Voice conversion with parallel/non-parallel data and synthetic speech detection

Tian, Xiaohai

2019

Tian, X. (2019). Voice conversion with parallel/non-parallel data and synthetic speech detection. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/89880>

<https://doi.org/10.32657/10220/47729>

VOICE CONVERSION WITH
PARALLEL/NON-PARALLEL DATA AND
SYNTHETIC SPEECH DETECTION

Xiaohai Tian

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

2019

VOICE CONVERSION WITH
PARALLEL/NON-PARALLEL DATA AND
SYNTHETIC SPEECH DETECTION

XIAOHAI TIAN

School of Computer Science and Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfilment of the requirement for the degree of
Doctor of Philosophy

2019

Statement of Originality

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

01/02/2019
Date

.....
Tian Xiaohai

Supervisor Declaration Statement

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

20/9-Feb-01
.....
Date

.....
A/P: Chng Eng Siong

Authorship Attribution Statement

This thesis contains material from 1 paper(s) published in the following peer-reviewed journal(s) where I was the first and/or corresponding author.

Chapter 3 is published as Xiaohai Tian, Siu Wa Lee, Zhizheng Wu, Eng Siong Chng and Haizhou Li. "An Exemplar-based Approach to Frequency Warping for Voice Conversion". *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 25.10 (2017): 1863-1876.

The contributions of the co-authors are as follows:

- A/Prof Eng Siong Chng provided the initial project direction and edited the manuscript drafts.
- I prepared the manuscript drafts. The manuscript was revised by Dr Siu Wa Lee, Dr. Zhizheng Wu and Dr. Haizhou Li.
- I co-designed the study with Dr. Zhizheng Wu.
- I performed all the data preparation, system development, conducted all the experiments and analyzed experimental results.

01/02/2019
Date

Tian Xiaohai

Abstract

The objective of voice conversion techniques is to convert a source speaker’s voice so that it sounds like that of a target speaker. Voice conversion belongs to a popular area of personalized speech generation. On one hand, it can be applied to solve problems, e.g. emotion conversion, improving the intelligibility of speech, or change whisper/murmur into speech. On the other hand, voice conversion also presents a threat to Automatic Speaker Verification (ASV) systems. Synthetic speech detection is a technique to discriminate between live and synthetic speech. In this sense, it provides a feasible way to improve robustness and protect ASV systems. This thesis focuses on the following two aspects: improving performance in voice conversion and studying countermeasures for synthetic speech detection.

For voice conversion, a typical system usually requires parallel data, which means that the source and target utterances contain the same linguistic content for conversion model training. As parallel data is always difficult to collect in practice, the challenge is to estimate a robust conversion function with limited parallel data or even non-parallel data. In this thesis, we propose two novel voice conversion methods to improve the system performance for both parallel and non-parallel data based voice conversion.

First, we propose a novel parallel data based voice conversion framework, in which we combine frequency warping and exemplar-based methods. Under an exemplar-based frequency warping (EFW) framework, each warping function and spectral residual is generated from dictionaries of warping function exemplars and spectral residual exemplars respectively. With sparsity constraint, we avoid statistical averaging effects due to Gaussian Mixture Models (GMM) and provide a more accurate warping function and residual compensation. Results from experiments on VOICES database show that the proposed method significantly improved speech quality compared with state-of-the-art parametric methods.

Second, we propose a non-parallel data based voice conversion framework that we call the Average Modeling Approach (AMA). The proposed approach makes use of a multi-speaker average model that maps speaker independent linguistic features to speaker dependent acoustic features. As linguistic features and acoustic features can be extracted from the same utterance, the proposed approach doesn't require parallel data for average model training and adaptation. Two average model adaptation approaches are introduced, namely model-based adaptation and feature-based adaptation. Experiments on the Voice Conversion Challenge 2018 (VCC2018) database confirmed the effectiveness of the proposed method.

For synthetic speech detection, we investigated the use of different feature representations to discriminate between live and synthetic speech. As low-dimensional features are typically used in synthetic techniques, detailed information has been abandoned in synthetic speech. This work mainly focused on the detailed information of high-dimensional features, especially their high-frequency parts, for synthetic speech detection. Dynamic features were also calculated to assess the effectiveness of temporal information to detect artifacts across frames. Results of experiments on the standard ASVspoof 2015 corpus suggest that high-dimensional features and dynamic features are useful for synthetic speech detection.

Acknowledgments

These past four years have been a wonderful experience for me. I could not have completed this work successfully without the love and support from a lot of people around me. Firstly, I would like to express my sincere thanks to my supervisor Dr. Chng Eng Siong (Nanyang Technological University, NTU) and co-supervisor Dr. Lee Siu Wa (Institute for Infocomm Research, I²R) for their support and advice. Their suggestions and discussions helped me to overcome the difficulties in my research.

I would also like to express my appreciation to Prof. Li Haizhou and Dr. Wu Zhizheng for all their guidance and collaboration. They were like co-supervisors to me, discussing research ideas and revising conference and journal papers, which are the important parts of my Ph.D research.

Moreover, I also want to thank to my colleagues in Human Language Technology department of I²R for their generous help. Also, thanks to all my colleagues at speech team and NTU-UBC Research Center of Excellence in Active Living for the Elderly (LILY) in NTU. It is a great pleasure to meet all these friends and discuss different technical aspects of the project and research work.

Finally, I want to express my warm gratitude to all my family and my wife for their constant love and encouragement.

Lists of Publications

Journal:

- **Xiaohai Tian**, Siu Wa Lee, Zhizheng Wu, Eng Siong Chng and Haizhou Li. “An Exemplar-based Approach to Frequency Warping for Voice Conversion”. IEEE/ACM Transactions on Audio Speech and Language Processing, 2017.

Conference:

- **Xiaohai Tian**, Zhizheng Wu, Siu-Wa Lee and Eng Siong Chng, ”Correlation-based FrequencyWarping for Voice Conversion”, in Proceedings of the International Symposium on Chinese Spoken Language Processing (ISCSLP) 2014.
- **Xiaohai Tian**, Zhizheng Wu, Siu-Wa Lee, Nguyen Quy Hy, Minghui Dong and Eng Siong Chng, ”Sparse representation for frequency warping based voice conversion”, in Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2015.
- **Xiaohai Tian**, Zhizheng Wu, Siu-Wa Lee, Nguyen Quy Hy, Minghui Dong and Eng Siong Chng, ”System Fusion for High-Performance Voice Conversion”, in Proceedings of INTERSPEECH 2015.
- Xiong Xiao, **Xiaohai Tian**, Steven Du, Haihua Xu, Eng Siong Chng and Haizhou Li, “Spoofing Speech Detection Using High Dimensional Magnitude and Phase Features: the NTU System for ASVspoof 2015 Challenge”, INTERSPEECH 2015.
- **Xiaohai Tian**, Steven Du, Xiong Xiao, Haihua Xu, Eng Siong Chng and Haizhou Li, ’Detecting Synthetic Speech Using Long Term Magnitude and Phase Information’, ChinaSip, 2015.

- **Xiaohai Tian**, Zhizheng Wu, Xiong Xiao, Eng Siong Chng and Haizhou Li, “Spoofing detection from a feature representation perspective”, International Conference on Acoustics, Speech, and Signal Processing (ICASSP) 2016.
- **Xiaohai Tian**, Zhizheng Wu, Xiong Xiao, Eng Siong Chng and Haizhou Li, “An investigation of spoofing speech detection under additive noise and reverberant conditions”, INTERSPEECH 2016.
- **Xiaohai Tian**, Xiong Xiao, Eng Siong Chng, Haizhou Li. “Spoofing speech detection using temporal convolutional neural network”. APSIPA 2016.
- **Xiaohai Tian**, Lei Meng, Siyuan Liu, Zhiqi Shen, Eng Siong Chng, Cyril Leung, Frank Guan, and Chunyan Miao. ”Novel Functional Technologies for Age-Friendly E-commerce.” In International Conference on Human Aspects of IT for the Aged Population 2017.
- **Xiaohai Tian**, Juchao Wang, Haihua Xu, Eng Siong Chng, Haizhou Li. “Average Modeling Approach to Voice Conversion with Non-Parallel Data”. Odyssey 2018.

Contents

Abstract	i
Acknowledgments	iii
Lists of Publications	iv
List of Figures	x
List of Tables	xiv
List of Abbreviations	xvi
1 Introduction	1
1.1 Background	1
1.2 Motivation	3
1.3 Contributions	4
1.4 Thesis Organization	6
2 Introduction to Voice Conversion and Synthetic Speech Detection	7
2.1 Overview of a Voice Conversion System	7
2.1.1 Speech Analysis and Synthesis	8
2.1.2 Parameterization	10
2.1.3 Feature Alignment	11
2.1.4 Feature Conversion	12
2.1.4.1 Spectral Feature Mapping	13
2.1.4.2 Prosodic Feature Conversion	13
2.1.5 Evaluation Matrices	14
2.1.5.1 Objective Evaluation	14
2.1.5.2 Subjective Evaluation	15
2.2 State-of-the-Art Spectral Feature Mapping Techniques for Voice Conversion	17

2.2.1	Codebook Mapping Method	17
2.2.2	Statistical Parametric Method	18
2.2.3	Neural Network Method	23
2.2.4	Frequency Warping Method	25
2.2.5	Exemplar-based Method	28
2.3	Synthetic Speech Detection	29
2.4	Summary	31

3 An Exemplar-based Approach to Frequency Warping for Parallel Data

	Voice Conversion	32
3.1	Motivation	32
3.1.1	Limitation of GMM-based Weighted Frequency Warping (WFW) and GMM-based Residual Compensation	33
3.2	Exemplar-based Frequency Warping with Residual Compensation	34
3.2.1	Dictionary Construction	36
3.2.2	Run-time Conversion	38
3.2.2.1	Exemplar-based Frequency Warping	40
3.2.2.2	Exemplar-based Residual Compensation	41
3.2.3	Exemplar-based Frequency Warping vs. Exemplar-based Sparse Representation	41
3.3	Evaluation	42
3.3.1	Database	42
3.3.2	System Setup	42
3.3.3	Evaluation Metrics	45
3.3.3.1	Objective Evaluation	45
3.3.3.2	Subjective Evaluation	46
3.3.4	Validate the configurations of EFW framework	46
3.3.4.1	Effects of Dictionary Size	46
3.3.4.2	Effects of Smoothing the Warping Function Dictionary .	47
3.3.4.3	CFW vs. AMF and DFW	49
3.3.4.4	GMM-based vs. Exemplar-based Residual Compensation	51

3.3.5	EFW Framework vs. State-of-the-Arts	53
3.3.5.1	Speech Quality Evaluation	53
3.3.5.2	Speaker Similarity Evaluation	55
3.4	Summary	56
4	Average Modeling Approach for Non-Parallel Data Voice Conversion	57
4.1	Motivation	57
4.2	Average Modeling Approach (AMA) to Voice Conversion	61
4.2.1	Phonetic PosteriorGram (PPG) based voice conversion	61
4.2.2	Average Modeling Approach (AMA)	63
4.3	Experimental Setup	66
4.3.1	Database and Feature Extraction	66
4.3.1.1	ASR System Training	66
4.3.1.2	Speaker Independent (SI) Average Model Training	66
4.3.2	Baselines and Setup	67
4.4	Evaluations	69
4.4.1	Objective Evaluation	69
4.4.1.1	Effect of Adaptation Setups on Model-based AMA	69
4.4.1.2	Comparative Studies Across Competitive Methods	70
4.4.2	Subjective Evaluation	70
4.4.2.1	M-AMA vs. F-AMA	71
4.4.2.2	AMAs vs. State-of-the-Arts	72
4.4.3	Evaluation Results in VCC2018	72
4.5	Summary	76
5	Synthetic Speech Detection from a Feature Representation Perspective	77
5.1	Motivation	77
5.2	Synthetic Speech Detection System	79
5.2.1	Feature Extraction	80
5.2.2	Feature Representation	83
5.2.3	Classifier	84
5.2.4	Fusion	84

5.3	Experiments	85
5.3.1	Database	85
5.3.2	Evaluation Metrics	86
5.3.3	Results on Development Set	86
5.3.4	Results on Evaluation Set	88
5.3.5	Fusion Results	90
5.4	Discussion	91
5.4.1	Magnitude vs Phase Features	91
5.4.2	High-Dimensional vs Low-Dimensional Features	92
5.4.3	High-Frequency vs Low-Frequency Features	93
5.4.4	Difficulty in Detecting Unit Selection-based Attacks	93
5.5	Summary	94
6	Conclusion and Future Direction	95
6.1	Contributions	95
6.1.1	An Exemplar-based Frequency Warping (EFW) Approach for Parallel Data Voice Conversion	96
6.1.2	Average Modeling Approach (AMA) for Non-Parallel Data Voice Conversion	97
6.1.3	Synthetic Speech Detection using High-Dimensional Features	98
6.2	Future Works	99
6.2.1	Voice conversion	99
6.2.2	Synthetic speech detection	100
	References	101

List of Figures

2.1	Block diagram of a voice conversion system.	8
2.2	Illustration of the VTLN-based frequency warping and piecewise linear frequency warping.	26
2.3	Block diagram of the run-time conversion for exemplar-based voice conversion.	29
2.4	Block diagram of a synthetic speech detection system with a speaker verification system.	30
3.1	Block diagram of the exemplar-based frequency warping with residual compensation framework. a) During dictionary construction, a source dictionary $\mathbf{A}$, a warping function dictionary $\mathbf{W}$, and a residual dictionary $\mathbf{R}$ are built; b) at run-time conversion, the activation matrix $\mathbf{H}$ is estimated between the run-time speech sample $\mathbf{X}$ and the source dictionary $\mathbf{A}$. The activation matrix $\mathbf{H}$ is then shared by the warping function dictionary $\mathbf{W}$ and the residual dictionary $\mathbf{R}$ when generating the final converted speech.	35
3.2	Illustration of the effects of temporal smoothing on warping function dictionaries estimated using DFW, AMF and CFW. (a), (c) and (e) shows a part of warping function dictionaries before smoothing, $\mathbf{W}'$, while (b), (d) and (f) are the respective warping function dictionaries after smoothing, $\mathbf{W}$	37
3.3	Spectral distortion as a function of dictionary size.	47
3.4	(a) Spectral distortion as a function of smoothing window size for warping function dictionary. (b) Mean square error (MSE) of delta feature as a function of smoothing window size for warping function dictionary.	48

3.5	Preference t-test result of speech quality with 95% confidence intervals for ECFW-ERC with and without smoothing the warping function dictionary.	49
3.6	Preference test results of speech quality with 95% confidence intervals for GMM-based and exemplar-based frequency warping methods without residual compensation. (a) GAMF vs. GCFW, (b) GDFW vs. GCFW, (c) EAMF vs. ECFW and (d) EDFW vs. ECFW.	50
3.7	Preference test results of speech quality with 95% confidence intervals for GMM-based and exemplar-based frequency warping methods with residual compensation. (a) GAMF-GRC vs. GCFW-GRC, (b) GDFW-GRC vs. GCFW-GRC, (c) EAMF-ERC vs. ECFW-ERC and (d) EDFW-ERC vs. ECFW-ERC.	51
3.8	Preference test results of speech quality with 95% confidence intervals for GMM-based and exemplar-based residual compensation with GCFW and ECFW systems.	52
3.9	Preference test results of speech quality with 95% confidence intervals for GMM-based and exemplar-based frequency warping frameworks.	54
3.10	Preference test results of speech quality with 95% confidence intervals for ECFW-ERC vs. parametric methods ML-GMM, ML-GMM-GV and DKPLS.	54
3.11	Comparison of speaker identification rates with 95% confidence intervals for different voice conversion methods.	55
4.1	Block diagram of non-parallel data voice conversion based on non-parallel data alignment.	58
4.2	Block diagram of non-parallel data voice conversion based on adaptation on parallel data based average model. An average model is first trained by the parallel data from multiple speaker pairs. Then the adaptation is performed by a small mount of non-parallel data from source and target speakers.	59
4.3	Block diagram of non-parallel data voice conversion based on speaker independent (SI) features. The conversion model is learned between the SI feature representations and speaker dependent (SD) acoustic features. . .	60

4.4	An example of the Phonetic Posteriorgram. The horizontal axis refers to the frame number, while the vertical axis refers to the phonetic classes. For example, in our case, the 42 dimension are corresponding to the 42 English phonemes.	61
4.5	Block diagram of PPG based voice conversion.	62
4.6	Block diagram of the model-based Average Modeling Approach (AMA) to voice conversion	64
4.7	Block diagram of feature-based Average Modeling Approach (AMA) to voice conversion	65
4.8	Comparison of Mel-Cepstral Distortions (MCDs) of different conversion methods. 'Source' indicates the MCD evaluated on source utterance without conversion.	70
4.9	Preference t-test result of speech quality with 95% confidence intervals between M-AMA and F-AMA.	71
4.10	Preference t-test result of speaker identity with 95% confidence intervals between M-AMA and F-AMA.	71
4.11	Preference t-test results of speech quality with 95% confidence intervals between proposed AMA and baseline methods.	72
4.12	Preference t-test results of speaker identity with 95% confidence intervals between proposed AMA and baseline methods.	73
4.13	Boxplot for the average MOS score of all speaker pairs. System N04 is the proposed F-AMA.	74
4.14	Boxplot for the average MOS score of same-gender conversion. System N04 is the proposed F-AMA.	74
4.15	Boxplot for the average MOS score of cross-gender conversion. System N04 is the proposed F-AMA.	75
4.16	The average results of similarity score over all speaker pairs. System N04 is the proposed F-AMA.	75
5.1	Architecture of the proposed synthetic speech detection system.	80
5.2	Demonstration of the 6 types of features for utterance D16_1000404, which is a natural speech from the development set.	81

5.3	Score distribution of the MGD feature of HD-D representation.	88
5.4	Score distribution of the LMS feature of HD-D representation.	90
5.5	Demonstration of the high- and low-dimensional representations of of IF and GD for utterance D16_1000404, which is a natural speech from the development set.	92

List of Tables

3.1	<i>The summary of three categories of systems in seven groups. In category (I), we report the state-of-the-art parametric approaches. In category (II), we report a cluster of GMM-based frequency warping techniques with and without residual compensation. In category (III), we compare the proposed EFW approaches with the GMM counterpart in category (II).</i>	43
3.2	<i>Spectral distortion of exemplar-based frequency warping methods with and without temporal smoothing (window size 1 vs. 15) on the evaluation set.</i>	48
4.1	<i>Mel-Cepstral Distortions (MCDs) as a function of different adaptation settings for model-based AMA.</i>	69
5.1	<i>Summary of the spoofing attacks of ASVspoof 2015 database.</i>	85
5.2	<i>Averaged EERs (%) of different features on the development set.</i>	86
5.3	<i>Averaged EERs (%) of different features on the evaluation set.</i>	89
5.4	<i>EERs (%) of fused system on both development and evaluation sets. Pre-system indicates the results of our previous work [1].</i>	89

List of Abbreviation

AMA	Average Modeling Approach
AMF	Automatic Mapping of Formants
AMT	Amazon Mechanical Turk
AS	Amplitude Scaling
ASR	Automatic Speech Recognition
ASV	Automatic Speaker Verification
BLSTM	Bidirectional Long Short-Term Memory
BPD	Baseband Phase Difference
CFW	Correlation-based Frequency Warping
CMU	Carnegie Mellon University
CWT	Continuous Wavelet Transform
DBN	Deep Belief Network
DFT	Discrete Fourier Transform
DFW	Dynamic Frequency Warping
DKPLS	Dynamic Kernel Partial Lease Squares
DNN	Deep Neural Network
DTW	Dynamic Time Warping
EER	Equal Error Rate
EFW	Exemplar-based Frequency Warping
FFT	Fast Fourier Transform
GD	Group Delay
GMM	Gaussian Mixture Model
GV	Global Variance
HMM	Hidden Markov Model
HNM	Harmonic plus Noise Model
IF	Instantaneous Frequency
IFT	Inverse Fourier Transform
IMFCC	Inverse Mel-Frequency Cepstral Coefficient
JD-GMM	Joint-Density Gaussian Mixture Model
LPC	Linear Predictive Coefficients
LFCC	Linear Frequency Cepstral Coefficient
LMS	Log Magnitude Spectrum
LSD	Log-Spectral Distortion
LSF	Line Spectral Frequency

MAP	Maximum A Posteriori
MCC	Mel-Cepstral Coefficient
MCD	Mel-Cepstral Distortion
MFCC	Mel-Frequency Cepstral Coefficient
MGC	Mel-Generalized Cepstral Coefficient
MGD	Modified Group Delay
ML	Maximum Likelihood
MLP	Multilayer Perceptron
MLPG	Maximum Likelihood Parameter Generation
MSE	Mean Square Error
MOS	Mean Opinion Score
NMF	Nonnegative Matrix Factorization
PLS	Partial Least Squares
PPG	Phonetic PosteriorGram
PSOLA	Pitch Synchronous Overlap and Add
RC	Residual Compensation
RLMS	Residual Log Magnitude Spectrum
RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
RPS	Relative Phase Shift
SPSS	Statistical Parametric Speech Synthesis
SPTK	Speech Signal Processing Toolkit
STFT	Short Time Fourier Transform
TTS	Text-To-Speech
VC	Voice Conversion
VQ	Vector Quantization
VTLN	Vocal Tract Length Normalization
WFW	Weighted Frequency Warping

CHAPTER 1

Introduction

1.1 Background

Speech, as the most common and effective form of human communication, is getting wider adoption in human-computer interaction. In general, two types of technologies are required, speech perception and speech generation. On one hand, speech perception is a technique for computer to extract the useful information from human. There are various research topics regarding this area, e.g. speech recognition [2], speaker recognition [3], natural language processing [4], etc. On the other hand, computer also need to generate the feedback to interact with human. As the purpose of feedback is to answer questions, make recommendations and perform actions, a personalized voice is more preferable to users. Thus, one important research topic in speech generation is to develop technologies to generate speech with various voice characteristics.

There are mainly two technologies to generate personalized speech, namely Text-To-Speech (TTS) synthesis and Voice Conversion (VC). TTS is a technique to generate speech signals for a given input text. It can be realized by either concatenating pre-recorded speech, namely unite selection [5–7], or generating speech from a statistical parametric model [8–10], namely Statistical Parametric Speech Synthesis (SPSS). Both of these TTS methods are able to generate intelligible and natural-sounding speech. However, to establish a TTS system, a large speech database of the target speaker along with manual work (e.g. manually checking of segmental boundary and prosodic annotation) is always required. If we need to generate speech with various speaking styles and emotions, the size of the speech database becomes exponentially larger and recording

such large amounts of speech for the database is cost inefficient and time consuming. Speaker adaptation technique for TTS is able to customize a TTS system towards the voice characteristics and speaking manner of a specific speaker with a small number of target speech samples [11–13]. However, this technique can be only applied on a trained SPSS model, which is still a tedious process.

Voice conversion, another technique to generate personalized speech, converts a source speaker’s speech to sound like that of a target speaker without changing the language content [14–16]. Generally, the conversion model can be trained using a few minutes speech samples of both the source and target speakers. Voice conversion has many applications, such as personalized speech synthesis [15], speech enhancement [17], speaking-aid [18] and spoofing attack [19]. Compared to TTS techniques, voice conversion requires much less training data and manual work. In this sense, voice conversion provides a cost effective way to create personalized voices.

A voice conversion is considered to be successful when the identity of the source speaker is transformed to that of the target speaker, with speech quality and linguistic content maintained. Now, finding the speaker identity related features for conversion becomes the major problem. From a human perceptual point of view, speech signals carry speaker identity information simultaneously at different levels [20, 21]. The ‘high-level’ aspects, referring to speaking style, are more related to linguistic characteristics, e.g. language or dialect, terminology usage and other non-speech characteristics that are attributes of a specific speaker. The ‘low-level’ aspects, referring to the actual sound of a speaker’s voice, are more related to acoustic characteristics, including both segmental and supra-segmental features, which are defined as below:

Segmental features: These are a kind of short-term feature related to vocal tract characteristics, such as timbre, spectral envelope, average pitch intensity, etc.

Supra-segmental features: These kind of features are concerned with acoustic properties of the domain over more than one segment. It is more related to the voice source characteristics (prosodic), such as pitch contour, the duration of words, syllables or phonemes, intonation contours or stress patterns, etc.

Although the ‘high-level’ aspects are closely related to speaker individuality, they are hard to extract and quantify from the speech. Hence, the current voice conversion technologies mainly focus on the transformation of ‘low-level’ (acoustic) features. In a desired voice conversion system, both segmental and supra-segmental features containing speaker identity information should be transformed. Due to the difficulty of analyzing and modeling supra-segmental features, the majority of previous works focus only on ssegmental level conversion. In this work, we also focus on segmental feature conversion, specifically, spectral feature mapping.

1.2 Motivation

One key challenge of voice conversion is to estimate a robust spectral feature mapping function to transform the source speaker’s characteristics to that of the target with limited amounts of training data. In addition, the generated speech should maintain speech quality. There are two major problems with previous voice conversion methods. The first issue is that the quality of converted speech always gets degraded due to the over-smoothing problem. Generally, this can be caused by the use of the minimum Mean Square Error (MSE) [14, 15] or the maximum likelihood [16] function as the optimization criteria for modeling, e.g. Gaussian Mixture Model (GMM). Such optimization criteria are doomed to lose spectral details and cause the converted speech sounds to be muffled [16, 22]. Frequency warping is one of the most successful methods to preserve spectral details and prevent the over-smoothing problem. However, the current state-of-the-art frequency warping systems also rely on GMM for warping function and residual spectrum¹ estimation [22, 24]. This introduces estimation error and degrades system performance. The second issue of most of the existing voice conversion techniques is that their training relies on parallel data, which requires that the source and target speakers must utter the same speech content. Such parallel data may not always be available in practice. There have been attempts to build voice conversion systems over non-parallel corpus. However, the quality of converted speech is moderate [25–28]. Therefore, it is

¹Residual spectrum refers to the amplitude difference between the warped spectrum and the target spectrum. In some studies, it is also called spectrum differential [23].

important to improve the performance of voice conversion with both parallel data and non-parallel data.

Speech, apart from communication, is also used as a low-cost biometric solution to verify speaker identity in order to safeguard personal information. However, recent improvements in TTS and voice conversion techniques present a threat to Automatic Speaker Verification (ASV) systems. An attacker can use TTS or voice conversion systems to impersonate a target speaker’s voice to attack an ASV system. To overcome such a challenge, synthetic speech detection provides a feasible way to improve the robustness of ASV systems against synthetic speech. This provides the motivation to investigate the countermeasures that can improve the reliability of synthetic speech detection.

1.3 Contributions

In this thesis, we address the two problems mentioned in Section 1.2 by: 1) improving voice conversion quality with parallel and non-parallel data and 2) studying countermeasures to detect synthetic speech. The three major contributions are listed below.

1. **An Exemplar-based Frequency Warping (EFW) Approach for Parallel Data Voice Conversion [29–31]:** In the first study, we introduce a framework based on exemplar-based frequency warping and Residual Compensation (RC) for parallel data voice conversion. In previous frequency warping framework, the GMM was adopted to generate the warping functions and spectral residuals, which introduced the over-smoothing effect and degraded system performance. The proposed framework exploits an exemplar-based technique to avoid the statistical averaging problem inherited from GMM-based approaches. Specifically, we use the exemplar-based technique to estimate the warping functions and residual vectors, where the target warping functions and residual vectors are estimated as a weighted linear combination of the dictionaries of exemplar warping functions and exemplar residual spectrogram respectively. As the linear combination weights are constrained to be sparse, the over-smoothing effect is prevented.
2. **Average Modeling Approach (AMA) for Non-Parallel Data Voice Conversion [32]:** In the second study, we introduce an average model adaptation

approach for non-parallel data voice conversion. The proposed AMA is a departure from previous adaptation-based approaches where the average models are trained on parallel data from multiple pairs of speakers. Unlike previous works, the proposed AMA learns the mapping between speaker independent linguistic features and acoustic features. As linguistic features and acoustic features can be extracted from the same utterance, non-parallel data can be used for both average model training and adaptation. At run-time, the linguistic features extracted from the given source speech are presented to the adapted model to generate converted speech. In this sense, the proposed method can easily make use of publicly available speech corpora for average model training, while reducing the required amount of target speech for adaptation. In addition, the AMA can also be easily applicable for many to one conversion.

3. **Synthetic Speech Detection using High-Dimensional Features [1, 33]:** In the final study, we focus on discriminating synthetic speech, e.g. voice conversion and TTS, from natural speech in order to protect speaker verification systems. As speech synthesis techniques typically exploit low-dimensional features, much detailed information has been abandoned in synthetic speech. These detailed information may be comprised of high-dimensional features extracted directly from the Fourier transform. Hence, detailed information carried in high-dimensional or high-frequency features might be useful for synthetic speech detection. In this study, we investigate the use of different feature representations for synthetic speech detection. Specifically, we focus on the high-dimensional and the high-frequency information of both magnitude and phase features to discriminate synthetic spoofing attacks and natural speech.

The first work on the exemplar-based approach to frequency warping discussed in Chapter 3 is published in the conferences: ISCSLP 2014 [29], ICASSP 2015 [30], and its extension is published in the journal: IEEE/ACM Transactions on Audio, Speech and Language Processing [31]. The second work on the average model approach discussed in Chapter 4 is published in the conference: Odyssey 2018 [32]. The third work on the synthetic speech detection discussed in Chapter 5 is published in the conferences: INTERSPEECH 2015 [1], ChinaSip [34] and ICASSP 2016 [33].

1.4 Thesis Organization

The thesis is organized as follows:

Chapter 2 provides a background on voice conversion, followed by a review of the state-of-the-art voice conversion techniques. The relationship and difference between techniques are discussed. Then an overview of synthetic speech detection is provided.

Chapter 3 proposes a parallel data based voice conversion framework, namely exemplar-based frequency warping, to improve GMM-based frequency warping framework by reducing the over-smoothing effect.

Chapter 4 proposes an average modeling approach for nonparallel data voice conversion.

Chapter 5 investigates synthetic speech detection from a feature representation perspective.

Finally, Chapter 6 concludes the thesis with a summary of the contributions and discusses future research directions.

CHAPTER 2

Introduction to Voice Conversion and Synthetic Speech Detection

In this chapter, we first provide an overview of a typical voice conversion system in Section 2.1, by describing the background review, including the framework for speech analysis and reconstruction, feature extraction, feature alignment, feature conversion and evaluation matrices. Section 2.2 reviews a range of state-of-the-art spectral feature mapping techniques for voice conversion. Synthetic speech detection is then introduced in 2.3. Finally, Section 2.4 presents a summary of the chapter.

2.1 Overview of a Voice Conversion System

Voice conversion is a supervised learning problem consisting of two phases, offline training and run-time conversion. Fig. 2.1 shows a block diagram of a voice conversion system. In the offline training phase (Figure 2.1 (a)), given the training data from source and target speakers, the speech signals are first analyzed and parameterized into feature vectors, $\mathbf{X}^{raw} \in \mathbb{R}^{D \times I}$ and $\mathbf{Y}^{raw} \in \mathbb{R}^{D \times J}$, where the D indicates the feature dimension, and I and J indicate the frame number of the source and target data respectively. The alignment process is applied to the source and supervisory target feature vectors resulting in the aligned feature pairs $[\mathbf{X}; \mathbf{Y}] = [\mathbf{X}^\top, \mathbf{Y}^\top]^\top$, where $\mathbf{X} \in \mathbb{R}^{D \times M}$ and $\mathbf{Y} \in \mathbb{R}^{D \times M}$. A supervised learning algorithm is then applied to analyze the aligned feature pairs and produce a conversion function, $\mathcal{F}(\cdot)$.

At run-time (Figure 2.1 (b)), testing speech is first passed to the same analysis and parameterization module as in the offline training phase. The trained conversion function

Figure 2.1: Block diagram of a voice conversion system.

is then used to convert new features of source speaker $\mathbf{x} \in \mathbb{R}^D$ to generate the converted target features $\tilde{\mathbf{y}} = \mathcal{F}(\mathbf{x})$, where $\tilde{\mathbf{y}} \in \mathbb{R}^D$. Finally, these converted target features are passed to a synthesis module to reconstruct the converted speech signal.

As described above, a typical voice conversion system mainly contains four modules: a speech analysis and synthesis module, a feature parameterization module, a feature alignment module and a feature conversion module. Therefore, this section provides a review of the typical approaches for each module in the literature. These are discussed in detail below.

2.1.1 Speech Analysis and Synthesis

The speech analysis and synthesis module aims to 1) analyze speech signal in different features for modeling and modification and 2) reconstruct the speech signal from the analyzed features.

A high quality speech analysis and synthesis framework is an essential part of a voice conversion system. As discussed in Section 1.1, a voice conversion system attempts to

convert speaker identity by transforming the acoustic and prosodic features. Hence, the framework for speech analysis and synthesis should have two main properties. First, it should be able to analyze speech signals into acoustic features (spectral) and prosodic features (pitch, duration), and reconstruct the signal from the analyzed parameters with high quality and naturalness. Second, it should allow flexible modifications of the acoustic and prosodic features without quality degradation.

With these perspectives in mind, a variety of speech analysis and synthesis frameworks are developed. These approaches can be broken down into two categories: non-parametric framework, e.g. Pitch Synchronous Overlap and Add (PSOLA) [35], and parametric framework, e.g. Harmonic plus Noise Model (HNM) [36] and Speech Transformation and Representation using Adaptive Interpolation of weiGHTed spectrum (STRAIGHT) [37]. They are briefly described below.

Pitch Synchronous Overlap and Add (PSOLA): is one of the most popular non-parametric speech synthesis techniques, which is able to modify the speech prosodic (pitch and duration) without introducing artifacts. PSOLA works by dividing a speech waveform into short segments using a sequence of pitch synchronous analysis windows. During synthesis, the pitch can be modified by changing the windows length; while the duration can be modified by repeating or eliminating the segments. This modification can be done either in the time domain (TD-PSOLA) or in the frequency domain (FD-PSOLA). Due to its good performance, this technique has been applied into unit selection based TTS [38] and frequency warping based voice conversion [38]. However, because of its non-parametric property, it is not suitable for statistical model synthesis approaches.

Harmonic plus Noise Model (HNM): is an example of a sinusoidal model. It decomposes the speech signal into a harmonic component and a noise-like component. The excitation of the harmonic component is represented as a sum of sine waves; while white Gaussian noise is used as the excitation of noise-like component. This technique succeeds in parameterizing speech using f_0 and spectrum [39], and has been successfully applied to TTS systems [39, 40] and voice conversion [41, 42].

STRAIGHT: is a pitch-adaptive spectral analysis combined with a surface reconstruction method. It decomposes a speech into three components: a pitch-adaptive smoothed spectrogram, a fundamental frequency (f_0) contour, and an aperiodicity map. As the periodicity interference is removed from the spectrogram representation, STRAIGHT is able to control the spectrogram and f_0 extraction more precisely. Due to the great flexibility for parameter manipulation and the high quality of reconstructed speech, it has been widely used in TTS and voice conversion tasks [16, 43–46].

2.1.2 Parameterization

After the speech analysis process, the speech signal is represented in prosodic and acoustic features. Generally, the prosodic features, e.g. f_0 , duration and intensity, are of low-dimension (e.g. 1 dimension), which can be used to estimate the prosodic conversion function directly. While, the acoustic features (spectral envelop), containing more spectral information, usually have high dimension (e.g. 513 dimension). Although such high-dimensional features have also been used in some voice conversion methods [22, 24, 38, 47], more commonly, the acoustic features will be parameterized into a compact form for feature alignment and spectral conversion function estimation.

The spectral feature after parameterization should be representative of speaker identity and appropriate for statistical modeling. Moreover, it should also be easy to convert back to raw acoustic feature for speech signal reconstruction while keeping speech quality and naturalness. Some popular parameterization methods used in voice conversion tasks are briefly described below.

Line Spectral Frequencies (LSF) [48, 49]: are used to represent Linear Prediction Coefficients (LPC), which is a smoothed spectral envelope, uniquely. Due to its robust representation for quantization and interpolation, the LSF feature has been widely used in previous voice conversion systems [22, 50, 51].

Cepstral Coefficients: represents the spectral envelope based on the result of taking the Inverse Fourier transform (IFT) of a log power spectrum in a finite number of coefficients. Mel-Frequency Cepstral Coefficients (MFCC), one of the most popular speech features, have been widely used in speech recognition and speaker recognition.

Mel-Cepstral Coefficients (MCC) and Mel-Generalized Cepstral Coefficients (MGC) is also popular in voice conversion tasks [16, 43–45].

2.1.3 Feature Alignment

In order to train a conversion function from the source to target speech features, it is necessary to align the training data into frame level source-target feature pairs. In practice, the training data can be either parallel or non-parallel. The parallel data indicates that the source and target training utterances contain the same linguistic content, while the non-parallel data do not have such constraints. Hence, two of the alignment techniques were developed for parallel and nonparallel data accordingly, which are briefly described below.

Parallel Data Alignment

Although the linguistic content of the parallel source-target utterances are the same, the speaking rate of two speakers is usually different. Hence, the utterances still need to be aligned in time. Dynamic time warping [52] (DTW) is one of the most popular techniques for parallel data alignment. To perform DTW, a distance matrix between the source and target feature sequences is first calculated. Then, the optimal alignment is obtained by searching the path with the minimum distance between the source and target feature sequences. Usually, DTW is applied on low-dimensional features, e.g. MCC and MFCC, with the Euclidean distance as the measure of feature distance.

To improve alignment accuracy, a few approaches are also proposed. A two step alignment is proposed in [45]. First, the source and target feature sequences are aligned by DTW. This initial alignment is used to estimate the voice conversion function, and generate the converted speech. Then, DTW is applied to align the converted and target feature sequences again to refine the alignment. An alternative approach is proposed in [22], where the utterances are first segmented into phonemes using Hidden Markov Model (HMM)-based forced alignment. Then, the DTW is preformed inside each phoneme pair to prevent inaccurate alignment across different phonemes.

Non-parallel Data Alignment

As the assumption of the availability of parallel training data is not always realistic, approaches are also proposed to perform alignment on non-parallel data.

One of the approaches for non-parallel data alignment is based on feature clustering. In an early approach [53], the source and target feature vectors were first segmented into the same number of classes. Then, class mapping was performed by comparing the distance between the centroids of source and target classes. Alignment was then obtained in the paired source-target clusters following a nearest-neighbor criterion. In a more recent work [54], a HMM-based speech recognizer was used to label all of the source and target frames in the state level. Given a labeled source sequence, alignment was performed by finding the longest match from the target sequences.

As an alternative to class-based approaches, an Iterative combination of a Nearest Neighbor search step and a Conversion step Alignment (INCA) is proposed in [25]. INCA iteratively refines the alignment from the basic nearest neighbor method, combined with voice conversion. Given the source and target feature vectors, initial alignment was first obtained by finding the nearest neighbor of each source feature vector in the target data set. Second, this initial alignment was used to estimate the voice conversion function and generate the converted speech. Then DTW was applied on converted and target feature sequences again to refine the alignment. The second and third steps were repeated until the conversion error converges.

2.1.4 Feature Conversion

Feature conversion is the central task of a voice conversion system. The purpose of this module is to convert the speaker identity of the source speaker to that of the target speaker while maintaining high speech quality. As discussed in Section 1.1, both acoustic and prosodic features are related to speaker identity. Hence, a successful speaker identity conversion includes both spectral feature mapping and prosodic feature conversion, which are discussed in detail below.

2.1.4.1 Spectral Feature Mapping

After the data processing steps in Sections 2.1.1, 2.1.2 and 2.1.3, the aligned source-target feature pairs $[\mathbf{X}; \mathbf{Y}] = [\mathbf{X}^\top, \mathbf{Y}^\top]^\top$ are obtained. Then the spectral feature mapping function $\mathcal{F}(\cdot)$ is learned from these aligned feature pairs, which can be expressed as

$$\mathbf{y} = \mathcal{F}(\mathbf{x}) + \mathbf{e} = \tilde{\mathbf{y}} + \mathbf{e}, \quad (2.1)$$

where $\mathbf{x}, \mathbf{y}, \tilde{\mathbf{y}}, \mathbf{e} \in \mathbb{R}^D$, D is the feature dimension, $\tilde{\mathbf{y}}$ is the converted feature, and $\mathbf{e}$ is the approximation error.

There exists a number of methods for the spectral feature mapping function $\mathcal{F}(\cdot)$ estimation, including codebook mapping method [55, 56], statistical parametric method [14–16, 45], neural network method [57–59], frequency warping method [22, 24, 38, 47] and exemplar-based method [46, 60–62]. Generally, codebook mapping method, frequency warping method and exemplar-based method accept the high-dimensional spectral feature (without parameterization) for mapping function $\mathcal{F}(\cdot)$ estimation. Statistical parametric and neural network methods perform estimates using more parameters and are used to model more complex transformations. Hence, low-dimensional features, e.g. MCC and LSF, are usually employed in such methods. The technical details of these approaches will be reviewed in Section 2.2.

2.1.4.2 Prosodic Feature Conversion

Prosody also plays an important role in the perception of speaker identity. It can be represented as difference features, e.g. pitch and duration. The prosodic features are affected by both short term as well as long term dependencies. However, voice conversion is realized with small amount of training data which limits the prosodic feature modeling. Moreover, the evaluation of prosody conversion is difficult. As the same person can utter the same sentence with different prosody, the objective measures, which compares the converted prosody to original target prosody, can be misleading. These make prosodic feature modeling and conversion to be a challenging topic in voice conversion.

Previous studies of prosody conversion mainly focus on pitch (f_0) transformation. A basic f_0 transformation technique shifts the global mean and variance of the source

speaker's f_0 , towards those of the target speakers [14]. To better match with human perception, this transformation is usually performed in logarithmic domain as

$$\widetilde{f0}_y = \exp((\log f0_x - \mu^x) \frac{\sigma^y}{\sigma^x} + \mu^y), \quad (2.2)$$

where μ^x and μ^y are respectively the mean of the source and target speakers' f_0 in logarithmic domain. σ^x and σ^y are the corresponding variances of the source and target speakers' f_0 in logarithmic domain. $\widetilde{f0}_y$ is the estimated f_0 of the target speaker. This basic f_0 transformation can modify speaker identity towards the target speaker efficiently and is widely used in a number of works [14–16, 22, 24, 42, 44–46, 57, 60, 63].

Alternatively, a Continuous Wavelet Transform (CWT) is proposed for f_0 modeling and transformation [64]. CWT decomposes the f_0 contour into different temporal scales. The f_0 modeling and transformation are performed at different prosodic levels. CWT has been proposed to use and adopted in different voice conversion frameworks such as DKPLS [65], neural network [66] and exemplar-based prosody conversion [67].

2.1.5 Evaluation Matrices

The voice conversion system can be evaluated by both objective and subjective tests. Specifically, objective evaluation is used to measure pitch and spectral distortion between converted and reference features, while subjective evaluation is used to examine the naturalness and speaker identity of converted speech.

2.1.5.1 Objective Evaluation

Root Mean Squared Error (RMSE) and correlation are usually employed to evaluate the performance of the pitch conversion, which is measured between the converted and target pitch contours.

Spectral distortion is used to evaluate the performance of spectral conversion. It can be measured on either low-dimensional features (e.g. MCCs), namely Mel-Cepstral Distortion (MCD) or high-dimensional features (e.g. log spectrum), namely Log-Spectral Distortion (LSD). These two types of measurements are reviewed in detail below.

Mel-Cepstral Distortion (MCD)

MCD [16] is used to measure spectral distortion for MCC features. For j^{th} frame, the MCD is calculated as:

$$\text{MCD[dB]} = 10/\ln 10 \sqrt{2 \sum_{i=1}^D (mc_{i,j} - mc_{i,j}^{conv})^2}, \quad (2.3)$$

where $mc_{i,j}$ and $mc_{i,j}^{conv}$ are the i^{th} dimension of the target and converted MCCs for frame j respectively. D is the dimension of the MCC feature. A lower MCD indicates a smaller distortion.

Log-Spectral Distortion (LSD)

LSD ratio [68] is used to measure distortion for the log spectrum feature. The distortion in the j^{th} frame of the log spectrum is calculated using

$$d(\mathbf{x}_j, \mathbf{y}_j) = \sum_{i=1}^D (\log \mathbf{x}_{j,i} - \log \mathbf{y}_{j,i})^2, \quad (2.4)$$

where D is the total number of the frequency bins.

A distortion ratio between the converted-to-target distortion and the source-to-target distortion is defined as

$$\text{LSD} = \frac{\sum_{j=1}^J d(\tilde{\mathbf{y}}_j, \mathbf{y}_j)}{\sum_{j=1}^J d(\mathbf{x}_j, \mathbf{y}_j)} \times 100\%, \quad (2.5)$$

where, $\mathbf{x}_j$ and $\mathbf{y}_j$ denote the source and target spectrum respectively. $\tilde{\mathbf{y}}_j$ is the converted spectrum. J is the total frame number. A lower LSD ratio indicates a smaller spectral distortion.

2.1.5.2 Subjective Evaluation

Generally, two types of subjective evaluations are carried out to access conversion performance in terms of speech quality and speaker similarity. These are described below.

Quality test

Mean Opinion Score (MOS) is one of the most popular measures for speech quality evaluation. Usually, MOS is a single number ranging from 1 to 5, where 1 denotes the lowest perceived quality, and 5 denotes the highest perceived quality. The MOS of a system is represented as the average of the scores from all listeners. The major problem of the MOS test is that each system is evaluated separately, and the difference in results between the systems may be small.

Alternatively, an AB preference test can also be used for speech quality evaluation between two systems. For each AB test pair, one speech sample is from the proposed method and another is from the baseline method. They are presented in a random order. A listener is asked to listen to both samples and decide which sample sounds better in terms of speech quality.

Similarity test

The XAB and ABX tests are commonly used to assess speaker similarity.

In both of these two tests, X, A and B share the same language content. A listener is asked to listen to sample X first, and samples of A and B are then presented in a random order. After listening, the listener should decide if the reference sample X is a) closer to A, b) closer to B or c) neither of the above, in terms of speaker individuality.

The difference between these two tests is that: in the XAB test, X is the speech sample from the evaluating method, and A and B are either the source or target sample; while in the ABX test, X is the speech sample from the target recording, and A and B are the conversion samples from two systems. Hence, the XAB test is used to evaluate the speaker similarity for a single system X, while the ABX test is used to compare the results between system A and B.

Note that, in both of these two tests, a higher identification rate for a given voice conversion system indicates a better similarity to the target speaker.

2.2 State-of-the-Art Spectral Feature Mapping Techniques for Voice Conversion

In the previous section, an overview of voice conversion system was provided, with a description of data preparation including speech signal analysis, parameterization, data alignment, feature conversion and evaluation. This section will review state-of-the-art spectral feature mapping techniques and their limitations. As discussed in Section 2.1.4.1, mapping function estimation can be grouped into five categories: codebook mapping, statistical parametric, neural network, frequency warping and exemplar-based methods. The approaches in these categories are discussed in detail below.

2.2.1 Codebook Mapping Method

This group of state-of-the-art approaches is concerned with finding the vector correspondences between two codebooks, from the source and target speakers, which determines the spectrum mapping between the source and target at the frame level.

An early codebook mapping based voice conversion approach was proposed in [55]. First, during the training phase, the codebooks of the source and target speaker were established from training utterances respectively. Vector Quantization (VQ) is then applied on training utterances at the frame level to reduce the number of code vectors. Second, DTW was employed to find vector correspondences between the two codebooks, which were accumulated as histograms. The mapping codebook was then represented as a linear combination of the target vectors, with the accumulated histograms as weighting functions. At run-time, the source feature vectors were quantized based on the source codebook. The converted feature vectors were determined by the mapping codebook.

The codebook mapping method can successfully convert speaker identity. However, in conventional VQ, only one of the vectors is used to represent the correspondence, which results in frame-to-frame discontinuities and degrades speech quality. To enhance the quality of converted speech, fuzzy VQ [56] and weighted average methods [50, 69] are proposed to generate a smooth transition across successive frames.

2.2.2 Statistical Parametric Method

This group of state-of-the-art approaches aims at finding a parametric model to form a source to target feature transformation. Feature transformation can be realized through either a linear conversion function, e.g. Gaussian Mixture Model (GMM) based approaches [14–16, 63], or a nonlinear conversion function, e.g. Dynamic Kernel Partial Least Squares (DKPLS) approach [45]. Both of these are reviewed in detail below.

Gaussian Mixture Model (GMM) based methods

- **Source GMM approach:** To avoid the problem of hard clustering, a GMM-based method using soft clustering is first introduced in [14] with an assumption that the GMM can be well fitted to the source feature vectors. Given the aligned source and target feature vectors, $\mathbf{X}$ and $\mathbf{Y}$, a single Gaussian distribution of $\mathbf{X}$ can be expressed as:

$$P(\mathbf{x}) = \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}^{(x)}, \boldsymbol{\Sigma}^{(x)}) \quad (2.6)$$

where $\mathbf{x} \in \mathbb{R}^D$, D is the feature dimension. $\mathcal{N}(\mathbf{x} | \boldsymbol{\mu}^{(x)}, \boldsymbol{\Sigma}^{(x)})$ is a Gaussian distribution defined by the mean vector $\boldsymbol{\mu}^{(x)}$ and the covariance matrix $\boldsymbol{\Sigma}^{(x)}$.

The conversion function of the jointly Gaussian of $[\mathbf{X}; \mathbf{Y}]$ can be expressed as

$$\mathcal{F}(\mathbf{x}) = \boldsymbol{\mu}^{(y)} + \boldsymbol{\Sigma}^{(xy)}(\boldsymbol{\Sigma}^{(x)})^{-1}(\mathbf{x} - \boldsymbol{\mu}^{(x)}) \quad (2.7)$$

where the target mean vector $\boldsymbol{\mu}^{(y)}$ and the cross-covariance matrix between source and target features $\boldsymbol{\Sigma}^{(xy)}$ can be expressed as

$$\boldsymbol{\mu}^{(y)} = E[\mathbf{y}] \quad (2.8)$$

$$\boldsymbol{\Sigma}^{(xy)} = E[(\mathbf{y} - \boldsymbol{\mu}^{(y)})(\mathbf{x} - \boldsymbol{\mu}^{(x)})^\top] \quad (2.9)$$

Extending the assumption from a single Gaussian to a GMM with K Gaussian components, the Equation (2.6) can be rewritten as

$$P(\mathbf{x}) = \sum_{k=1}^K \alpha_k \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k^{(x)}, \boldsymbol{\Sigma}_k^{(x)}) \quad (2.10)$$

where $\boldsymbol{\mu}_k^{(x)}$, $\boldsymbol{\Sigma}_k^{(x)}$ and α_k are the mean vector, the covariance matrix and the weight assigned to the k^{th} Gaussian component, respectively. $\sum_{k=1}^K \alpha_k = 1$ and $\alpha_k > 0$.

Now the conversion function $\mathcal{F}(\cdot)$ of Equation (2.7) is updated as

$$\mathcal{F}(\mathbf{x}) = \sum_{k=1}^K p_k(\mathbf{x}) [\boldsymbol{\mu}_k^{(y)} + \boldsymbol{\Sigma}_k^{(xy)} (\boldsymbol{\Sigma}_k^{(x)})^{-1} (\mathbf{x} - \boldsymbol{\mu}_k^{(x)})] \quad (2.11)$$

where $p_k(\mathbf{x})$ is the posterior probability of the source vector $\mathbf{x}$ generated from the k^{th} Gaussian component, expressed as

$$p_k(\mathbf{x}) = \frac{\alpha_k \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k^{(x)}, \boldsymbol{\Sigma}_k^{(x)})}{\sum_{k=1}^K \alpha_k \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k^{(x)}, \boldsymbol{\Sigma}_k^{(x)})}. \quad (2.12)$$

During training, the mean vectors $\boldsymbol{\mu}_k^{(y)}$ and cross-covariance matrices $\boldsymbol{\Sigma}_k^{(xy)}$ are estimated by minimizing the least squares error of converted spectrum $\mathcal{F}(\mathbf{x})$ and the corresponding aligned target vectors $\mathbf{y}$.

- **Joint-Density GMM (JD-GMM) approach:** In source GMM approach, two separate GMM are trained to model source and target features. To simplify the training process, a JD-GMM [15] is proposed to jointly model the distribution of the aligned source and target feature pairs $\mathbf{Z} = [\mathbf{X}; \mathbf{Y}] = [\mathbf{X}^\top, \mathbf{Y}^\top]^\top$. To represent the joint-density Gaussian mixture distribution, the Equation (2.10) can be rewritten as

$$P(\mathbf{z}) = \sum_{k=1}^K \alpha_k \mathcal{N}(\mathbf{z} | \boldsymbol{\mu}_k^{(z)}, \boldsymbol{\Sigma}_k^{(z)}) \quad (2.13)$$

where $\mathbf{z} \in \mathbb{R}^{2D}$, $\boldsymbol{\mu}_k^{(z)} \in \mathbb{R}^{2D}$ and $\boldsymbol{\Sigma}_k^{(z)} \in \mathbb{R}^{2D \times 2D}$ are expressed as

$$\mathbf{z} = \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix}, \quad \boldsymbol{\mu}_k^{(z)} = \begin{bmatrix} \boldsymbol{\mu}_k^{(x)} \\ \boldsymbol{\mu}_k^{(y)} \end{bmatrix}, \quad \boldsymbol{\Sigma}_k^{(z)} = \begin{bmatrix} \boldsymbol{\Sigma}_k^{(xx)} & \boldsymbol{\Sigma}_k^{(xy)} \\ \boldsymbol{\Sigma}_k^{(yx)} & \boldsymbol{\Sigma}_k^{(yy)} \end{bmatrix}. \quad (2.14)$$

During GMM training, the target mean $\boldsymbol{\mu}_k^{(y)} \in \mathbb{R}^D$ and the cross-covariance matrix between source and target features $\boldsymbol{\Sigma}_k^{(xy)} \in \mathbb{R}^{D \times D}$ are estimated. Then the

Equation (2.11) and (2.12) can be rewritten as

$$\mathcal{F}(\mathbf{x}) = \sum_{k=1}^K p_k(\mathbf{x}) [\boldsymbol{\mu}_k^{(y)} + \boldsymbol{\Sigma}_k^{(yx)} (\boldsymbol{\Sigma}_k^{(xx)})^{-1} (\mathbf{x} - \boldsymbol{\mu}_k^{(x)})] \quad (2.15)$$

$$p_k(\mathbf{x}) = \frac{\alpha_k \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k^{(x)}, \boldsymbol{\Sigma}_k^{(xx)})}{\sum_{k=1}^K \alpha_k \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k^{(x)}, \boldsymbol{\Sigma}_k^{(xx)})} \quad (2.16)$$

- Maximum Likelihood GMM (ML-GMM) approach:** In conventional GMM-based conversion [14, 15], each feature frame is converted independently. This may result in the discontinuity of feature trajectories and degrade the speech quality. To refine the estimated feature trajectories, a parameter generation algorithm is employed in [16]. In this approach, both static $[\mathbf{X}; \mathbf{Y}]$ and dynamic features pairs $[\Delta \mathbf{X}; \Delta \mathbf{Y}]$ are used in JD-GMM training. Hence, the joint vector $\mathbf{z}$ is written as $\mathbf{z} = [[\mathbf{x}^\top, \Delta \mathbf{x}^\top]^\top, [\mathbf{y}^\top, \Delta \mathbf{y}^\top]^\top]^\top \in \mathbb{R}^{4D}$. Then the JD-GMM training process is employed as Equation (2.13). At run-time, the feature conversion is performed based on the maximum-likelihood estimation of a spectral parameter trajectory. Then Equation (2.15) can be rewritten as

$$\mathcal{F}(\mathbf{x}) = \arg \max p(\mathbf{Y} | \mathbf{X}, \mathcal{N}^{(\mathbf{Z})}) \quad (2.17)$$

where $\mathcal{N}^{(\mathbf{Z})}$ is the parameter set of GMM.

By using dynamic features in the conversion framework, temporal information across frames is considered in feature prediction, which results in a smooth feature trajectories and improves the quality of converted speech. Due to its robust performance [16], this approach has become a popular baseline in recent works [42, 44, 46, 70]. This baseline is also used in our works.

- Problems in GMM-based approaches:** While GMM-based methods could generate speech highly similar to the target, they are known to suffer from over-smoothing [16, 22, 71] and over-fitting [14, 44, 71] problems.

Over-smoothing problem is due to the use of the minimum mean square error [14, 15] or the maximum likelihood [16] function as the optimization criteria. As a result, statistical averaging arises and the converted speech fails to capture the desired

details of temporal and spectral dynamics. Additionally, GMM-based methods generally employ low-resolution features such as Mel Cepstral Coefficients (MCC) or Line Spectral Frequencies (LSF) to avoid the curse of dimensionality. Such low-resolution features, however, are doomed to lose spectral details. These two aspects, statistical averaging and low-resolution features, cause the converted speech of most GMM-based methods to sound muffled [16, 22]. To enhance the temporal variation and alleviate the over-smoothing problem, Global Variance (GV) [16, 63] and Modulation Spectrum (MS) [72] were proposed.

Over-fitting occurs when a model achieves good performance on the training data while having poor predictability on the evaluation set. This problem is mainly caused by the complexity of the established model compared to the limited size of the training data. In GMM-based methods, over-fitting problem is observed when the number of mixtures of the GMM is too high or full covariance matrix is estimated [14, 44, 71]. One solution to reduce over-fitting is to use the diagonal covariance matrix in GMM estimation. In [44], a partial least squares (PLS) regression is proposed to deal with the over-fitting problem. This is reviewed in detail below.

Partial Least Square (PLS) based methods

With assumption that the source and target features can be represented by a small number of latent variables, PLS regression based approach [44] models the relationship between the aligned source-target feature pairs $[\mathbf{X}; \mathbf{Y}]$ in a low-dimensional feature space. By choosing a suitable number of latent components, PLS approach can prevent over-fitting problem effectively.

First, the aligned source-target feature pairs $[\mathbf{X}; \mathbf{Y}]$ can be expressed as

$$\mathbf{X} = \mathbf{P}\mathbf{W}^\top + \mathbf{E}_x \quad (2.18)$$

$$\mathbf{Y} = \mathbf{Q}\mathbf{U}^\top + \mathbf{E}_y \quad (2.19)$$

where, $\mathbf{P} \in \mathbb{R}^{D \times h}$ and $\mathbf{Q} \in \mathbb{R}^{D \times h}$ are the loading matrices of the source and target speaker; $\mathbf{W}^\top \in \mathbb{R}^{h \times M}$ and $\mathbf{U}^\top \in \mathbb{R}^{h \times M}$ are the low-dimensional latent variable matrices of the source and target speaker, respectively; and $\mathbf{E}_x \in \mathbb{R}^{D \times M}$ and

$\mathbf{E}_y \in \mathbb{R}^{D \times M}$ are the transformation errors. Here, M denotes the frame number, while D and h denote the feature dimension and the number of PLS components respectively. Commonly, $h < D$, which enables a robust transformation with a small amount of training data.

Then the transformation is defined as

$$\mathbf{Y} = \beta \mathbf{X} + \epsilon, \quad (2.20)$$

where, β is the regression matrix defined by $\mathbf{P}$ and $\mathbf{Q}$, and ϵ is the estimation error.

An early approach, which combines the PLS with JD-GMM, was proposed in [44]. Given the aligned source-target feature pairs $[\mathbf{X}; \mathbf{Y}]$, the joint distribution can be established by Equation (2.13). The conversion function for k^{th} Gaussian in Equation (2.15) is defined as

$$\mathcal{F}_k(\mathbf{x}) = \boldsymbol{\mu}_k^{(y)} + \boldsymbol{\Sigma}_k^{(xy)} (\boldsymbol{\Sigma}_k^{(xx)})^{-1} (\mathbf{x} - \boldsymbol{\mu}_k^{(x)}). \quad (2.21)$$

Using PLS regression model (Equation (2.20)) to predict the conversion on each Gaussian, then Equation (2.15) can be rewritten as

$$\mathcal{F}(\mathbf{x}) = \sum_{k=1}^K p_k(\mathbf{x}) \mathcal{F}_k(\mathbf{x}) \approx \sum_{k=1}^K p_k(\mathbf{x}) \beta_k \mathbf{x}. \quad (2.22)$$

Experiments conducted on the CMU ARCTIC database [73] showed that the PLS method outperformed the JD-GMM baseline in terms of both quality and similarity.

In the subsequent work, a Dynamic Kernel Partial Least Squares (DKPLS) approach [45] is proposed to model nonlinear relationship between source and target features. Given the aligned source-target feature pairs $[\mathbf{X}; \mathbf{Y}]$, the source feature $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_m, \dots, \mathbf{x}_M]$ is first clustered into C reference vectors, denoted as $\boldsymbol{\mu} = [\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_c, \dots, \boldsymbol{\mu}_C]$. Then the source features are projected to a high-dimensional Gaussian kernel space, defined as

$$t_{c,m} = e^{-\frac{\|\mathbf{x}_m - \boldsymbol{\mu}_c\|^2}{2\delta^2}} \quad (2.23)$$

where $t_{c,m}$ is the element of kernel matrix according to the source feature $\mathbf{x}_m$ and the reference vector $\boldsymbol{\mu}_c$. δ is the scaling parameter. In the implementation [45],

the $2\delta^2$ was set to 10. The kernel vector $\mathbf{t}_m$ corresponding to source feature $\mathbf{x}_m$ is represented as

$$\mathbf{t}_m = [\mathbf{t}_{1,m}, \dots, \mathbf{t}_{c,m}, \dots, \mathbf{t}_{C,m}]^\top \quad (2.24)$$

In order to take the dynamic information into account for conversion, the adjacent kernel vectors are concatenated to form a new vector. The dynamic kernel matrix $\mathbf{O}$ according to the source features $\mathbf{X}$ is represented as

$$\mathbf{O} = [\mathbf{o}_1, \dots, \mathbf{o}_m, \dots, \mathbf{o}_M] \quad (2.25)$$

where $\mathbf{o}_m$ is the m^{th} dynamic kernel vector $\mathbf{o}_m = [\mathbf{t}_{m-1}^\top, \mathbf{t}_m^\top, \mathbf{t}_{m+1}^\top]^\top$.

Then the PLS transformation in Equation (2.20) can be rewritten as

$$\mathbf{Y} = \beta \mathbf{O} + \epsilon. \quad (2.26)$$

Evaluation results on the VOICES database [74] suggest that compared to the GMM and PLS baselines, the DKPLS approach further improved the converted speech in terms of both speech quality and speaker identity.

As discussed above, statistical parametric method can model the relationship between source and target features effectively. However, due to the model capability, only low-dimensional features are generally employed, which fails to model the spectral details and long-term information revealed by high-dimensional features, thereby resulting in a muffled speech.

2.2.3 Neural Network Method

This group of state-of-the-art approaches is aiming at finding a non-linear mapping between source and target feature vectors using different neural networks.

In recent years, neural networks have shown the potential in achieving good performance in many speech areas, e.g. speech recognition [75–77], speech synthesis [10, 78, 79], voice conversion [57–59], etc.

A neural network consists of a large number of neurons, also named nodes. Each neuron is a non-linear transformation. Let's take a single layer feed forward neural

network as an example. Given a source feature vector $\mathbf{x}$ as a neural network input, the transformation can be expressed as

$$F(\mathbf{x}) = G(\mathbf{W} * \mathbf{x} + b) \quad (2.27)$$

where $\mathbf{W}$ and b are the weight and the bias vector respectively. $G(\cdot)$ is the non-linear activation function, e.g. sigmoid function, hyperbolic tangent function ($\tanh$) and Rectified Linear Unit (ReLU) function. To form a complex model, several interconnected neurons can be combined in a layer-by-layer manner, which is known as Multilayer Perceptron (MLP) [80] or Deep Neural Network (DNN) [75].

Early approaches for voice conversion were usually implemented with feed forward networks where the connections between the neurons did not form a cycle. One of the pioneer works for feed forward neural network based voice conversion was proposed in [81], where a network with two hidden layers (8 hidden units) was used for formant frequencies (3 dimensional) conversion. Later a Radial Basis Function (RBF) network was proposed to transform the the LPC spectral envelope [82]. The work proposed in [57] investigated different architectures of feed forward network for MCCs feature mapping. The results [57] showed that feed forward neural networks are able to convert the spectral features effectively. However, under this type of network, the spectral features are modeled independently, which ignores the temporal dependencies of speech sequences. To deal with this problem, a minimum sequential error criterion [59] is introduced to replace the frame-based error for neural network training. This criterion brings a wide acoustic context into model training and helps the continuity of the generated speech. Results of experiments on the CMU ARCTIC database [73] also confirmed the effectiveness of the proposed approach [59].

More recently, a number of different neural network architectures have been proposed for spectral feature mapping [58, 83, 84]. For example, a Restricted Boltzmann Machine (RBM) is proposed to model the joint distributions of the source and target spectral features [83], while a Deep Belief Network (DBN) is proposed in [84] to convert the speech features in a high-order eigen space rather than the cepstrum space. A more advanced neural network, namely Recurrent Neural Networks (RNN), is also adopted for voice conversion [85, 86]. One of the approaches is proposed in [85], where a Bidirectional Long

Short-Term Memory (BLSTM) based RNN is used for spectral feature conversion. The advantage of this approach is that the network can capture the temporal information in both forward and backward directions, which helps feature estimation and improves the naturalness of the converted speech. Results of experiments on the CMU ARCTIC database indicated that BLSTM significantly outperforms DNN baselines in terms of both quality and similarity.

Neural network models show the capability of modeling a much more complex transformation from source to target speaker and have been proved to outperform statistical parametric voice conversion approaches [57–59, 73, 83]. However, due to model complexity, it usually requires a certain amount of data (e.g. more than 100 utterances) for model training, which is not always available in practice.

2.2.4 Frequency Warping Method

This group of state-of-the-art approaches aims at finding an optimal warping function that warps the frequency axis of the source spectrum towards the target spectrum. A successful frequency warping voice conversion usually consists of two steps: 1) estimating the warping function that transforms the source spectrum in the frequency axis; 2) compensating for the energy residual in each frequency band.

Warping function estimation

Given a pair of source-target spectrum $[\mathbf{x}; \mathbf{y}]$, an invertible warping function $\mathbf{w}$ is first estimated. The warped spectrum $\hat{\mathbf{y}}$ is expressed as

$$\hat{\mathbf{y}} = \mathbf{w}(\mathbf{x}), \quad (2.28)$$

where $\mathbf{x} \in \mathbb{R}^D$ and $\hat{\mathbf{y}} \in \mathbb{R}^D$ are high-resolution spectra, $\mathbf{w} \in \mathbb{R}^D$ is a vector which also serves as the warping function to evaluate $\hat{\mathbf{y}}$ by $\mathbf{w}(\mathbf{x})$.

In recent literature, the warping function $\mathbf{w}$ is either realized by a single parameter, e.g. Vocal Tract Length Normalization (VTLN) based approaches [42, 47, 87–90] (see Figure 2.2 (a)), or represented as a piecewise linear function [22, 24, 38] (see Figure 2.2 (b)).

Figure 2.2: Illustration of the VTLN-based frequency warping and piecewise linear frequency warping.

VTLN aims to normalize a speaker’s voice in order to remove individual speaker characteristics. It shows potential in achieving good performance in speech recognition and may also be suitable for voice conversion. Early attempts at VTLN-based approaches were presented in [47, 87]. In these works, the performance of different VTLN approaches for voice conversion were investigated, including two segments piecewise linear function, power function, quadratic function and bilinear function. The work proposed in [42] improves bilinear frequency warping by an iterative algorithm. However, results presented in studies [47, 87, 91] showed that the single parameter VTLN approach may not adequately describe the mapping of spectral details. Hence, the piecewise linear warping function has become a mainstream solution.

The goal of the piecewise linear warping function is to align a set of frequencies between the source and target spectrum by minimizing the spectral distance between the converted and target spectrum. The set of frequencies can be all the frequency bins or some pivot frequencies. A Dynamic Frequency Warping (DFW) approach is proposed in [38], where an optimal warping function is obtained by minimizing the spectral distance between the converted and target spectrum over all the frequency bins. However, the search space of DFW is not constrained, one-to-many and many-to-one mapping of frequency bins can happen, which may result in an over-

stretched or over-compressed alignment. Alternatively, the warping function can be estimated based on formants frequencies. In practice, the formants frequency and the corresponding mapping can be found either manually [92] or automatically [22]. One example is Automatic Mapping of Formants (AMF) [22], which can be regarded as a special case of DFW. Instead of aligning all the frequency bins, AMF aligns only at the formant frequencies, which are calculated in advance from Line Spectral Frequencies (LSF) vectors. Between the formant frequencies, linear alignment is assumed. In this way, the search space of AMF is reduced. Operating on formant frequencies, AMF reduces the over-stretched or over-compressed problems caused by DFW to obtain a better warping function [30]. More recently, a Correlation-based Frequency Warping (CFW) [29] is proposed to improve the AMF, where it allows for a more relaxed alignment between the source and target around the formants. Additionally, the correlation coefficient is employed as the optimization criterion to estimate the warping functions, instead of the spectral distance. Results of experiments carried out on the VOICES database showed that the CFW approach significantly outperformed the DFW and AMF baselines in terms of the converted speech quality.

To estimate the warping functions for a source target speaker pair, generally, the feature vectors are first separated into clusters. The same warping function is shared within an acoustic class [38]. This leads to the discontinuity of the converted spectra across classes. To mitigate this, a Weighted Frequency Warping (WFW) is proposed in [22], which combines GMM with a frequency warping method to generate smooth warping functions across frames.

Amplitude scaling

As the warping process does not remove any spectral details, frequency warping methods facilitate the quality of converted speech. However, frequency warping itself only transforms the frequency axis without considering the amplitude of the spectrum. Hence, such approaches suffer from speaker similarity problem. To address this, Amplitude Scaling (AS) [24, 42], also known as energy correction filter [22], is proposed to compensate for the amplitude difference between the warped and the target spectrum. The converted spectrum $\tilde{\mathbf{y}}$ is express as

$$\tilde{\mathbf{y}} = \hat{\mathbf{y}}\mathbf{r}, \quad (2.29)$$

where $\hat{\mathbf{y}} \in \mathbb{R}^D$ is the warped spectrum from Equation (2.28), $\mathbf{r} \in \mathbb{R}^D$ is the residual vector which refers to the amplitude difference between the warped spectrum and the target spectrum.

In the WFW [22] approach, the residual $\mathbf{r}$ is defined by a smoothed energy correction filter calculated between the converted spectra from a GMM-based method and a frequency warping based method. Alternatively, in a more recent work [24], the residual $\mathbf{r}$ between the converted and target spectrum is modeled for each acoustic class by a GMM. Results reported in [22, 24] indicated that the frequency warping with amplitude scaling approaches significantly outperformed the GMM baselines in terms of both quality and speaker similarity.

Operating on high-dimensional features, frequency warping method is able to preserve the spectral details and generate high quality speech. However, current approaches still relies on GMM method to estimate warping functions and residual spectra, which reduces the variation of converted speech and limits the speech quality.

2.2.5 Exemplar-based Method

This group of state-of-the-art approaches is aiming at generating the target speech using a weighted linear combination of a set of exemplars from the training data of the target speaker. The set of exemplars is also known as a dictionary.

Recently, exemplar-based sparse representation was studied [93–95], which shows the potential of achieving good performance in voice conversion [46, 60–62, 70]. Typically, an exemplar-based voice conversion system consists of two steps [60]: offline dictionary construction and run-time conversion. During dictionary construction, the paired source-target dictionary, $[\mathbf{A}; \mathbf{B}]$, is established by the aligned source to target spectra. During run-time conversion, as shown in Figure 2.3, given a source spectrogram $\mathbf{X}$, the Nonnegative Matrix Factorization (NMF) [96] is employed to estimate the combination weights, also known as activation matrix $\mathbf{H}$.

$$\mathbf{X} \approx \mathbf{A}\mathbf{H}. \quad (2.30)$$

Figure 2.3: Block diagram of the run-time conversion for exemplar-based voice conversion.

Then the converted spectrogram $\tilde{\mathbf{Y}}$ is generated as a weighted linear combination of target exemplars in $\mathbf{B}$, expressed as

$$\tilde{\mathbf{Y}} = \mathbf{B}\mathbf{H}. \quad (2.31)$$

With the sparsity constraint on the combination weights, the over-smoothing problem is greatly reduced. In addition, as the converted speech is generated from the target exemplars, which are high-resolution features (e.g. spectrum), this group of approaches produces relatively high quality speech.

In a recent work, the exemplar-based voice conversion framework was improved by introducing spectral compression and residual compensation techniques [46]. Spectral compression is used to emphasize the low-intensity spectrum for activation matrix estimation and spectrum conversion. While, residual compensation is employed to compensate for the modeling error of activation matrix estimation to further enhance system performance. Results of experiments carried on the VOICES database showed that the proposed exemplar-based framework significantly outperformed ML-GMM and PLS baselines while was comparable with the DKPLS and ML-GMM-GV baselines [46].

2.3 Synthetic Speech Detection

Another focus of this thesis is to improve the robustness of Automatic Speaker Verification (ASV) systems against synthetic spoofing attacks. ASV systems aim to verify claimed

identity based on a given speech signal. However, state-of-the-art ASV systems are not designed for spoofing attacks, e.g. impersonation, replay, speech synthesis and voice conversion, as reviewed in [19]. Specifically, attacks realized by speech synthesis and voice conversion, which provide easily accessible ways to generate high quality speech of the target speaker, impose a genuine threat to ASV systems [97]. One way to address the threat from synthetic spoofing attacks is to improve the robustness of ASV systems by combining the detection process and verification process [98]. An alternative way is to build some standalone detection systems using different countermeasures, and this is also the focus of this thesis.

Figure 2.4: Block diagram of a synthetic speech detection system with a speaker verification system.

A typical synthetic speech detection system, as shown in Figure 2.4, consists of two components, namely feature extraction and classifier. These are discussed in detail below.

Feature extraction: To build a synthetic speech detection system, the first step is to extract the features from a speech signal. The main objective of feature extraction is to represent the speech signal as features that capture the artifacts related to the synthetic speech. In previous studies, different features were used for synthetic speech detection. For example, Mel-Frequency Cepstrum Coefficient (MFCC) [99, 100], Linear Frequency Cepstral Coefficients (LFCC) [101], Subband Centroid Frequency Coefficient (SCFC) [102] are used to detect the artifacts of magnitude. To detect the

artifacts of phase, a number of phase features were also used, e.g. Modified Group Delay (MGD) [103–105], cosine-normalized phase [103, 105], Relative Phase Shift (RPS) [106–109], etc. However, these features are generally low-dimensional, where detailed information of high-dimensional features is discarded in feature calculation, resulting in a degradation of the system performance.

Classifier: After feature extraction, a classifier is built to discriminate between synthetic and natural speech. A number of classifiers are used in previous works, e.g. Joint Factor Analysis (JFA) [99, 104], Probabilistic Linear Discriminative Analysis (PLDA) [99, 104] and Support Vector Machine (SVM) based classifiers [110, 111]. Among them, GMM is one of the most popular classifier used in many works [99, 104, 112]. Given a feature sequence $\mathbf{X}$, the log-likelihood ratio $\Lambda(\mathbf{X})$ is calculated as

$$\Lambda(\mathbf{X}) = \log L(\mathbf{X}|\lambda_n) - \log L(\mathbf{X}|\lambda_s), \quad (2.32)$$

where L is the likelihood function, and λ_n and λ_s are GMMs for natural and synthetic speech respectively. The likelihood ratio $\Lambda(\mathbf{X})$ is used to decide whether the speech is natural speech based on a threshold.

2.4 Summary

This chapter has given an introduction to typical voice conversion systems, state-of-the-art spectral feature mapping techniques and synthetic speech detection. This was provided by first giving an overview of a typical voice conversion system including the framework for speech analysis and reconstruction, feature extraction, feature alignment, feature conversion and its evaluation metrics. Then, a range of state-of-the-art spectral feature mapping techniques and their limitations were reviewed. These methods also gave background knowledge and work as the baselines of our work in Chapter 3 and 4. Finally, synthetic speech detection was briefly introduced.

The next three chapters are going to introduce our proposed approaches. In Chapter 3 and Chapter 4, two voice conversion approaches are proposed, namely an exemplar-based approach to frequency warping for voice conversion with parallel data and an average modeling approach to voice conversion with non-parallel data. Finally, in Chapter 5, we investigate the use of different feature representations to discriminate between live and synthetic speech.

CHAPTER 3

An Exemplar-based Approach to Frequency Warping for Parallel Data Voice Conversion

In the previous chapter, an introduction to voice conversion systems was given, including a review of state-of-the-art feature mapping technologies and their limitations. Among them, the one of great interest to us is the frequency warping based technique, which is able to preserve spectral details during conversion and prevent over-smoothing from crossing frames. In this chapter, we present a novel framework for parallel data based voice conversion, which combines frequency warping and exemplar-based methods. Thanks to the sparse constraint, the warping function and residual vectors are generated by a linear interpolation of a small set of warping function and residual exemplars respectively. This will reduce the statistical averaging effects inherited from GMM. In addition, by directly applying frequency warping on the high-dimensional spectrum, the proposed method maintains spectral details during conversion.

The chapter is organized as follows. Section 3.1 first discusses the problems of conventional warping function methods, and provides the motivation for exemplar-based frequency warping. The exemplar-based frequency warping framework is then proposed in Section 3.2. Experiments are carried out in Section 3.3 to compare the proposed approach against the state-of-the-art baselines.

3.1 Motivation

In general, frequency warping based voice conversion consists of two parts: 1) warp the source spectrum to that of the target speaker through a warping function on the frequency

axis; 2) compensate for the amplitude difference between the warped spectrum and the target spectrum. Typically, the warping function is estimated by spectral class. The same warping function, shared within an acoustic class, leads to the discontinuity of converted spectra across classes. To avoid hard decisions, GMM is adopted to generate the warping function by interpolating warping functions from different acoustic classes [22]. A GMM-based amplitude scaling approach is then proposed in [24], where the amplitude difference between the source and target spectra is modeled for each acoustic class by a GMM. However, one of the major issues of GMM is that it attempts to minimize the difference between the converted and target features, or to maximize the joint likelihood of the source and target features. These optimization criteria often introduce statistical averaging, which leads to over-smoothing effect in the converted speech.

The rest of this section discusses the limitations of the GMM-based frequency warping framework and provides the motivation of using exemplar-based frequency warping to address the over-smoothing problem.

3.1.1 Limitation of GMM-based Weighted Frequency Warping (WFW) and GMM-based Residual Compensation

In the implementation of WFW [22], two types of feature representations are used, the low-dimensional features are typically Mel-Cepstral Coefficients (MCC) or Line Spectral Frequencies (LSF), and the high-dimensional features are spectra. The low-dimensional features are used for GMM modeling and deriving warping functions, and the high-dimensional features are used to perform run-time conversion, which keeps the spectral details.

Given the aligned source-target spectra pairs, we first learn a GMM with K Gaussian components. Then the K warping functions are derived from the mean vectors of corresponding Gaussian components, denoted as $\{\mathbf{w}_1, \dots, \mathbf{w}_k, \dots, \mathbf{w}_K\}$.

At run-time, for each input frame $\mathbf{x}$, its corresponding weighted warping function $\mathbf{w} \in \mathbb{R}^D$ is estimated by

$$\mathbf{w} = \sum_{k=1}^K p_k(\mathbf{x}) \mathbf{w}_k, \quad (3.1)$$

where $p_k(\mathbf{x})$ and $\mathbf{w}_k \in \mathbb{R}^D$ are the posterior probability and the corresponding warping function of $\mathbf{x}$ in mixture k .

With $\mathbf{w}$, the warped spectrum $\widehat{\mathbf{y}}$ can be obtained by Equation (2.28). To compensate for the amplitude difference between $\widehat{\mathbf{y}}$ and the reference target spectrum, a GMM-based Residual Compensation (GRC) technique was proposed in [24].

Let $\mathbf{x}_n$, $\mathbf{y}_n$ and $\widehat{\mathbf{y}}_n$ be the n^{th} frame of the source, target and estimated target in training data respectively. The residual spectrum $\mathbf{r}_n \in \mathbb{R}^D$ can be computed in log-scale by

$$\log \mathbf{r}_n = \log \mathbf{y}_n - \log \widehat{\mathbf{y}}_n. \quad (3.2)$$

Then, the residual compensation vector $\mathbf{r} \in \mathbb{R}^D$ for input frame $\mathbf{x}$ is calculated in log-scale by

$$\log \mathbf{r} = \sum_{k=1}^K p_k(\mathbf{x}) \left[\sum_{n=1}^N \log \mathbf{r}_n \cdot \gamma_{n,k} \right], \quad (3.3)$$

where N is the number of aligned training frames, $\gamma_{n,k}$ is the occupation probability of the n^{th} frame belonging to the k^{th} Gaussian component. $\sum_{n=1}^N \log \mathbf{r}_n \cdot \gamma_{n,k}$ is the mean residual spectral vector of k^{th} Gaussian component in log-scale.

A compensated warped spectrum is then obtained by

$$\widetilde{\mathbf{y}} = \exp(\log \widehat{\mathbf{y}} + \log \mathbf{r}). \quad (3.4)$$

Under the GMM-based framework, the estimation of warping function Equation (3.1) and residual compensation vector Equation (3.3) suffer from statistical averaging due to posterior probability $p_k(\mathbf{x})$, which is known to be a source of over-smoothing [46].

To deal with the over-smoothing issue in GMM-based framework, we propose to use exemplar-based technique to estimate the warping functions and residual vectors, namely exemplar-based frequency warping (EFW) approach. Specifically, Non-negative Matrix Factorization (NMF) technique has been used to compute weights of exemplars. Due to sparsity constraint, we avoid the statistical averaging problem inherited from GMM-based approaches [22] in warping functions and residual spectra calculation. The technical details of proposed EFW framework will be presented in section 3.2.

3.2 Exemplar-based Frequency Warping with Residual Compensation

In this section, an exemplar-based approach to frequency warping and residual compensation is introduced. The framework consists of two modules: a) dictionary construction;

Figure 3.1: Block diagram of the exemplar-based frequency warping with residual compensation framework. a) During dictionary construction, a source dictionary $\mathbf{A}$, a warping function dictionary $\mathbf{W}$, and a residual dictionary $\mathbf{R}$ are built; b) at run-time conversion, the activation matrix $\mathbf{H}$ is estimated between the run-time speech sample $\mathbf{X}$ and the source dictionary $\mathbf{A}$. The activation matrix $\mathbf{H}$ is then shared by the warping function dictionary $\mathbf{W}$ and the residual dictionary $\mathbf{R}$ when generating the final converted speech.

b) run-time conversion. Figure 3.1 illustrates the workflow of the proposed framework. During dictionary construction, the source and target parallel utterances are first aligned to construct the paired source-target dictionary on the high-dimensional spectrum. For each source feature vector, also known as an exemplar in this work, we construct a warping function to warp the source to target spectrum. The residual error after warping is retained and kept as an exemplar's residual vector. Hence, three dictionaries are established in the dictionary construction process, namely, the source, warping function and residual dictionaries. At run-time conversion, the input spectrum is first decomposed into a linear combination of exemplars from the source dictionary. The combination weights are constrained to be sparse to reduce statistical averaging. These weights are further used to generate an interpolated warping function and spectral residual for each input spectrum. Finally, the interpolated warping function and spectral residual are applied to the input spectrum to generate the output spectrum, which is then passed through a synthesis filter to reconstruct the target speech.

Our framework benefits from three unique properties, 1) we use sparse representation to avoid the over-smoothing issue in a GMM-based framework; 2) the warping functions are applied on a high-dimensional spectrum and across multiple frames to maintain both spectral and temporal quality; 3) the exemplar-based residual compensation is applied to improve the high-dimensional spectrum reconstruction process. Similar to the GMM-based frequency warping framework, we avoid hard clustering by estimating the warping functions over multiple frames to ensure a smooth spectral trajectory [22]. The algorithms and process in Figure 3.1 are described in further detail next.

3.2.1 Dictionary Construction

Given a collection of source-target parallel speech utterances, we first extract the MCC and spectrum features. Then, DTW [52] is applied on the MCC features to align the parallel utterances. Using the found frame-wise alignment, we form the paired source-target spectra dictionary $[\mathbf{A}; \mathbf{B}] = [\mathbf{A}^\top; \mathbf{B}^\top]^\top$, where $\mathbf{A}$ and $\mathbf{B}$ are the source and target dictionaries of dimension $\mathbb{R}^{D \times N}$ respectively, M is the dimension of the spectra, and N is the number of aligned frames. Mathematically

$$\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n, \dots, \mathbf{a}_N], \quad (3.5)$$

$$\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_n, \dots, \mathbf{b}_N], \quad (3.6)$$

where $\mathbf{a}_n, \mathbf{b}_n \in \mathbb{R}^D$ are called exemplars, which are the n^{th} spectra that form the source-target pair. In this work, only the voiced frames are contained in $\mathbf{A}$ and $\mathbf{B}$.

From $[\mathbf{A}; \mathbf{B}]$, we can now derive the warping function dictionary $\mathbf{W}$. Specifically, between each paired exemplars $[\mathbf{a}_n; \mathbf{b}_n]$, a warping function $\mathbf{w}'_n \in \mathbb{R}^D$ can be determined by the frequency warping methods described in Section 2.2.4, such as DFW, AMF and CFW. The set of N warping functions are then used to form a warping function dictionary $\mathbf{W}' \in \mathbb{R}^{D \times N}$.

As the warping functions are estimated from the source and target exemplars independently, these functions may vary widely across frames. It happens even within a continuous voiced segment. Abrupt changes of warping functions across adjacent frames may lead to discontinuity in the converted spectrogram, which is not desirable. To reduce

Figure 3.2: Illustration of the effects of temporal smoothing on warping function dictionaries estimated using DFW, AMF and CFW. (a), (c) and (e) shows a part of warping function dictionaries before smoothing, $\mathbf{W}'$, while (b), (d) and (f) are the respective warping function dictionaries after smoothing, $\mathbf{W}$.

such effects, we post-process the warping function dictionary by taking the contextual information into consideration [46]. Specifically, for each segment, the warping functions of $\mathbf{W}'$ are smoothed by a moving average filter (a.k.a. temporal smoothing window). We denote the n^{th} smoothed warping function as $\mathbf{w}_n$. The set of N smoothed warping functions are then used to form a smoothed warping function dictionary $\mathbf{W} \in \mathbb{R}^{D \times N}$, which is denoted as

$$\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_n, \dots, \mathbf{w}_N]. \quad (3.7)$$

Figure 3.2 shows an example of the warping function dictionaries before and after temporal smoothing. It is observed that the smoothing of warping function dictionary can prevent the discontinuity across adjacent frames effectively. The smoothed warping function dictionary $\mathbf{W}$ is then used for the residual dictionary calculation.

The above warping function dictionary $\mathbf{W}$ establishes an approximate relationship between the source and target exemplars. When we apply the warping function on the source exemplar, an amplitude difference, which is also called the residual, will be presented between the warped and the target spectrum. To improve conversion performance, we retain these residual difference between the warped spectrum of n^{th} source exemplar to the reference target exemplar in log-scale, which is similar to Equation (3.2). Mathematically, this is

$$\log \mathbf{r}_n = \log \mathbf{b}_n - \log \widehat{\mathbf{b}}_n, \quad (3.8)$$

where $\mathbf{b}_n$ is the n^{th} target exemplar, $\widehat{\mathbf{b}}_n$ is the corresponding warped spectrum which is obtained by Equation (2.28), and $\mathbf{r}_n \in \mathbb{R}^D$.

To maintain the nonnegative values in expressing the residual spectrum, we form the residual dictionary, $\mathbf{R} \in \mathbb{R}^{D \times N}$, with a set of N residual vectors as a nonnegative matrix

$$\mathbf{R} = [\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_n, \dots, \mathbf{r}_N]. \quad (3.9)$$

Hence, at the completion of the dictionary construction stage, the source dictionary $\mathbf{A}$, warping function dictionary $\mathbf{W}$ and residual dictionary $\mathbf{R}$ are formed.

3.2.2 Run-time Conversion

At run-time, the three constructed dictionaries, $\mathbf{A}$, $\mathbf{W}$ and $\mathbf{R}$, are used for voice conversion as shown in Figure 3.1 (b).

Let $\mathbf{x} \in \mathbb{R}^D$ be a speech frame from the source speaker to be converted. We first decompose $\mathbf{x}$ into a linear combination of source exemplars in $\mathbf{A}$ as

$$\mathbf{x} \approx \sum_{n=1}^N h_n \mathbf{a}_n = \mathbf{A} \mathbf{h}, \quad (3.10)$$

where $\mathbf{h} \in \mathbb{R}^N$ is the activation vector, of which its n^{th} element h_n denotes the weight associated with exemplar $\mathbf{a}_n$ in $\mathbf{A}$.

To estimate $\mathbf{h}$, the Nonnegative Matrix Factorization (NMF) technique [96, 113] is used.

$$\mathbf{h} = \arg \min_{\mathbf{h} \geq 0} d(\mathbf{x}, \mathbf{A} \mathbf{h}) + \lambda \|\mathbf{h}\|_1, \quad (3.11)$$

where $d(\mathbf{x}, \mathbf{A}\mathbf{h})$ is the Kullback-Leibler (KL) divergence between $\mathbf{x}$ and $\mathbf{A}\mathbf{h}$, λ is the sparsity penalty factor, and $\|\mathbf{h}\|_1$ is L_1 norm on the activation vector. The details of NMF can be found in [113] [46]. In our case, given $\mathbf{A}$ and $\mathbf{x}$, which are matrices of positive values, the estimated $\mathbf{h}$ is also of positive values, such that Equation (3.11) is satisfied.

The activation vector $\mathbf{h}$ in Equation (3.10) describes how an input speech frame $\mathbf{x}$ can be approximated by a sparse linear combination of source exemplars in $\mathbf{A}$. Because the source and target dictionaries, $\mathbf{A}$ and $\mathbf{B}$, are aligned with parallel content, the activation vector $\mathbf{h}$ can be used to construct $\tilde{\mathbf{y}}$ [46, 60],

$$\tilde{\mathbf{y}} = \mathbf{B}\mathbf{h}. \quad (3.12)$$

Instead of constructing $\tilde{\mathbf{y}}$ from $\mathbf{B}$ directly, in this work, we estimated the converted spectrum $\hat{\mathbf{y}}$ and residual spectrum $\hat{\mathbf{r}}$ separately. Note that we have the converted dictionary $\hat{\mathbf{B}}$ and residual dictionary $\mathbf{R}$ in Figure 3.1. From Equation (3.8) and Equation (3.9), the target dictionary $\mathbf{B}$ can be expressed as

$$\log \mathbf{B} = \log \hat{\mathbf{B}} + \log \mathbf{R}, \quad (3.13)$$

which can be rewritten in linear-scale as

$$\mathbf{B} = \hat{\mathbf{B}} \circ \mathbf{R}, \quad (3.14)$$

where, $\circ$ denotes the element-wise multiplication. Then, substituting Equation (3.14) into Equation (3.12) yields

$$\tilde{\mathbf{y}} = (\hat{\mathbf{B}} \circ \mathbf{R})\mathbf{h}. \quad (3.15)$$

Due to the sparsity constraint of $\mathbf{h}$, Equation (3.15) can be further approximated as

$$\tilde{\mathbf{y}} = (\hat{\mathbf{B}} \circ \mathbf{R})\mathbf{h} \approx (\hat{\mathbf{B}}\mathbf{h}) \circ (\mathbf{R}\mathbf{h}). \quad (3.16)$$

We now express Equation (3.16) in log-scale as

$$\log \tilde{\mathbf{y}} \approx \log(\hat{\mathbf{B}}\mathbf{h}) + \log(\mathbf{R}\mathbf{h}), \quad (3.17)$$

where, $\hat{\mathbf{B}}\mathbf{h}$ denotes the warped spectrum $\hat{\mathbf{y}}$, and $\mathbf{R}\mathbf{h}$ denotes the residual vector $\hat{\mathbf{r}}$. Next we justify the run-time implementation of Equation (3.17).

3.2.2.1 Exemplar-based Frequency Warping

The warped spectrum $\hat{\mathbf{y}}$ in Equation (3.17) can be further written as a linear combination of exemplars of converted dictionary

$$\hat{\mathbf{y}} = \hat{\mathbf{B}}\mathbf{h} = \sum_{n=1}^N h_n[\mathbf{w}_n(\mathbf{a}_n)], \quad (3.18)$$

where $\mathbf{w}_n(\mathbf{a}_n) \in \mathbb{R}^D$ denotes the n^{th} exemplar in $\hat{\mathbf{B}}$.

As illustrated in Section 3.2.1, the warping function dictionary $\mathbf{W}$ preserves the relationship between $\mathbf{A}$ and $\mathbf{B}$. Therefore, given a speech frame $\mathbf{x}$, it is logical to assume that the same activation vector $\mathbf{h}$ can also be used to interpolate the $\mathbf{W}$ to obtain the warping function $\hat{\mathbf{w}} \in \mathbb{R}^D$ in a similar way as Equation (3.1)

$$\hat{\mathbf{w}} = \sum_{n=1}^N h_n \mathbf{w}_n = \mathbf{W}\mathbf{h}. \quad (3.19)$$

Then, Equation (3.18) can be rewritten as

$$\hat{\mathbf{y}} \approx \hat{\mathbf{w}}\left(\sum_{n=1}^N h_n \mathbf{a}_n\right) = \hat{\mathbf{w}}(\mathbf{x}). \quad (3.20)$$

As each speech frame requires an independent factorization and warping process, for a given input utterance with Q frames of spectrogram, i.e. $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_q, \dots, \mathbf{x}_Q] \in \mathbb{R}^{D \times Q}$, we can construct its corresponding activation matrix $\mathbf{H} = [\mathbf{h}_1, \dots, \mathbf{h}_q, \dots, \mathbf{h}_Q] \in \mathbb{R}^{N \times Q}$. Mathematically, this is

$$\mathbf{X} \approx \mathbf{A}\mathbf{H}. \quad (3.21)$$

Thus, the warping functions $\hat{\mathbf{W}}$ of $\mathbf{X}$ is calculated by

$$\hat{\mathbf{W}} = \mathbf{W}\mathbf{H}. \quad (3.22)$$

Finally, we apply the warping functions $\hat{\mathbf{W}} = [\hat{\mathbf{w}}_1, \dots, \hat{\mathbf{w}}_q, \dots, \hat{\mathbf{w}}_Q] \in \mathbb{R}^{D \times Q}$ on the source spectrogram $\mathbf{X}$ to obtain the warped spectrogram $\hat{\mathbf{Y}} = [\hat{\mathbf{y}}_1, \dots, \hat{\mathbf{y}}_q, \dots, \hat{\mathbf{y}}_Q] \in \mathbb{R}^{D \times Q}$.

3.2.2.2 Exemplar-based Residual Compensation

To improve the quality of generated $\hat{\mathbf{Y}}$, Exemplar-based Residual Compensation (ERC) is applied to compensate for the amplitude difference. In view of the intrinsic relationship between source dictionary $\mathbf{A}$ and residual dictionary $\mathbf{R}$, we use the same activation vector $\mathbf{h}$ in Equation (3.10) to derive a spectral residual for input speech frame $\mathbf{x}$. For a given spectrogram $\mathbf{X}$, the residual spectrogram, as expressed in the second term of Equation (3.17), can be calculated as

$$\hat{\mathbf{R}} = \mathbf{R}\mathbf{H}. \quad (3.23)$$

Finally, the converted spectrogram $\tilde{\mathbf{Y}} = [\tilde{\mathbf{y}}_1, \dots, \tilde{\mathbf{y}}_Q] \in \mathbb{R}^{D \times Q}$ is obtained by combining the residual spectrogram $\mathbf{R} \in \mathbb{R}^{D \times Q}$ and the warped spectrogram $\hat{\mathbf{Y}} \in \mathbb{R}^{D \times Q}$ in the log-scale

$$\tilde{\mathbf{Y}} = \exp(\log \hat{\mathbf{Y}} + \log \hat{\mathbf{R}}). \quad (3.24)$$

3.2.3 Exemplar-based Frequency Warping vs. Exemplar-based Sparse Representation

Both the proposed exemplar-based frequency warping method and the exemplar-based sparse representation [46] originate from the exemplar-based idea. Their differences lie in the conversion function and the process.

During dictionary construction, the exemplar-based sparse representation framework [46] establishes a paired source-target dictionary $[\mathbf{A}; \mathbf{B}]$, that defines the frame-to-frame mapping correspondence, while the exemplar-based frequency warping framework establishes a warping function dictionary $\mathbf{W}$ in addition to $\mathbf{A}$ and $\mathbf{B}$.

At run-time conversion, the exemplar-based sparse representation framework [46] estimates the linear combination weights of exemplars $\mathbf{H}$ from the source dictionary. $\mathbf{H}$ is then applied to the target dictionary to realize the conversion. Although the exemplar-based frequency warping framework also estimates $\mathbf{H}$ from the source dictionary, $\mathbf{H}$ is applied to the warping function dictionary to compose $\hat{\mathbf{W}}$, which is then used to generate $\hat{\mathbf{Y}}$.

For residual compensation, the exemplar-based sparse representation framework [46] utilizes two residual dictionaries, from both the source and target dictionaries. A linear transformation is estimated between these two residual dictionaries. At run-time

conversion, the source residual spectrogram is mapped to obtain the target residual spectrogram [46] with the linear transformation. In the exemplar-based frequency warping framework, we only obtain one residual dictionary $\hat{\mathbf{R}}$ during dictionary construction (See Figure 3.1). At run-time conversion, we propose a new strategy to obtain the target residual spectrogram by interpolating the residual dictionary $\mathbf{R}$ with the same activation matrix $\mathbf{H}$ used for the warping function.

3.3 Evaluation

In this section, we conducted two sets of experiments to validate the proposed framework. We first conducted experiments to find the optimal configuration. Then, we evaluated the proposed EFW framework with the optimal configuration by comparing it against the state-of-the-art baselines.

3.3.1 Database

The VOICES database [74] was used in all the experiments to report system performance. The data from two male (*jal* and *jcs*) and two female (*leb* and *sas*) speakers were used for experiments, that make up 12 speaker pairs, 4 intra-gender and 8 inter-gender conversions. 20 parallel utterances of each speaker were used as the training set, another 10 utterances were used as the development set, and another 20 utterances were used as the evaluation set. All speech signals were sampled at 16kHz and the average length was around 4 seconds. Hence, the total length of 20 training utterances was around 80 seconds.

We used STRAIGHT to extract the 513-dimensional spectrum [37], aperiodicity measures and $\log F_0$ with 5 ms frame shift. The MCC in 25 dimensions and the LSF in 14 dimensions were also extracted from the STRAIGHT spectrum.

3.3.2 System Setup

We validated the proposed ideas by comparing against three categories of the state-of-the-art techniques. To understand how the component modules work, we also reported the contrastive results for their different combinations. Specifically, twenty one combinations

Table 3.1: The summary of three categories of systems in seven groups. In category (I), we report the state-of-the-art parametric approaches. In category (II), we report a cluster of GMM-based frequency warping techniques with and without residual compensation. In category (III), we compare the proposed EFW approaches with the GMM counterpart in category (II).

Category	Group	Conversion method	Weighted frequency warping	Residual compensation	Number of Gaussians	Smoothing window for WF dictionary	Feature for training	Feature for conversion	LSD ratio (%)
(I) Parametric approaches	(a)	ML-GMM ML-GMM-GV DKPLS	N/A	N/A	64	N/A	24-dim MCCs with delta	24-dim MCCs with delta	66.38 66.60 66.21
	(b)	GAMF GDFW GCFW	GMM (Eq. (3))	N/A	N/A	N/A	14-dim LSFs	513-dim spectrum	97.13 86.34 96.79
(II) GMM-based frequency warping approaches	(c)	GAMF-GRC GDFW-GRC GCFW-GRC	GMM (Eq. (3))	GMM (Eq. (5))	32	N/A	14-dim LSFs	513-dim spectrum	71.96 72.25 71.08
	(d)	GAMF-ERC GDFW-ERC GCFW-ERC	GMM (Eq. (3))	Exemplar (Eq. (25))	32	N/A	14-dim LSFs	513-dim spectrum	61.38 64.73 61.60
	(e)	EAMF EDFW ECFW	Exemplar (Eq. (24))	N/A	N/A	15	513-dim spectrum	513-dim spectrum	94.99 94.77 96.57
	(f)	EAMF-GRC EDFW-GRC ECFW-GRC	Exemplar (Eq. (24))	GMM (Eq. (5))	32	15	513-dim spectrum	513-dim spectrum	71.80 73.59 70.26
(III) Exemplar-based frequency warping approaches	(g)	EAMF-ERC EDFW-ERC EFW(ECFW-ERC)	Exemplar (Eq. (24))	Exemplar (Eq. (25))	N/A	15	513-dim spectrum	513-dim spectrum	60.50 63.71 58.70

of methods in seven groups were studied under the three categories as summarized in Table 3.1, that included the GMM method with global variance (GV) enhancement and Dynamic Kernel Partial Least Square (DKPLS) method. Note that the exemplar-based sparse representation method does not outperform the GMM method with GV enhancement [46]. Therefore, exemplar-based sparse representation method was not included as a baseline.

- (I) *Parametric approaches:* We implemented the JD-GMM with maximum likelihood parameter conversion [16] (ML-GMM) and its variant with GV enhancement (ML-GMM-GV) [114]. ML-GMM-GV used GV that is derived from the target speaker to reduce the muffled effects. Both static and its dynamic features were used in these implementations. We also implemented the dynamic kernel partial least squares regression (DKPLS) method [45]. We used the same configuration as in [45], with the context window size was set to 3, and the number of latent components and reference vectors were 35 and 200 respectively. ML-GMM-GV and DKPLS represent the state-of-the-art of parametric approaches.
- (II) *GMM-based frequency warping approaches:* We implemented the WFW method [22], where we used LSF features to train GMM and Automatic Mapping of Formants (AMF) to derive the warping functions. We call the first technique GAMF. We also implemented two variants of WFW where AMF was replaced by Dynamic Frequency Warping (DFW) [38] and Correlation-based Frequency Warping (CFW) [29], that we called GDFW and GCFW respectively. We also studied the effect of GMM-based residual compensation (Equation (3.3)) and exemplar-based residual compensation (Equation (3.23)), denoted as GRC and ERC, in conjunction with the above three frequency warping schemes respectively.
- (III) *Exemplar-based frequency warping approaches:* We implemented the proposed exemplar-based frequency warping method (EFW), with three types of warping functions, AMF, DFW and CFW. We call these three setups of EFW as EAMF, EDFW and ECFW respectively. We also studied the effect of GRC and ERC, in conjunction with the above three frequency warping schemes respectively.

All the conversion methods shared the same frame alignment. We first obtained the alignment by applying DTW over MCC features. In the aligned frame pairs, some source and target frames were repeated due to the warping effect. We only retained the frame pairs that had both distinctive source or target frames in the dictionary and removed the rest. Note that for parametric approaches, we used all the aligned frames for training. However, for frequency warping approaches, we involved only the voiced frames in establishing the source-target dictionaries and estimating the warping functions. For different speaker pairs, the number of exemplars ranged between 6,700 and 7,100.

At run-time conversion, the aperiodicity coefficients were carried over from source to target speakers, while the F_0 was converted by a global linear transformation in log-scale [115].

We empirically set the sparsity penalty factor λ to 0.1. To verify the sparsity of the activation weights, we calculated the average top weights over the development utterances for all speaker pairs. For each speaker pair, we sorted the activation weight vectors of frames in a descending order. Then, we averaged the activation weights over all the frames of the development utterances. We observed that, for all the speaker pairs, the top 100 averaged weights accounted for over 75% overall weights. This confirmed the effectiveness of the sparsity constraint and was consistent with the results in [46].

As suggested in [46, 116], a spectral compression factor p was used in EFW approaches to maintain the nonnegative nature and emphasize the low intensity frequency bands of the spectral features. From the results on the development data, we empirically set p to 0.4.

3.3.3 Evaluation Metrics

Both objective and subjective evaluations were conducted to perform a systematic comparison. Specifically, objective evaluation was used to measure spectral distortion between the converted and reference target speech, while subjective evaluation was used to examine naturalness and speaker identity respectively.

3.3.3.1 Objective Evaluation

The log spectral distortion (LSD) ratio [68] was employed to measure spectral distortion, as discussed in Section 2.1.5.1. The average LSD ratio over all evaluation pairs was

reported. A lower LSD ratio indicates a smaller spectral distortion. As the unvoiced frames were not transformed in frequency warping approaches, only voiced frames were involved in the LSD ratio calculation.

3.3.3.2 Subjective Evaluation

We also conducted AB preference tests and XAB tests as the subjective evaluation to confirm the findings in the objective evaluation. We reported the assessment in terms of speech quality and speaker similarity.

Amazon Mechanical Turk (AMT), a widely used crowdsourcing platform [117–119], was used in all the listening tests. The evaluation set was constructed by 12 speaker pairs with 20 converted sentences for each pair. In the test, 20 sentences were randomly selected from the 240 testing samples. To prevent cheating, three golden standard pairs were randomly mixed with the testing samples [119], [46]. 20 paid subjects participated in all the listening tests.

T-test with a 95% confidence interval was used in all the subjective evaluations to measure the significance of the differences between different voice conversion methods.

3.3.4 Validate the configurations of EFW framework

In this section, we examined four configurations of EFW framework, including the effects of dictionary size, smoothing the warping function dictionary, different frequency warping methods and exemplar-based residual compensation.

3.3.4.1 Effects of Dictionary Size

Addressing the problem of limited training data, we would like to know the effects of different training data size. Note that 20 training utterances offer about 7,000 exemplars in the dictionaries. We studied the effects by reducing the number of exemplars in the dictionaries, that is equivalent to reducing the number of training utterances.

Figure 3.3 presents the LSD ratios as a function of the number of exemplars in the dictionary from the total of about 7,000 to a small sub-sample of 500. It was observed that the LSD ratio decreases as the number of exemplars increases in both development and evaluation data sets.

Figure 3.3: Spectral distortion as a function of dictionary size.

According to Section 3.2.2, activation matrix estimation is the main computational loads on the EFW framework during run-time conversion. It was observed in the experiments that the computational cost of this module was almost linearly correlated with the size of the source and target dictionaries, i.e. the system becomes approximately two times faster when dictionary size was halved.

To compare fairly with baseline methods, all exemplars were used in the EFW framework in the following experiments.

3.3.4.2 Effects of Smoothing the Warping Function Dictionary

As discussed in Section 3.2.1, inter-frame discontinuity may appear in the warping function dictionary $\mathbf{W}'$. Here we proposed a temporal smoothing window to address this problem. To validate our proposal, we conducted experiments using ECFW-ERC to assess the effectiveness of the smoothed warping function dictionary.

Objective evaluation. Firstly, we examined the effect of the temporal smoothing window size for the warping function dictionary. The experiments were conducted on both development and evaluation sets. The LSD ratio was measured with different smoothing window sizes from 1 (5ms) to 19 (95ms). As shown in Figure 3.4 (a), experiments on both the development and evaluation sets show that LSD ratios improve as the smoothing window size increases. After a window size of 15 (75ms), we

Figure 3.4: (a) Spectral distortion as a function of smoothing window size for warping function dictionary. (b) Mean square error (MSE) of delta feature as a function of smoothing window size for warping function dictionary.

observe that the LSD ratio plateaus out. As shown in Figure 3.4 (b), we also observe a similar trend for delta feature measurement. Therefore, we set the window size as 15 (75ms) for the rest of the experiments.

Table 3.2: *Spectral distortion of exemplar-based frequency warping methods with and without temporal smoothing (window size 1 vs. 15) on the evaluation set.*

Conversion method	Window size	LSD ratio (%)	Window size	LSD ratio (%)
EAMF	1	95.28	15	94.99
EDFW	1	95.20	15	94.77
ECFW	1	96.73	15	96.57
EAMF-ERC	1	63.46	15	60.50
EDFW-ERC	1	70.32	15	63.71
ECFW-ERC	1	61.00	15	58.70

Secondly, we examined the performance of the smoothed warping function dictionary under the EFW framework across different frequency warping methods. As shown in Table 3.2, by using the smoothed warping function dictionary, the LSD ratios of all the three variants of EFW methods decrease. This confirms the need for temporal smoothing in the construction of the warping function dictionary.

Figure 3.5: Preference t-test result of speech quality with 95% confidence intervals for ECFW-ERC with and without smoothing the warping function dictionary.

Subjective evaluation. We also conducted an AB preference evaluation to validate the effectiveness of the smoothed warping function dictionary using ECFW-ERC. We set the smoothing window size to 1 for no inter-frame smoothing (win 1), and 15 for inter-frame smoothing (win 15). As shown in Figure 3.5, the ECFW-ERC (win 15) achieves significantly better speech quality than the ECFW-ERC (win 1), with the preference scores of $(55 \pm 4.5)\%$ and $(45 \pm 4.5)\%$ respectively. Both objective and subjective results confirm the effectiveness of smoothing the warping function dictionary for the EFW framework. Hereafter, all the experiments were reported with the smoothed warping function dictionary whenever it is used.

3.3.4.3 CFW vs. AMF and DFW

We compared the performance of different frequency warping methods, AMF, DFW and CFW, with and without residual compensation. The objective tests were reported in Table 3.1 group (b), (c), (e) and (g).

Objective evaluation. We observe that under the GMM-based frequency warping framework the three frequency warping techniques produced similar results within group (b) and (c) of Table 3.1, with GDFW achieving the lowest LSD ratio without residual compensation, and GCFW-GRC achieving the lowest LSD ratio with residual compensation. All results suggest that residual compensation contributed significantly while correlation-based frequency warping with residual compensation (GCFW-GRC) achieves the best overall LSD ratio at 71.08%. We apply GMM-based Residual Compensation (GRC) in group (c) that is formulated in Equation (3.3) as part of the GMM-based frequency warping framework [24]. Equation (3.1) and Equation (3.3) share the same posterior probability of Gaussian components.

A similar pattern is observed within group (e) and (g) of Table 3.1. We apply Exemplar-based Residual Compensation (ERC) in group (g) which is a part of the EFW framework as illustrated in Figure 3.1, where Equation (3.22) and Equation (3.23) share the same activation matrix. In group (e), EDFW achieves the lowest LSD ratio of 94.77% without residual compensation. In group (g), ECFW-ERC achieves the overall lowest LSD ratio of 58.70%.

Results in group (c) and (g) confirm the effectiveness of correlation-based frequency warping.

Subjective evaluation. Two sets of subjective evaluations were performed to assess the speech quality between CFW and the other two frequency warping methods, AMF and DFW.

Figure 3.6: Preference test results of speech quality with 95% confidence intervals for GMM-based and exemplar-based frequency warping methods without residual compensation. (a) GAMF vs. GCFW, (b) GDFW vs. GCFW, (c) EAMF vs. ECFW and (d) EDFW vs. ECFW.

Firstly, we conducted an AB preference test to assess the speech quality between CFW and the other frequency warping methods, AMF and DFW, without residual compensation in group (b) and (e) of Table 3.1. Figure 3.6 presents the preference scores with 95% confidence intervals. Figure 3.6 (a) and Figure 3.6 (b) suggest that

Figure 3.7: Preference test results of speech quality with 95% confidence intervals for GMM-based and exemplar-based frequency warping methods with residual compensation. (a) GAMF-GRC vs. GCFW-GRC, (b) GDFW-GRC vs. GCFW-GRC, (c) EAMF-ERC vs. ECFW-ERC and (d) EDFW-ERC vs. ECFW-ERC.

the speech quality of GCFW outperforms that of GAMF and GDFW under the GMM-based frequency warping framework. Similar results are also observed in Figure 3.6 (c) and Figure 3.6 (d), which suggest that ECFW outperforms EAMF and EDFW under the exemplar-based frequency warping framework as well.

Secondly, we conducted an AB preference test to assess the speech quality between CFW and other frequency warping methods with residual compensation in group (c) and (g) of Table 3.1. We observe in Figure 3.7 that CFW consistently outperforms AMF and DFW in all the tests, although the difference between EAMF-ERC ($(47.5 \pm 3.35)\%$) and ECFW-ERC ($(52.5 \pm 3.35)\%$) is not significant. The results are also consistent with those reported in the objective evaluation (See Table 3.1).

3.3.4.4 GMM-based vs. Exemplar-based Residual Compensation

We consider that exemplar-based residual compensation provides a more detailed modelling than GMM-based residual compensation. To validate the idea, we examined the two residual compensation methods in combination with both GMM-based and exemplar-

based frequency warping frameworks, and reported the objective tests in group (c), (d), (f) and (g) of Table 3.1.

Objective evaluation. We observe that the Exemplar-based Residual Compensation (ERC) outperforms its GMM-based counterpart (GRC) across all experiments, by around 10% in terms of LSD ratio reduction.

Figure 3.8: Preference test results of speech quality with 95% confidence intervals for GMM-based and exemplar-based residual compensation with GCFW and ECFW systems.

Subjective evaluation. We further conducted an AB preference test to compare the two residual compensation methods under GMM-based and exemplar-based frequency warping frameworks. Note that CFW achieves consistently good results in previous experiments. Here we only implemented GCFW and ECFW in this study. We observe in Figure 3.8 (a) that the GCFW-GRC outperforms GCFW-ERC which contradicts the objective evaluation results in group (c) and (d) of Table 3.1, where GCFW-ERC has a lower LSD ratio. We understand that subjective evaluation provides us a different perspective from objective evaluation. We also observe in Figure 3.8 (b) that ECFW-ERC outperforms ECFW-GRC, but their preference scores fall into each other’s confidence intervals ($(51 \pm 4.16)\%$ and $(49 \pm 4.16)\%$), which means they are not statistically significant. Subjective evaluation seems to favor the implementation where the warping function and the residual compensation share the same linear combination weights. Specifically, we have the posterior probability of Gaussian components as the linear combination weights in the GMM-based framework, and the activation matrix in the exemplar-based framework.

3.3.5 EFW Framework vs. State-of-the-Arts

In this section, we validated the proposed EFW framework with optimal configuration against the baseline methods in terms of both speech quality and speaker similarity.

3.3.5.1 Speech Quality Evaluation

In this work, we advocated the use of 1) exemplar-based frequency warping (EFW) method, and 2) exemplar-based residual compensation (ERC) method. We have confirmed that ERC is superior to the GMM-based baselines. In this section, we conducted objective and subjective evaluations to assess the speech quality between the proposed EFW method with the state-of-the-art baselines.

Objective evaluation. From Table 3.1, we clearly see that all EFW methods (Table 3.1 group (g)) outperform their GMM-based counterparts (Table 3.1 group (c)), with ECFW-ERC giving the lowest LSD ratio of 58.70%. The results suggest that exemplar-based frequency warping and residual compensation methods offer a clear advantage over GMM-based methods individually and collectively.

It is worth noting that the ECFW-ERC implementation outperforms the state-of-the-art parametric approaches, ML-GMM, ML-GMM-GV and DKPLS, as reported in Table 3.1.

Subjective evaluation. Then we conducted subjective evaluations to assess speech quality between the proposed EFW method with the state-of-the-art baselines.

First, subjective evaluations were conducted to assess speech quality between GMM-based and exemplar-based frequency warping frameworks. In Section 3.3.4.4, experiments suggest that it is advantageous for the warping function and residual compensation to share the same framework, either GMM-based or exemplar-based. Here we only evaluated the three typical comparative pairs, GAMF-GRC vs. EAMF-ERC, GDFW-GRC vs. EDFW-ERC and GCFW-GRC vs. ECFW-ERC. We observe in Figure 3.9 that for all the three frequency warping methods, AMF, DFW and CFW, the exemplar-based framework performs significantly better than the GMM-based framework. These results also resonate with the objective evaluations in Table 3.1.

Figure 3.9: Preference test results of speech quality with 95% confidence intervals for GMM-based and exemplar-based frequency warping frameworks.

Hence, we conclude that the EFW framework is superior to the GMM-based frequency warping framework in terms of speech quality.

Finally, we compared the ECFW-ERC with the statistical parametric methods.

Figure 3.10: Preference test results of speech quality with 95% confidence intervals for ECFW-ERC vs. parametric methods ML-GMM, ML-GMM-GV and DKPLS.

In [46], exemplar-based sparse representation method was benchmarked against ML-GMM, ML-GMM-GV and DKPLS. Similarly, we would like to conduct an AB preference tests between the proposed EFW method with these three baselines. The preference scores with 95% confidence intervals reported in Figure 3.10 clearly show

that the proposed ECFW-ERC outperforms ML-GMM, ML-GMM-GV and DKPLS by a large margin, while the exemplar-based sparse representation is not superior to ML-GMM-GV [46].

We are glad to see that all results in Figure 3.6, 3.7, 3.9 and 3.10 resonate with those in Table 3.1 in the sense that better subjective test results come with lower LSD ratios. Figure 3.8 (a) does present a different view that we consider it as an outlier.

3.3.5.2 Speaker Similarity Evaluation

Figure 3.11: Comparison of speaker identification rates with 95% confidence intervals for different voice conversion methods.

We conducted XAB test to assess the speaker similarity for each voice conversion method. From the objective and subjective evaluations, we note that residual compensation helps consistently. In this test, we only involved systems with residual compensation. Figure 3.11 reports the results of identification rate for the voice conversion methods of group (a), (c) and (g) in Table 3.1. We note that EDFW-ERC achieves the lowest identification rate of $(67.50 \pm 9.8)\%$, which is significantly lower than GDFW-GRC and ML-GMM-GV with the identifications of $(83.75 \pm 5.43)\%$ and $(85 \pm 7.17)\%$ respectively.

While the identification rates of other methods fall into each other’s confidence intervals. In other words, they are not significantly different.

To summarize, subjective evaluation confirms that the proposed exemplar-based frequency warping framework achieves a significantly higher quality of speech than the state-of-the-art baselines, while maintaining a similar speaker identification rate.

3.4 Summary

This chapter proposed a voice conversion framework based on exemplar-based frequency warping and residual compensation for parallel data voice conversion. Under exemplar-based frequency warping (EFW) framework, both warping functions and residual spectra were generated using a weighted linear combination of a set of exemplars from the warping function and residual dictionaries respectively. There are mainly three advantages of the EFW over GMM-based frequency warping framework:

- As the linear combination weights are constrained to be sparse, over-smoothing problem in GMM-based framework [22] was reduced;
- The proposed EFW framework benefited from both exemplar-based sparse representation and frequency warping to improve spectral details;
- The exemplar-based residual compensation (ERC) that shared the same activation matrix with the warping function reduced spectral distortion and enhanced converted speech quality.

Experiments were conducted on the VOICES database, with a comparison between EFW and the state-of-the-art baselines. Both objective and subjective results confirm the effectiveness of the proposed methods. Specifically, EFW achieved significantly higher speech quality than state-of-the-art methods while maintaining similar speaker identity.

CHAPTER 4

Average Modeling Approach for Non-Parallel Data Voice Conversion

As discussed in Chapter 2, conventional voice conversion techniques require source-target parallel speech data for model training. However, such parallel data may not always be available in practice. In this chapter, we propose a non-parallel data approach for voice conversion that we call the average model adaptation approach. The proposed approach makes use of an average model that maps speaker-independent linguistic features to speaker dependent acoustic features. An average model is trained with speech from multi-speakers and should not have any bias depending on speaker identity. This model is then served as an initial model used to adapt towards target by using a small amount of speech data of target speaker. As linguistic features and acoustic features can be extracted from the same utterance, the proposed approach doesn't require parallel data for either average model training or adaptation.

The chapter is organized as follows. Section 4.1 first discusses the problems of conventional non-parallel data based voice conversion methods, and provides the motivation for the average model adaptation approach. The average model approach is then proposed in Section 4.2. Experiments and evaluations are then carried out in Section 4.3 and 4.4 to compare the proposed method against the state-of-the-art baselines.

4.1 Motivation

Most of the voice conversion techniques rely on parallel data to train the conversion model. However, such parallel data is always expensive to collect and sometimes may

not be available in practice. In previous studies, there were mainly three ways to perform voice conversion with non-parallel data: 1) non-parallel data alignment, 2) adaptation on parallel data based average model and 3) conversion with speaker independent features. A brief review of non-parallel data based voice conversion techniques along with a discussion on their limitations is now given below.

Non-parallel Data Alignment:

Figure 4.1: Block diagram of non-parallel data voice conversion based on non-parallel data alignment.

As discussed in Section 2.1.3, non-parallel data alignment aims to find paired source-target frames from non-parallel corpus. Then, the conversion model is trained on the aligned source-target feature pairs, as shown in Figure 4.1. A simple approach is to cluster the source and target feature vectors into classes, then the aligned features can be obtained by following a nearest-neighbor criterion inside corresponding source-target class pairs. However, a performance deterioration of 15% to 25% was reported, compared to the parallel data training case [53]. In another approach, an HMM-based speech recognizer was used to label all source and target data. Alignment was then performed by finding the longest matching sequences between the source and target corpus [54]. To further improve the alignment accuracy, an iterative alignment approach was introduced in [25, 120], namely ‘Iterative combination of a Nearest Neighbor search step and a Conversion step Alignment method’ (INCA). According to the objective results, this alignment method was able to achieve performance comparable to that of parallel data cases [25]. However, the alignment performance will be affected by the data size of the source and target speakers.

Figure 4.2: Block diagram of non-parallel data voice conversion based on adaptation on parallel data based average model. An average model is first trained by the parallel data from multiple speaker pairs. Then the adaptation is performed by a small amount of non-parallel data from source and target speakers.

Adaptation on Parallel Data based Average Model:

An alternative method aims to learn a general model from pre-defined speaker pairs with parallel data. Adaptation is then applied on the general model towards the target speaker pair with a small amount of non-parallel data. An example of adaptation on parallel data based average model is given in Figure 4.2. In practice, different adaptation techniques are employed for the model adaptation. The Maximum A Posterior (MAP) was first proposed in [26], which adapted the mean of the pre-trained GMM conversion function, while in [27], the eigenvoice technique was proposed to realize mean adaptation for GMM-based voice conversion. Recently, an i-vector and average model adaptation approach was used for LSTM-based voice conversion [28], where the i-vectors of both the source and target speakers were presented as a part of the input for average model training. Given a new speaker pair, their i-vectors augmented with source spectral features are presented to the average model to generate the target speaker's voice. Although these methods demonstrate the ability to perform adaptation with non-parallel data, they still rely on parallel data from pre-defined speaker pairs for the average model training, which is also difficult to collect in practice.

Figure 4.3: Block diagram of non-parallel data voice conversion based on speaker independent (SI) features. The conversion model is learned between the SI feature representations and speaker dependent (SD) acoustic features.

Conversion with Speaker Independent Features:

A recent trend in non-parallel data approaches is to make use of speaker independent feature to bridge between the different speakers for voice conversion. One example can be found in [121], where a speech signal was decomposed into speaker dependent and speaker independent features. The conversion was then realized by replace the speaker dependent features with that of the target. Alternatively, the conversion model can be learned between the speaker independent feature representations and speaker dependent acoustic features [122, 123], as shown in Figure 4.3. For example, a Phonetic PosteriorGram (PPG) based approach was proposed in [123], where an automatic speech recognition (ASR) system is used to extract speaker independent PPG features. Recently, the variational auto-encoder is also adopted to learn speaker independent latent representations [122]. The evaluation carried out on the CMU ARCTIC database showed that the proposed approach outperformed the parallel data baseline in terms of both speech quality and speaker similarity [123]. Hence, this type of non-parallel data approach is also the focus of this chapter.

4.2 Average Modeling Approach (AMA) to Voice Conversion

In this section, an average modeling approach (AMA) is introduced for non-parallel data voice conversion. This is motivated by the strategy used in Phonetic PosteriorGram (PPG) based voice conversion, where a speech signal was encoded into both speaker dependent (acoustic) and speaker independent (linguistic) features. Then the conversion model was trained between linguistic to acoustic features of the same speaker. The proposed AMA extends the PPG approach from a single speaker model to an average model, where non-parallel data from multiple speakers was used for model training. Therefore, we can utilize the publicly available speech data for average model training, while less target speech is required for adaptation. This section first introduces the idea of PPG-based voice conversion, and then describes the two average modeling approaches for non-parallel data voice conversion.

4.2.1 Phonetic PosteriorGram (PPG) based voice conversion

Figure 4.4: An example of the Phonetic Posteriorgram. The horizontal axis refers to the frame number, while the vertical axis refers to the phonetic classes. For example, in our case, the 42 dimension are corresponding to the 42 English phonemes.

Among the non-parallel based attempts, one of the successful approaches is the PPG-based VC [123], which models the relationship between PPG features, that is called the linguistic feature, to the acoustic features. PPG [124], as shown in Figure 4.4, is a sequence of probability vectors estimated with a general purpose automatic speech recognition (ASR) system according to the given acoustic features, e.g. filterbank and MFCC. As the ASR system is designed to generate the outputs invariant to the input speaker, PPG feature is considered to be speech content dependent while speaker independent. Hence, the conversion model can be trained between PPG features and their corresponding acoustic features of target speaker, where the source speaker information is not required in the training process. Figure 4.5 illustrates the training and conversion process.

Figure 4.5: Block diagram of PPG based voice conversion.

During offline training, as shown in Figure 4.5(a), two types of features are extracted from target speaker t , the speaker independent linguistic feature PPG $\mathbf{L}^t \in \mathbb{R}^{D1 \times N}$, and the speaker dependent acoustic feature MCC $\mathbf{A}^t \in \mathbb{R}^{D2 \times N}$. We have N to denote the number of frames, and $D1$ and $D2$ to denote the dimension of linguistic feature and acoustic feature respectively. As $\mathbf{L}^t$ and $\mathbf{A}^t$ are extracted from the same utterance, these two feature sequences are initially aligned. Then, the feature mapping $\mathcal{F}^t(\cdot)$ of

target speaker is trained by the bidirectional long short-term memory (BLSTM) using the Back-Propagation Through Time (BPTT) algorithm, denoted as

$$\mathbf{A}^t = \mathcal{F}^t(\mathbf{L}^t) + \mathbf{e}, \quad (4.1)$$

At run-time, the trained BLSTM model is used for voice conversion as shown in Figure 4.5 (b). Given an utterance from the source speaker, we first extract the PPG features $\mathbf{L}^s \in \mathbb{R}^{D_2 \times M}$ using the same ASR system. As the $\mathbf{L}^s$ is already normalized between speakers, the converted MCCs $\mathbf{A}^{conv}$ are generated by

$$\mathbf{A}^{conv} = \mathcal{F}^t(\mathbf{L}^s). \quad (4.2)$$

While the PPG-based voice conversion approach doesn't require parallel data for source-target pairs, it still requires a large amount of speech samples from the target speaker for conversion model training, and that is not practical in many real-world applications. Moreover, for each target speaker, the entire model training must be repeated, which is usually computationally expensive.

4.2.2 Average Modeling Approach (AMA)

Inspired by [123], this section proposes a novel average modeling approach (AMA), where a speaker independent (SI) average modeling technique is introduced as a solution to both the non-parallel and limited training data problems. Here, we propose two average model adaptation approaches: 1) to adapt the average model with a small mount of target data (model-based average modeling approach), 2) to present speaker identity as the input to the average model to generate target speech (feature-based average modeling approach).

Model-based AMA

The objective of Model-based AMA is to adjust the parameters of a SI average model towards a unseen speaker. This is achieved by using a small mount of adaptation data of the target speaker to refine the model. As a result, the adapted model can better predict feature distribution for a given speaker.

The Model-based AMA consists of three steps: (1) average model training, (2) adaptation and (3) conversion. Figure 4.6(a) shows the average model training process. Given the speech data of multiple speakers $\{c_1, \dots, c_K\}$, we first extract PPG

Figure 4.6: Block diagram of the model-based Average Modeling Approach (AMA) to voice conversion .

$\{\mathbf{L}^{c_1}, \dots, \mathbf{L}^{c_K}\}$ and MCC features $\{\mathbf{A}^{c_1}, \dots, \mathbf{A}^{c_K}\}$ respectively. The paired PPG and MCC features from all the speakers, $\{\{\mathbf{L}^{c_1}, \mathbf{A}^{c_1}\}, \dots, \{\mathbf{L}^{c_K}, \mathbf{A}^{c_K}\}\}$, are used to train the SI average model following the same training criterion as that for the single speaker model.

For a specific target speaker, we adjust the trained SI average model by a small amount of target data (e.g. less than 100 utterances), as shown in Figure 4.6(b). The adaptation can be achieved by fine-tuning either a part of the model or the whole network, which will be studied in detail in Section 4.4.1.1. After adaptation, the feature mapping $\mathcal{F}^t(\cdot)$ of target speaker is obtained.

Similar to the process in [123], at run-time, we first extract the linguistic feature $\mathbf{L}^s$, from source speech. Then we convert $\mathbf{L}^s$ to acoustic features $\mathbf{A}^{\text{conv}}$ by Equation (4.2), as shown in Figure 4.6(c).

Feature-based AMA

Figure 4.7: Block diagram of feature-based Average Modeling Approach (AMA) to voice conversion .

As described above, in model-based AMA, an adaptation step is required to adjust the SI average model towards the target speaker. In this sense, the adaptation needs to be repeated for different targets. Alternatively, we propose a more straightforward approach, which presents speaker identity (SpeakerID) feature as part of the input of the neural networks, namely feature-based AMA. SpeakerID is a vector encoding the speaker identity information. We suppose the SpeakerID defined a “speaker space” that we can manipulate elements of this space to control the speaker identity of the generated output speech. In practice, the SpeakerID can be either i-vector or the speaker code represented by a one-hot vector.

During the training of an average model, we introduce SpeakerID as a part of the input by augmenting it with PPG features (see Figure 4.7(a)). With SpeakerID, acoustic features from different speakers can be distinguished. At run-time conversion, target speakerID augmented with PPG features are presented to the average model to generate the voice of the target speaker, as shown in Figure 4.7(b).

4.3 Experimental Setup

In this section, the experimental setup are presented, including database description, feature extraction and AMA model training and adaptation. Several baseline methods are also implemented for comparison. These are described below.

4.3.1 Database and Feature Extraction

Both the Wall Street Journal corpus (WSJ) [125] and the VCC2018 [126] corpus¹ were used in the system implementation.

The WSJ corpus consists of 37,318 utterances, including 18,596 speech samples from 140 female speakers and 18,722 speech samples from 142 male speakers. The VCC2018 corpus consists of 12 speakers, including 8 source speakers and 4 target speakers, with 81 utterances for each speaker.

The ASR system was trained on the WSJ corpus; while the speaker independent (SI) average models were trained on a combination of the speech samples of 4 target speakers in VCC2018 corpus and the WSJ corpus. All the audio files were resampled at 16 kHz.

4.3.1.1 ASR System Training

To train the ASR system, a 40-dimensional filterbank features were extracted using a 25 ms hamming window with 5 ms shift. The input layer was composed of filterbank features and their dynamics with a context window of 11 frames and 1,320 dimensions (120×11). The DNN model contained 5 hidden layers with 1,024 hidden units in each layer and the soft-max output layer had 42 units, representing the 42 phoneme. The frame accuracy of the ASR system was 76.03%. Kaldi toolkit [127] was used to train the DNN-HMM ASR system.

4.3.1.2 Speaker Independent (SI) Average Model Training

To ensure the quality of the average models, gender dependent models were trained in our systems. For the female average model, we chose 17,828 utterances for training and

¹<http://www.vc-challenge.org/>

another 930 non-overlapping utterances for validation. For the male average model, we chose 17,894 utterances for training and 990 non-overlapping utterances for validation.

WORLD vocoder [128] was used to extract the 513-dimensional spectrum, 1-dimensional aperiodicity coefficients and F_0 with 5 ms frame step. Then 40-dimensional MCCs were calculated from the spectrum using the Speech Signal Processing Toolkit (SPTK)². The 42-dimensional PPG features were extracted by the ASR system trained in Section 4.3.1.1.

- **Model-based AMA:** The average model consisted of two BLSTM layers of 1,024 hidden units, one feed forward layer of 1,024 hidden units and a linear output layer. The network input was PPG features (42-dim); the dimension of the output was 127, which was the vuv flag (1-dim) combining with MCC (40-dim), lf0 (1-dim) and bap (1-dim) with their dynamic and accelerate features.
- **Feature-based AMA:** The average model architecture and output were the same as that for model-based AMA implementation. The network input was the combination of PPG features (42-dim) with speakerID. In our system, one-hot vector was used to represent the speakerID. Hence, speakerID was 142 and 144 for female and male average model respectively.

For all average model training, the minibatch size was set to 10, while the momentum and learning rate were set as 0.9 and 0.002 respectively.

4.3.2 Baselines and Setup

Voice conversion experiments were conducted on the VCC2018 database. There were a total of totally 8 source speakers and 4 target speakers.

To fairly compare with the parallel data baseline, 4 source speakers with parallel data were selected, consisting of two female speakers *VCCSF1* and *VCCSF2*, and two male speakers *VCCSM1* and *VCCSM2*. 4 target speakers were selected, consisting of two female speakers *VCCTF1* and *VCCTF2*, and two male speakers *VCCTM1* and *VCCTM2*. The training and evaluation data sets contain 81 utterances and 35 utterances from each

²<https://sourceforge.net/projects/sp-tk/>

speaker respectively. For the feature extraction, we used the same configuration as in Section 4.3.1. The details of baselines and proposed methods were introduced as follows.

- **ML-GMM:** We implemented the joint-density Gaussian mixture model with maximum likelihood parameter conversion. The source and target MCC features were aligned using Dynamic Time Warping (DTW) [52]. 81 parallel utterances from source-target speaker pairs were selected for model training. Both static and their corresponding dynamic features were used in this implementation. The mixtures number of GMM is set to 64. The global variance enhancement [114] was performed to post-filter the converted results for listening test.
- **PPG-LSTM:** The PPG-based BLSTM system with non-parallel data [123]. The network contained two BLSTM layers with 256 hidden units in each layer. Network input and output were the same as for the fine-tune based adaptation described in Section 4.3.1.2. The minibatch size was set to 10, while momentum and learning rate were set as 0.9 and 0.002 respectively. 70 and another 11 non-overlapping utterances from target speaker were selected for training and validation respectively.
- **M-AMA:** refers to the proposed model-based AMA. The average model was described in Section 4.3.1.2. The minibatch size set to 10, the momentum and learning rate of 0.9 and 0.002 respectively. We will examine the effect of adaptation with different settings in Section 4.4.1.1.
- **F-AMA:** refers to the proposed feature-based AMA. The average model architecture was described in Section 4.3.1.2.

During conversion, aperiodicity coefficients were copied from source data, while F_0 was converted by a global linear transformation in log-scale. The Maximum Likelihood Parameter Generation (MLPG) algorithm was employed to refine the spectral parameter trajectory [129], followed by spectral enhancement post-filtering in the cepstral domain. The Merlin [130] was used for both average model and conversion model training.

4.4 Evaluations

In this section, experiments were conducted to compare the performance of the AMA method with a range of state-of-the-art baselines through both objective and subjective evaluations. In addition, the results that were submitted to VCC2018 are also reported.

4.4.1 Objective Evaluation

Mel-Cepstral Distortion (MCD) [16] was employed to measure the distortion, as discussed in Section 2.1.5.1. We only report the averaged MCDs of all the conversion pairs for the experiments. A lower MCD indicates a smaller distortion.

4.4.1.1 Effect of Adaptation Setups on Model-based AMA

Table 4.1: *Mel-Cepstral Distortions (MCDs) as a function of different adaptation settings for model-based AMA.*

Model-based AMA adaptation			MCD (dB) with different number of adaptation utterances		
Output layer	DNN layer	BLSTM layer	20 utts	50 utts	70 utts
Y			6.72	6.57	6.55
Y	Y		6.59	6.46	6.42
Y	Y	Y	6.38	6.28	6.28

Firstly, we examine the effect of different adaptation setups in model-based AMA. Specifically, we performed adaptation on different hidden layers and varied the number of adaptation utterances.

The results are presented in Table 4.1. Across the three adaptation setting, the adaptation over the whole network consistently outperformed the other two approaches with lower MCDs.

We then compared the performance of these adaptation settings over different amounts of adaptation data, e.g. 20, 50 and 70 utterances for training with 3, 8 and 11 non-overlapping utterances for validation respectively. As shown in Table 4.1, MCD decreased as the adaptation utterances increased, while the improvement between 50 and 70 adaptation utterances was marginal. The setting which achieved the lowest distortion, with

MCD of 6.28 dB, was that of using 70 utterances to adapt the whole network. Hence, this adaptation setting was used in the rest of the experiments.

4.4.1.2 Comparative Studies Across Competitive Methods

Then, we further compared the performance of AMA approaches with the baseline methods, ML-GMM-GV and PPG-LSTM. The MCD results are showed in Figure 4.8. We observed that M-AMA performed similarly to the PPG-LSTM approach. The averaged MCDs over all the testing pairs of these two approaches were 6.28 dB and 6.27 dB respectively, while ML-GMM and F-AMA achieved the best (6.06 dB) and worst (6.43 dB) results in terms of MCD.

Figure 4.8: Comparison of Mel-Cepstral Distortions (MCDs) of different conversion methods. 'Source' indicates the MCD evaluated on source utterance without conversion.

According to the objective evaluation, the proposed method did not outperform the baselines. However, the objective results were not always correlated with the subjective evaluation. Such observations were reported in previous works [22, 30, 31]. In the next section, we will examine the subjective evaluation to assess the effectiveness of the proposed AMA.

4.4.2 Subjective Evaluation

AB preference test and XAB test, as discussed in Section 2.1.5.2, were conducted to assess speech quality and speaker similarity respectively. 20 sample pairs were randomly

selected from the 560 paired samples ($16 \times 35 = 560$). 10 subjects participated in each test. Amazon Mechanical Turk (AMT) [117–119], was used in all the listening tests.

4.4.2.1 M-AMA vs. F-AMA

In this section, preference listening tests were conducted to assess both speech quality and speaker identity between model-based AMA and feature-based AMA.

Figure 4.9: Preference t-test result of speech quality with 95% confidence intervals between M-AMA and F-AMA.

Firstly, we conducted an AB preference test to assess the speech quality between M-AMA and F-AMA. Figure 4.9 presents the preference scores with 95% confidence intervals. The results suggested that the F-AMA achieved significantly better performance than M-AMA in terms of speech quality, with the preference score of $(71.5 \pm 8.9)\%$ and $(28.5 \pm 8.9)\%$ respectively.

Hence, we concluded that feature-based AMA is superior to model-based AMA in terms of both speech quality and speaker identity.

Figure 4.10: Preference t-test result of speaker identity with 95% confidence intervals between M-AMA and F-AMA.

Then, we conducted a XAB preference test to assess speaker identity between M-AMA and F-AMA. Again, the results, as shown in Figure 4.10, suggested that the F-AMA achieved significantly better performance than M-AMA in terms of speaker identity, with the preference score of $(55.5 \pm 13.23)\%$ and $(21 \pm 8.13)\%$ respectively.

4.4.2.2 AMAs vs. State-of-the-Arts

In this section, we validated the proposed average modeling approaches with optimal configuration against the baseline methods in terms of both speech quality and speaker identity.

Figure 4.11: Preference t-test results of speech quality with 95% confidence intervals between proposed AMA and baseline methods.

The subjective results of quality preference tests are presented in Figure 4.11. The results, based on preference scores with 95% confidence intervals, suggested that the proposed approaches, F-AMA and M-AMA, significantly outperformed the baseline systems, ML-GMM-GV and PPG-LSTM.

The subjective results of speaker identity are presented in Figure 4.12. It was observed that the proposed average modeling approaches, M-AMA and F-AMA, consistently outperformed PPG-LSTM and ML-GMM-GV in all the tests, although the difference between M-AMA ($(46.5 \pm 9.60)\%$) and PPG-LSTM ($(43.5 \pm 8.91)\%$) is not significant.

The subjective results confirmed the effectiveness of the proposed average modeling approaches, M-AMA and F-AMA, in terms of both speech quality and speaker similarity.

4.4.3 Evaluation Results in VCC2018

23 teams participated in VCC2018 [126], including 18 normal submission teams (represented as N03 to N20) and 5 delay submission teams (represented as D01 to D05). S00

Figure 4.12: Preference t-test results of speaker identity with 95% confidence intervals between proposed AMA and baseline methods.

and T00 refer to the natural speech of the source and target speakers, and B01 refers to the sprocket baseline. We submitted the results of the F-AMA system to the HUB task (parallel data), with the system denoted as N04.

Perceptual evaluations were carried out to evaluate the converted samples for all submitted systems in terms of naturalness and speaker similarity. In total, 267 listeners (146 male and 121 female) participated in the evaluation.

Results for Speech Quality

Figure 4.13 showed the average MOS score plot of all the participating teams for the HUB task. Overall, system N10 performed best with a MOS score of 4.06, which was significantly higher than other systems. Our system commands the 7th place in terms of MOS with a score of 3.21.

The detailed results for the same- and cross-gender conversion are presented in Figure 4.14 and Figure 4.15. Generally, it was observed that most of the systems achieved better performance in same-gender conversion compared with cross-gender conversion,

Figure 4.13: Boxplot for the average MOS score of all speaker pairs. System N04 is the proposed F-AMA.

Figure 4.14: Boxplot for the average MOS score of same-gender conversion. System N04 is the proposed F-AMA.

while N10 performed equally well. Our system achieved the 7th place in both same- and the cross-gender conversion, with MOS scores of 3.33 and 3.09 respectively.

Results for Speaker Similarity

Figure 4.16 shows the MOS and Similarity score plots of all the participating teams for the HUB task³. It was observed that our system commands the 10th place in terms of similarity with a score of 59%. Again, N10 outperformed the other systems with a similarity score of 85%.

³Figure 4.13, Figure 4.14, Figure 4.15 and Figure 4.16 are extracted from the official results release

Figure 4.15: Boxplot for the average MOS score of cross-gender conversion. System N04 is the proposed F-AMA.

Figure 4.16: The average results of similarity score over all speaker pairs. System N04 is the proposed F-AMA.

According to the evaluation results above, N10 significantly outperformed the other systems in terms of both speech quality and speaker similarity, with an average MOS of 4.06 and a similarity score of 85%. Note that a speaker-dependent WaveNet vocoder [131] was used in this system for the waveform reconstruction from the predicted STRAIGHT features, including f_0 and spectrum. WaveNet has showed potential in achieving good performance in speech generation tasks [131, 132]. This may be one of the possible directions for our future work.

4.5 Summary

This chapter addressed the challenging task of voice conversion for non-parallel data condition. The proposed average modeling approach was motivated by the Phonetic PosteriorGram (PPG) based approach [123], where the conversion model was trained by mapping speaker-independent PPG features to speaker dependent acoustic features. The proposed AMA was a departure from previous adaptation-based approaches where the average models were trained on parallel data of multiple speaker pairs. Two adaptation techniques, namely model-based AMA (M-AMA) and feature-based AMA (F-AMA), were implemented to adapt the average model towards target with a small amount of target speech. As linguistic features and acoustic features can be extracted from the same utterance, the proposed approach doesn't require parallel data for either average model training or adaptation. The proposed AMA had three advantages,

- Without the need of parallel training data, we can easily make use of publicly available speech corpora for average model training;
- Using model adaptation technique instead of full fledged training, we reduced the required amount of target speech;
- As the source data is not required during the training and adaptation, AMA can be easily applicable for many to one conversion.

Experiments were conducted on the VCChallenge 2018 database, with comparisons of the AMA approach against both the PPG and the ML-GMM-GV approaches. Subjective results confirmed the effectiveness of the proposed methods. Specifically, the F-AMA outperformed the baselines significantly in terms of both quality and similarity. The effectiveness of AMA was further confirmed in VCC2018 through both objective and subjective experiments.

CHAPTER 5

Synthetic Speech Detection from a Feature Representation Perspective

Chapter 3 and 4 have proposed two novel approaches for both parallel and non-parallel data based voice conversion. In this chapter, we turn our attention to the investigation of synthetic speech detection, which aims to improve the robustness of ASV systems against synthetic spoofing attacks. The approach proposed in this chapter is motivated by the limitations of the current state-of-the-art synthesis techniques, in particular those of low-dimensional feature based approaches where detailed information of high-dimensional features is discarded during feature generation. In this work, we investigate the use of high-dimensional features, especially in the high-frequency region, for synthetic speech detection. In addition, dynamic features are also calculated to assess the effectiveness of using temporal information to detect artifacts across frames.

The chapter is organized as follows. Section 5.1 first reviews the state-of-the-art synthetic speech detection approaches. Our proposed system is then detailed in Section 5.2. Experiments and discussions are then carried out in Section 5.3 and 5.4 to compare different features and their various representations for synthetic speech detection.

5.1 Motivation

Speech, as a low-cost biometric solution, is used to verify speaker identity to safeguard personal information. However, most of the existing Automatic Speaker Verification (ASV) systems are designed to verify speaker identity based on human voices, which may be vulnerable to diverse spoofing attacks [19]. In particular, recent improvement

in TTS and voice conversion techniques offers an effective way to impersonate a target speaker’s voice, which imposes a significant threat to ASV systems [97]. To overcome such a challenge, synthetic speech detection provides a feasible way to improve the robustness of ASV systems and protect the systems.

Currently, a common way for synthetic speech detection is to build some standalone detection systems using different countermeasures which detect synthetic speech based on feature representations. Hence, the most important aspect of these methods is to extract reliable features that represent the discriminative information between synthetic and natural speech consistently.

Generally, given a speech signal, the Short-Time Fourier Transform (STFT) is first applied to obtain the magnitude and phase spectrum. Then, different feature representations are derived from these two parts.

Magnitude-based features: Filter bank based cepstral coefficients, as one of the most popular magnitude spectrum features, are widely used in speech tasks. There are mainly three types of filter bank cepstral features introduced in previous works, e.g. Mel-Frequency Cepstral Coefficients (MFCCs), Inverted Mel-Frequency Cepstral Coefficients (IMFCCs) and Linear Frequency Cepstral Coefficients (LFCCs). The main difference among these features is the selection of the filter bank, which is an array of band-pass filters that split the log power spectrum into frequency sub-bands. In MFCCs [99, 100], the log power spectrum is mapped onto the non-linear Mel frequency scale. To emphasize the high frequency region, an inverted Mel filter bank is used to extract the IMFCCs [133]. Alternatively, the LFCCs [101] are extracted with a filter bank in linear scale. A comparison work between these features for synthetic speech detection is performed in [102].

Phase-based features: Considering that phase information is usually neglected by many synthetic techniques, phase-based features are also used for synthetic speech detection. For example Modified Group Delay (MGD) [103–105] and cosine-normalized phase [103, 105] are two popular phase-based features used in previous works. A relative phase shift feature was proposed [106–108] and demonstrated to achieve promising performance. Recently, a Cochlear Filter Cepstral Coefficients plus Instantaneous

Frequency (CFCCIF) [134] was proposed to combine both magnitude and phase based features to detect synthetic speech. Evaluations carried out on the ASVspoof 2015 database confirmed the effectiveness of the proposed feature.

As synthetic techniques typically exploit low-dimensional features, much detailed information has been ignored in synthetic speech, which results in artifacts. However, the features used in previous works are also low-dimensional features which have been processed by dimension reduction. Unfortunately, such a process may lose the artifacts in synthetic speech and affect the effectiveness of detection systems.

In this work, we investigate synthetic speech detection from a feature representation perspective with the following hypotheses:

- The high-dimensional feature is more effective than the low-dimensional feature, as high-dimensional features contain more detailed information of magnitude and phase;
- The high-frequency region is effective for synthetic speech detection, as this region is rarely modelled by speech synthesis or voice conversion;
- The dynamic temporal information is important for synthetic speech detection, even for high-dimensional features;
- Features derived from different sources are complementary and a fusion of these features can improve detection performance.

To verify these hypotheses, we conducted a series of experiments using six features and the same neural network-based classifier. For each feature we extracted high-dimensional, and their corresponding low-dimensional, features. We also employed the low- and high-frequency regions of the high-dimensional features to assess their effectiveness.

5.2 Synthetic Speech Detection System

In this section, a synthetic speech detection system is introduced, with the aim of utilizing high-dimensional features and their high-frequency region for synthetic speech detection.

Figure 5.1: Architecture of the proposed synthetic speech detection system.

The system contains three part, a) feature extraction, b) classifier and c) fusion. The internal blocks of dedicated module are shown in Figure 5.1. This section first introduces the different features and their various representations, and then describes the neural network classifier and a score level system fusion used in the detection system.

5.2.1 Feature Extraction

Six types of feature are used in this work, including two magnitude-based features, namely Log Magnitude Spectrum (LMS) and Residual Log Magnitude Spectrum (RLMS); and four phase-based features, namely Instantaneous Frequency derivative (IF), Baseband Phase Difference (BPD), Group Delay (GD) and Modified Group Delay (MGD). All the features are evaluated using STFT on the speech signal. The speech is sampled at 16KHz and the STFT analysis window is 25ms with 15ms overlap. The Hamming window and direct current (DC) offset is applied on each analysis frame. Given a speech signal at frame t , the complex spectrum is:

$$X(t, \omega) = |X(t, \omega)|e^{j\theta(t, \omega)}, \quad (5.1)$$

Figure 5.2: Demonstration of the 6 types of features for utterance D16_1000404, which is a natural speech from the development set.

where, $|X(t, \omega)|$ and $\theta(t, \omega)$ are the magnitude and phase spectrum at frame t and frequency ω respectively. The magnitude and phase spectrum are further used to generate different magnitude- and phase-based features respectively. The FFT length is chosen to be 512 and the dimension of all the original features are 256. The features are summarized as follows:

- **LMS**: The log magnitude spectrum feature, as shown in Figure 5.2 (a), which contains formant information, harmonic structure and all the spectral details of speech signals. To reduce the dynamic range of the magnitude spectrum, the

logarithmic is used. In the case of LMS both large and small magnitude frequency components are visible. It can be expressed as $LMS(t, \omega) = \log(|X(t, \omega)|)$.

- **RLMS**: To reduce the impact of formant information and better analyse harmonic structure and spectral details, the residual log magnitude spectrum feature is used in this work. The inverse Linear Predictive Coding (LPC) filter is employed to estimate the glottal waveform from the speech signal; then the LMS is extracted from the residual waveform. Figure 5.2 (b) shows an example of RLMS from natural speech.
- **IF**: Instantaneous frequency [135] is the derivative of the phase along the time axis and is defined as:

$$\begin{aligned} IF(t, \omega) &= \frac{1}{2\pi} \frac{d\theta(t, \omega)}{dt} \\ &\approx \frac{1}{2\pi} (\theta(t, \omega) - \theta(t-1, \omega)). \end{aligned} \quad (5.2)$$

Each element of IF is added as integer multiples of π to make its value within the interval $[-\pi, \pi]$. The IF feature captures the phase temporal information. Unlike the original phase spectrum that hardly shows any patterns, there are clear patterns in the IF spectrum, as shown in Figure 5.2 (c), making it possible to be used as a feature.

- **BPD**: Baseband phase difference [136] is a phase feature extracted from baseband STFT, which can also provide a clear pattern to present phase information. For t -th frame, the BPD is calculated as:

$$BPD(t, \omega) = \text{princ}(IF(t, \omega) - \Omega_k n), \quad (5.3)$$

where $\Omega_k = 2\pi k/N$ is the normalized angular frequencies. Figure 5.2 (d) shows an example of BPD from natural speech.

- **GD**: Group delay [137] is a representation of filter phase response, which is defined as the negative derivative of the Fourier transform phase. It is a frame-based feature, used to capture the phase distortion along frequency axis.

$$GD(t, \omega) = \text{princ}\{\theta(t, \omega) - \theta(t, \omega - 1)\}, \quad (5.4)$$

where $\text{princ}(\cdot)$ represents the principal value operator, mapping the input onto $[-\pi; \pi]$ interval by adding integer numbers of 2π . Figure 5.2 (e) gives an example of BPD from natural speech.

- **MGD**: A variation of GD, which can obtain a clearer phase pattern than GD, as shown in Figure 5.2 (f). The MGD [137] feature of frame t is calculated as:

$$\tau(t, \omega) = \frac{X_R(t, \omega)Y_R(t, \omega) + X_I(t, \omega)Y_I(t, \omega)}{|S(t, \omega)|^{2\gamma}}, \quad (5.5)$$

$$\text{MGD}(t, \omega) = \frac{\tau(t, \omega)}{|\tau(t, \omega)|} |\tau(t, \omega)|^\alpha, \quad (5.6)$$

where $X_R(t, \omega)$ and $X_I(t, \omega)$ denote the real and image part of STFT for $x(n)$; while $Y_R(t, \omega)$ and $Y_I(t, \omega)$ denote the real and image part of STFT for $nx(n)$ respectively. $S(t, \omega)$ is the smoothed spectrum of $|X(t, \omega)|$. Based on experimental results, the values of γ and α are set to 1.2 and 0.4 respectively.

5.2.2 Feature Representation

In order to investigate the effectiveness of high-dimensional features and the relative contribution of the low- and high-frequency regions for synthetic speech detection, different feature representations are used in this study, as summarized below:

- **HD**: The original high-dimensional features which are extracted directly from the Fourier transform without dimension reduction, as described in Section 5.2.1. The dimension of the HD feature is 256.
- **LD**: The low-dimensional representation of the HD feature. The Mel-frequency scaled filter banks are computed over the corresponding features to generate 23 filter bank coefficients.
- **HD-LF**: The low-frequency range of the HD feature, consisting of the first half of 128 frequency bins.
- **HD-HF**: The high-frequency range of the HD feature, consisting of the second half of 128 frequency bins.

The effectiveness of temporal information for synthetic speech detection has been shown in many works [34, 102]. In [1], we use 51 frames and achieved good performance. Such a high-dimensional feature, however, will increase the difficulty of classifier training. Hence, the dynamic features, including delta and acceleration coefficients, of all the representations noted above are extracted in this study to capture the temporal information, namely HD-D, LD-D, HD-LF-D, HD-HF-D. The low-dimensional representations of certain features, especially MFCC for Log Magnitude Spectrum (LMS) and MGDCC for Modified Group Delay (MGD), are widely used in previous studies [99, 103, 104, 112], which are set as baselines in our experiments.

5.2.3 Classifier

As described in Section 5.2.2, high-dimensional features combined with their delta and acceleration coefficients are used for synthetic speech detection. Hence, the input dimension of the detection system is 768, which is relatively high and hardly modeled by conventional classifiers in speech processing, such as the GMM-based classifier.

Due to its capability to model high-dimensional features, a neural network based classifier is adopted for synthetic speech attack detection. Each of the features mentioned above could be an input vector to train its own classifier. The neural network, consisting of one hidden layer with 2,048 sigmoid nodes, is used to perform frame-wise classification for given testing utterance. We treat the detection task as a two classes classification problem for neural network training. For each input frame, two posterior probabilities are generated depending on whether the frame is from natural or synthetic speech respectively. During testing, the posterior probabilities of natural speech will be used as the score for synthetic speech detection. These two posterior probabilities sum to 1. The posterior probabilities of all the frames are estimated and averaged over the test utterance.

5.2.4 Fusion

Considering that different features may be complementary, a score level fusion is applied. Previous studies [1, 102] have shown that specific features, while effective for certain types of artifacts, are not sensitive to other types of artifacts. For example, the main

purpose of the LMS and RLMS features is to capture the artifacts of magnitude, while IF, BPD, GD and MGD are expected to detect the distortion of phase. Hence, system fusion is employed to leverage and harmonize the merits of different features. To avoid over-adjusting scores to fit development data, the scores of all systems are simply averaged to produce the final score.

5.3 Experiments

In this section, experiments are conducted to compare the performance of the different features and their various representations for synthetic speech detection, including both magnitude and phase based features that are widely used in previous works. The systems are tested on the same standard ASVspoof 2015 database consisting of 10 types of spoofing attacks, with training performed using only the first 5 types of spoofing attacks.

5.3.1 Database

Table 5.1: *Summary of the spoofing attacks of ASVspoof 2015 database.*

	Spoofing algorithm	Features	Vocoder	Known?
S1	Frame selection VC	MGC	STRAIGHT	Yes
S2	C1 VC	MGC	STRAIGHT	Yes
S3	HMM TTS	MGC	STRAIGHT	Yes
S4	HMM TTS	MGC	STRAIGHT	Yes
S5	GMM VC	MGC	MLSA	Yes
S6	GMM VC	MGC	STRAIGHT	No
S7	GMM VC	LSP	STRAIGHT	No
S8	Tensor VC	MGC	STRAIGHT	No
S9	KPLS VC	MGC	STRAIGHT	No
S10	Unit selection TTS	Waveform	None	No

The ASVspoof 2015 database [138] was used to assess the performance of different features for synthetic speech detection. This database consists of three subsets, namely the training set, the development set and the evaluation set. There are a total of 10 types of spoofing attacks, namely S1 to S10, as shown in Table 5.3.1. The training and

development sets only contain the spoofing attacks generated by the first five methods (S1-S5), while the evaluation set consists of the spoofing attack generated by both five *known* methods (S1-S5) and five *unknown* methods (S6-S10). Note that, vocoders and low-dimensional features are used in all the first nine methods (S1-S9)¹, while waveform-based unit selection is used to generate S10. More details about the database can be found in [138].

5.3.2 Evaluation Metrics

Equal Error Rate (EER) is used to evaluate system performance. The EER is obtained by selecting an operating point which gives the equal miss rate and false alarm rate. In practice, the EERs of each feature for different spoofing attacks are computed using the Bosaris toolkit². For feature-based systems, we only report the results on the evaluation set. For each speech synthesis/voice conversion method of the development set (S1-S5) and the evaluation set (S1-S10), the averaged EERs of the fused systems are reported.

5.3.3 Results on Development Set

Table 5.2: *Averaged EERs (%) of different features on the development set.*

Feature	LMS	RLMS	IF	BPD	GD	MGD
LD	8.24	6.12	21.69	4.24	22.05	21.05
LD-D	2.96	5.29	17.57	3.08	12.58	23.22
HD	0.42	0.71	0.64	0.79	0.91	0.19
HD-D	0.04	0.39	0.05	0.09	0.07	0.02
HD-LF	7.34	2.48	5.72	6.84	9.41	3.47
HD-LF-D	1.68	1.66	1.46	1.70	1.53	0.72
HD-HF	0.16	1.68	3.02	3.62	4.34	3.16
HD-HF-D	0.03	1.05	1.45	2.57	2.28	0.70

We conducted experiments on the development set to evaluate the performance of different systems, the average scores of all the spoofing types are shown in Table 5.2. We first examined the effectiveness of high-dimensional features in synthetic speech detection. As expected, the high-dimensional (HD) features always outperformed their corresponding low-dimensional (LD) representation. Similar outcomes were observed when dynamic

¹Our system achieved the best performance on S1-S9 in the ASVspoof 2015 challenge.

²<https://sites.google.com/site/bosaristoolkit/>

features were included, in particular HD-D achieved lower error rates than LD-D. This is because low-dimensional features are not as capable of capturing as much detailed information as high-dimensional features. It also implies that the detailed magnitude and phase information of speech signals contain more artificial cues for synthetic speech detection.

We then compared the performance of the low- and the high-frequency regions of the high-dimensional features. The results showed that HD-HF features always outperform the corresponding HD-LF features. This suggests that the high-frequency region of synthetic voice contains more artifacts than the low-frequency counterpart. Low-dimensional features, such as Mel-frequency scaled filter banks, have high resolution at low frequencies and low resolution at high frequencies. They take into consideration human perception sensitivity with respect to frequencies and therefore are best for speech/speaker recognition. However, they are not designed to be sensitive to the artifacts in synthetic speech signals. Some interesting observations were found by using dynamic features: 1) For magnitude-based features, LMS and RLMS, the high-frequency part (HD-HF-D) outperformed its low-frequency counterpart (HD-LF-D). Especially, for the LMS feature, the HD-HF-D feature (EER of 0.03%) obtained an even better detection accuracy compared to the HD-D feature (EER of 0.04%), which is much better than the HD-LF-D feature with an EER of 1.68%. This suggests that the high-frequency region is the informative part of the LMS feature. 2) For phase-based features, IF, BPD, GD and MGD, the performance of the HD-LF-D and HD-HF-D converge when using their dynamic features, and the HD-LF-D performed better in some cases.

In general, dynamic features outperformed their static counterparts in most cases. This indicates the importance of temporal information in synthetic speech detection. We also note that compared to the high-dimensional features such as HD, HD-HF and HD-LF, the inclusion of dynamic information for low-dimensional features resulted in less improvements being achieved. Moreover, compared to high-frequency (HD-HF) features, low-frequency (HD-LF) features obtained more benefits from dynamic features. Overall, the best performance achieved on the development set was obtained by the high-dimensional MGD feature, with its dynamic feature representation (HD-D) achieving an EER of 0.02%.

Figure 5.3: Score distribution of the MGD feature of HD-D representation.

5.3.4 Results on Evaluation Set

The results on the evaluation set are presented in Table 5.3. Here, we separated the results into three parts, the results of known attacks (S1-S5), the results of unknown attacks using vocoders and low-dimensional features (S6-S9) and the results for unknown attacks generated by waveform (S10).

We first examined the effectiveness of high-dimensional features. As expected, the high-dimensional (HD) features always outperformed their corresponding low-dimensional (LD) representation. Similar outcomes were observed when dynamic features were included. In particular, HD-D achieved lower error rates than LD-D. This is because low-dimensional features are not capable of capturing as much detailed information as high dimensional features. This implies that the detailed magnitude and phase information of speech signals contain more artificial cues for synthetic speech detection.

As the spoofing data of the known attacks (S1-S5) is available in training data, we observed that the results on the development set were similar to that of the S1 to S5 on the evaluation set. The best performance on S1-S5 of the evaluation set was also obtained by the MGD feature of HD-D achieving an EER of 0.0%.

With that, we now compare performance across different types of attack. Even though S6-S9 were using similar conversion or synthesis techniques, slightly higher error rates were observed for S6-S9 than for S1-S5. This is because S1-S5 attacks were available for

Table 5.3: Averaged EERs (%) of different features on the evaluation set.

Feature	S1-S5 (Known)						S6-S9 (Unknown)						S10 (Unknown)					
	LMS	RLMS	IF	BPD	GD	MGD	LMS	RLMS	IF	BPD	GD	MGD	LMS	RLMS	IF	BPD	GD	MGD
LD	8.53	5.18	17.79	4.57	19.52	20.70	10.27	7.34	14.28	4.39	19.16	21.57	40.56	47.08	40.58	40.58	32.8	45.88
LD-D	3.08	4.72	15.66	3.32	11.16	23.60	3.88	6.45	16.00	2.96	18.87	25.19	38.14	46.52	42.37	29.59	39.01	48.31
HD	0.06	0.76	0.58	0.72	0.71	0.11	0.06	0.81	0.74	0.92	0.48	0.30	41.42	40.79	39.05	37.61	41.83	40.31
HD-D	0.02	0.34	1.31	0.10	0.05	0.00	0.01	0.36	1.31	0.09	0.03	0.02	35.24	30.8	25.56	30.67	33.9	38.54
HD-LF	5.17	2.10	5.35	5.19	8.49	2.25	3.61	3.55	8.84	8.15	12.56	4.37	42.92	48.76	47.12	47.39	48.61	47.76
HD-LF-D	0.91	1.27	1.22	1.39	1.21	0.29	0.78	2.63	1.59	1.69	2.17	0.45	43.53	48.82	37.77	39.23	46.33	46.02
HD-HF	0.09	1.66	2.84	3.44	4.20	2.24	0.14	2.92	3.03	3.27	4.17	4.00	41.79	41.75	32.63	33.51	38.93	47.59
HD-HF-D	0.03	0.94	1.5	2.59	2.06	0.67	0.05	1.74	1.24	1.90	2.49	4.71	36.68	31.31	27.07	35.51	35.6	43.09

Table 5.4: EERs (%) of fused system on both development and evaluation sets. Pre-system indicates the results of our previous work [1].

	Development set						Evaluation set											
	S1	S2	S3	S4	S5	Average	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	Average	
Feature	S1	S2	S3	S4	S5	Average	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	Average	
LD	0.87	2.78	1.06	0.84	1.20	2.55	0.85	2.22	0.61	0.61	7.74	9.38	2.41	0.41	0.81	41.38	6.64	
LD-D	0.45	1.83	0.84	0.67	3.98	1.56	0.42	2.07	0.51	0.53	4.98	6.87	1.51	0.27	0.34	40.16	5.77	
HD	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.03	0.02	0.00	0.00	0.00	37.42	3.75	
HD-D	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	27.79	2.78	
HD-LF	2.41	1.83	0.03	0.07	1.73	1.21	1.34	1.44	0.04	0.08	1.54	1.77	1.45	0.26	1.05	45.95	5.49	
HD-LF-D	0.25	0.23	0.01	0.02	0.24	0.15	0.10	0.10	0.00	0.01	0.17	0.22	0.18	0.01	0.07	41.90	4.28	
HD-HF	0.04	0.00	0.00	0.01	0.30	0.07	0.03	0.04	0.02	0.03	0.27	0.29	0.01	0.00	0.01	36.20	3.69	
HD-HF-D	0.00	0.00	0.00	0.00	0.12	0.02	0.00	0.00	0.00	0.00	0.13	0.07	0.00	0.00	0.00	28.82	2.90	
Pre-system	0.04	0.00	0.00	0.44	0.14	0.12	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	26.10	2.62

Figure 5.4: Score distribution of the LMS feature of HD-D representation.

training, while S6-S9 were only available during evaluation. However, as the vocoders and low-dimensional features used for generation were similar, the difference in results between them was not as large as expected. Again, the MGD feature with its HD-D representation achieved good performance on this set (EER of 0.02%). This implies the effectiveness of phase information. The LMS feature with HD-D representation performed best with an EER of 0.01%. The score distribution of the MGD and LMS feature with their HD-D representations on the 10 types of synthetic spoofing attacks and natural speech are presented in Figure 5.3 and Figure 5.4, respectively. A score of 0 denotes synthetic speech, while 1 denotes natural speech.

For unknown attacks (S10), performance degraded dramatically. This was due to the fact that all the features used in our systems aimed to detect artifacts of either magnitude or phase. However, S10 attacks were generated by a unit selection algorithm, which used original waveforms directly. Hence, the artifacts only appeared in the transition positions, which increased the difficulty for detection.

5.3.5 Fusion Results

The results of fused systems on both development and evaluation sets were presented in Table 5.4. From the results, we have the following observations. First, compared to systems using a single feature, the EERs of the fused system were reduced significantly on both development and evaluation sets. Particularly, for the fused system of HD-D,

all the spoofing attacks in the development set and most of the spoofing attacks in the evaluation set (S1-S9) were detected with EERs of 0.0%. Second, all the fused systems failed to detect S10 generated by waveform concatenation. We know that in unit selection most of the speech samples are actually natural speech. The artifacts only exist between the units. Third, the results once again confirmed the effectiveness of high-dimensional features, high-frequency information and dynamic features for synthetic speech detection. Finally, we confirmed that all the results were consistent with [1]. Specifically, for the development set and (S1-S9) of the evaluation set, our system outperformed the previous system. The best results were achieved by the fusion system of HD-D with an EER of 2.78%.

5.4 Discussion

Several interesting observations regarding the experimental results are now discussed, including the comparison of system performance between magnitude and phase based features, high- and low-dimensional features, high- and low-frequency features and the difficulty in unit selection-based attacks detection.

5.4.1 Magnitude vs Phase Features

From the perspective of voice conversion and speech synthesis, artifacts are introduced in both magnitude and phase domains. For magnitude, artifacts are mainly due to over-smoothing in both temporal and frequency domains. Over-smoothing in temporal structure lowers the variation of the spectral trajectory for synthetic speech while over-smoothing in the frequency domain can result in missing spectral details. These outcomes have been intensively reported in both voice conversion [139] and speech synthesis [9].

Even though phase has shown to be useful for speech perception [140], it is very hard to find stable patterns for modeling. Hence, in most vocoders, the original/natural phase is discarded and replaced with a minimum phase, leaving obvious artifacts.

The success of using magnitude and phase-based features in synthetic speech detection confirms that artifacts exist in both these two parts.

5.4.2 High-Dimensional vs Low-Dimensional Features

For magnitude, low-dimensional features are commonly used in voice conversion and speech synthesis. However, the limitation of low-dimensional features for capturing detailed spectral information is reported in [46, 141]. It was noticed that the use of low-dimensional features resulted in the loss of spectral details in synthetic speech signals. Hence, there should be more artifact cues in spectral details for synthetic speech detection. This is confirmed by the results shown in the previous section, which indicates that low-dimensional features perform much worse than high-dimensional counterparts.

For phase-based features, such as IF, GD and MGD, the performance of their low-dimensional (LD) representations are much worse than their HD counterparts. Although, after processing, some patterns are observed in the HD phase features, in general, the patterns are still not as clear as those of magnitude-based features. The patterns become blurred after dimension reduction using filter banks, as shown in Figure 5.5.

Figure 5.5: Demonstration of the high- and low-dimensional representations of IF and GD for utterance D16_1000404, which is a natural speech from the development set.

5.4.3 High-Frequency vs Low-Frequency Features

In general, low-dimensional features, used in most speech synthesis and voice conversion approaches, model the low-frequency of speech information more accurately than their high-frequency counterpart. Additionally, due to the smoothing effect of modelling, spectral details can be further lost. Hence, with magnitude-based features, artifacts are easier to find in the high-frequency region. Our results, as shown in Section 5.3.3 and 5.3.4, confirm the effectiveness of high-frequency features. Similar conclusion can also be found in [102].

The situation is a bit different in phase-based features. As phase features are not modeled in speech synthesis or in voice conversion, they do not contain the smoothing effects of modeling and the issues concerning low-dimensional features. However, the phase generated by vocoder is still different from that of natural speech. Unlike the results of magnitude-based features, with phase-based features implemented with dynamic features, low-frequency (HD-LF) outperforms high-frequency (HD-HF). The performance of high-dimensional features (HD) is much better than for either low-frequency or high-frequency. This suggests that both low- and high-frequency of phase-based features contribute to synthetic speech detection.

5.4.4 Difficulty in Detecting Unit Selection-based Attacks

Similar to all the other systems submitted to the ASVspoof 2015 challenge [138], our system didn't perform as expected in S10, which is an unit-selection based attack. The unit-selection based attack is produced by concatenating the time-domain waveform directly without any vocoding and feature extraction techniques, which doesn't carry much artifacts from the perspective of feature representations.

According to a human perception study [142], human listeners can easily detect a unit-selection attack, as human ears can detect the artifacts of discontinuity at the waveform concatenating points. We will investigate techniques to detect discontinuities of synthetic speech in future work.

5.5 Summary

This chapter investigated high-dimensional features and high-frequency information to discriminate between synthetic and natural speech. In speech synthesis, artifacts may have occurred in different stages: 1) Low-dimensional features were typically exploited in synthetic techniques, where the detailed information, especially in the high frequency region, has been abandoned in the synthetic speech; 2) Frame-level feature generation is generally performed in synthetic techniques, where the discontinuity may introduce across frames; 3) Using vocoder for speech generation, original/natural phase was discarded and replaced with a minimum phase. In order to detect the artifacts mentioned above, we proposed to use both magnitude and phase features in our system. For each feature, we extracted their high- and low-dimensional features and corresponding dynamic features.

Experiments were conducted on the ASVspoof 2015 database consisting of a total of 10 types of spoofing attacks (S1-S10). The results showed that, our proposed high-dimensional feature with dynamic information (HD-D) outperformed the other feature representations. Based on the experimental results, we concluded that

- high-dimensional features outperform their corresponding low-dimensional representations;
- high-frequency and low-frequency information are complementary in discriminating synthetic and natural speech;
- dynamic features are useful for synthetic speech detection;
- fusion of different systems improve the detection performance.

Additionally, by applying score level system fusion, all synthetic speech using vocoders and low-dimensional features were successfully detected while poor performance was observed for unit selection-based synthetic speech. Therefore, in future, we will investigate more robust feature to detect such kind of spoofing attacks.

CHAPTER 6

Conclusion and Future Direction

This thesis has focused on the topic of voice conversion and its counterpart, synthetic speech detection. Voice conversion aims to convert the speaker identity of one speaker (source) to another speaker (target) without changing linguistic information. The challenge of voice conversion is to estimate a robust conversion function to transform the source speaker’s characteristics to the target’s with a limited amount of training data. However, many state-of-the-art voice conversion techniques do not perform well with such limited data constraint. To deal with this problem, sparse representation and average model adaptation techniques are adopted in our proposed approaches to improve the system performance under both parallel and non-parallel data conditions. Simultaneously, with the development of speech synthesis technologies, automatic speaker verification (ASV) systems face significant threats from synthetic speech attacks. To protect the ASV systems and improve their robustness against synthetic speech attacks, this thesis has also developed novel countermeasures for synthetic speech detection. To conclude the work, Section 6.1 first summarizes the contributions proposed in this work. Section 6.2 discusses some of the future directions that can be explored.

6.1 Contributions

This thesis consists of three contributions to improve the system performance of voice conversion and synthetic speech detection. For voice conversion, two novel approaches have been developed for both parallel and non-parallel data. In particular, an exemplar-based frequency warping framework is proposed to address the over-smoothing problem

under limited parallel data condition. And an average model adaptation approach is proposed for non-parallel data voice conversion, where publicly available speech corpora can easily be used for model training and relax the limited data constraint. For synthetic speech detection, we make use of high-dimensional features, especially high-frequency regions, to detect the artifacts of synthetic speech, which can reduce the error rate of state-of-the-art countermeasures. The contributions presented in this thesis are summarized as below.

6.1.1 An Exemplar-based Frequency Warping (EFW) Approach for Parallel Data Voice Conversion

The first contribution proposed a novel framework based on exemplar-based frequency warping and residual compensation for parallel data voice conversion, with the work published at IEEE/ACM Transactions on Audio, Speech and Language Processing journal [31], and ISCSLP 2014 [29] and ICASSP 2015 conferences [30]. EFW combined frequency warping and exemplar-based methods, where both the warping functions and residual vectors were represented as a weighted linear combination of warping function and residual vector exemplars respectively. With sparsity constraint, the over-smoothing problem of warping function and residual vector in GMM-based framework was reduced. Additionally, benefiting from frequency warping, high-dimensional features were directly used for conversion. As a result, the EFW preserved the desired spectral details and generated high quality speech.

Experiments were carried out using 12 speaker pairs from the VOICES database, with a comparison of EFW to the state-of-the-art baseline methods from both statistical parametric and frequency warping techniques. The objective results showed that EFW achieved the lowest LSD ratio of 58.70%, which outperformed all the baselines. The subjective results based on the preference tests suggested that EFW achieved significantly higher speech quality than state-of-the-art methods while maintaining similar speaker identity, which also confirmed the effectiveness of the proposed method.

Detailed experiments were also carried out to evaluate the different aspects that contributed to EFW. These included performing warping function estimation with and

without the proposed warping function dictionary smoothing and varying the type of frequency warping methods used. The objective results showed that by using the smoothed warping function dictionary, the LSD ratio of EFW decreased from 61% to 58.70%. The subjective results based on the preference tests also suggested that the warping function dictionary smoothing step significantly improved converted speech quality. Additionally, both the objective and subjective evaluation results confirmed that the EFW framework outperformed the GMM-based frequency warping framework for all three frequency warping methods in terms of converted speech quality.

6.1.2 Average Modeling Approach (AMA) for Non-Parallel Data Voice Conversion

The second contribution proposed an average modeling approach (AMA) for non-parallel data voice conversion, with the work published in the Odyssey 2018 conference [32]. This idea followed the strategy used in PPG-based voice conversion, where the conversion model was trained between linguistic to acoustic features of the same speaker. The proposed AMA extended the PPG approach from a single speaker model to an average model, where non-parallel data from multiple speakers were used for model training. For a specific target, we can perform the adaptation either at model level (M-AMA) or at feature level (F-AMA). Therefore, parallel data was not required for both average model training and adaptation. The advantage of AMA over the PPG approach was that publicly available speech corpora could easily be used for average model training, and the required amount of target speech for adaptation was reduced.

Evaluations were carried out using 16 speaker pairs from the VCChallenge 2018 database, with comparisons of the AMA approach against both the PPG and the ML-GMM-GV approaches. Subjective results based on the preference tests showed that M-AMA achieved significantly higher speech quality than the baselines in maintaining similar speaker identity, while F-AMA outperformed the baselines significantly in terms of both quality and similarity. The results of F-AMA were also submitted to VCChallenge 2018, where perceptual evaluations were carried out to rate the converted speech from 23 participants in terms of naturalness and similarity. Our system achieved the 7th place in terms of naturalness with a MOS score of 3.21 and the 10th place in terms of

speaker similarity with a score of 59%. These results further confirmed the effectiveness of proposed method.

6.1.3 Synthetic Speech Detection using High-Dimensional Features

The final contribution of synthetic speech detection focused on discriminating synthetic speech from natural speech to protect speaker verification systems, with the work published at the INTERSPEECH 2015 [1] and ICASSP 2016 conferences [33]. Traditional automatic speaker verification (ASV) systems were targeted at verifying the speaker identity of different human speech samples. System performance degraded dramatically when faced with diverse spoofing attacks, e.g. synthetic speech. In this work, we investigated different feature representations for synthetic speech detection. Due to the low-dimensional features were typically used for the synthetic speech generation, the detailed information has been abandoned in the synthetic speech. Hence, this work mainly focused on the detailed information of high-dimensional feature representations, especially their high-frequency parts, for synthetic speech detection. In addition, dynamic features were also utilized to detect artifacts across frames.

Experiments were carried out on the standard ASVspoof 2015 corpus consisting of 10 types of synthetic speech. After system fusion, our proposed high-dimensional feature with dynamic information (HD-D) achieved average EERs of 0.0% on the vocoder based synthetic speech (S1-S9) and 27.79% on the unit-selection based synthetic speech (S10). The averaged EER over all the 10 type of synthetic spoofing attacks is 2.89%, which outperformed other feature representations. This system had also been submitted for the ASVspoof 2015 challenge and obtained the EERs of 0.003% and 5.231% on the known and unknown attacks task respectively. The averaged result showed that our system commands the 4th place with a EER of 2.62% among 16 participant teams.

Detailed analysis was also carried out based on the experimental results, which indicated that 1) the artifacts of synthetic speech existed in both magnitude and phase-based features; 2) for both magnitude and phase-based features, the results of the high-dimensional representations consistently outperformed their low-dimensional counterparts; 3) for magnitude-based features, artifacts were easier to detect in the high-frequency region; while for phase-based features, both the low- and the high-frequency

regions contributed to synthetic speech detection; 4) dynamic features were effective for synthetic speech detection; 5) compared to the vocoder based synthetic speech (S1-S9), unit-selection based synthetic speech (S10) is more difficult to be detected.

6.2 Future Works

The objectives of this thesis are: 1) to develop algorithms to improve voice conversion performance, and 2) to study countermeasures for synthetic speech detection to protect ASV systems. As a result, two novel algorithms, EFW and AMA, are developed to improve system performance for parallel and non-parallel data based voice conversion respectively. In addition, we also propose the use of high-dimensional features for synthetic speech detection, which has been shown to successfully detect vocoder based synthetic speech. For future research, some possible directions are listed and discussed below.

6.2.1 Voice conversion

WaveNet: In this thesis, both EFW and AMA voice conversion systems are built based on traditional vocoders, e.g. STRAIGHT and WORLD. In this sense, the quality of converted speech is limited by the vocoder. Recently, WaveNet, a network that directly generates speech waveforms without vocoder, has shown potential in achieving good performance in speech generation tasks [132]. Following this idea, a WaveNet vocoder [131] is proposed, where the acoustic features of existing vocoders are fed as auxiliary features of WaveNet to generate high quality speech signals.

For future work, the use of WaveNet for voice conversion should be explored. For example, WaveNet can serve as a vocoder of EFW and AMA to post-process converted features and generate high quality speech. In addition, it is also possible to develop an end-to-end voice conversion system, where WaveNet, as a part of AMA, is jointly optimized with the conversion model.

Prosody conversion: As discussed in Section 1.1, prosody, e.g. pitch contour and duration, plays an important role in the perception of speaker identity. However, most state-of-the-art voice conversion systems simply perform a linear transformation for pitch conversion, while duration is copied from the source speech directly.

Hence, prosody conversion is one of the possible directions to further improve the performance of voice conversion systems. For example, a sequence-to-sequence model and attention mechanism [143, 144], which learns a model to encode a variable-length sequence into a fixed-length vector representation and to decode that fixed-length vector representation back into a variable length sequence, can be applied for duration conversion.

6.2.2 Synthetic speech detection

Noise conditions: At present, synthetic speech detection databases are usually established under clean condition. However, noisy conditions, which are inevitable in real world scenarios, were seldom considered in previous studies. Hence, a concern arises whether currently developed spoofing detection algorithms/systems are still effective under noisy conditions.

Preliminary studies under additive and reverberant noise conditions were conducted in [145], where a database in both additive noise and reverberant environments is generated based on the ASVspoof 2015 challenge database. The results of using the classifier trained from clean data showed that 1) system performance degraded in all the additive noise scenarios and further deteriorated as SNR decreased; 2) compared to reverberation noise, additive noises affected the detection system more seriously; 3) system performance varied significantly under different noise scenarios and phase-based features were more robust than magnitude-based features.

To improve the performance of current detection systems under real-world conditions, one direction is to study multi-condition training for noisy environment adaptation in robust synthetic speech detection [146]. An alternative direction is to investigate noise robust feature representations.

References

- [1] 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 *Proc. INTERSPEECH*, 2015.
- [2] L. R. Rabiner, “A tutorial on hidden markov models and selected applications in speech recognition,” *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.
- [3] J. P. Campbell Jr, “Speaker recognition: A tutorial,” *Proceedings of the IEEE*, vol. 85, no. 9, pp. 1437–1462, 1997.
- [4] C. D. Manning, C. D. Manning, and H. Schütze, *Foundations of statistical natural language processing*. MIT press, 1999.
- [5] A. J. Hunt and A. W. Black, “Unit selection in a concatenative speech synthesis system using a large speech database,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 373–376, IEEE, 1996.
- [6] A. W. Black and P. A. Taylor, “Automatically clustering similar units for unit selection in speech synthesis,” 1997.
- [7] R. A. Clark, K. Richmond, and S. King, “Multisyn: Open-domain unit selection for the festival speech synthesis system,” *Speech Communication*, pp. 317–330, 2007.
- [8] A. W. Black, H. Zen, and K. Tokuda, “Statistical parametric speech synthesis,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2007.
- [9] H. Zen, K. Tokuda, and A. W. Black, “Statistical parametric speech synthesis,” *Speech Communication*, pp. 1039–1064, 2009.

- [10] H. Ze, A. Senior, and M. Schuster, “Statistical parametric speech synthesis using deep neural networks,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7962–7966, IEEE, 2013.
- [11] M. Tamura, T. Masuko, K. Tokuda, and T. Kobayashi, “Speaker adaptation for HMM-based speech synthesis system using MLLR,” in *the third ESCA/COCOSDA Workshop (ETRW) on Speech Synthesis*, 1998.
- [12] J. Yamagishi, T. Kobayashi, N. Yuji, K. Ogata, and J. Isogai, “Analysis of speaker adaptation algorithms for HMM-based speech synthesis and a constrained SMAPLR adaptation algorithm,” 2009.
- [13] J. Yamagishi and T. Kobayashi, “Average-voice-based speech synthesis using HSMM-based speaker adaptation and adaptive training,” *IEICE TRANSACTIONS on Information and Systems*, vol. 90, no. 2, pp. 533–543, 2007.
- [14] Y. Stylianou, O. Cappé, and E. Moulines, “Continuous probabilistic transform for voice conversion,” *IEEE Transactions on Speech and Audio Processing*, pp. 131–142, 1998.
- [15] A. Kain and M. W. Macon, “Spectral voice conversion for text-to-speech synthesis,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1998.
- [16] T. Toda, A. W. Black, and K. Tokuda, “Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory,” *IEEE Transactions on Audio, Speech, and Language Processing*, pp. 2222–2235, 2007.
- [17] T. Toda, M. Nakagiri, and K. Shikano, “Statistical voice conversion techniques for body-conducted unvoiced speech enhancement,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 20, no. 9, pp. 2505–2517, 2012.
- [18] A. B. Kain, J.-P. Hosom, X. Niu, J. P. van Santen, M. Fried-Oken, and J. Staehely, “Improving the intelligibility of dysarthric speech,” *Speech communication*, vol. 49, no. 9, pp. 743–759, 2007.

REFERENCES

- [19] Z. Wu, N. Evans, T. Kinnunen, J. Yamagishi, F. Alegre, and H. Li, “Spoofing and countermeasures for speaker verification: a survey,” *Speech Communication*, vol. 66, pp. 130–153, 2015.
- [20] D. G. Childers and C. Lee, “Vocal quality factors: Analysis, synthesis, and perception,” *the Journal of the Acoustical Society of America*, vol. 90, no. 5, pp. 2394–2410, 1991.
- [21] H. Kuwabara and Y. Sagisak, “Acoustic characteristics of speaker individuality: Control and conversion,” *Speech communication*, vol. 16, no. 2, pp. 165–173, 1995.
- [22] D. Erro, A. Moreno, and A. Bonafonte, “Voice conversion based on weighted frequency warping,” *IEEE Transactions on Audio, Speech, and Language Processing*, pp. 922–931, 2010.
- [23] K. Kobayashi, T. Toda, G. Neubig, S. Sakti, and S. Nakamura, “Statistical singing voice conversion with direct waveform modification based on the spectrum differential,” *INTERSPEECH*, 2014.
- [24] E. Godoy, O. Rosec, and T. Chonavel, “Voice conversion using dynamic frequency warping with amplitude scaling, for parallel or nonparallel corpora,” *IEEE Transactions on Audio, Speech, and Language Processing*, pp. 1313–1323, 2012.
- [25] D. Erro, A. Moreno, and A. Bonafonte, “INCA algorithm for training voice conversion systems from nonparallel corpora,” *IEEE Transactions on Audio, Speech, and Language Processing*, pp. 944–953, 2010.
- [26] C.-H. Lee and C.-H. Wu, “MAP-based adaptation for speech conversion using adaptation data selection and non-parallel training,” in *INTERSPEECH*, 2006.
- [27] T. Toda, Y. Ohtani, and K. Shikano, “Eigenvoice conversion based on gaussian mixture model,” in *INTERSPEECH*, 2006.
- [28] J. Wu, Z. Wu, and L. Xie, “On the use of I-vectors and average voice model for voice conversion without parallel data,” in *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, 2016.

- [29] X. Tian, Z. Wu, S. W. Lee, and E. S. Chng, “Correlation-based frequency warping for voice conversion,” in *9th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, 2014.
- [30] X. Tian, Z. Wu, S. W. Lee, N. Q. Hy, E. S. Chng, and M. Dong, “Sparse representation for frequency warping based voice conversion,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2015.
- [31] X. Tian, S. W. Lee, Z. Wu, E. S. Chng, and H. Li, “An exemplar-based approach to frequency warping for voice conversion,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 10, pp. 1863–1876, 2017.
- [32] X. Tian, J. Wang, H. Xu, E. S. Chng, and H. Li, “Average modeling approach to voice conversion with non-parallel data,” 2018.
- [33] X. Tian, Z. Wu, X. Xiao, E. S. Chng, and H. Li, “Spoofing detection from a feature representation perspective,” in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2016.
- [34] X. Tian, S. Du, X. Xiao, H. Xu, E. S. Chng, and H. Li, “Detecting synthetic speech using long term magnitude and phase information,” in *Proc. IEEE China Summit and International Conference on Signal and Information Processing (ChinaSIP)*, pp. 611–615, IEEE, 2015.
- [35] E. Moulines and F. Charpentier, “Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones,” *Speech communication*, vol. 9, no. 5-6, pp. 453–467, 1990.
- [36] J. Laroche, Y. Stylianou, and E. Moulines, “HNS: Speech modification based on a harmonic+ noise model,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 550–553, 1993.
- [37] H. Kawahara, I. Masuda-Katsuse, and A. de Cheveigné, “Restructuring speech representations using a pitch-adaptive time–frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds,” *Speech communication*, pp. 187–207, 1999.

- [38] H. Valbret, E. Moulines, and J.-P. Tubach, “Voice transformation using PSOLA technique,” *Speech Communication*, pp. 175–187, 1992.
- [39] D. Erro, I. Sainz, E. Navas, and I. Hernáez, “Hnm-based MFCC + F0 extractor applied to statistical speech synthesis,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4728–4731, IEEE, 2011.
- [40] D. Erro, I. Sainz, E. Navas, and I. Hernáez, “Improved hnm-based vocoder for statistical synthesizers,” in *Twelfth Annual Conference of the International Speech Communication Association*, 2011.
- [41] Y. Stylianou and O. Cappe, “A system for voice conversion based on probabilistic classification and a harmonic plus noise model,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 1, pp. 281–284, 1998.
- [42] D. Erro, E. Navas, and I. Hernaez, “Parametric voice conversion based on bilinear frequency warping plus amplitude scaling,” *IEEE Transactions on Audio, Speech, and Language Processing*, pp. 556–566, 2013.
- [43] T. Toda, H. Saruwatari, and K. Shikano, “Voice conversion algorithm based on Gaussian mixture model with dynamic frequency warping of STRAIGHT spectrum,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2001.
- [44] E. Helander, T. Virtanen, J. Nurminen, and M. Gabbouj, “Voice conversion using partial least squares regression,” *IEEE Transactions on Audio, Speech, and Language Processing*, pp. 912–921, 2010.
- [45] E. Helander, H. Silén, T. Virtanen, and M. Gabbouj, “Voice conversion using dynamic kernel partial least squares regression,” *IEEE Transactions on Audio, Speech, and Language Processing*, pp. 806–817, 2012.
- [46] Z. Wu, T. Virtanen, E. S. Chng, and H. Li, “Exemplar-based sparse representation with residual compensation for voice conversion,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, pp. 1506–1521, 2014.

REFERENCES

- [47] D. Sundermann and H. Ney, “VTLN-based voice conversion,” in *IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, 2003.
- [48] F. Itakura, “Line spectrum representation of linear predictor coefficients of speech signals,” *The Journal of the Acoustical Society of America*, vol. 57, no. S1, pp. S35–S35, 1975.
- [49] F. Soong and B. Juang, “Line spectrum pair (LSP) and speech data compression,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 9, pp. 37–40, 1984.
- [50] O. Turk and L. M. Arslan, “Robust processing techniques for voice conversion,” *Computer Speech & Language*, vol. 20, no. 4, pp. 441–467, 2006.
- [51] E. Helander, J. Nurminen, and M. Gabbouj, “LSF mapping for voice conversion with very small training sets,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4669–4672, 2008.
- [52] H. Sakoe and S. Chiba, “Dynamic programming algorithm optimization for spoken word recognition,” *IEEE Transactions on Acoustics, Speech and Signal Processing*, pp. 43–49, 1978.
- [53] H. Ney, D. Suendermann, A. Bonafonte, and H. Höge, “A first step towards text-independent voice conversion,” in *Eighth International Conference on Spoken Language Processing*, 2004.
- [54] H. Ye and S. Young, “Voice conversion for unknown speakers,” in *Eighth International Conference on Spoken Language Processing*, 2004.
- [55] M. Abe, S. Nakamura, K. Shikano, and H. Kuwabara, “Voice conversion through vector quantization,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 655–658, 1988.
- [56] K. Shikano, S. Nakamura, and M. Abe, “Speaker adaptation and voice conversion by codebook mapping,” in *IEEE International Symposium on Circuits and Systems*, pp. 594–597, 1991.

- [57] S. Desai, E. V. Raghavendra, B. Yegnanarayana, A. W. Black, and K. Prahallad, “Voice conversion using artificial neural networks,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2009.
- [58] L.-H. Chen, Z.-H. Ling, L.-J. Liu, and L.-R. Dai, “Voice conversion using deep neural networks with layer-wise generative training,” *IEEE/ACM Transactions on Speech and Audio Processing*, pp. 1859–1872, 2014.
- [59] F.-L. Xie, Y. Qian, Y. Fan, F. K. Soong, and H. Li, “Sequence error (SE) minimization training of neural network for voice conversion,” in *INTERSPEECH*, 2014.
- [60] R. Takashima, T. Takiguchi, and Y. Ariki, “Exemplar-based voice conversion in noisy environment,” in *Spoken Language Technology workshop (SLT)*, 2012.
- [61] R. Aihara, T. Fujii, T. Nakashika, T. Takiguchi, and Y. Ariki, “Small-parallel exemplar-based voice conversion in noisy environments using affine non-negative matrix factorization,” *EURASIP J. Audio, Speech and Music Processing*, p. 32, 2015.
- [62] Z. Jin, A. Finkelstein, S. DiVerdi, J. Lu, and G. J. Mysore, “Cute: A concatenative method for voice conversion using exemplar-based unit selection,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016.
- [63] H. Benisty and D. Malah, “Voice conversion using GMM with enhanced global variance,” in *INTERSPEECH*, 2011.
- [64] M. S. Ribeiro and R. A. Clark, “A multi-level representation of f0 using the continuous wavelet transform and the discrete cosine transform,” in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pp. 4909–4913, IEEE, 2015.
- [65] G. Sanchez, H. Silen, J. Nurminen, and M. Gabbouj, “Hierarchical modeling of f0 contours for voice conversion,” in *INTERSPEECH*, 2014.

- [66] F.-L. Xie, Y. Qian, F. K. Soong, and H. Li, “Pitch transformation in neural network based voice conversion,” in *2014 9th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, pp. 197–200, IEEE, 2014.
- [67] H. Ming, D. Huang, M. Dong, H. Li, L. Xie, and S. Zhang, “Fundamental frequency modeling using wavelets for emotional voice conversion,” in *International Conference on Affective Computing and Intelligent Interaction (ACII)*, pp. 804–809, 2015.
- [68] H. Ye and S. Young, “High quality voice morphing,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2004.
- [69] L. M. Arslan, “Speaker transformation algorithm using segmental codebooks (STASC),” *Speech Communication*, vol. 28, no. 3, pp. 211–226, 1999.
- [70] Z. Wu, T. Virtanen, T. Kinnunen, E. S. Chng, and H. Li, “Exemplar-based voice conversion using non-negative spectrogram deconvolution,” in *8th ISCA Speech Synthesis Workshop*, 2013.
- [71] L. Mesbahi, V. Barreaud, and O. Boeffard, “Comparing gmm-based speech transformation systems,” in *Eighth Annual Conference of the International Speech Communication Association*, 2007.
- [72] S. Takamichi, T. Toda, A. W. Black, and S. Nakamura, “Modulation spectrum-based post-filter for GMM-based voice conversion,” in *Asia-Pacific Signal and Information Processing Association, 2014 Annual Summit and Conference (AP-SIPA)*, 2014.
- [73] J. Kominek and A. W. Black, “The CMU arctic speech databases,” in *Fifth ISCA Workshop on Speech Synthesis*, 2004.
- [74] A. B. Kain, *High resolution voice transformation*. PhD thesis, OGI School of Science and Eng., Portland, Oregon, USA, 2001.
- [75] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, *et al.*, “Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups,” *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

- [76] G. E. Dahl, D. Yu, L. Deng, and A. Acero, “Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition,” *IEEE Transactions on audio, speech, and language processing*, vol. 20, no. 1, pp. 30–42, 2012.
- [77] M. L. Seltzer, D. Yu, and Y. Wang, “An investigation of deep neural networks for noise robust speech recognition,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7398–7402, IEEE, 2013.
- [78] Y. Qian, Y. Fan, W. Hu, and F. K. Soong, “On the training aspects of deep neural network (dnn) for parametric tts synthesis,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3829–3833, IEEE, 2014.
- [79] Z. Wu, P. Swietojanski, C. Veaux, S. Renals, and S. King, “A study of speaker adaptation for dnn-based speech synthesis,” in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.
- [80] K. Hornik, M. Stinchcombe, and H. White, “Multilayer feedforward networks are universal approximators,” *Neural networks*, vol. 2, no. 5, pp. 359–366, 1989.
- [81] M. Narendranath, H. A. Murthy, S. Rajendran, and B. Yegnanarayana, “Transformation of formants for voice conversion using artificial neural networks,” *Speech communication*, vol. 16, no. 2, pp. 207–216, 1995.
- [82] T. Watanabe, T. Murakami, M. Namba, T. Hoya, and Y. Ishida, “Transformation of spectral envelope for voice conversion based on radial basis function networks,” in *Seventh International Conference on Spoken Language Processing*, 2002.
- [83] Z. Wu, E. S. Chng, and H. Li, “Conditional restricted boltzmann machine for voice conversion,” in *China Summit and International Conference on Signal and Information Processing (ChinaSIP)*, 2013.
- [84] T. Nakashika, R. Takashima, T. Takiguchi, and Y. Ariki, “Voice conversion in high-order eigen space using deep belief nets,” in *Interspeech*, pp. 369–372, 2013.

- [85] L. Sun, S. Kang, K. Li, and H. Meng, “Voice conversion using deep bidirectional long short-term memory based recurrent neural networks,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4869–4873, 2015.
- [86] T. Nakashika, T. Takiguchi, and Y. Ariki, “Voice conversion using rnn pre-trained by recurrent temporal restricted boltzmann machines,” *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 23, no. 3, pp. 580–587, 2015.
- [87] D. Sundermann, H. Ney, and H. Hoge, “VTLN-based cross-language voice conversion,” in *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2003.
- [88] M. Eichner, M. Wolff, and R. Hoffmann, “Voice characteristics conversion for TTS using reverse VTLN,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2004.
- [89] A. Přibilová and J. Přibil, “Non-linear frequency scale mapping for voice conversion in text-to-speech system with cepstral description,” *Speech Communication*, pp. 1691–1703, 2006.
- [90] R. Vích and M. Vondra, “Pitch synchronous transform warping in voice conversion,” in *Cognitive Behavioural Systems*, 2012.
- [91] M. Vondra and R. Vích, “Speech identity conversion,” in *Nonlinear Speech Modeling and Applications*, 2005.
- [92] Z. Shuang, R. Bakis, S. Shechtman, D. Chazan, and Y. Qin, “Frequency warping based on mapping formant parameters,” in *ICSLP*, 2006.
- [93] T. Virtanen, “Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria,” *IEEE Transactions on Audio, Speech, and Language Processing*, pp. 1066–1074, 2007.

- [94] B. King, C. Févotte, and P. Smaragdis, “Optimal cost function and magnitude power for NMF-based speech separation and music interpolation,” in *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, 2012.
- [95] D. Kitamura, H. Saruwatari, H. Kameoka, Y. Takahashi, K. Kondo, and S. Nakamura, “Multichannel signal separation combining directional clustering and non-negative matrix factorization with spectrogram restoration,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, pp. 654–669, 2015.
- [96] D. D. Lee and H. S. Seung, “Algorithms for non-negative matrix factorization,” in *Advances in neural information processing systems*, 2001.
- [97] Z. Wu, A. Khodabakhsh, C. Demiroglu, J. Yamagishi, D. Saito, T. Toda, and S. King, “SAS: A speaker verification spoofing database containing diverse attacks,” in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2015.
- [98] E. Khoury, T. Kinnunen, A. Sizov, Z. Wu, and S. Marcel, “Introducing i-vectors for joint anti-spoofing and speaker verification,” in *Proc. INTERSPEECH*, 2014.
- [99] 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 *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4401–4404, IEEE, 2012.
- [100] T. Satoh, T. Masuko, T. Kobayashi, and K. Tokuda, “A robust speaker verification system against imposture using an HMM-based speech synthesis system,” in *Proc. EUROSPEECH*, pp. 759–762, 2001.
- [101] J.-F. Bonastre, D. Matrouf, and C. Fredouille, “Artificial impostor voice transformation effects on false acceptance rates,” in *Proc. INTERSPEECH*, pp. 2053–2056, 2007.
- [102] M. Sahidullah, T. Kinnunen, and C. Hanilçi, “A comparison of features for synthetic speech detection,” in *Proc. INTERSPEECH*, 2015.

- [103] Z. Wu, C. E. Siong, and H. Li, “Detecting converted speech and natural speech for anti-spoofing attack in speaker recognition,” in *Proc. INTERSPEECH*, 2012.
- [104] Z. Wu, T. Kinnunen, E. S. Chng, H. Li, and E. Ambikairajah, “A study on spoofing attack in state-of-the-art speaker verification: the telephone speech case,” in *Proc. Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pp. 1–5, IEEE, 2012.
- [105] Y. Liu, Y. Tian, L. He, J. Liu, and M. T. Johnson, “Simultaneous utilization of spectral magnitude and phase information to extract supervectors for speaker verification anti-spoofing,” in *Proc. INTERSPEECH*, 2015.
- [106] P. L. De Leon, I. Hernaez, I. Saratxaga, M. Pucher, and J. Yamagishi, “Detection of synthetic speech for the problem of imposture,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4844–4847, IEEE, 2011.
- [107] J. Sanchez, I. Saratxaga, I. Hernaez, E. Navas, and D. Erro, “The AHOLAB RPS SSD spoofing challenge 2015 submission,” in *Proc. INTERSPEECH*, 2015.
- [108] L. Wang, Y. Yoshida, Y. Kawakami, and S. Nakagawa, “Relative phase information for detecting human speech and spoofed speech,” in *Proc. INTERSPEECH*, 2015.
- [109] J. Sanchez, I. Saratxaga, I. Hernaez, E. Navas, D. Erro, and T. Raitio, “Toward a universal synthetic speech spoofing detection using phase information,” *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 810–820, 2015.
- [110] P. L. De Leon, M. Pucher, and J. Yamagishi, “Evaluation of the vulnerability of speaker verification to synthetic speech,” 2010.
- [111] P. L. De Leon, M. Pucher, J. Yamagishi, I. Hernaez, and I. Saratxaga, “Evaluation of speaker verification security and detection of HMM-based synthetic speech,” *IEEE Transactions on Audio, Speech, and Language Processing*, 2012.

- [112] Z. Wu, X. Xiao, E. S. Chng, and H. Li, “Synthetic speech detection using temporal modulation feature,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7234–7238, IEEE, 2013.
- [113] J. F. Gemmeke, T. Virtanen, and A. Hurmalainen, “Exemplar-based sparse representations for noise robust automatic speech recognition,” *IEEE Transactions on Speech and Audio Processing*, pp. 2067–2080, 2011.
- [114] T. Toda, T. Muramatsu, and H. Banno, “Implementation of computationally efficient real-time voice conversion,” in *INTERSPEECH*, 2012.
- [115] D. T. Chappell and J. H. Hansen, “Speaker-specific pitch contour modeling and modification,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1998.
- [116] B. King, C. Févotte, and P. Smaragdis, “Optimal cost function and magnitude power for NMF-based speech separation and music interpolation,” in *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, IEEE, 2012.
- [117] M. K. Wolters, K. B. Isaac, and S. Renals, “Evaluating speech synthesis intelligibility using amazon mechanical turk,” in *7th ISCA Speech Synthesis Workshop (SSW7)*, 2010.
- [118] F. Ribeiro, D. Florêncio, C. Zhang, and M. Seltzer, “Crowdmos: An approach for crowdsourcing mean opinion score studies,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2011.
- [119] S. Buchholz and J. Latorre, “Crowdsourcing preference tests, and how to detect cheating,” in *INTERSPEECH*, 2011.
- [120] H. Silén, J. Nurminen, E. Helander, and M. Gabbouj, “Voice conversion for non-parallel datasets using dynamic kernel partial least squares regression,” *Convergence*, p. 2, 2013.

- [121] T. Nakashika, T. Takiguchi, and Y. Minami, “Non-parallel training in voice conversion using an adaptive restricted boltzmann machine,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, pp. 2032–2045, 2016.
- [122] C.-C. Hsu, H.-T. Hwang, Y.-C. Wu, Y. Tsao, and H.-M. Wang, “Voice conversion from unaligned corpora using variational autoencoding wasserstein generative adversarial networks,” *Interspeech*, 2017.
- [123] L. Sun, K. Li, H. Wang, S. Kang, and H. Meng, “Phonetic posteriorgrams for many-to-one voice conversion without parallel data training,” in *IEEE International Conference on Multimedia and Expo (ICME)*, IEEE, 2016.
- [124] T. J. Hazen, W. Shen, and C. White, “Query-by-example spoken term detection using phonetic posteriorgram templates,” in *IEEE Workshop on Automatic Speech Recognition & Understanding (ASRU), 2009.*, pp. 421–426, IEEE, 2009.
- [125] D. B. Paul and J. M. Baker, “The design for the wall street journal-based CSR corpus,” in *Proceedings of the workshop on Speech and Natural Language*, Association for Computational Linguistics, 1992.
- [126] J. Lorenzo-Trueba, J. Yamagishi, T. Toda, D. Saito, F. Villavicencio, T. Kinnunen, and Z. Ling, “The voice conversion challenge 2018: Promoting development of parallel and nonparallel methods,” *Submitted to Odyssey*, 2018.
- [127] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, *et al.*, “The kaldi speech recognition toolkit,” in *IEEE 2011 workshop on automatic speech recognition and understanding*, IEEE Signal Processing Society, 2011.
- [128] M. Morise, F. Yokomori, and K. Ozawa, “World: a vocoder-based high-quality speech synthesis system for real-time applications,” *IEICE TRANSACTIONS on Information and Systems*, pp. 1877–1884, 2016.
- [129] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, “Speech parameter generation algorithms for HMM-based speech synthesis,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2000.

- [130] Z. Wu, O. Watts, and S. King, “Merlin: An open source neural network speech synthesis system,” *Proc. SSW, Sunnyvale, USA*, 2016.
- [131] A. Tamamori, T. Hayashi, K. Kobayashi, K. Takeda, and T. Toda, “Speaker-dependent WaveNet vocoder,” in *Proc. INTERSPEECH*, vol. 2017, pp. 1118–1122, 2017.
- [132] A. Van Den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior, and K. Kavukcuoglu, “WaveNet: A generative model for raw audio,” in *SSW*, p. 125, 2016.
- [133] 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*, vol. 4, no. 2, pp. 114–122, 2007.
- [134] T. B. Patel and H. A. Patil, “Combining evidences from mel cepstral, cochlear filter cepstral and instantaneous frequency features for detection of natural vs. spoofed speech,” in *Proc. INTERSPEECH*, 2015.
- [135] L. D. Alsteris and K. K. Paliwal, “Short-time phase spectrum in speech processing: A review and some experimental results,” *Digital Signal Processing*, vol. 17, no. 3, pp. 578–616, 2007.
- [136] M. Krawczyk and T. Gerkmann, “STFT phase reconstruction in voiced speech for an improved single-channel speech enhancement,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 12, pp. 1931–1940, 2014.
- [137] B. Yegnanarayana and H. A. Murthy, “Significance of group delay functions in spectrum estimation,” *IEEE Transactions on Signal Processing*, vol. 40, no. 9, pp. 2281–2289, 1992.
- [138] 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 *Proc. INTERSPEECH*, 2015.

REFERENCES

- [139] Y. Stylianou, “Voice transformation: a survey,” in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2009.
- [140] K. K. Paliwal and L. D. Alsteris, “On the usefulness of STFT phase spectrum in human listening tests,” *Speech Communication*, vol. 45, no. 2, pp. 153–170, 2005.
- [141] C. Valentini-Botinhao, Z. Wu, and S. King, “Towards minimum perceptual error training for DNN-based speech synthesis,” in *Proc. INTERSPEECH*, 2015.
- [142] M. Wester, Z. Wu, and J. Yamagishi, “Human vs machine spoofing detection on wideband and narrowband data,” in *Proc. INTERSPEECH*, 2015.
- [143] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, *et al.*, “Tacotron: A fully end-to-end text-to-speech synthesis model,” *arXiv preprint*, 2017.
- [144] R. Skerry-Ryan, E. Battenberg, Y. Xiao, Y. Wang, D. Stanton, J. Shor, R. J. Weiss, R. Clark, and R. A. Saurous, “Towards end-to-end prosody transfer for expressive speech synthesis with tacotron,” *arXiv preprint*, 2018.
- [145] 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*, pp. 1715–1719, 2016.
- [146] H. Yu, A. Sarkar, D. A. L. Thomsen, Z.-H. Tan, Z. Ma, and J. Guo, “Effect of multi-condition training and speech enhancement methods on spoofing detection,” in *Sensing, Processing and Learning for Intelligent Machines (SPLINE), 2016 First International Workshop on*, pp. 1–5, IEEE, 2016.