the cases of insufficient training data, covariance matrices are often restricted to being diagonal (i.e., to have zeros everywhere except for the main diagonal) or even isotropic, which is a subset of diagonal matrices that have equal values along the main diagonal.

If we choose the MMI training criterion then our goal would be to maximize the posterior probability of the correct class given the training data across all classes, as follows:

$$\mathcal{Q}_{\text{MMI}} = \sum_k \sum_{\mathbf{x}_n \in \mathcal{C}_k} \log p(\mathcal{C}_k|\mathbf{x}_n) = \sum_k \sum_{\mathbf{x}_n \in \mathcal{C}_k} \log \frac{p(\mathbf{x}_n|\mathcal{C}_k)}{p(\mathbf{x}_n)}\,. \tag{3.13}$$

This is a general-purpose criterion that favors models with a high likelihood of the correct class and low likelihoods of the incorrect classes. Quite often, real-life problems have their own performance measures that are not completely aligned with the optimization of a general-purpose objective functions [51]. Such cases benefit the development of task-specific objective functions. In **II**, we derived such an objective function and its optimization method for the National Institute of Standards and Technology (NIST) LRE 2015 [52].

[1] We do not consider a univariate case because one-dimensional frame- or utterance-level features are almost never used to represent a speech utterance, and the authors have never seen Gaussian models applied directly on top of waveforms

4 SPOOFING AND COUNTERMEASURES FOR SPEAKER VERIFICATION

A *spoofing attack* against a biometric system is a deliberate attempt to purposefully fool the system in some way to gain unauthorized access to the protected environment. The SO/IEC 30107-1:2016[1] standard calls it a *presentation attack*. Methods against such attacks are known as *anti-spoofing* solutions or *spoofing countermeasures*.

4.1 TYPES OF VOICE SPOOFING ATTACKS

One of the most challenging issues for building robust spoofing detectors is in the rich and changing nature of these attacks. Various attack types and subtypes may differ from one another more than from natural speech (see Fig. 4.1), thus making it challenging to develop a perfect anti-spoofing system that can detect unseen attack types. The currently known spoofing attacks could be categorized into four major types [53]: *impersonation* [54,55], *voice conversion* [56], *speech synthesis* [16] and *replay* [57] attacks. Except for the first type where a person tries to imitate the other person, voice conversion, speech synthesis and replay are all machine-generated attacks.

4.1.1 Human-generated attacks

Impersonation is the only type of spoofing attack that does not require the usage of machines and is done solely through a person, an *impersonator*, who attempts to mimic the speech characteristics of another person, a *target speaker*. The need for impersonators to first study the target speaker's voice and then to produce spoofed utterances at real-time speed at a maximum makes the data generation process highly resource demanding compared with the machine-generated speech. For this reason, impersonation is the least studied type of attack, with only a handful of impersonators and target speakers considered in each study [17,54,55]. The general belief, however, is that impersonators focus more on imitating prosody and other high-level traits of speech but do not necessarily alter short-term speech characteristics as much. As a result, impersonated speech might sound convincing to a human ear but less effective for spoofing *automatic speaker verification* (ASV) systems that rely on short-term spectral features. A promising and largely unexplored direction for circumventing the dependency on professional impersonators might be to use *crowd-sourcing* to find speakers among the general population who sound similar to a specific target speaker [59]. At the present time, however, very little is known about the effectiveness of such an approach. For the above-mentioned reasons, impersonation attacks are not studied further in this dissertation. Instead, the focus will be on technically induced attacks that are consistently shown to be effective in degrading the accuracy of unprotected ASV systems to unacceptable levels [60–62].

[1] https://www.iso.org/standard/53227.html

4.1.2 Machine-generated attacks

The majority of the research in the field of speaker anti-spoofing focuses on machine-generated attacks, for two main reasons. First, unlike human-generated attacks, whose effectiveness depends largely on the impersonator, such attacks consistently cause increased error rates for stand-alone ASV systems lacking any protection [60–62]. Second, relative speed and the ease of spoofing data generation allows one to produce many spoofed training and test utterances, thus enabling the study of a wide variety of modeling techniques and at the same time leading to increased statistical confidence in the estimated error rates [63].

Voice conversion (VC) algorithms modify the speech utterance of a given *source* speaker (who we will call an *attacker* in the context of anti-spoofing) to make him or her sound similar to a target speaker. Historically, VC originated from *parallel* approaches [64, 65] that required pairs of utterances with the same lexical content from an attacker and a target speaker to train a conversion method. *Non-parallel* approaches either require parallel data only for reference speakers [66] to train the model or do not require any parallel data whatsoever [67,68]. VC attacks are among the most studied in the field of ASV at the moment [8]. Some of the most effective solutions for detecting such attacks are based on feature engineering [69] and the fusion of different features [70]. In particular, it is known that most *vocoders*, an

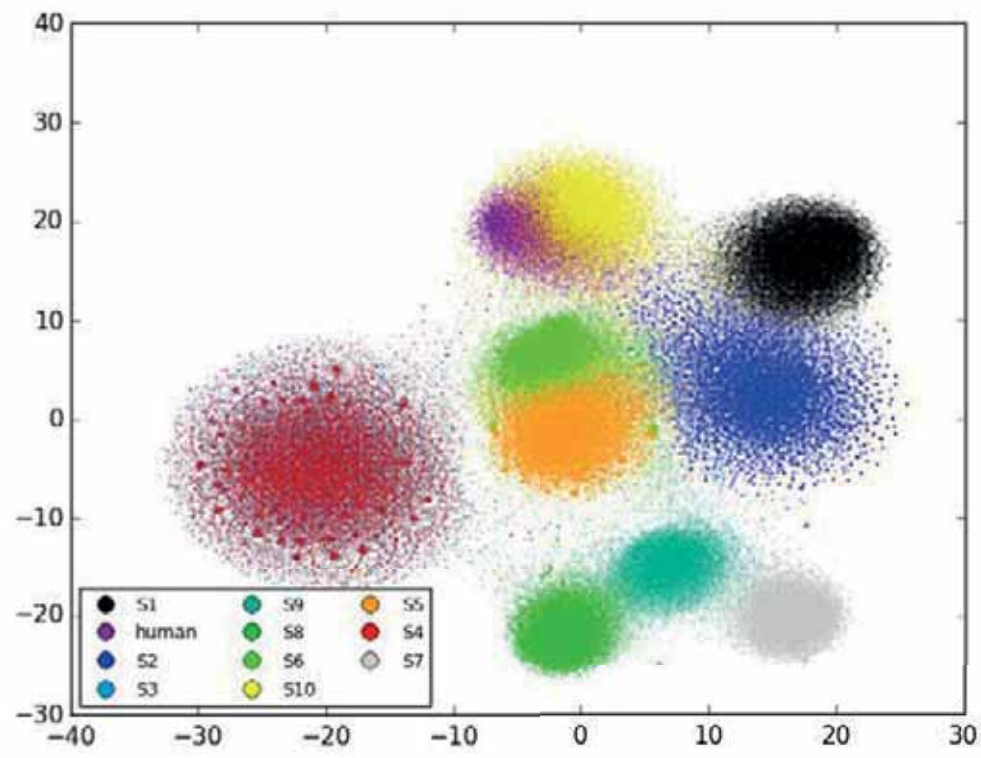

Figure 4.1: Visualization of ASVspoof15 [3] data, where both development and evaluation parts of the corpus have been jointly mapped to 2D using t-SNE algorithm [58]. Each point represents an i-vector.

essential component of VC that is responsible for generating speech waveforms, discard phase information; hence phase-aware features [71,72] have displayed high spoofing attack detection rates.

Speech synthesis systems, also known as *text-to-speech* (TTS) systems, first use training data to build a speech model, which is then used for transforming any text in a given language into speech. Many different approaches to TTS exist, but at the moment, the two most prominent techniques are based on either a parametric statistical TTS via *hidden Markov models* (HMMs) [73] or *recurrent neural networks* (RNNs) [74], and *unit selection* (US) [75]. Another new and promising approach that is based on deep learning and generates an audio waveform sample by sample, WaveNet[2], shows top scores in subjective sound quality tests [76]. At the moment, it is still too computationally demanding to gain widespread usage, but as deep learning oriented hardware improves exponentially quickly and WaveNet's generation process becomes sufficiently optimized, it has the potential to become one of the toughest spoofing methods to deal with.

TTS spoofing attacks were found to be highly effective against unprotected ASV systems (i.e., without any anti-spoofing components) [61], with their effectiveness as a spoofing technique increasing with the amount of training data used for TTS training [77], as might be expected. As well as for the VC, parametric TTS methods use a vocoder to generate the final speech waveform. This is arguably their weakest aspect at the moment in the context of anti-spoofing, as the detection of the most popular vocoders can be considered a solved problem [8], at least in the context of a clean speech (experiments with noisy speech are much less successful, especially for low *signal-to-noise* ratios [7,78]). On the other hand, US TTS methods output a piece-wise collection of natural speech units (that have been selected based on how well they represent a target utterance and how well consecutive units fit with one another [75]), thus avoiding vocoder limitations, and pose a serious threat to many anti-spoofing systems [8]. Recently, some countermeasures have shown promising results against US TTS [29,79,80], but all experiments have been conducted with a particular implementation of a US algorithm on a single corpus, so it remains unknown how the proposed countermeasures will generalize to other high-quality US TTS methods.

A *replay* attack takes place when an attacker records a target speaker or intercepts the target speaker's recorded utterance by replaying it to the ASV system to be identified as a target speaker. Among all machine-generated attacks, replay requires the least amount of technical expertise yet provides high levels of false acceptance errors for ASV systems both with [30,81] and without countermeasures [82]. The main reason that replay attacks are so challenging is that such attacks are, in fact, natural speech that has passed through additional recording and playing stages. In theory, this should create additional channel effects that can be detected unless high-end equipment, used to collect and playback an attack, reduces them sufficiently.

4.2 ASVSPOOF 2015

Although the potential devastating effects of a spoofing attack on a speaker verification system have been known for a long time, this threat has not been at the center of attention of either research or engineering communities. Occasional unsystematic studies on this topic were conducted on small, private corpora, thus causing

[2] `https://deepmind.com/blog/wavenet-generative-model-raw-audio/`

the research findings to lack consistency, and making the countermeasure results incomparable. The first serious attempt to both highlight the importance of anti-spoofing and to present a relatively large, publicly available development corpus was done[3] in 2015 as a first edition of the *automatic speaker verification spoofing and countermeasures* (ASVspoof) challenge [8]. It focused on VC and TTS attack detection. The challenge involved 10 attacks in total — two parametric TTS methods, one US TTS method, and seven VC methods — split between development and evaluation sets. The author was involved in the preparation of the challenge and focused on designing proper protocols for training and evaluation.

One of the main conclusions of the challenge was that a front-end is more important than a back-end is: successful submissions were represented by a variety of different classifiers: GMM [41, 72], SVM [42], and different architectures of neural networks [70, 83]. This might be explained by two facts: (1) the natural speech utterances for this challenge were collected in an anechoic chamber[4], and hence, the challenge data lacked noise and did not require any specialized modeling techniques, and (2) only one spoofing attack, a unit selection attack generated using MaryTTS[5], largely influenced the evaluation performance in this challenge, and the successful systems were the ones able to extract informative cues at the front-end level; the choice of a classifier for subsequent modeling of these cues is not crucial.

4.3 STAND-ALONE ANTI-SPOOFING

As discussed above, two main components of building a successful anti-spoofing detector are (1) feature engineering [24], and (2) classifier engineering [**II**], [84]. As for the former, multiple attempts have been made to design highly specialized features for a particular corpus/attack type that have failed to generalize to new cases [70,85,86]. Only a handful of features have shown consistent performance over a broad range of data. Two features among them that became a *de facto* standard are MFCC [22, 23] and CQCC [26, 27, 29] Regarding classifier engineering, the research community has not been nearly as inventive as for the feature engineering. This is perhaps because the most common research background is in signal processing, or the dataset sizes are smaller than in conventional ASV research, hence prohibiting the use of certain machine learning approaches.

The field of pattern recognition has a long and deep-rooted tradition of thinking that classification systems should include separately treated feature extraction (front-end) and classifier (back-end) components. Inspired by the overwhelming success of *deep learning* in multiple areas of machine learning, the emphasis is, however, gradually shifting toward *end-to-end* systems [87] that simultaneously learn both a feature representation and a suitable classifier for the given task. At the current moment, a certain confusion exists regarding the term "end-to-end system". In the speaker anti-spoofing field, a waveform represents the original input and a score vector that is later compared against a threshold to produce a final decision represents the original output. Hence, a truly end-to-end anti-spoofing system should start with a waveform and produce a score vector of spoofing classes. One successful implementation of such a system [88] consists of a particular combination of convolutional, long short-term memory and deep neural network layers—called

[3] `http://www.asvspoof.org/index2015.html`

[4] `https://wiki.inf.ed.ac.uk/CSTR/SASCorpus`

[5] `http://mary.dfki.de/`

CLDNN model—which was originally proposed for speech recognition [89]. This model was applied for BTAS 2016 [90] challenge data. It was found to outperform the result of the challenge winner by 30%. The majority of studies in this field on end-to-end systems do not, however, use waveforms as an input but rather start with some intermediate data representation, usually a few processing steps prior to well-known frame-level features. The most common starting point is the spectrogram [91, 92].

4.4 INTEGRATION OF ANTI-SPOOFING INTO AN ASV SYSTEM

Spoofing detection has little practical meaning by itself and becomes important only as a component of the joint biometric authentication system. By far the most popular solution for combining an anti-spoofing module and a speaker verification module is *score-level fusion* [93]. Here, a score acts as an interface between actual anti-spoofing and verification classifiers on one side, and fusion and decision-making systems on the other side. This provides the utmost flexibility to design both classifiers independently from one another and a decision-making system: this allows to use different front- and back-end suited for particular applications, to seamlessly combine the results that come from different people or even different organizations.

Score-level fusion can be divided into *cascaded fusion* that has independent thresholds for anti-spoofing and verification tasks, resulting in trial acceptance only if its scores pass both thresholds, and *parallel fusion* that combines both anti-spoofing and verification scores in a classifier and outputs a score that requires only a singe threshold. At this stage of development, it is important to avoid the over-fitting of this classifier because this might compromise the security of the entire biometric system even if stand-alone anti-spoofing and speaker verification detectors function properly. For this reason, simple classifiers, such as *logistic regression* [94], are preferred. A number of studies compared both fusion approaches [95, 96]. In terms of a false acceptance rate, cascaded fusion seems to outperform parallel fusion, whereas metrics that take into account both false acceptance and false rejection errors yield controversial results.

An alternative to score-level fusion is *feature-level fusion*, where a single classifier is designed and (or) trained in such a way as to be able to model both natural and synthetic variabilities at the same time and to output a single score. In [2] and **II**, the author studied several architectures of such classifiers and compared them with a score-level logistic regression approach. The main motivation behind this was to explore if it is possible to automatically find and use certain dependencies between speaker and spoofing factors of variation that are lost when we compress a high-dimensional feature vector (or vectors) into a scalar score. The results in **II** indicate that this in indeed a promising direction both in terms of performance and generalization to unseen attacks.

5 PERFORMANCE EVALUATION

The evaluation of detection systems, described in this dissertation, is statistical in nature and was performed after processing a large number of trials. A *trial* is a combination of enrollment and test utterances. The processing of each trial via a system results in a numerical *score*, which is a quantitative representation of hypotheses that we want to accept or reject.

5.1 TWO-CLASS PROBLEMS

Binary tasks are those that test for only two hypotheses: the *null* hypothesis and the *alternative* hypothesis. For example, in speaker verification, the null hypothesis assumes that the same speaker spoke all trial utterances, whereas the alternative hypothesis assumes that enrollment and test utterances originate from different speakers. In spoofing detection, the null hypothesis assumes that a trial utterance belongs to natural speech, whereas the alternative hypothesis assumes that a trial utterance belongs to artificial speech. Each score in a binary classification task is just a scalar value, usually in the form of a *log-likelihood ratio* (LLR) between the two hypotheses. In this way, a zero score does not favor any of the hypotheses; larger positive scores express increasing preference for the null hypothesis, and vice versa for negative scores. In the end, this value has to be converted into a "hard" decision: either a trial belongs to a class represented by a null hypothesis (we will call it a *target* trial), or a class represented by an alternative hypothesis (we will call it an *imposture* trial). This operation is done via *thresholding* and is subject to two kinds of errors: *false rejection*, when a trial from a null hypothesis class was mistakenly assigned to an alternative hypothesis class and *false acceptance*, when a trial from an alternative hypothesis class was assigned to a null one. These two types of errors are reversely dependent on each other and have to be taken into account together. Both false rejection and false acceptance errors are possible outcomes of a particular trial. The aggregated values across all trials are known as the *false rejection rate* (FRR) and *false acceptance rate* (FAR), sometimes known as *miss* and *false alarm* rates respectively. They are computed as follows:

$$\mathrm{FRR}(\mathcal{S}_{\mathrm{tgt}}|\theta) = \frac{1}{|\mathcal{S}_{\mathrm{tgt}}|} \sum_{s \in \mathcal{S}_{\mathrm{tgt}}} \mathrm{I}(s < \theta), \tag{5.1}$$

$$\mathrm{FAR}(\mathcal{S}_{\mathrm{imp}}|\theta) = \frac{1}{|\mathcal{S}_{\mathrm{imp}}|} \sum_{s \in \mathcal{S}_{\mathrm{imp}}} \mathrm{I}(s \geq \theta)\,, \tag{5.2}$$

where $\mathcal{S}_{\mathrm{tgt}}$ and $\mathcal{S}_{\mathrm{imp}}$ are the sets of scores for target and imposture trials respectively, θ is a threshold to make "hard" decisions, $\mathrm{I}(\bullet)$ is an indicator function, and $|\bullet|$ denotes the number of elements in a given set.

Some tasks, such as the joint speaker verification and spoofing detection considered in **II**, might contain imposture trials of different natures: a typical evaluation protocol contains both *zero-effort* imposture trials, where natural speech utterances come from different speakers and *spoofing* impostor trials, where the test utterances

contain deliberate spoofing. The corresponding FARs are computed exactly in the same way (5.2) and are called the *zero-effort false acceptance rate* (FAR(Z)) and *spoofing false acceptance rate* (FAR(S)). If need be, a further granularization of the FAR(S) could be achieved by splitting a spoofing trial set according to different attack types, assuming such metadata are available.

Another measure used in this dissertation to evaluate a *stand-alone* countermeasure performance is the *spoofing detection error* (SDE), which measures the total error rate of a system and was defined in **II** as

$$\mathrm{SDE}(\mathcal{S}_{\mathrm{tgt}}, \mathcal{S}_{\mathrm{imp}}|\theta) = \frac{|\mathcal{S}_{\mathrm{tgt}}|\mathrm{FRR}(\mathcal{S}_{\mathrm{tgt}}|\theta) + |\mathcal{S}_{\mathrm{imp}}|\mathrm{FAR}(\mathcal{S}_{\mathrm{imp}}|\theta)}{|\mathcal{S}_{\mathrm{tgt}}| + |\mathcal{S}_{\mathrm{imp}}|}\,. \tag{5.3}$$

The evaluation measures described so far are all threshold-dependent ones. As a result, they measure the *discrimination* and *calibration* abilities of a system [97]. The latter refers to the ability to set up good decision thresholds for a particular application of a system, whereas the former refers to the ability to differentiate target trials from impostor trials. These application-specific requirements might not be known during the development stage. Therefore, it makes sense to focus only on the discrimination abilities of a system during development. The most common threshold-independent measure used in biometrics is the *equal error rate* (EER), which is a value of $\mathrm{FRR}(\bullet|\hat{\theta})$ for such $\hat{\theta}$ that $\mathrm{FRR}(\bullet|\hat{\theta}) = \mathrm{FAR}(\bullet|\hat{\theta})$. In practice, because the FRR and FAR change in discrete steps, exact equality is not guaranteed and interpolations, such as the *receiver operating characteristic convex hull* (ROCCH-EER) method [98, 99], are recommended.

So far, we have treated false rejection and false acceptance errors as being equally important (or equally dangerous) to the system, but for most biometric systems, especially those that focus on security, this is clearly not true. A false acceptance error, which in a security context means letting an intruder into the system, usually has a higher penalty than a false rejection error does. One of the evaluation measures that incorporate this knowledge together with the probabilities of both legitimate and adversarial access attempts is a *detection cost function* (DCF):

$$\mathrm{DCF}(\mathcal{S}_{\mathrm{tgt}}, \mathcal{S}_{\mathrm{imp}}|\theta) = C_{\mathrm{tgt}}P_{\mathrm{tgt}}\mathrm{FRR}(\mathcal{S}_{\mathrm{tgt}}|\theta) + C_{\mathrm{imp}}(1 - P_{\mathrm{tgt}})\mathrm{FAR}(\mathcal{S}_{\mathrm{imp}}|\theta)\,, \tag{5.4}$$

where C_{tgt} and C_{imp} are the costs of each types of error, and P_{tgt} is the prior probability of the legitimate access attempts. As discussed above, during the development stage, we try not to focus too much on calibration of a system and hence use a minimum of the DCF over all possible threshold values as an evaluation metric.

5.2 MULTI-CLASS PROBLEMS

In contrast to the binary tasks, we are required to check the whole set of hypotheses in this case. Spoken language identification is a typical example, where a different hypothesis represents every valid language, with the possible addition of another hypothesis that represents unknown languages. For such tasks, a score is a vector of the dimensionality equal to the number of hypotheses in question. This change affects in multiple ways the computation of the evaluation metrics. Some of them now require normalization, such as the LLR or DCF. Meanwhile, others undergo more profound modifications, such as the FAR, which now depends on two sets of

class scores instead of just one set. [1]:

$$\mathrm{LLR}(\mathcal{C}_i|s_n) = \log \frac{s_n^{(i)}}{\frac{1}{M}\sum\limits_{\substack{j\in\{1,\ldots,M\}\\ j\neq i}} p(s_n|\mathcal{C}_j)}\,, \tag{5.5}$$

$$\mathrm{FRR}(\mathcal{S}_i|\theta) = \frac{1}{\mathcal{S}_i}\sum_{s_n\in\mathcal{S}_i} \mathrm{I}\{\mathrm{LLR}(\mathcal{C}_i|s_n) < \theta\}, \tag{5.6}$$

$$\mathrm{FAR}(\mathcal{S}_i,\mathcal{S}_j|\theta) = \frac{1}{\mathcal{S}_i}\sum_{s_n\in\mathcal{S}_i} \mathrm{I}\{\mathrm{LLR}(\mathcal{C}_j|s_n) \geqslant \theta\}\,, \tag{5.7}$$

where M is the number of valid classes: $\mathcal{C}_1,\ldots,\mathcal{C}_M$ and a score vector of n-th trial, $s_n = (s_n^{(1)},\ldots,s_n^{(M)})^\mathsf{T}$. Assuming that both types of errors have an equal cost and that all classes have equal prior probabilities, we arrive at the following:

$$\mathrm{DCF}(\mathcal{S}_1,\ldots,\mathcal{S}_M|\theta) = \frac{1}{2M}\left(\sum_{i=1}^{M}\mathrm{FRR}(\mathcal{S}_i|\theta) + \frac{1}{M-1}\sum_{\substack{(i,j)\in\{1,\ldots,M\}^2\\ i\neq j}}\mathrm{FAR}(\mathcal{S}_i,\mathcal{S}_j|\theta)\right). \tag{5.8}$$

In publication **III**, the author used a modified version of (5.8) as an objective function during optimization. In particular, the indicator functions in equations (5.6)–(5.7) were substituted with *hinge functions* to make the objective function piece-wise differentiable, and the result underwent a *weak-sense auxiliary function* [49] optimization process.

[1] `https://www.nist.gov/sites/default/files/documents/2016/10/06/lre15_evalplan_v23.pdf`

6 SUMMARY OF PUBLICATIONS AND RESULTS

Publications **I** and **II** focused on ASV, whereas publication **III** focused on language recognition. Moreover, publication **II** addressed the problem of spoofed speech detection in the context of ASV, proposing a *back-end* approach, where i-vectors were used as input data and we represented artificial speech artifacts via a special latent subspace.

Table 6.1: Summary of the speech corpora

Corpus	Description	Used in publication
I4U'12	Female subset of I4U dataset prepared for NIST SRE'12. It contains 578 training speakers with 21,216 segments, 459 test speakers that give 10,524 target trials and 6,061,824 non-target trials.	**I**
SRE'06	The corpus is based on the core task "1conv4w-1conv4w" of NIST'06. It consists of telephony speech. The training set was modified via the *copy-synthesis* approach to generate MCEP- and LPC-coded speech without undergoing any specific VC technique. The test set comprises 2 VC techniques: *joint-density Gaussian mixture model* (JD-GMM) method and a simplified *unit selection* (US) method. The training set consists of 12,372 utterances of natural speech and their MCEP and LPC modifications. There are: 3,946 genuine trials, 2,747 zero-effort impostor trials, 2,747 MCEP- and LPC-based JD-GMM impostor trials, and 2,747 FS impostor trials.	**II**
LRE'15	This is a multi-language corpus that contains 20 languages from 6 language clusters: Arabic, Chinese, English, Slavic, French, and Iberian. It contains both conversational telephone speech and broadcast narrowband speech data. All our experiments with this corpus used a modified version of the development set, where long utterances were cut into several smaller ones in addition to being used on their own. The total amount of development and evaluation utterances, respectively, are 168,791 and 164,334.	**III**
I2R Dev	The corpus is based on data composed of LREs 96, 03, 05, 07, 09 and 11. It was developed to represent the LRE'15 corpus as closely as possible, but due to the lack of suitable data, it contains only 13 different languages from the Arabic, Chinese, English, and Slavic language clusters. The development part contains 19,326 utterances and the evaluation part contains 13,947 utterances.	**III**

6.1 SUMMARY OF PUBLICATIONS

In **I**, we reviewed and analyzed three variants of PLDA and their application to the ASV task. We provided a unified formulation, equations, and implementations[1] for all three methods.

In **II**, we took a back-end approach to artificial speech detection, namely *voice conversion* (VC). One of the most desired and challenging properties of artificial speech detectors is their robustness against *unseen* types of attacks. Hence, we focused on *generative* models due to their innate ability to provide better generalization compared with discriminative models. Specifically, we modeled the *synthesis-channel* subspace to perform speaker verification and anti-spoofing jointly in the i-vector space, which is a well-established technique for speaker modeling. The proposed approach enabled us to integrate speaker verification and anti-spoofing tasks into a unified system without resorting to any late (score) fusion techniques. To validate the proposed approach, we studied both *vocoder-matched* and *vocoder-mismatched* ASV and VC spoofing detection, using a subset of the NIST 2006 speaker recognition evaluation (SRE) corpus processed through simple voice conversion techniques.

In **III**, we worked on a different task, language recognition, and proposed a new method to boost discriminative capabilities of the PLDA model without losing its generative advantages. We showed a sequential projection and training steps leading to a classifier that operates in the original i-vector space but is discriminatively trained in a low-dimensional PLDA latent subspace. To achieve this goal, we used the extended Baum-Welch technique to optimize the model with respect to two alternative objective functions for discriminative training. The first one is the well-known MMI objective, whereas the second one is a novel objective designed to directly approximate the language detection cost defined in the NIST LRE campaigns. We evaluated the performance using NIST LRE 2015 and the "Fantastic 4" development dataset comprised of the utterances from previous LREs. The proposed approximation method of the cost function and PLDA subspace training are applicable for a broad range of tasks.

6.2 SUMMARY OF RESULTS

In **I**, we compared the *standard*, *simplified* and *two-covariance* PLDA variants, and analyzed them in terms of the predictive power they provide. Our experimental results suggest that it is better to use the simplest possible model suited for the particular application. We presented scalable training algorithms for all the three models, including a publicly available implementation.

In **II**, we separately evaluated the accuracy of speaker verification, the spoofing detector and the joint system. Concerning **standalone speaker verification**, the i-vector PLDA approach (EER = 0.81% for female and EER = 0.54% for male) outperformed the two other techniques at differentiating target and zero-effort impostor trials. Under spoofing, however, its overall FAR increased by a factor of 28 for the female subset and by a factor of 47 for the male subset, confirming that i-vector PLDA systems without countermeasures are highly vulnerable to voice conversion attacks. Concerning the **standalone spoofing detector**, we found the cosine scoring of i-vectors and two-stage PLDA systems to be the most stable across different conditions. Regarding the two types of attacks, linear predictive (LP) coded attacks

[1] `https://sites.google.com/site/fastplda/`

were easier to detect even in the mismatched case, whereas mel-cepstral (MCEP) coded attacks were more challenging. Concerning the experiments on **joint speaker verification and anti-spoofing**, the new joint approach of modeling an additional synthesis-channel subspace outperformed other considered methods and showed promising generalization abilities to the unseen cases.

In **III**, training models in the low-dimensional PLDA latent subspace consistently outperformed training them in a standard LDA space for all six language clusters. The proposed technique yielded a 5% relative improvement for MMI optimization and 9% for direct cost optimization. The combination of the PLDA latent subspace and direct cost optimization led to a 9% relative improvement over the best generative system and a 10% relative improvement over our best discriminative system reported earlier.

7 CONCLUSION

This PhD dissertation work has addressed a number of basic research problems in the domain of speaker and language recognition. The work involved the development of a new discriminative training method to improve upon state-of-the-art language recognition, including a novel way to optimize the NIST LRE cost function. Additionally, it developed a new type of framework for automatic speaker verification with a built-in spoofing attack (specifically, voice conversion attacks) resistance. The methods were extensively evaluated on standard NIST benchmark tests using standardized evaluation measures and were compared with other competing state-of-the-art methods. In this concluding section, the author will not repeat the findings that were summarized in the previous chapter. Instead, however, we take a brief look at the limitations of the work in addition to looking into certain future horizons.

The methods studied in this work are all based on factor analysis and hence inherit its limitations, namely the need for carefully selected and preprocessed features, and the linearity of input-output mapping (that can be partially alleviated by using mixtures of factor analysers [100]). The domination of factor analysis models in the field of speaker and language recognition for a decade shows, however, that these constraints were not sufficient for being a limiting factor. However, it is still clear that the current trend is slowly shifting toward neural networks, much deeper nonlinear models that can automatically learn suitable feature representations. Current state-of-the-art systems have very clever ways to integrate both approaches into the same pipeline but, as the amount of knowledge and understanding of neural network principles grows, neural networks might completely substitute factor analysis models.

The prospects of speaker anti-spoofing also look gloomy. The success of current countermeasures could be explained by crude spoofing methods that try to mimic only the most distinguishable characteristics of natural speech and leave behind many easily identifiable artifacts in spoofed speech, such as the complete disregard of phase information or the concatenation of independent waveform blocks. The latest generation of speech synthesis methods, spearheaded by WaveNet [76], does not seem to have such obvious flaws. It is an open question as to how well current and future countermeasures will be able to cope with it. The same could be said about replay attacks because any replay attack is in essence a natural speech utterance that just undergoes an additional recording and playback stages. It is not clear if it would be possible to tell a difference between natural speech recorded once from the one recorded twice given a sufficiently high quality of recording and playback devices. A natural solution against playback attacks of such quality would be a usage of new randomized phrases for each verification attempt but this would not suffice against a more sofisticated attacks based on speech synthesis or voice conversion. In summary, the author envisions that the progress in spoofing methods will eventually lead to the reduced applicability of speaker recognition technologies in certain areas, especially as a *unimodal* verification procedure without additional security checks like *verbal information verification* [101] which is a process of verifying

spoken utterances against the information stored in a given personal data profile. Similarly to the effects of *generative adversarial learning* [102] in *deep learning*, a further arms race between spoofing and anti-spoofing solutions will probably lead to the increased robustness and accuracy of *stand-alone* speaker recognition performance in the areas that is likely to survive and thrive, such as *diarization* [103] or police surveillance.

BIBLIOGRAPHY

[1] T. Pekhovsky and A. Sizov, "Comparison between supervised and unsupervised learning of probabilistic linear discriminant analysis mixture models for speaker verification," *Pattern Recognition Letters* **34**, 1307–1313 (2013).

[2] E. Khoury, T. Kinnunen, A. Sizov, Z. Wu, and S. Marcel, "Introducing i-vectors for joint anti-spoofing and speaker verification," in *Interspeech* (2014), pp. 61–65.

[3] Z. Wu, T. Kinnunen, N. Evans, J. Yamagishi, C. Hanilçi, M. Sahidullah, and A. Sizov, "ASVspoof 2015: the First Automatic Speaker Verification Spoofing and Countermeasures Challenge," in *Interspeech* (2015), pp. 2037–2041.

[4] K. A. Lee, V. Hautamäki, A. Larcher, W. Rao, et al., "Fantastic 4 system for NIST 2015 Language Recognition Evaluation," (2015), `http://arxiv.org/abs/1602.01929`.

[5] K. A. Lee, H. Li, L. Deng, et al., "The 2015 NIST Language Recognition Evaluation: the Shared View of I2R, Fantastic4 and SingaMS," in *Interspeech* (2016), pp. 3211–3215.

[6] A. Sizov, K. A. Lee, and T. Kinnunen, "Discriminating Languages in a Probabilistic Latent Subspace," in *Odyssey: the Speaker and Language Recognition Workshop* (2016), pp. 81–88.

[7] C. Hanilçi, T. Kinnunen, M. Sahidullah, and A. Sizov, "Spoofing detection goes noisy: An analysis of synthetic speech detection in the presence of additive noise," *Speech Communication* **85**, 83–97 (2016).

[8] Z. Wu, J. Yamagishi, T. Kinnunen, C. Hanilçi, M. Sahidullah, A. Sizov, N. Evans, and M. Todisco, "ASVspoof: the automatic speaker verification spoofing and countermeasures challenge," *IEEE Journal of Selected Topics in Signal Processing* **11**, 588–604 (2017).

[9] K. A. Lee and S. I. Group, "The I4U Mega Fusion and Collaboration for NIST Speaker Recognition Evaluation 2016," in *Interspeech* (2017), pp. 1328–1332.

[10] D. Reynolds and R. Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models," *IEEE Transactions on Speech and Audio Processing* **3**, 72–83 (1995).

[11] H. Li, B. Ma, and K. A. Lee, "Spoken language recognition: from fundamentals to practice," *Proceedings of the IEEE* **101**, 1136–1159 (2013).

[12] V. Shchemelinin and K. Simonchik, "Examining vulnerability of voice verification systems to spoofing attacks by means of a TTS system," in *International Conference on Speech and Computer* (2013), pp. 132–137.

[13] Z. Wu, A. Larcher, K.-A. Lee, E. Chng, T. Kinnunen, and H. Li, "Vulnerability evaluation of speaker verification under voice conversion spoofing: the effect of text constraints.," in *Interspeech* (2013), pp. 950–954.

[14] R. G. Hautamäki, T. Kinnunen, V. Hautamäki, and A.-M. Laukkanen, "Automatic versus human speaker verification: The case of voice mimicry," *Speech Communication* **72**, 13–31 (2015).

[15] W. Shang and M. Stevenson, "Score normalization in playback attack detection," in *ICASSP* (IEEE, 2010), pp. 1678–1681.

[16] P. L. De Leon, V. R. Apsingekar, M. Pucher, and J. Yamagishi, "Revisiting the security of speaker verification systems against imposture using synthetic speech," in *ICASSP* (IEEE, 2010), pp. 1798–1801.

[17] Y. W. Lau, M. Wagner, and D. Tran, "Vulnerability of speaker verification to voice mimicking," in *Intelligent Multimedia, Video and Speech Processing, 2004. Proceedings of 2004 International Symposium on* (2004), pp. 145–148.

[18] T. Kinnunen, Z.-Z. Wu, K. A. Lee, F. Sedlak, E. S. Chng, and H. Li, "Vulnerability of speaker verification systems against voice conversion spoofing attacks: The case of telephone speech," in *ICASSP* (IEEE, 2012), pp. 4401–4404.

[19] P. Patrick, G. Aversano, R. Blouet, M. Charbit, and G. Chollet, "Voice forgery using ALISP: indexation in a client memory," in *ICASSP* (IEEE, 2005), pp. 17–20.

[20] S. K. Ergünay, E. Khoury, A. Lazaridis, and S. Marcel, "On the vulnerability of speaker verification to realistic voice spoofing," in *Biometrics Theory, Applications and Systems (BTAS)* (IEEE, 2015), pp. 1–6.

[21] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-End Factor Analysis for Speaker Verification," *IEEE Trans. on Audio, Speech, and Language Processing* **19**, 788–798 (2011).

[22] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Transactions on Acoustics, Speech, and Signal Processing* **28**, 357–366 (1980).

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

[24] M. Sahidullah, T. Kinnunen, and C. Hanilçi, "A comparison of features for synthetic speech detection," in *Interspeech* (2015), pp. 2087–2091.

[25] S. Chakroborty, A. Roy, and G. Saha, "Improved closed set text-independent speaker identification by combining MFCC with evidence from flipped filter banks," *International Journal of Signal Processing* **4**, 114–122 (2007).

[26] J. Youngberg and S. Boll, "Constant-Q signal analysis and synthesis," in *ICASSP* (IEEE, 1978), pp. 375–378.

[27] J. Brown, "Calculation of a constant Q spectral transform," *The Journal of the Acoustical Society of America* **89**, 425–434 (1991).

[28] J. Powell, *How music works* 2010).

[29] M. Todisco, H. Delgado, and N. Evans, "A new feature for automatic speaker verification anti-spoofing: Constant Q cepstral coefficients," in *Odyssey: the Speaker and Language Recognition Workshop* (2016), pp. 249–252.

[30] G. Lavrentyeva, S. Novoselov, E. Malykh, A. Kozlov, O. Kudashev, and V. Shchemelinin, "Audio-replay attack detection countermeasures," *arXiv preprint arXiv:1705.08858* (2017).

[31] J. Makhoul, "Linear prediction: A tutorial review," *Proceedings of the IEEE* **63**, 561–580 (1975).

[32] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, et al., *The HTK book* 2006).

[33] B. Bielefeld, "Language identification using shifted delta cepstrum," in *Fourteenth Annual Speech Research Symposium* (1994).

[34] P. Torres-Carrasquillo, E. Singer, M. Kohler, R. Greene, D. Reynolds, and J. Deller Jr, "Approaches to language identification using Gaussian mixture models and shifted delta cepstral features," in *Interspeech* (2002).

[35] W. Campbell, D. Sturim, and D. Reynolds, "Support vector machines using GMM supervectors for speaker verification," *IEEE Signal Processing Letters* **13**, 308–311 (2006).

[36] P. Torres-Carrasquillo, N. Dehak, E. Godoy, D. Reynolds, F. Richardson, S. Shum, E. Singer, and D. Sturim, "The MITLL NIST LRE 2015 Language Recognition System," in *Odyssey: the Speaker and Language Recognition Workshop* (2016), pp. 196–203.

[37] S. Novoselov, A. Kozlov, G. Lavrentyeva, K. Simonchik, and V. Shchemelinin, "STC anti-spoofing systems for the ASVspoof 2015 challenge," in *ICASSP* (IEEE, 2016), pp. 5475–5479.

[38] G. Saon, H. Soltau, D. Nahamoo, and M. Picheny, "Speaker adaptation of neural network acoustic models using i-vectors.," in *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)* (2013), pp. 55–59.

[39] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *ICASSP* (IEEE, 2014), pp. 1695–1699.

[40] "Deep Neural Networks for extracting Baum-Welch statistics for Speaker Recognition," in *Odyssey: the Speaker and Language Recognition Workshop* (2014), pp. 293–298.

[41] T. Patel and H. Patil, "Combining evidences from mel cepstral, cochlear filter cepstral and instantaneous frequency features for detection of natural vs. spoofed speech," in *Interspeech* (2015), pp. 2062–2066.

[42] S. Novoselov, A. Kozlov, G. Lavrentyeva, K. Simonchik, and V. Shchemelinin, "STC anti-spoofing systems for the ASVspoof 2015 challenge," in *ICASSP* (IEEE, 2016), pp. 5475–5479.

[43] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using Adapted Gaussian mixture models," in *Digital Signal Processing* (2000).

[44] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the royal statistical society. Series B (methodological)* 1–38 (1977).

[45] C. Bishop, *Pattern Recognition and Machine Learning* 2006).

[46] S. Prince and J. Elder, "Probabilistic Linear Discriminant Analysis for Inferences About Identity," in *IEEE Int. Conf. on Computer Vision (ICCV)* (2007).

[47] P. Kenny, "Bayesian speaker verification with heavy-tailed priors," in *Odyssey: the Speaker and Language Recognition Workshop* (2010), pp. 61–70.

[48] D. Martınez, O. Plchot, L. Burget, O. Glembek, and P. Matejka, "Language recognition in ivectors space," *Interspeech* 861–864 (2011).

[49] D. Povey, *Discriminative training for large vocabulary speech recognition*, PhD thesis (University of Cambridge, 2005).

[50] V. N. Vapnik, *The Nature of Statistical Learning Theory* 1995).

[51] D. Povey and P. Woodland, "Minimum phone error and I-smoothing for improved discriminative training," in *ICASSP* (IEEE, 2002), pp. 105–108.

[52] H. Zhao, D. Bansé, G. Doddington, C. Greenberg, J. Hernandez-Cordero, J. Howard, L. Mason, A. Martin, D. Reynolds, and E. Singer, "Results of The 2015 NIST Language Recognition Evaluation," in *Interspeech* (2016), pp. 3206–3210.

[53] N. Evans, T. Kinnunen, and J. Yamagishi, "Spoofing and countermeasures for automatic speaker verification.," in *Interspeech* (2013), pp. 925–929.

[54] Y. W. Lau, D. Tran, and M. Wagner, "Testing voice mimicry with the YOHO speaker verification corpus," in *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems* (2005), pp. 15–21.

[55] R. G. Hautamäki, T. Kinnunen, V. Hautamäki, and A.-M. Laukkanen, "Comparison of human listeners and speaker verification systems using voice mimicry data," in *Odyssey: the Speaker and Language Recognition Workshop* (2014), pp. 137–144.

[56] B. Pellom and J. Hansen, "An experimental study of speaker verification sensitivity to computer voice-altered imposters," in *ICASSP* (IEEE, 1999), pp. 837–840.

[57] J. Lindberg and M. Blomberg, "Vulnerability in speaker verification-a study of technical impostor techniques," in *Sixth European Conference on Speech Communication and Technology* (1999).

[58] L. Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research* **9**, 2579–2605 (2008).

[59] S. Panjwani and A. Prakash, "Crowdsourcing attacks on biometric systems," in *Symposium On Usable Privacy and Security* (2014), pp. 257–269.

[60] T. Masuko, T. Hitotsumatsu, K. Tokuda, and T. Kobayashi, "On the security of HMM-based speaker verification systems against imposture using synthetic speech," in *Eurospeech* (1999).

[61] P. L. De Leon, M. Pucher, and J. Yamagishi, "Evaluation of the vulnerability of speaker verification to synthetic speech," (2010).

[62] F. Alegre, R. Vipperla, N. Evans, and B. Fauve, "On the vulnerability of automatic speaker recognition to spoofing attacks with artificial signals," in *Signal processing conference (EUSIPCO), 2012 proceedings of the 20th European* (2012), pp. 36–40.

[63] G. Doddington, M. Przybocki, A. Martin, and D. Reynolds, "The NIST speaker recognition evaluation – overview, methodology, systems, results, perspective," *Speech Communication* **31**, 225–254 (2000).

[64] M. Abe, S. Nakamura, K. Shikano, and H. Kuwabara, "Voice conversion through vector quantization," *Journal of the Acoustical Society of Japan (E)* **11**, 71–76 (1990).

[65] A. Kain and M. Macon, "Spectral voice conversion for text-to-speech synthesis," in *ICASSP* (IEEE, 1998), pp. 285–288.

[66] A. Mouchtaris, J. Van der Spiegel, and P. Mueller, "Nonparallel training for voice conversion based on a parameter adaptation approach," *IEEE Trans. on Audio, Speech, and Language Processing* **14**, 952–963 (2006).

[67] T. Nakashika, T. Takiguchi, and Y. Minami, "Non-parallel training in voice conversion using an adaptive restricted Boltzmann machine," *IEEE/ACM Trans. on Audio, Speech, and Language Processing* **24**, 2032–2045 (2016).

[68] T. Kinnunen, L. Juvela, P. Alku, and J. Yamagishi, "Non-parallel voice conversion using i-vector PLDA: towards unifying speaker verification and transformation," in *ICASSP* (IEEE, 2017), pp. 5535–5539.

[69] F. Alegre, A. Amehraye, and N. Evans, "Spoofing countermeasures to protect automatic speaker verification from voice conversion," in *ICASSP* (IEEE, 2013), pp. 3068–3072.

[70] X. Xiao, X. Tian, S. Du, H. Xu, E. S. Chng, and H. Li, "Spoofing speech detection using high dimensional magnitude and phase features: The NTU approach for ASVspoof 2015 challenge," in *Interspeech* (2015), pp. 2052–2056.

[71] L. Wang, Y. Yoshida, Y. Kawakami, and S. Nakagawa, "Relative phase information for detecting human speech and spoofed speech," in *Interspeech* (2015), pp. 2092–2096.

[72] M. J. Alam, P. Kenny, G. Bhattacharya, and T. Stafylakis, "Development of CRIM system for the automatic speaker verification spoofing and countermeasures challenge 2015," in *Interspeech* (2015), pp. 2072–2076.

[73] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis," in *Sixth European Conference on Speech Communication and Technology* (1999).

[74] Y. Fan, Y. Qian, F.-L. Xie, and F. Soong, "TTS synthesis with bidirectional LSTM based recurrent neural networks," in *Interspeech* (2014), pp. 1964–1968.

[75] A. Hunt and A. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," in *ICASSP* (IEEE, 1996), pp. 373–376.

[76] A. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499* (2016).

[77] V. Shchemelinin and K. Simonchik, "Examining Vulnerability of Voice Verification Systems to Spoofing Attacks by Means of a TTS System," in *Proceeding of the 15th International Conference on Speech and Computer* (2013), pp. 132–137.

[78] X. Tian, Z. Wu, X. Xiao, E. S. Chng, and H. Li, "An investigation of spoofing speech detection under additive noise and reverberant conditions," in *Interspeech* (2016), pp. 1715–1719.

[79] K. Sriskandaraja, V. Sethu, E. Ambikairajah, and H. Li, "Front-End for Antispoofing Countermeasures in Speaker Verification: Scattering Spectral Decomposition," *IEEE Journal of Selected Topics in Signal Processing* **11**, 632–643 (2017).

[80] J. Alam and P. Kenny, "Spoofing Detection Employing Infinite Impulse Response - Constant Q Transform-based Feature Representations," in *European Signal Processing Conference* (2017), pp. 111–115.

[81] T. Kinnunen, M. Sahidullah, H. Delgado, M. Todisco, N. Evans, J. Yamagishi, and K. A. Lee, "The ASVspoof 2017 Challenge: Assessing the Limits of Replay Spoofing Attack Detection," in *Interspeech* (2017), pp. 2–6.

[82] J. Villalba and E. Lleida, "Speaker verification performance degradation against spoofing and tampering attacks," in *FALA 2010* (2010), pp. 131–134.

[83] J. Villalba, A. Miguel, A. Ortega, and E. Lleida, "Spoofing detection with DNN and one-class SVM for the ASVspoof 2015 challenge," in *Interspeech* (2015), pp. 2067–2071.

[84] C. Hanilçi, T. Kinnunen, M. Sahidullah, and A. Sizov, "Classifiers for synthetic speech detection: A comparison," in *Interspeech* (2015), pp. 2057–2061.

[85] J. Sanchez, I. Saratxaga, I. Hernaez, E. Navas, and D. Erro, "The AHOLAB RPS SSD spoofing challenge 2015 submission," in *Interspeech* (2015), pp. 2042–2046.

[86] P. Korshunov and S. Marcel, "Cross-database evaluation of audio-based spoofing detection systems," in *Interspeech* (2016), pp. 1705–1709.

[87] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proceedings of the 31st International Conference on Machine Learning (ICML-14)* (2014), pp. 1764–1772.

[88] H. Dinkel, N. Chen, Y. Qian, and K. Yu, "End-to-end spoofing detection with raw waveform CLDNNS," in *ICASSP* (IEEE, 2017), pp. 4860–4864.

[89] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on* (2015), pp. 4580–4584.

[90] P. Korshunov, S. Marcel, H. Muckenhirn, A. Gonçalves, A. S. Mello, R. V. Violato, F. O. Simões, M. Neto, M. de Assis Angeloni, J. A. Stuchi, et al., "Overview of BTAS 2016 speaker anti-spoofing competition," in *Biometrics Theory, Applications and Systems (BTAS), 2016 IEEE 8th International Conference on* (2016), pp. 1–6.

[91] G. Lavrentyeva, E. Novoselov, Sergey Malykh, A. Kozlov, O. Kudashev, and V. Shchemelinin, "Audio replay attack detection with deep learning frameworks," in *Interspeech* (2017), pp. 82–86.

[92] C. Zhang, C. Yu, and J. H. Hansen, "An Investigation of Deep-Learning Frameworks for Speaker Verification Antispoofing," *IEEE Journal of Selected Topics in Signal Processing* **11**, 684–694 (2017).

[93] N. Brummer, J. Cernocky, M. Karafiát, D. A. van Leeuwen, P. Matejka, P. Schwarz, A. Strasheim, et al., "Fusion of heterogeneous speaker recognition systems in the STBU submission for the NIST speaker recognition evaluation 2006," *IEEE Trans. on Audio, Speech, and Language Processing* **15**, 2072–2084 (2007).

[94] S. Pigeon, P. Druyts, and P. Verlinde, "Applying logistic regression to the fusion of the NIST'99 1-speaker submissions," *Digital Signal Processing* **10**, 237–248 (2000).

[95] I. Chingovska, A. Anjos, and S. Marcel, "Anti-spoofing in action: joint operation with a verification system," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops* (2013), pp. 98–104.

[96] M. Sahidullah, H. Delgado, M. Todisco, H. Yu, T. Kinnunen, N. W. Evans, and Z.-H. Tan, "Integrated Spoofing Countermeasures and Automatic Speaker Verification: An Evaluation on ASVspoof 2015.," in *Interspeech* (2016), pp. 1700–1704.

[97] N. Brummer and D. A. Van Leeuwen, "On calibration of language recognition scores," in *Odyssey: the Speaker and Language Recognition Workshop* (2006), pp. 1–8.

[98] F. Provost and T. Fawcett, "Robust classification for imprecise environments," *Machine learning* **42**, 203–231 (2001).

[99] N. Brümmer and E. de Villiers, "The BOSARIS toolkit user guide: Theory, algorithms and code for binary classifier score processing," *Documentation of BOSARIS toolkit* (2011).

[100] Z. Ghahramani, G. E. Hinton, et al., "The EM algorithm for mixtures of factor analyzers," (1996).

[101] Q. Li, B.-H. Juang, and C.-H. Lee, "Automatic verbal information verification for user authentication," *IEEE transactions on speech and audio processing* **8**, 585–596 (2000).

[102] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems* (2014), pp. 2672–2680.

[103] X. Anguera, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland, and O. Vinyals, "Speaker diarization: A review of recent research," *IEEE Trans. on Audio, Speech, and Language Processing* **20**, 356–370 (2012).

A ERRATA

In **I**, there is an error on page 4 stating that for the standard PLDA model to be equivalent to the two-covariance model or the simplified PLDA model the following equality should hold: $M = D - 1$. The correct equation is as follows:

$$M = \frac{1 + 2D - \sqrt{8D + 1}}{2}. \tag{A.1}$$

On the same page, the author of this thesis made a claim that the standard PLDA is the most general one, without a precise definition of "generality". By this statement, the intention was to highlight that the two other models can be obtained as special cases of the standard PLDA model. An alternative view would be that the least constrained formulation is the most general which would point to the two-covariance model.

ALEKSANDR SIZOV

This work contains three case studies that developed and advanced methods for speaker verification and language identification, with a special emphasis on channel robustness and the ability to withstand attempts of unauthorized access to systems protected by speaker verification, called spoofing.

UNIVERSITY OF
EASTERN FINLAND

uef.fi

PUBLICATIONS OF
THE UNIVERSITY OF EASTERN FINLAND
Dissertations in Forestry and Natural Sciences

ISBN 978-952-61-2687-6
ISSNL 1798-5668