TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

Elina Helander

Mapping Techniques for Voice Conversion

Julkaisu 1052 • Publication 1052

Tampere 2012

Tampereen teknillinen yliopisto. Julkaisu 1052
Tampere University of Technology. Publication 1052

Elina Helander

Mapping Techniques for Voice Conversion

Thesis for the degree of Doctor of Science in Technology to be presented with due permission for public examination and criticism in Tietotalo Building, Auditorium TB109, at Tampere University of Technology, on the 19th of June 2012, at 12 noon.

Tampereen teknillinen yliopisto - Tampere University of Technology
Tampere 2012

Thesis advisor

Prof. Moncef Gabbouj
Department of Signal Processing
Tampere University of Technology
Tampere, Finland

Pre-examiners

Prof. Yannis Stylianou
Department of Computer Science
University of Crete
Crete, Greece

Dr. Tomi Kinnunen
Speech and Image Processing Unit
University of Eastern Finland
Joensuu, Finland

Opponent

Prof. Paavo Alku
Department of Signal Processing and Acoustics
Aalto University
Espoo, Finland

Custos

Prof. Ari Visa
Department of Signal Processing
Tampere University of Technology
Tampere, Finland

ISBN 978-952-15-2842-2 (printed)
ISBN 978-952-15-2890-3 (PDF)
ISSN 1459-2045

Abstract

SPEAKER identity plays an important role in human communication. In addition to the linguistic content, speech utterances contain acoustic information of the speaker characteristics. This thesis focuses on *voice conversion*, a technique that aims at changing the voice of one speaker (a source speaker) into the voice of another specific speaker (a target speaker) without changing the linguistic information. The relationship between the source and target speaker characteristics is learned from the training data. Voice conversion can be used in various applications and fields: text-to-speech systems, dubbing, speech-to-speech translation, games, voice restoration, voice pathology, etc.

Voice conversion offers many challenges: which features to extract from speech, how to find linguistic correspondences (alignment) between source and target features, which machine learning techniques to use for creating a mapping function between the features of the speakers, and finally, how to make the desired modifications to the speech waveform. The features can be any parameters that describe the speech and the speaker identity, e.g. spectral envelope, excitation, fundamental frequency, and phone durations. The main focus of the thesis is on the design of suitable mapping techniques between frame-level source and target features, but also aspects related to parallel data alignment and prosody conversion are addressed.

The perception of the quality and the success of the identity conversion are largely subjective. Conventional statistical techniques are able to produce good similarity between the original and the converted target voices but the quality is usually degraded. The objective of this thesis is to design conversion techniques that enable successful identity conversion while maintaining the original speech quality.

Due to the limited amount of data, statistical techniques are usually utilized in extracting the mapping function. The most popular technique is based on a Gaussian mixture model (GMM). However, conventional GMM-based conversion suffers from many problems that result in degraded speech quality. The problems are analyzed in this thesis, and a technique that combines GMM-based conversion

with partial least squares regression is introduced to alleviate these problems. Additionally, approaches to solve the time-independent mapping problem associated with many algorithms are proposed.

The most significant contribution of the thesis is the proposed novel dynamic kernel partial least squares regression technique that allows creating a non-linear mapping function and improves temporal correlation. The technique is straightforward, efficient and requires very little tuning. It is shown to outperform the state-of-the-art GMM-based technique using both subjective and objective tests over a variety of speaker pairs. In addition, quality is further improved when aperiodicity and binary voicing values are predicted using the same technique.

The vast majority of the existing voice conversion algorithms concern the transformation of the spectral envelopes. However, prosodic features, such as fundamental frequency movements and speaking rhythm, also contain important cues of identity. It is shown in the thesis that pure prosody alone can be used, to some extent, to recognize speakers that are familiar to the listeners. Furthermore, a prosody conversion technique is proposed that transforms fundamental frequency contours and durations at syllable level. The technique is shown to improve similarity to the target speaker's prosody and reduce roboticness compared to a conventional frame-based conversion technique.

Recently, the trend has shifted from text-dependent to text-independent use cases meaning that there is no parallel data available. The techniques proposed in the thesis currently assume parallel data, i.e. that the same texts have been spoken by both speakers. However, excluding the prosody conversion algorithm, the proposed techniques require no phonetic information and are applicable for a small amount of training data. Moreover, many text-independent approaches are based on extracting a sort of alignment as a pre-processing step. Thus the techniques proposed in the thesis can be exploited after the alignment process.

Errata

Page 70:
Section 7.2, 2nd paragraph, starting with "Speaker-pair specific MCD …"

DKPLS obtained lower MCD for each speaker pair compared to ML-GMM, except for one speaker pair with five training sentences. The difference between the techniques with that particular pair, however, was not statistically significant. In addition, there were also three other pairs out of 48 pairs that did not obtain statistically significant difference when comparing DKPLS to ML-GMM with five training sentences. With 20 sentences there was one pair that did not obtain statistically significant different MCD as stated. However, when comparing DKPLS to GMM-F, the difference between their MCD means was found to be statistically significant for each speaker pair. This applied to both training data cases.

P6 Publication 6: Section III-B, last paragraph

"The processing steps 2-12…" should be *"The processing steps 3-12…"*.

P7 Publication 7: Appendix

In Step 2), the variable **c** should be capitalized, i.e. there should be $\mathbf{C}=\mathbf{X}\mathbf{Y}^T$ instead of $\mathbf{c}=\mathbf{X}\mathbf{Y}^T$.

"The processing steps 2-12…" should be *"The processing steps 3-12…"*.

Contents

List of Publications

This thesis is a compound thesis based on the following seven publications. In the text, these publications are referred to as [P1], [P2], etc.

[P1] **Elina E. Helander and Jani Nurminen**, On the importance of pure prosody in the perception of speaker identity. In *Proceedings of the 8th Annual Conference of the International Speech Communication Association, Interspeech*, Antwerp, Belgium, Aug. 2007, pp. 2665–2668.

[P2] **Elina E. Helander and Jani Nurminen**, A novel method for prosody prediction in voice conversion. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, Honolulu, Hawaii, USA, Apr. 2007, pp. IV-509–IV-512.

[P3] **Elina Helander, Jan Schwarz, Jani Nurminen, Hanna Silén, and Moncef Gabbouj**, On the impact of alignment on voice conversion performance. In *Proceedings of the 9th Annual Conference of the International Speech Communication Association, Interspeech*, Brisbane, Australia, Sep. 2008, pp. 1453–1456.

[P4] **Elina Helander, Jani Nurminen, and Moncef Gabbouj**, Analysis of LSF frame selection in voice conversion. In *Proceedings of the 12th International Conference on Speech and Computer, SPECOM*, Moscow, Russia, Oct. 2007, pp. 651–656.

[P5] **Elina Helander, Hanna Silén, Joaquin Míguez, and Moncef Gabbouj**, Maximum a posteriori voice conversion using sequential Monte Carlo methods. In *Proceedings of the 10th Annual Conference of the International Speech Communication Association, Interspeech*, Makuhari, Chiba, Japan, Sep. 2010, pp. 1716–1719.

[P6] **Elina Helander, Tuomas Virtanen, Jani Nurminen, and Moncef Gabbouj**, Voice conversion using partial least squares regression. *IEEE Transactions on Audio, Speech, and Language Processing*, Special Section on Voice Transformation, vol. 18, no. 5, pp. 912–921, Jul. 2010.

[P7] **Elina Helander, Hanna Silén, Tuomas Virtanen, and Moncef Gabbouj**, Voice conversion using dynamic kernel partial least squares regression. *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 3, pp. 806–817, Mar. 2012.

Abbreviations

ACR	Absolute category rating
ANN	Artificial neural network
A/S	Analysis/synthesis
BAP	Band aperiodicity
CART	Classification and regression tree
CCR	Comparison category rating
CV	Cross-validation
DCT	Discrete cosine transform
DKPLS	Dynamic kernel partial least squares
DPLS	Dynamic partial least squares
DTW	Dynamic time warping
EM	Expectation maximization
F_0	Fundamental frequency
F-F	Female-to-female
F-M	Female-to-male
GMM	Gaussian mixture model
HMM	Hidden Markov model
HNM	Harmonic plus noise model
IDCT	Inverse discrete cosine transform
KPLS	Kernel partial least squares
LP	Linear prediction
LPC	Linear prediction coefficient
LSF	Line spectral frequency
MAP	Maximum a posteriori
MCC	Mel-cepstral coefficient
MCD	Mel-cepstral distortion
M-F	Male-to-female
MFCC	Mel-frequency cepstral coefficient
MGC	Mel-generalized cepstral coefficient
ML	Maximum likelihood

M-M	Male-to-male
MMSE	Minimum mean squared error
MOS	Mean opinion score
MSE	Mean squared error
MV	Mean-variance
NIPALS	Non-linear iterative partial least squares
OLS	Ordinary least squares
PCR	Principal component regression
PDF	Probability density function
PLS	Partial least squares
RMSE	Root mean squared error
SD	Spectral distortion
SIMPLS	Simple iterative partial least squares
SVR	Support vector regression
TTS	Text-to-speech
VAD	Voice activity detection
VC	Voice conversion
VR	Variance ratio
VT	Voice transformation
VTLN	Vocal tract length normalization

Chapter 1

Introduction

Speech is the most important communication form between humans. In everyday life we automatically decode speech into language regardless of who speaks. In a similar way, we have the ability to recognize different speakers in spite of the linguistic content of the speech. The physical and physiological properties of the speech production organs and learned speaking habits affect the voice individuality of a speaker. Voice individuality helps us to identify the person to whom we are talking without seeing the speaker. A familiar speaker can be recognized even from a single word like "hello" over the telephone [Kre11].

Separating the speaker identity from the lexical content is easy for humans but still somewhat difficult for machines. The major increase in computer speed and storage has offered a new possibility for people to communicate with computers through speech-based human-computer interfaces. Computers are able to recognize and synthesize speech in order to interact with people that have different voice characteristics. In speech recognition, voice individuality is considered as an obstacle, and speaker normalization and adaptation techniques are used to compensate the acoustic differences resulting from different voice qualities.

In text-to-speech (TTS) synthesis, a given text is analyzed and "spoken" by a machine. Nowadays most TTS systems are either based on cutting and pasting segments of speech from a large recorded database (called *unit selection speech synthesis*) [Hun96] or by creating speech from statistical models trained from speech parameters [Tok02]. In the first case, the TTS voice is restricted to be the speaker that recorded the database (a so-called master speaker). Recordings require time, effort, and storing capabilities. Hence, there are usually only a few alternative voices to select from. With the help of *voice conversion* (VC), that is the topic of this thesis, only a small set of recordings of the desired speaker is needed to make the synthesizer speak with the desired speaker's voice. For example text messages or e-mails can be read aloud with the sender's own voice.

Although the most evident application is a TTS system, it is by no means the only one. Potential application areas include dubbing, voice restoration,

language learning, chat rooms, games, and voice pathology. The most extreme case is cross-lingual VC where the source and target do not speak the same language [Sün06b, Err10a]. The approach can be used for example in dubbing with the original actors' voice or in speech-to-speech translation. Using speech-to-speech translation systems people who do not speak the same language can interact with the help of speech recognition, language processing, translation, and speech synthesis [Gu06]. Instead of using a generic TTS voice, the message can be spoken by the user's own voice although he/she cannot speak the language.

Voice conversion has been a topic of interest during the last two decades. In spite of the research, the commercial usage of the technology has been limited due to the unsatisfactory speech quality. There is a compromise between the identity conversion and quality; better identity conversion usually requires more signal modifications that may cause more distortions. Furthermore, the perception of the quality and speaker identity are largely subjective. There is no unique correct conversion result; a person can utter a given sentence in slightly different ways that are still inherent to the speaker. Due to these reasons, time-consuming listening tests must be used in the voice conversion system evaluation.

1.1 Overview of Voice Conversion

This thesis defines voice conversion as automatic modification of speech spoken by one speaker (a *source* speaker) to give an impression that it was spoken by another specific speaker (a *target* speaker). Also multiple source speakers can be used, but in this thesis, there is only a single source speaker. Voice conversion should not be confused with a term *voice transformation* (VT) that refers to various modifications that are applied to speech signals. Sometimes these terms are used interchangeably in the literature. The main difference between VC and VT is that in VC, there is always a specific target speaker that a VC system tries to mimic. This thesis concentrates on VC.

The conventional VC process consists of two phases: training and conversion. In training, a mapping model from source features to target features is created based on training data from both speakers. In the conversion phase, any unknown utterance from the source speaker can be converted to sound like it was spoken by the target speaker. This *stand-alone voice conversion* represents the core idea of VC regardless of the application and it is depicted in Figure 1.1.

If VC is used in a unit selection TTS system, the initial input is text. The text is analyzed by the TTS system, the chosen segments (units) from the master TTS speaker are parameterized and converted to match the target speaker characteristics learned from the training data. The interaction between a VC system and a TTS system has not been well studied and most studies concentrate on stand-alone VC.

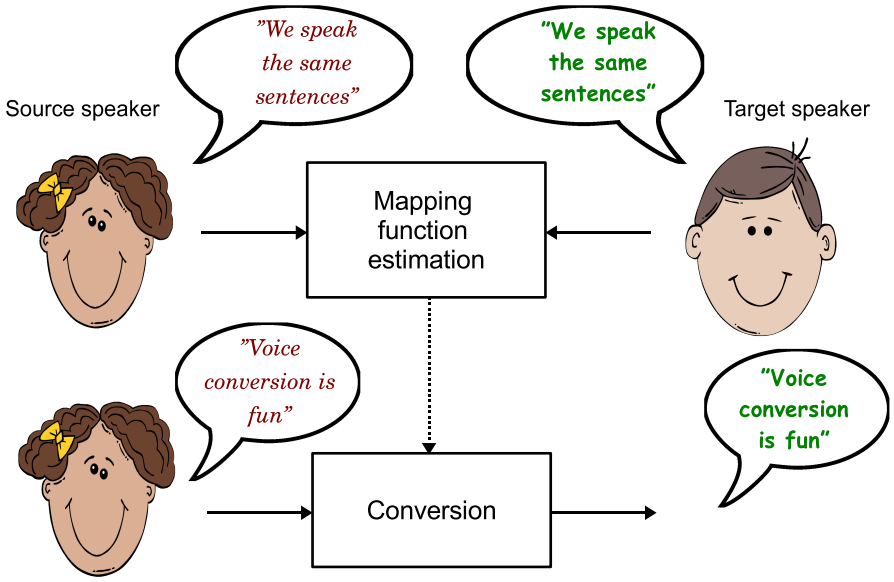


Figure 1.1: Stand-alone voice conversion.

Voice conversion systems can be text-dependent or text-independent. Conventional VC systems as shown in Figure 1.1 are usually text-dependent meaning that they require the same texts (parallel data) to be recorded from both the source and target speakers, but the linguistic content is not explicitly used. The parallel sentences can be aligned for example with the help of dynamic time warping [Rab93].

Nowadays the trend has shifted towards text-independent (non-parallel) voice conversion. One alternative is to use a TTS system to synthesize the same texts that the target speaker has spoken [Dux06b, Tot08]. Most techniques proposed to cope with non-parallel data in a standard VC framework are based on finding phonetic or acoustic similarities from the source and the target data. Some of them exploit techniques commonly used for speech recognition; statistical adaptation techniques [Mou06] and vocal tract length normalization [Sün03]. Some text-independent approaches require linguistic knowledge of the data, e.g. phonetic labels were used in [Tao10]. Alternatively, the data can be automatically divided into acoustic classes [Sün06a]. The alignment technique proposed in [Err10a] is an iterative procedure that first aligns the data using nearest neighbor of each source vector in the target acoustic space, estimates a conversion function, converts source vectors and repeats the nearest neighbor procedure for the converted source samples, makes the alignment again and so on.

In practice, the performance of a voice conversion system is rather dependent on the particular source-target speaker pair. The most common problem formulation is to have data from only one source and one target speaker available, as it is assumed in this thesis. However, there are voice conversion approaches that

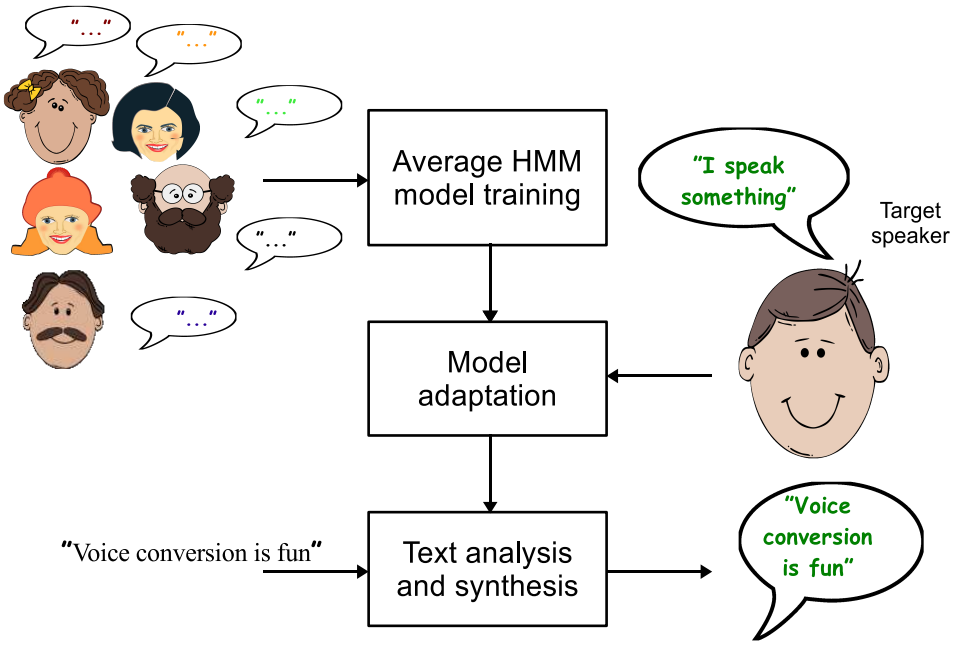


Figure 1.2: Voice conversion in HMM-based speech synthesis is obtained by adapting an average voice model trained from many "source" speakers.

exploit speech from more than two speakers. Figure 1.2 illustrates the conversion process in statistical parametric speech synthesis that is most often used as a synonym for hidden Markov model (HMM) based speech synthesis [Tok02]. The "conversion" process follows the idea of speaker adaptation used in speech recognition; an average voice model trained from multi-speaker data is adapted with speech data from the target speaker [Yam09a]. This gives an ability to generate a large set of voices without considerable effort and restrictions on the training data [Yam10]. Detailed linguistic content of the training data is usually required, but the use of a speech recognizer together with a speech synthesis system can remove this requirement [Yam09b].

Furthermore, the use of eigenvoices [Tod07b, Oht10] is an example of an approach utilizing speech from many speakers. In the eigenvoice method, originally developed for speaker adaptation in speech recognition [Kuh00], the parameters of any speaker are formed as a linear combination of eigenvoices. The main benefit of the eigenvoices is that only a little amount of training data for the target is required and the linguistic content does not have to be known. As a prerequisite, the approach requires a large amount of speakers (80 is used in [Tod07b]) with parallel training data.

A fundamental question is that what speech features should be modified and how, for obtaining believable identity transformation while maintaining a reasonably good speech quality. The voice individuality is a combination of many parameters [Kuw95]. The characteristics of a speaker are usually visible at seg-

mental and suprasegmental level. Segmental level is mainly related to a speaker's speech production system anatomy that defines the timbre and the fundamental frequency (F_0) of the voice whereas suprasegmental level refers to intonation and speaking style. In addition, the choice of words can be considered as a speaker-specific property.

Speech signal is usually parameterized into excitation and spectral envelope for enabling modifications and mapping function estimation at the feature level. Most VC systems modify only segmental properties such as the timbre (the spectral envelope) and the F_0 scale at frame level. A disbenefit of many VC mapping techniques is that they ignore temporal continuity of speech features. Furthermore, some of them require a lot of tuning to find an optimal configuration for different sizes of training data or do not perform well with small databases.

Techniques proposed for VC have been successfully used in other feature mapping approaches, such as emotion conversion [Kaw99b], bandwidth expansion of narrowband speech [Par00], acoustics-to-articulatory mapping [Tod08], speech enhancement [Mou07], and body-conducted speech conversion [Tod09].

1.2 Objective and Scope of the Thesis

This thesis concentrates on stand-alone voice conversion as depicted in Figure 1.1. A major challenge is to utilize the limited training data to find an effective mapping between source and target features. Most mapping approaches are based on statistical conversion functions. The most popular approach, GMM-based conversion [Kai98, Sty98, Tod07a], provides a reasonable identity transformation at the cost of quality. Problems related to GMM-based voice conversion are addressed and alleviated in this thesis. In addition, a new mapping approach based on kernel transformed source data is proposed. The thesis specifically addresses the overfitting problem common in all model fitting tasks and the temporal continuity of converted speech features that is ignored by many mapping techniques. The overall objective of the thesis is to propose mapping techniques that allow good identity conversion but preserve the quality well with a small amount of training data. Furthermore, the techniques in the journal articles [P6, P7] are simple to tune and implement. These issues can be considered important advantages for example in small hand-held devices.

Quality is mainly related to segmental level and most of the research has been focused on converting the spectral envelope features. The emphasis of the thesis is on spectral conversion as well, but to make a complete voice conversion system, also algorithms for converting parameters related to excitation/residual are proposed. Moreover, an important cue of a speaker identity is prosody, that is a suprasegmental phenomenon visible at F_0 contours or sound durations at syllable, word, sentence, or even paragraph level. However, conventionally F_0 is

converted at segmental level. This thesis includes a technique for F_0 and duration prediction at syllable level.

In the literature, it is common to use several tens or even hundreds of parallel sentences that cannot be considered realistic. The fundamental idea of VC is to have only a small amount of training data available, at least from the target side. In this thesis, a small amount means 5–30 parallel sentences. Non-parallel alignment methods are beyond the scope of the thesis, but most of them carry out alignment as a pre-processing step. The core mapping techniques proposed in this thesis are applicable also for non-parallel cases after the alignment process.

1.3 Author's Contributions

The author has carried out the majority of the research in each of the included publications [P1–P7]. Other authors have mainly contributed in writing except in [P3] Jan Schwarz organized a listening test and calculated objective alignment accuracy results without conversion and Hanna Silén provided the codes for dynamic time warping. Prof. Joaquín Míguez has proposed a cost optimization technique using particle filters [Míg12] and the idea of applying the technique to voice conversion came to my mind when taking his course. This resulted in a common publication [P5].

To summarize, the most important contributions of the thesis are the following:

- Provide investigation on familiar speaker identification on the basis of a pure prosodic signal [P1] (Section 2.2).
- Provide analysis on how standard alignment technique based on dynamic time warping succeeds for aligning parallel data and what should be taken into account [P3] (Section 3.2.2).
- Provide a study to determine whether it would be possible to select real target speech segments (frames) for generating high quality speech in a unit selection manner used in TTS systems [P4] (Section 4.1).
- Propose to use a cost function for post-processing of converted speech feature sequence that is solved using a cost optimization technique based on particle filtering [P5] (Section 4.7).
- Propose a new prosody prediction technique for F_0 [P2] (Section 4.9.2). The technique has been granted a patent [Hel11].
- Address the problems of GMM-based VC and propose to combine GMM-based VC with partial least squares (PLS) regression in order to avoid overfitting [P6] (Chapter 5).

- Improve the temporal continuity of GMM-based VC with posterior probability smoothing [P6] (Chapter 5.4).
- Propose a novel approach for providing non-linear and temporally continuous mapping using dynamic kernel partial least squares (DKPLS) regression [P7] (Section 6.1).
- Address the importance of cross-validation order when using PLS for temporally correlated data such as speech features [P7] (Section 6.3).
- Propose techniques for aperiodicity and voicing prediction that use DKPLS and information from other features [P7] (Section 6.4).

1.4 About Notation

The variable n denotes for aligned source and target frame indices, i.e. a training data pair index. The original previous and next frame of frame n are denoted by $n-$ and $n+$, since n–1 and n+1 may not correspond to consecutive frames of frame n after training data selection. For the conversion phase, sections 4.5.1 and 4.7 employ variable t to denote the temporal frame index within a sentence and t–1 and t+1 are really the preceding and the next frame of frame t.

The bold-face symbols are used to denote for vectors and matrices. The original feature vectors for the source and the target in frame n are denoted by $\mathbf{x}_n$ and $\mathbf{y}_n$, respectively. Generally, the upper case letters denote for matrices but in this thesis, there are two exceptions; variables $\mathbf{X}_n$ and $\mathbf{Y}_n$ denote for the n^{th} input and output vector for prediction, respectively. In addition, they are used in the context of a joint-density GMM model that includes dynamics (Section 4.5.1).

1.5 Organization of the Thesis

This thesis is organized as follows. Chapter 2 briefly introduces speech production and parameterization as well as speaker identity perception. An overview of a voice conversion system and its evaluation is given in Chapter 3. Mapping techniques are devoted to an own chapter, Chapter 4. Chapters 5 and 6 describe two novel mapping techniques in detail. Chapter 7 gives results on objective spectral mapping performance for different mapping algorithms proposed or revised in the thesis. The main conclusions of the thesis are provided in Chapter 8 together with discussion and future work.

Chapter 2

Speech Feature Extraction and Speaker Identity

2.1 Speech Production and Perception

SPEECH production is a complex process in which a large number of muscles take part [Ben08]. The airflow is produced in the lungs and it is passed to the vocal folds of larynx. When the vocal folds vibrate during articulation, the resulting sound is *voiced*, otherwise; the sound is *unvoiced*. The vibration rate of vocal folds is referred to as F_0 or pitch and it is related to the length and mass of the vocal folds. The shorter and thinner the vocal folds, the higher is the pitch. Pitch is actually a perceptual factor whereas F_0 is the acoustic correlate of pitch [Hua01]. In many studies as well in this thesis, these terms are used interchangeably.

Vocal folds produce a glottal wave that consists of F_0 and its harmonics. The glottal wave travels through the upper respiratory tract, where pharyngeal, oral, and nasal cavities act as resonators. Different resonances occur depending on the position and shape of the lips, jaw, tongue, soft palate, and other speech organs. Harmonics near the resonances become emphasized resulting in *formants* that show up as broad peaks in the spectrum. The human speech production organs are shown in Figure 2.1.

The sounds of a language, i.e. the inventory of phonemes, can be divided into two classes: vowels and consonants. This classification applies to almost all languages in the world [Hua01]. Phonemes can be further divided into subgroups (nasals, fricatives, etc.) based on certain articulatory properties.

An example of a speech segment is shown in Figure 2.2a. The sampling rate of the signal is 16 kHz and according to the Nyquist theorem, the highest frequency that the signal can contain is 8 kHz. The segment is taken from the word "cash" (*/k//ae//sh/*). The waveform consists of two phonemes, a vowel */ae/* and an unvoiced consonant */sh/*. The pitch period of the vowel */ae/* is about 7.4 ms

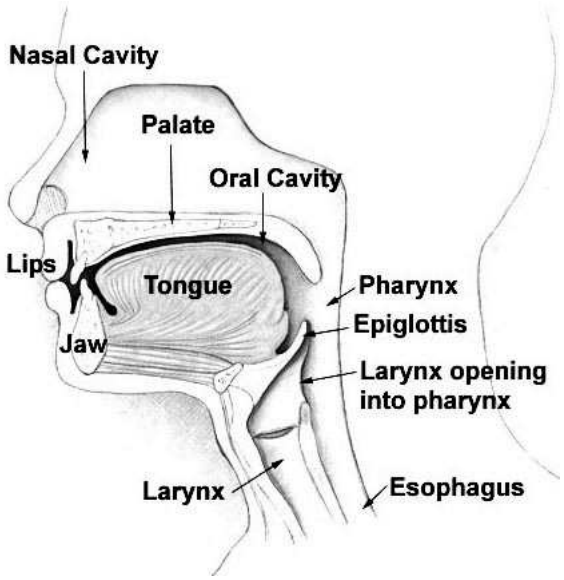


Figure 2.1: The human speech production organs [Wik11].

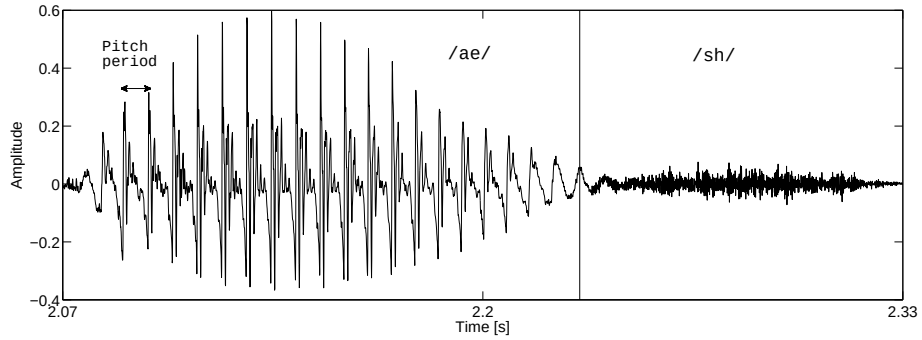
corresponding to an F_0 of 136 Hz. For the unvoiced consonant, no pitch period is observable.

Figure 2.2b shows a spectrogram, a long-term time-varying spectral representation of the speech signal. The darker a band is, the more energy the signal contains at a given frequency. The energy of the voiced sound is concentrated on lower frequencies and it is unevenly distributed because of the formants. During the unvoiced sound */sh/*, the energy is distributed evenly over higher frequencies.

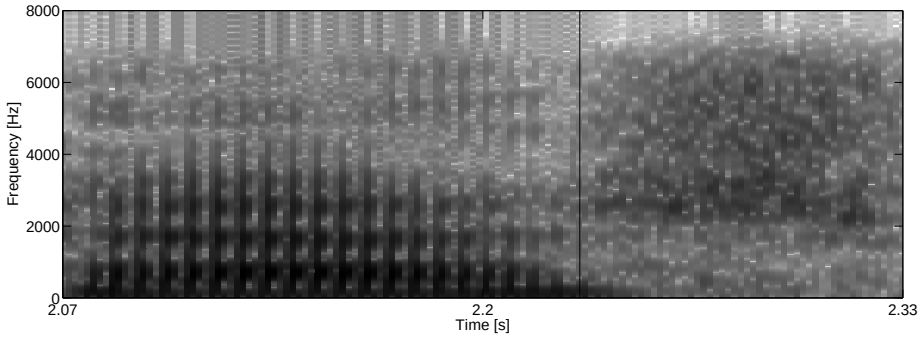
The ears and the brain are the major components of the speech perception system. The acoustic signal is transformed into a mechanical vibration pattern on the basilar membrane in the ear and then passed to the brain where various types of information is extracted. The cochlea of the inner ear acts as a spectrum analyzer. A lot of research has been devoted to derive frequency scales that follow the human perception. Two well-known scales are Bark scale and Mel scale. The Mel scale is approximately linear below 1 kHz and logarithmic above. The Mel scale is widely used for speech feature extraction in many applications.

2.2 Speaker Identity Perception

The success of voice conversion depends on how subjects perceive the quality and the speaker identity in the converted sample. The correspondence between acoustic correlates of speaker individuality and the human perception is not fully understood [Lav01], but this understanding would be relevant and beneficial for various fields of speech technology, especially for voice conversion.



(a)



(b)

Figure 2.2: An example of (a) speech waveform of a vowel */ae/*, and an unvoiced consonant */sh/* and (b) the corresponding spectrogram representation (approximate phone boundary is indicated with a vertical line).

Every speaker has a unique vocal anatomy and hence unique vocalizations are produced by different speakers. The effect of vocal tract is clearly visible in voiced sounds meaning that they contain more speaker-specific information. In particular, vowels and nasals are more effective than other sounds for both perceptual speaker identification [Ami07] and automatic speaker recognition [Eat94].

The first and the second formant are mainly related to the phoneme identity whereas the third and the fourth formant are related to the speaker [Kuw95]. In addition to the third and the fourth formant, the most important acoustic features related to speaker identity include F_0 and the closing phase of the glottal wave [Lav01]. Formant center frequencies affect more than F_0 and a 5% shift in center formant frequencies destroys the speaker identity [Kuw95]. Shifting higher formants is more destructive, but its effect varies between different speakers. Speaker individualities appear mostly in the spectral envelopes of vowels in the region of 1700–2500 Hz [Kit07].

Prosody is also an important aspect of voice individuality. Indeed, it has been observed that imitators are not able to fool automatic speaker recognition

systems but may fool people [Zet04]. This results from the fact that speaker recognition systems use mainly segmental information and not prosodic information as humans do. In [P1] the importance of prosody was investigated in the case of familiar speakers. A listener was asked to recognize a speaker he/she knows on the basis of a very coarse signal expressing the speaker's prosody. The coarse signal form was originally proposed for prosody evaluation in TTS systems [Son98]. During voiced sections, the stripped signal consists of a single sinusoid whose frequency and amplitude follow the F_0 and energy contour, respectively. Unvoiced and silent regions are represented as silence.

The recognition decision was made from a group of two or three speakers that had their average F_0 level close to each other [P1]. It was concluded that it is possible to identify familiar people on the basis of pure prosody. This indicates that prosody conversion is likely to be important when the target speaker is familiar to the listeners. The recognition rate of familiar speakers was slightly improved if the texts of the sinewave signals were provided [P1].

It can be summarized that vocal tract information (e.g. formants) is more important than the glottal wave information for identification, but the specific parameter importance varies from speaker to speaker and from listener to listener [Lav01]. Nevertheless, it is apparent that listeners exploit all sources of information, such as pitch, loudness, voice quality and their variation over time.

2.3 Speech Signal Representations

The production of speech involves a time-varying vocal tract system with time-varying glottal source. The speech signal is thus non-stationary, but many signal processing algorithms presume stationary signals. Speech signals are assumed to be stationary enough in blocks of 10–30 ms.

Short-time Fourier analysis is a basic tool for analyzing speech. Speech signal is decomposed into a series of short overlapping segments (frames), meaning that all samples within e.g. 20 ms window are gathered together at say 5–10 ms steps and a discrete Fourier transform (DFT) for each frame is calculated. An example of a DFT magnitude spectrum extracted from a vowel is shown in Figure 2.3 with dashed line. The spectrum is affected by the speaker's F_0 and the formant structure. Using a source-filter model, the effect of F_0 and vocal tract (formants) can be separated. The source or excitation represents the air flow at the vocal folds and the filter represents the resonances of the vocal tract. Usually speech recognizers can ignore the excitation since phoneme information is mainly included in the filter. However, for VC, there is a need to modify both the excitation (pitch) and the filter (timbre).

Although formants as such carry a lot of speaker-specific information and have been used as VC features in some studies e.g. [Nar95, Ren04], their estimation and modification is difficult. Representations obtained by straightforward

mathematical rules are more popular. This section introduces commonly used representations for speech signals.

2.3.1 Linear Prediction and Line Spectral Frequencies

Linear prediction (LP) is one of the most important speech analysis techniques. In LP, an all-pole filter from a short-time speech segment is estimated. The current sample s_n is predicted as a linear combination of its past p samples

$$\hat{s}_n = \sum_{k=1}^{p} a_k s_{n-k} \tag{2.1}$$

where a_k is the k^{th} coefficient of the all-pole filter $A(z)$. The coefficients of the LP filter can be estimated in various ways, for example using the autocorrelation method with Levinson-Durbin recursion [Hua01].

Linear prediction coefficients (LPCs) can be further transformed into line spectral frequencies (LSFs) and a fully reversible conversion back to LPCs is retained. LSFs, also called as line spectral pairs, are obtained by computing the roots of two polynomials, $P(z)$ and $Q(z)$, that are defined as

$$P(z) = A(z) + z^{-(p+1)}A(z^{-1}) \tag{2.2}$$

$$Q(z) = A(z) - z^{-(p+1)}A(z^{-1}) \tag{2.3}$$

Figure 2.3 shows an example of a short-time Fourier spectrum of a Hanning-windowed speech segment. The resulting 18^{th} order LP spectrum and the corresponding LSFs are also shown in Figure 2.3.

LSFs offer a robust representation for quantization and modification purposes. They have a close relationship to formants; if two or more LSFs are close to each other, this indicates a formant. Due to these properties, LSFs have been popular features for VC [Ars99, Nur06, Tur06, Err10b, Tao10] [P2, P3].

2.3.2 Cepstral Features

Another way to separate the source from the filter is via cepstrum. A (power) cepstrum is the result of taking the inverse Fourier transform of the log-magnitude Fourier spectrum. The *Mel-frequency cepstrum coefficients* (MFCCs) [Dav80] are widely used features in speech and speaker recognition. They parameterize the rough shape of the spectral envelope in a perceptually meaningful way. For each short-time frame, the power spectrum is calculated with DFT and the linear frequency scale is substituted with the Mel scale

$$f^{\text{mel}} = 2595 \log_{10}(1 + f/700) \tag{2.4}$$

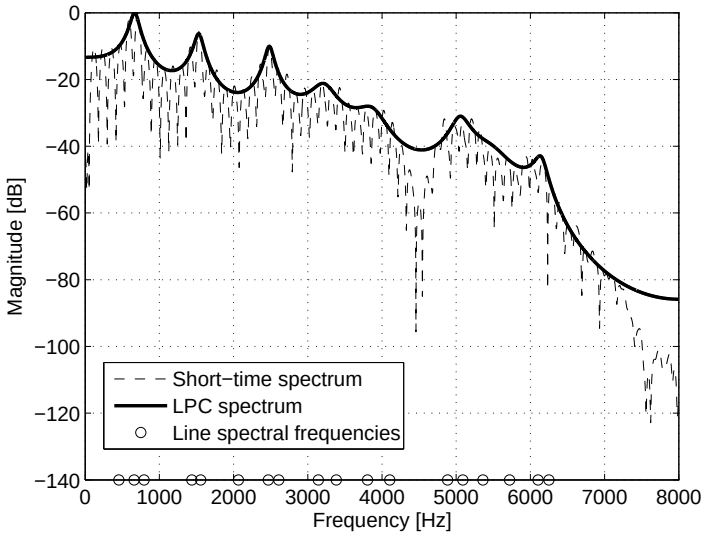


Figure 2.3: Spectrum of a speech segment (dashed line) and the resulting 18^{th} order LP spectrum (solid line) with corresponding line spectral frequencies (circles at the bottom of x-axis).

where f is the frequency in linear scale. The scale conversion is implemented using a bank of triangular filters uniformly distributed on the Mel scale. MFCCs are obtained by applying a discrete cosine transform (DCT) to the logarithmic filterbank energies as

$$c_d^{\mathrm{MFCC}} = \sum_{k=0}^{K-1} S_k \cos\left(\frac{\pi d(k-0.5)}{K}\right) \tag{2.5}$$

where S_k is the logarithmic energy output of the k^{th} filter, K is the number of filters and c_d^{MFCC} denotes for the d^{th} MFCC. For sampling rate of 16 kHz, usually 24 filters are used. DCT provides decorrelation of the features and the energy is usually concentrated on the first coefficients. For speech recognition, it is common to retain only the first coefficients (for example 13 out of 24).

2.3.3 Generalized Mel-Cepstral Analysis Method

Standard LPCs give information of the formants (peaks) but not the valleys (spectral zeros) in the spectrum whereas cepstral processing weights peaks and valleys equally. The generalized Mel-cepstral analysis method [Tok94] unifies both of them and gives flexibility to balance between them. The procedure is controlled by two parameters, α and γ. The parameter γ balances between the cepstral (γ=0) and linear prediction representation (γ=–1) whereas the parameter α controls the frequency resolution of the spectrum (α=0 for linear scale and

α=0.42 for approximating the Mel scale for sampling frequency of 16 kHz). The procedure results in Mel-generalized cepstral coefficients (MGCs). To obtain the cepstral coefficients, a cost function based on unbiased estimation of log spectrum [Tok95] is applied. In the SPTK toolkit [SPT], the Newton-Raphson method is used to minimize the cost function.

Mel-cepstral coefficients (MCCs) (γ=0) are a special case of MGCs and they have been popular in VC [Tod07a, Tot08, Des10] [P5, P6, P7]. The spectrum is modeled using the D^{th} order MCCs c_d, $d = 0, 1, \ldots, D$, as

$$H(z) = \exp \sum_{d=0}^{D} c_d \tilde{z}^{-d} \tag{2.6}$$

where $\tilde{z}^{-1}$ is defined by a first order all-pass function

$$\tilde{z}^{-1} = \frac{z^{-1} - \alpha}{1 - \alpha z^{-1}} \tag{2.7}$$

and α is 0.42 for sampling frequency of 16 kHz.

Figure 2.4 shows a speech spectrum obtained from the STRAIGHT analysis/synthesis system [Kaw99a] (Section 3.1) for a speech segment sampled at 16 kHz. Three types of 24^{th} order MGCs ({γ=0, α=0}, {γ=0, α=0.42}, and {γ=–0.5, α=0.42}) were extracted from the spectrum. The corresponding three spectra are shown in Figure 2.4. When α is set to 0.42 (non-linear scaling) the modeling at low frequencies is more accurate than at higher frequencies whereas linear scaling gives equal weight for all frequencies. In the case of γ=–0.5, the analysis algorithm gives more weight to peaks than the valleys.

2.4 Summary

Speech production and perception are complex processes. The interaction of the brain with vocal tract and auditory system is not fully understood. Different vocal tract anatomies and learned speaking habits make each person's voice unique. Learned speaking habits affect prosodic characteristics and to an extent, prosody can alone be used to identify a familiar speaker [P1].

Acoustic descriptors of speech include for example resonances of the vocal tract (formants) and vibration rate of the vocal cords (F_0). To separate the effect of F_0 from the vocal tract (spectral envelope), linear prediction or cepstral techniques can be used. Frequency axis can be transformed into a perceptually motivated scale, such as Mel scale. Mel-generalized analysis method combines linear prediction and cepstral processing in a unified manner.

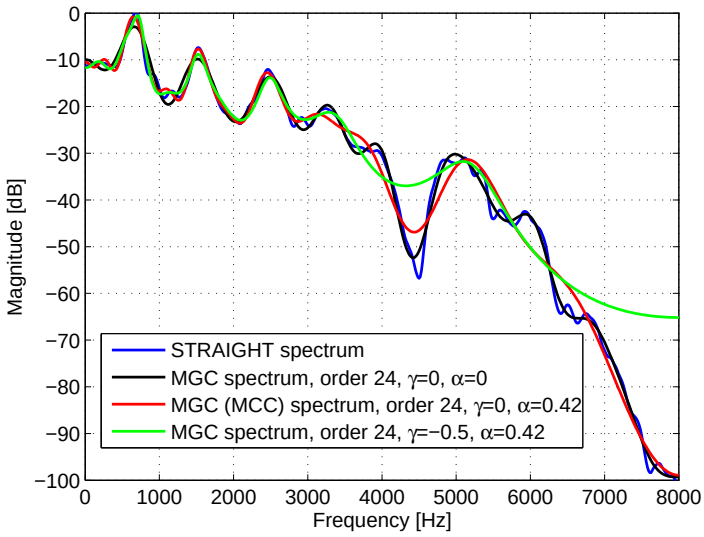


Figure 2.4: STRAIGHT spectrum (blue line) and the corresponding 24^{th} order Mel-generalized cepstral coefficient spectra with different settings for α and γ.

Chapter 3

Voice Conversion System

The voice conversion process consists of two parts: training and conversion. A general block diagram of the training phase is shown in Figure 3.1. The system is given speech databases from both source and target speaker. In this thesis, the speech databases contain utterances with the same lexical content (a parallel corpus). An analysis/synthesis (A/S) system, e.g. LPC codec, STRAIGHT, or codec based on harmonic-plus-noise model, is used to analyze the speech waveforms and provide frame-level estimates of parameters related to the used speech model. Usually the parameters include F_0, spectral envelope, and excitation in some form. Spectral envelope is typically parameterized for example into LSFs or MCCs explained in Section 2.3. A/S frameworks are described in Section 3.1.

The utterances of two speakers are aligned at frame level in order to obtain correspondence between different speech sounds. In the case of parallel data, this is accomplished for example by dynamic time warping. The alignment process is described in Section 3.2. Using the aligned source and target features, a mapping function between the spectral, excitation, or F_0 features is estimated. A variety of techniques exists for obtaining the mapping function. They are not discussed in this chapter but are devoted to an own chapter, Chapter 4.

In the conversion phase depicted in Figure 3.2, any unknown utterance from the source speaker can be transformed to sound like the target speaker. The utterance is analyzed and parameterized with the same A/S framework as in training and the conversion function obtained in the training phase is applied to the features. An inverse parameterization of the converted features is conducted. A time-domain waveform is generated from a set of converted parameters using the A/S framework.

The success of any new technology depends on the user's opinion on the perceived quality. The VC quality comprises two factors: the overall sound quality and the success of identity conversion. No standards exist for VC system evaluation. Most of the system evaluation strategies are borrowed from speech coding. Chapter 3.3 describes evaluation methodologies.

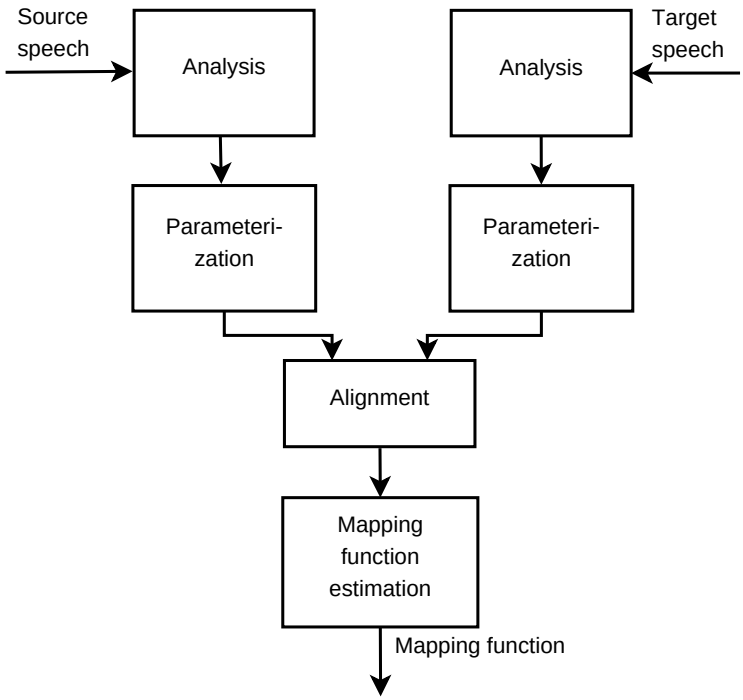


Figure 3.1: Training phase.

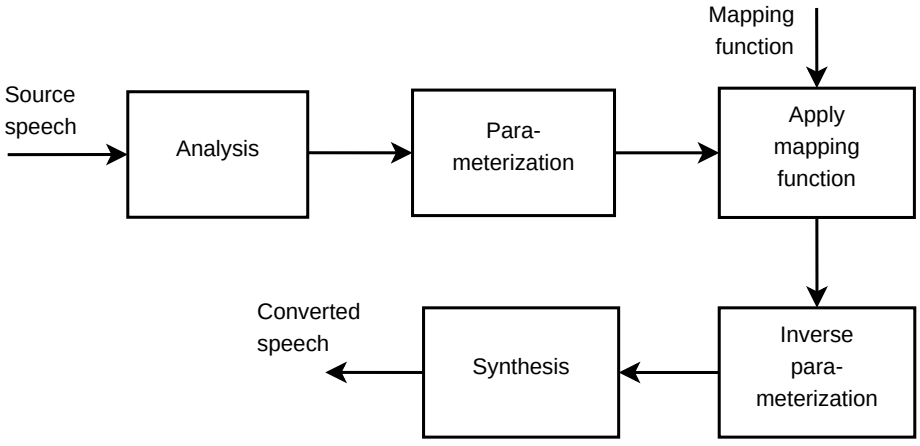


Figure 3.2: Conversion phase.

3.1 Analysis/Synthesis Framework

An essential part for successful VC is a high-quality A/S framework. The spectral envelope, F_0, and segmental durations should be easily modifiable. The frameworks reviewed in this section (LP-based codec, STRAIGHT, and harmonic-plus-noise model based codec) allow both spectral and prosodic modifications.

In LP-based codecs, the spectral envelope is represented with an all-pole filter. The simplest LPC codec, called LPC vocoder, models the excitation to consist of white noise for unvoiced segments or a sequence of impulses spaced at pitch period for voiced segments. If the codec decides that the current frame is voiced, an estimate of F_0 is needed. In synthesis, the excitation is passed through a time-varying all-pole filter.

The correct voiced-unvoiced estimation is essential for high-quality speech modification. If a voiced segment is declared as unvoiced, the synthesized sound is rough and less intelligible. On the other hand, classifying unvoiced segment as voiced results in metallic sounding speech. However, this two-category hard decision model is too simple, since for example voiced fricatives such as */z/* contain both periodic and non-periodic components. A *mixed excitation* is a weighted sum of both an impulse train and noise. In speech coding, a mixed excitation can be obtained by filtering the speech signal into frequency bands (e.g. 0–1 kHz, 1–2 kHz, 2–4 kHz, 4–6 kHz, and 6–8 kHz) and estimating the average voicing strength in these bands [Kon04]. Voicing strength can be estimated using the normalized correlation coefficients around the pitch period.

A simplified alternative is to exploit the idea of split-band linear predictive coding [Atk97], where the excitation signal is divided into two parts separated by a marker; the lower part of the spectrum is declared voiced and the upper part unvoiced. This results in a form of mixed excitation. Nurminen et al. [Nur06] modeled the residual in a more sophisticated way. They assumed that the excitation signal is as a sum of sine waves with frequency of F_0 harmonics for voiced frames and a fixed value for unvoiced frames. Sinusoids evolve randomly or linearly in time corresponding to unvoiced parts and voiced parts of the residual spectrum, respectively. The model and the corresponding A/S system described in [Nur06] was used in [P2, P3].

Instead of using sinusoidal modeling only for the residual, the original speech segment can be decomposed as a sum of sinusoids that are characterized with frequency, amplitude and phase. A special case of sinusoidal representation is harmonic modeling where frequencies of the sinusoids are multiples of F_0. A *harmonic plus noise model* (HNM) [Sty05] is an example of a sinusoidal model. HNM-based codecs have been a popular VC framework [Sty98, Shu06, Err10b]. In HNM, speech is decomposed into periodic (harmonic) and non-periodic (noise) components. A variety of techniques exist for extracting the parameters of the harmonic part. The noise component was extracted by subtracting the harmonic part from the original speech in the time-domain [Sty05].

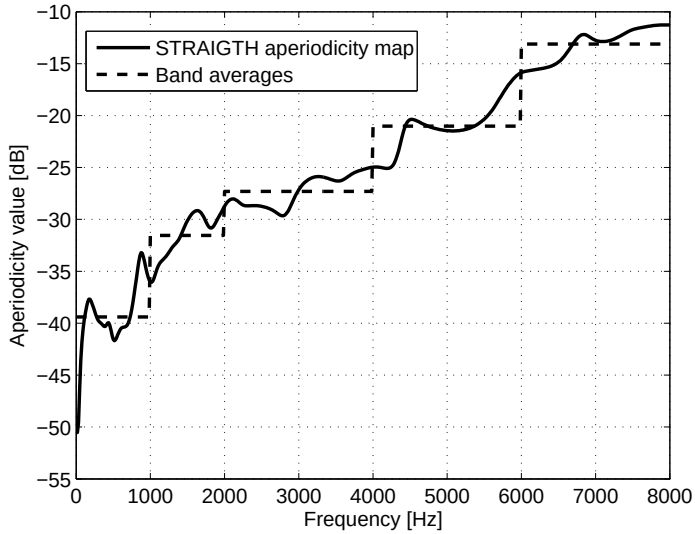


Figure 3.3: STRAIGHT aperiodicity map (solid line) and the corresponding band averages (0–1 kHz, 1–2 kHz, 2–4 kHz, 4–6 kHz, and 6–8 kHz).

The amplitudes and phases of the harmonics are not suitable as such for feature conversion, since their dimensionality varies according to F_0. Harmonic amplitudes and frequencies were parameterized into LSFs in [Err10b, Sty98].

STRAIGHT [Kaw99a] is a high-quality A/S system that has been a popular framework for VC [Che03, Oht06, Tod07a, Des10] [P5, P6, P7]. A speech waveform is decomposed into F_0 contour and a spectrum with carefully tuned algorithms. The estimation of the spectrum is done pitch-adaptively and the success of the F_0 extraction affects the estimation process. An example of STRAIGHT spectrum was given in Section 2.3.3 (Figure 2.4).

A mixed excitation can be used for STRAIGHT [Kaw06]. To generate signals with a mixed excitation, an aperiodicity index for each spectrum component is estimated. It is computed as a ratio of lower and higher spectral envelopes connecting all the valley points and all the peaks of the spectrum, respectively. The index ranges from –60 dB to 0 dB, –60 dB referring to a totally voiced and 0 dB to a totally unvoiced segment. An example of STRAIGHT aperiodicity map is shown in Figure 3.3 with 5-band average parameterization.

Table 3.1 gives an overview of the information provided by STRAIGHT and the parameterization that is used in [P5, P6, P7], with [P5, P7] or without [P6] the aperiodicity map. The parameters are updated at 5 ms steps.

In speech coding, a perceptual weighting filter [Hua01, Koi95] is used to shape the noise spectrum in order to hide it under the speech signal. The postfiltering procedure helps in emphasizing the formants and it can also be applied to converted speech features, that tend to become oversmoothed. Postfiltering is used in the sample generation for the listening tests in [P2, P3, P5, P6, P7].

Table 3.1: The common parameterization used for STRAIGHT features. The number of elements is shown in parenthesis for each feature in a frame.

STRAIGHT	Parameterization
Spectrum (513)	24^{th} order MCCs (25)
F_0 (1)	Logarithmic F_0 (1)
Aperiodicity map (513)	Band averages (5)

The correct estimation of the pitch plays a major role in A/S systems. A variety of techniques exist for pitch estimation. The main principle of time-domain algorithms is to find the pitch period by comparing the similarity between the original signal and its shifted version in the region of interest [Hua01]. Frequency-domain methods, such as harmonic peak detection and spectrum similarity methods, operate directly on the speech spectrum and are computationally more complex. The combination of time- and frequency-domain algorithms (e.g. [McA90]) has become popular due to the growing interest in sinusoidal speech coders.

In practice, A/S systems contain a lot of heuristics for pitch and voicing estimation. Problems in the estimation degrade the re-synthesis quality. The performance is rather dependent on the conditions. For example STRAIGHT [Kaw99a] gives excellent results on clean speech and modal voices but may fail with creaky non-modal voices [Sil09]. Codecs designed for speech coding purposes, on the other hand, need to operate also on noisy conditions, but may not offer high quality.

3.2 Parallel Corpus Alignment

The same speech utterance produced by two speakers is rarely realized at the same speaking rate. The utterances must thus be aligned in time in order to preserve linguistic correspondence. The simplest way for obtaining the alignment is linear normalization [Rab93]. It is based on the assumption that speaking rate variation is proportional to the duration of the utterance and independent of individual sounds. Linear normalization performs reasonably well for mono-syllabic words but not for poly-syllabic words [Whi76]. Thus satisfactory results can not be expected when aligning utterances that consist of multiple words.

The most popular approach for aligning two sentences at the frame-level is dynamic time warping (DTW). DTW is discussed in more detail in the following subsections. Other alternatives for parallel data include for example HMM-based alignments [Err10b] or sentence HMMs [Ars99]. If phoneme boundaries are available from manual alignment or speech recognizer, these can be used as anchor

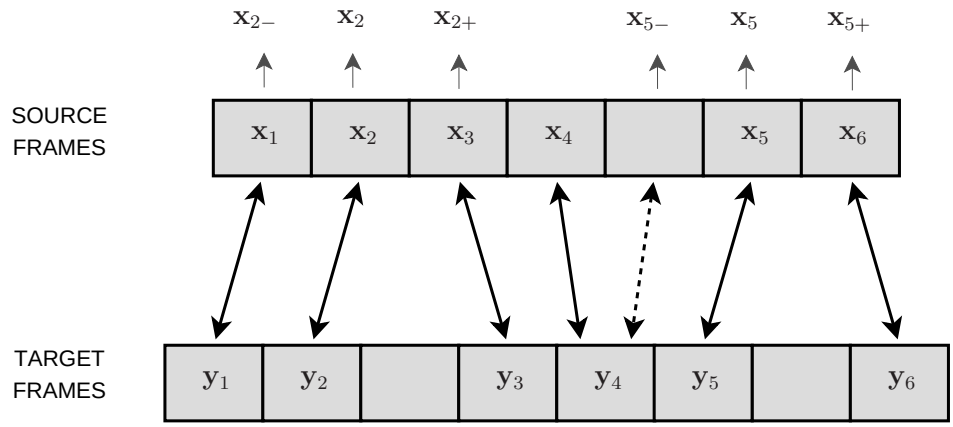


Figure 3.4: Overview of the training data alignment and selection procedure.

points for the alignment algorithm. In Chapter 1, also some non-parallel data alignment methods were referred but they are beyond the scope of the thesis.

As a result of the alignment, each source frame is ideally matched with exactly one target frame. With DTW, multiple source frames may become matched with a certain target frame or vice versa, but the best pair can be selected based on e.g. the minimum distance.

An example of alignment and training data gathering is shown in Figure 3.4. The figure also illustrates the notation; index n denotes the selected training data and the indices of the previous and the next frame of n are denoted by n_- and n_+, respectively. In Figure 3.4, the fifth frame of the source is not included in the training data, since it was matched with the same target frame as the fourth source frame.

3.2.1 Dynamic Time Warping

The idea of DTW is to obtain an optimal alignment between two sequences in terms of a distance function. DTW is a dynamic programming algorithm for time-aligning two feature sequences $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{T_x}\}$ and $\mathcal{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_{T_y}\}$.

Before conducting the alignment, a distance matrix between source and target features is calculated. A common choice is to use Euclidean distance as a distance measure and MFCCs as alignment features as was done in [P3]. MFCCs can be augmented with their first-order dynamics (deltas). The alignment features can be different from the features used in the conversion. In this thesis MCCs are used as spectral conversion features. They perform similarly to MFCCs and were used as alignment features in [P6, P7].

Some restrictions and constraints on the warping function are necessary in order to provide a meaningful comparison between two speech feature sequences:

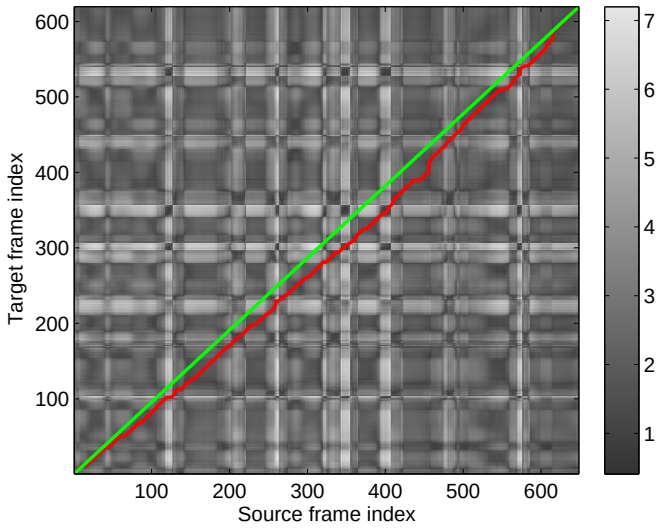


Figure 3.5: Example of a distance matrix and alignment given by DTW (red line) and linear interpolation (green line).

endpoint constraints, monotonicity constrains, local and global continuity constraints, and slope weighting [Rab93].

For local constraint of type I, the following is evaluated at each point of a grid

$$\varphi(i,j) = d(\mathbf{x}_i, \mathbf{y}_j) + \min(\zeta\varphi(i-1,j-1), \varphi(i,j-1), \varphi(i-1,j)) \tag{3.1}$$

where $\varphi(i,j)$ is the cumulative distance of point (i,j), $d(\mathbf{x}_i, \mathbf{y}_j)$ is the distance at the current point, and ζ is a positive slope weighting factor. The total cost of the optimal alignment is returned by $\varphi(T_x, T_y)$ and the optimal alignment can be obtained by path backtracking.

A classic DTW operates on an equally weighted case (ζ=1 in (3.1)). This makes the path bias towards the diagonal, since the cost of one cell corresponds to the cost of two cells (a horizontal and a vertical step) [Mül07]. In speech recognition, ζ is typically set to 2 [Rab93] that gives equal emphasis on performing one diagonal step or combining a horizontal and a vertical step. However, in this thesis ζ is set to $\sqrt{2}$ in order to give a slightly more weight on diagonal transitions.

An example of alignment given by DTW (ζ=$\sqrt{2}$ in (3.1)) and alignment obtained by linear normalization are shown in Figure 3.5 together with the distance matrix. The silent segments at the beginning and at the end were removed before obtaining the alignments. The distance matrix was calculated from MCCs that give similar results to MFCCs.

It is somewhat paradoxical that in VC, one attempts to capture the differences between the source and the target speaker characteristics but the alignment process is guided by global minimization of the differences between the source

and the target features [P3]. One alternative to compensate the differences between different speakers was proposed in [P6]: first the alignment was carried out using the original features and a conversion function was estimated using PLS regression (Section 5.2) between the aligned features for the sentence. The source features of the sentence were then converted to mimic the target. The alignment process was repeated for aligning target features with the converted source features. A computationally more complex approach was employed in [Sty98]. Instead of applying a simplified mapping function calculated for each sentence, alignments from all sentences were used to estimate a similar mapping function as for the final conversion phase. The process was repeated until no changes in the alignments occurred.

3.2.2 The Effect of Alignment

Both the training and objective evaluation of a conventional VC system rely on parallel data. However, the effect of alignment on voice conversion performance has not been addressed before [P3].

In [P3], three main alignment cases were compared: alignment with manual labels, alignment given by DTW, and linear normalization based alignment. In the first case, the alignment was made based on manually labeled phoneme boundaries. The boundaries were used as anchor points and it was assumed that the "correct" alignment goes through these labels.

In objective alignment evaluation, several DTW configurations were tested related to forcing the endpoints, different local constrains, data removal and the use of simple voice activity detection (VAD) [P3]. The objective alignment accuracy was measured by comparing the starting time of each phoneme given by a certain alignment scheme to the corresponding starting time given by the manual labels. The misalignment times were averaged over all phonemes and speaker pairs. In addition, the percentage of misalignments greater than 20 ms, 50 ms, and 100 ms were calculated in [P3].

A VC system was built based on alignments given by multiple alignment schemes and spectral distortion values were calculated. In a listening test, a VC system based on alignments given by DTW was compared to a voice conversion system based on alignments given by the manually annotated labels and alignments provided by linear interpolation. The DTW configuration in the listening test exploited simple VAD, forced endpoints and data removal as suggested by the objective results. In data removal, unvoiced-voiced pairs were discarded as well as pairs where at least one of the frames was assumed to be silent.

The conclusions according to the alignment are summarized as follows:

- As expected, the converted speech quality was degraded if linear interpolation was used. This applied both to the subjective quality and the objective measures.

- DTW endpoint constraints are beneficial in most of the cases whereas the use of a specific global or local constraint is not important.
- When using endpoint constraints, the presence of the silence segments at the beginning and at the end of utterances (endpoints) must be taken into account. Either both speakers should have a small amount of silence at endpoints or the silences should be removed using simple VAD techniques.
- DTW provides a globally optimal solution and thus non-optimal local pairs can occur. Removing for example pairs that include silence is likely to improve the effectiveness of a conversion function.
- The use of DTW instead of manual labels did not degrade subjective quality and the spectral distortion was degraded only about 0.05 dB.

3.3 Evaluation Methods

"Speech quality is a complex psychoacoustic outcome of the human perception process" [Ben08]. The quality of a speech signal is necessarily a subjective measure. The most straightforward way of measuring the quality is to have a group of people listening to a speech sample and rating its quality. This is costly and time consuming, and therefore objective measures have been developed. However, subjective tests should always be used when determining the final quality.

There are several difficulties related to objective quality evaluation in VC. First of all, for the testing data, the objective results are typically based on the alignment which is not perfect. Furthermore, numbers may not express the true information on the overall sound quality. The errors are usually calculated on frame-by-frame basis which does not take the temporal continuity into account. In some cases, e.g. crosslingual VC, there is no parallel data available for testing.

In addition to quality evaluation, the success of speaker identity transformation must be assessed. There are no well-established tests for evaluating the success of identity conversion. In this section, the commonly used objective quality measures and subjective evaluation tests are revised and discussed.

3.3.1 Objective Performance Measures

The performance is objectively measured by comparing the converted speech features to the original target features, if available. The error criterion should follow the human perception.

In speech coding, the success of the coding process is related to spectral distortion measure. The spectral distortion measure for LSFs and MCCs, the most widely used features for representing the spectrum in VC, are given below.

The spectral distortion (SD) between two LP spectra S_1 and S_2 is defined as

$$SD = \sqrt{\frac{1}{f_u - f_l} \int_{f_l}^{f_u} [10 \log_{10}(S_1(e^{\frac{j2\pi f}{f_s}})) - 10 \log_{10}(S_2(e^{\frac{j2\pi f}{f_s}}))]^2 \, \mathrm{d}f} \tag{3.2}$$

where f_l and f_u denote the lower and upper frequency limits and f_s is the sampling frequency. Equation (3.2) is mainly used for narrowband speech, say for a frequency range 125–3100 Hz [Kon04]. In practice the integral (3.2) is approximated by sampling the power spectrum with N points. Spectral distortion was used as a performance measure in [P3, P4].

Although the power spectrum is approximated by sampling, the computational complexity is still rather high. A lower computational complexity can be achieved by basic MSE techniques that are modified to take the perceptual effects into account. A weighted MSE between the original LSF vector $\mathbf{f}^{\mathrm{lsf}}$ and the converted LSF vector $\hat{\mathbf{f}}^{\mathrm{lsf}}$ is given by

$$\mathrm{d}(\mathbf{f}^{\mathrm{lsf}}, \hat{\mathbf{f}}^{\mathrm{lsf}}) = \sum_{i=1}^{p} w_i (f_i^{\mathrm{lsf}} - \hat{f}_i^{\mathrm{lsf}})^2 \tag{3.3}$$

where w_i is the weighting factor for i^{th} LSF. Different weighting schemes for narrow-band speech have been proposed, e.g. [Pal93]. More weight is put on the first LSFs, and LSFs that are close to each other indicating a formant.

The Mel-cepstral distortion (MCD) between the converted target and the original target is calculated as [Tod07a]

$$\mathrm{sd}^{\mathrm{mel}}[\mathrm{dB}] = \frac{10}{\ln 10} \sqrt{2 \sum_{d=1}^{D} (c_d - \hat{c}_d)^2} \tag{3.4}$$

where c_d and $\hat{c}_d$ are the d^{th} MCC of the original and converted target, respectively, and D is the order of MCCs. The first (0^{th}) MCC term is not included in (3.4), since it describes the energy of the frame and is usually copied from the source. The MCD is a widely used perceptual objective error criterion in many statistical speech synthesis and voice conversion studies. It was used to assess the spectral conversion performance in [P5, P6, P7].

Usually an averaged spectral distortion value is given, i.e. the results are averaged over all converted frames. However, the human hear is sensitive to occasional large abrupt changes. In speech coding, the quality criterion also defines what kind of outliers and how many of them are accepted. In [Pal93], the following criteria were used to define transparency: 1) the average SD is less than 1 dB, 2) there are no outlier frames having SD above 4 dB, and 3) less than 2% of frames have SD in the range from 2 to 4 dB. In [P4], these quality criteria were used to examine the upper limit for the use of real target data frames.

3.3.2 Identity Evaluation Using a Speaker Recognition System

A speaker identification system can be used to objectively assess the success of the identity transformation. If the target speaker is identified correctly, the conversion is considered successful. The benefit is that no parallel data is required.

A simple speaker identification system can be built as described in [Rey95]. A GMM is trained for both the source and the target speaker. A converted utterance is recognized to be spoken by the speaker whose model produces the highest cumulative likelihood for the utterance. On the other hand, the binary recognition result does not imply that the converted sentence would be close to the target, it only tells that it was *closer* to the target than the source. A more interesting case is to have models from a variety of male and female speakers available among which the selection is made. In [Far10] the selection was made among four speakers. In [Mou06] and [P7], twelve speakers were used.

In addition to a binary recognition result, a continuous objective measure is defined [Ars99]

$$\theta_{st} = \log \frac{p(\hat{\mathbf{y}}|\lambda_{\text{tgt}})}{p(\hat{\mathbf{y}}|\lambda_{\text{src}})} \tag{3.5}$$

where λ_{tgt} and λ_{src} are the GMMs for the target and the source speaker, respectively and $\hat{\mathbf{y}}$ is the converted target. A high value of θ_{st} indicates a good system whereas a negative value indicates that the conversion result is closer to the source. The measure was used in [P7]. It can be noted that Equation (3.5) is a log-likelihood ratio that derives from the Bayes classifier. Assuming perfect estimation of the class densities, equal class priors and symmetrical misclassification costs, the value θ=0 gives the Bayes optimal threshold for binary classification. The log-likelihood ratio is used in most speaker verification systems, e.g. [Rey00].

3.3.3 Subjective Evaluation

A number of rating procedures exists for assessing the speech quality in a listening test. They can be divided into absolute category rating (ACR), comparison category rating (CCR), and degradation category rating tests [Ben08]. The first two are used in VC evaluations. The results are presented as a mean score or an average number of votes obtained by each system. Furthermore, 95% confidence intervals on the mean are often given to allow assessment of whether the differences between different systems are significant.

A mean opinion score test (MOS) used in [P7] is an ACR test. It is the most common procedure for quality testing in telecommunications. Listeners rate a sample using a five-level scale (1=Bad, 2=Poor, 3=Fair, 4=Good, 5=Excellent). The average of all scores obtained by a particular system is the system's MOS.

In addition to quality, the success of identity conversion is evaluated. Identity can be assessed for example using a five-level scale from 1 denoting for definitely different to 5 denoting for definitely identical [Err10b]. The listeners must initially hear some examples from the target speaker.

In speech coding, a codec receiving a MOS of 3.5–4.0 is interpreted natural and adequate for telecommunications and a MOS over 4 is near transparent. This kind of interpretation cannot be used in evaluating a TTS system [Hua01] nor a VC system. Moreover, comparing the results from different MOS tests is difficult. Hence, the comparison of MOS obtained by different methods in different studies is not meaningful. On the other hand, the MOS test is suitable for ranking different TTS systems [Hua01] and presumably also different VC systems.

A drawback of a MOS test is that the difference between systems may be small. An alternative is to use a CCR test. In its simplest form, a listener is asked to choose a preference between two samples generated by two different systems. This test procedure was used in [P5, P6]. Also there can be an alternative that the samples are equal as it was used in [P2, P3]. An extended version is to judge the quality of the second sample relative to the first sample in seven categories (3=Much better, 2=Better, 1=Slightly better, 0=About the same, –1=Slightly worse, –2=Worse, –3=Much worse).

In an ABX test, a listener hears three samples, A, B, and X. The listener is asked to choose which one of the samples, A or B, is closer to X. This kind of test can be used to assess the identity as it was done in [P7]. A and B include samples from the original source and target speakers and X is the converted sample. If X is closer to the target speaker, the system is successful. The sentence in X can be the same as in A and B, but different prosodies may affect the result. In most inter-gender transformations, the desired result is easy to obtain. In intra-gender transformation, on the other hand, listeners may have difficulties even without any modifications: the recognition rate was about 70% when X contained an original sample of a different sentence from either the source or the target [Tod07a].

3.3.4 Evaluating Prosody Conversion

The evaluation of prosody conversion is difficult. Some studies such as [Lol08] use only objective error criteria. However, objective measures such as RMSE of the converted F_0 compared to the original target F_0 can be misleading, since there is not only one type of acceptable prosody. The same person can utter the same sentence at different prosody at different times. The aim of prosody conversion could rather be to generate "believable" prosody for the target speaker [P2]. Thus, a listening test should always be used in final prosody conversion evaluation. The subjects can be asked to choose the one of two sample files that mimicked a certain speaker's prosody the best [P2].

Prosody is likely to play a more important role when the listeners know the speaker. So far, VC system evaluations have not specifically considered speakers that are familiar to the listeners.

3.4 Summary

A high-quality analysis/synthesis system is essential for successful voice conversion. Spectral envelope as well as F_0 and durations should be easily modifiable. Many voice conversion algorithms assume that the data is aligned before estimating the mapping function, but the effect of parallel data alignment has not been studied before [P3]. Parallel data is most often aligned with DTW that performs reasonably well when considering silence parts and end points. There are no well-established techniques for assessing voice conversion systems. Identity and quality are typically evaluated in separate tests. How to weight these two goals has not been considered in the field.

Chapter 4

Mapping Techniques

SPEAKER identity should be converted as accurately as possible while maintaining high speech quality. Due to the limited amount of training data, most voice conversion systems learn statistical transformation functions from a set of frame-level paired feature vectors. For feature transformation, a conversion function maps the source feature vector $\mathbf{x}_n$ (size Dx1) into the target feature vector $\mathbf{y}_n$ (size Dx1) for each frame n.

In most VC studies, the aim is to find a conversion function $\mathcal{F}(\cdot)$ that minimizes the prediction error ϵ over all N pairs of training samples as

$$\epsilon = \sum_{n=1}^{N} \|\mathbf{y}_n - \mathcal{F}(\mathbf{x}_n)\|^2 \tag{4.1}$$

Before presenting statistical techniques for spectral mapping, the use of real target data is considered in Section 4.1 and approaches based on frequency warping are shortly reviewed in Section 4.2. The techniques presented in sections 4.3, 4.6, 4.4, and 4.7 can be applied to any speech features, but they are reviewed in the context of spectral features. Excitation and prosody (mainly F_0) conversion are considered in sections 4.8 and 4.9, respectively. The spectral envelope energy is assumed to be copied from the source.

4.1 The Use of Real Target Data

A perfect identity conversion could be achieved by using real speech data from the target, so why not to choose target speech frames in a unit selection manner? Sündermann et al. [Sün06a] used real target data in text-independent VC. An optimal target frame sequence was searched from the target database using a cost function based on an acoustic cost and a join cost similarly to unit selection speech synthesis. The acoustic cost was the distance between the source frame to be converted and a target frame candidate and the join cost accounted for

Table 4.1: The mean spectral distortion using 5, 10, 20, 50, and 100 training sentences for covering the acoustic feature space of a speaker with the speaker's own data and the amount of outliers.

	5	10	20	50	100
Mean SD [dB]	2.23	2.00	1.80	1.59	1.46
2 dB outliers [%]	58.5	46.1	34.7	21.4	14.0
4 dB outliers [%]	2.63	0.95	0.38	0.10	0.04

the continuity between consecutive target frame candidates. This technique may result in finding frames that are closest to the source. Dutoit et al. [Dut07] first converted LSF features using a conventional GMM-based approach and then searched from the target speech database for the closest match to the converted LSF vector in order to obtain more "realistic" target LSFs.

Obviously the selection process in [Sün06a] or the conversion process in [Dut07] affects the result. The upper limit for frame-based selection approaches was examined in [P4]. Provided that the selection process is perfect, we investigated whether it would be possible to cover a speaker's acoustic space (10-dimensional narrowband LSF vectors) at transparent quality. The definition of transparency was extracted from speech coding and it was described in Section 3.3.1. The results from [P4] are summarized in Table 4.1 for seven speakers from CMU ARCTIC database [Kom03]. It can be concluded that the database sizes of 5–100 sentences are not enough for representing the speaker's acoustic space at transparent quality even in the case of perfect selection and narrowband clean data.

The results obtained in statistical VC mapping techniques are far from transparent, but guiding the frame selection process is not straightforward and can cause major errors. Furthermore, if there is a large amount of data, one may consider building a unit selection speech synthesis voice without voice conversion.

4.2 Warping-Based Approaches

The idea of frequency warping is to find an optimal warping function that is used to warp the frequency axis of a pair of amplitude spectra in a way that the spectral distance between them is minimized. In speech recognition, vocal tract length normalization (VTLN) can be used to compensate for the effects of different vocal tract lengths. VTLN is usually limited to a single parameter α. Different warping functions related to α can be established and their use in VC was studied in [Sün03], but dynamic or non-linear warping of the frequency axis is more common in VC [Val92, Pri06, Shu06, Err10b].

Non-linear frequency warping can be based on finding a mapping function that minimizes the distance between two spectra [Val92]. Alternatively, a mapping function can be estimated based on formants [Shu06, Err10b]. Formants were found manually in [Shu06] and automatically in [Err10b]. The spectral envelopes associated with the mean vectors of a joint density GMM were found useful for extracting warping functions in [Err10b]. The source spectrum was warped with a warping function that was a weighted sum of different warping functions extracted for each Gaussian.

The main benefit of warping-based approaches is that they maintain good speech quality without oversmoothing effects. Nevertheless, the simple reallocation of formants on the new frequency axis does not provide proper identity transformation. This can be adequate if only certain voice characteristics such as general gender-transformation or adult-to-child transformation is considered as in [Pri06]. For obtaining a specific target, also formant intensity, bandwidth and the spectral tilt must be modified [Err10b]. In [God11], amplitude scaling functions for the spectral envelope were extracted for each acoustic class. Speaker identity can be improved by combining the frequency-warped source spectra with parts of the target spectra selected from the training data [Shu08].

The problems of warping-based approaches arise in preserving the shape of modified spectral peaks and controlling the bandwidths of close formants. Proper controlling of the formant amplitudes is also challenging.

Figure 4.1 shows an example of a 16^{th} order LP spectrum from two speakers, a female (red line) and a male (blue line), extracted from the vowel of the word "this". The sampling frequency is 16 kHz. The formants (the peaks) are clearly visible for both speakers in this case. In addition, Figure 4.1 includes the warped male spectrum (blue dashed line) with a scalar warping factor of 1.19. The warping factor was obtained by finding the slope between the first five formant positions of the speakers. Although warping can match reasonably well the formant locations of the two speakers, the relative formant heights are different.

4.3 Mapping Codebooks

One of the earliest VC systems was proposed in [Abe88]. The idea is to vector quantize the source and target feature vectors with their own codebook and then find a mapping codebook between the two codebooks based on DTW-aligned source and target features. This was accomplished by creating a two-dimensional histogram based on the codevector correspondence of the source-target pairs. The target codebook was a linear combination of the target codevectors, using the histogram as a weighting function. The method is straightforward and can capture the speaker identity well, but it suffers from frame-to-frame discontinuities and poor prediction capability on new data.

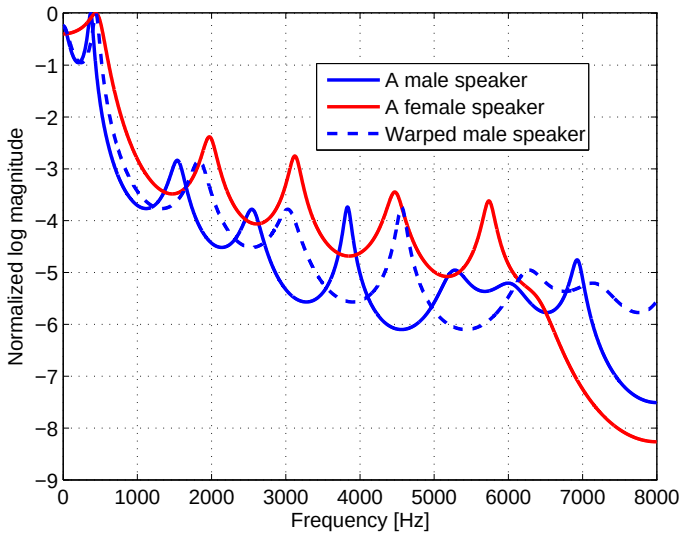


Figure 4.1: The logarithmic normalized LP spectrum of a male (blue line) and a female speaker (red line) and a warped spectrum of the male (blue dashed line).

Arslan [Ars99] tackled the problem of discrete representation of the acoustic space by using a weighted sum of target codewords. In order to convert a source vector, a set of weights is determined depending on a similarity measure between the source vector and the set of centroids in the source codebook. The conversion is realized by using the weights to linearly combine the corresponding centroids in the target codebook. While improving the continuity with respect to the basic codebook approach, the method is subject to oversmoothing by summing over a range of spectral envelopes. Further improvements are described in [Tur06].

In addition to codebooks containing direct source-target features pairs, a codebook can provide continuous mapping inside a cluster. In [Val92] and [P5], different linear transformation functions for different classes/clusters were used. In the local linear transformation approach [Pop12], each spectral vector was converted with an individual linear transformation determined in the least squares sense from a subset of nearby codewords. The approaches in [Val92, Pop12] can alleviate the oversmoothing effect but are susceptible for discontinuities. Trellis structured vector quantization [Esl11] deals with the problem of discontinuities by using dynamic programming to find the optimal sequence of target codewords.

4.4 Gaussian Mixture Model Based Mapping

A GMM is a weighted sum of M Gaussian components for modeling the distribution $p(\mathbf{x})$ as

$$p(\mathbf{x}) = \sum_{m=1}^{M} \boldsymbol{\alpha}_m \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_m^{(x)}, \boldsymbol{\Sigma}_m^{(x)}) \tag{4.2}$$

where $\boldsymbol{\alpha}_m$ is the prior probability of the m^{th} Gaussian component and the term $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_m^{(x)}, \boldsymbol{\Sigma}_m^{(x)})$ denotes for the multivariate normal distribution with mean vector $\boldsymbol{\mu}_m$ and covariance matrix $\boldsymbol{\Sigma}_m$, i.e.

$$\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_m^{(x)}, \boldsymbol{\Sigma}_m^{(x)}) = \frac{1}{(2\pi)^{D/2}\sqrt{|\boldsymbol{\Sigma}_m|}} \exp\left[-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_m^{(x)})^{\mathrm{T}} \boldsymbol{\Sigma}_m^{-1} (\mathbf{x} - \boldsymbol{\mu}_m^{(x)})\right] \tag{4.3}$$

where superscript T denotes for transpose.

The most popular way to estimate the GMM parameters is the expectation maximization (EM) algorithm [Dem77] that has been also been used in the experiments of this thesis.

4.4.1 Source GMM

Stylianou et al. [Sty98] proposed to fit a GMM to the source feature vectors and then estimate a conversion function. The conversion function between the source and the target data is assumed to be linear for each Gaussian and is of the form

$$\mathcal{F}(\mathbf{x}_n) = \sum_{m=1}^{M} \omega_{m,n}(\boldsymbol{\beta}_m \mathbf{x}_n + \mathbf{b}_m) \tag{4.4}$$

where $\boldsymbol{\beta}_m$ is a linear transform matrix, $\mathbf{b}_m$ is a bias term for cluster m and

$$\omega_{m,n} = \frac{\boldsymbol{\alpha}_m \mathcal{N}(\mathbf{x}_n; \boldsymbol{\mu}_m^{(x)}, \boldsymbol{\Sigma}_m^{(x)})}{\sum_{i=1}^{M} \boldsymbol{\alpha}_i \mathcal{N}(\mathbf{x}_n; \boldsymbol{\mu}_i^{(x)}, \boldsymbol{\Sigma}_i^{(x)})} \tag{4.5}$$

The solution for $\boldsymbol{\beta} = [\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \ldots, \boldsymbol{\beta}_m, \ldots, \boldsymbol{\beta}_M]$ can be found using a least squares approach

$$\boldsymbol{\beta} = ((\mathbf{X}\mathbf{X}^{\mathrm{T}})^{-1}\mathbf{X}\mathbf{Y}^{\mathrm{T}})^{\mathrm{T}} \tag{4.6}$$

where

$$\mathbf{X} = [\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n, \ldots, \mathbf{X}_N] \tag{4.7}$$

$$\mathbf{Y} = [\mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_n, \ldots, \mathbf{Y}_N] \tag{4.8}$$

$$\mathbf{X}_n = [\omega_{1,n}\mathbf{x}_n^{\mathrm{T}}, \omega_{2,n}\mathbf{x}_n^{\mathrm{T}}, \ldots, \omega_{M,n}\mathbf{x}_n^{\mathrm{T}}]^{\mathrm{T}} \tag{4.9}$$

$$\mathbf{Y}_n = \mathbf{y}_n \tag{4.10}$$

Note that the bias terms are excluded from the solution. Before extracting $\boldsymbol{\beta}$ (4.6), both $\mathbf{X}$ and $\mathbf{Y}$ are mean-centered.

In practice, the pseudoinverse of $\mathbf{X}\mathbf{X}^{\mathrm{T}}$ in (4.6) should be used if the number of Gaussians is set rather high compared to the size of the training data. Without the pseudoinverse, the inverse matrix becomes non-positive definite and inaccurate results over new data are obtained. In [Sty98], each feature dimension is transformed independently if covariance matrices of the source GMM are diagonal. Additionally, different scaling of variables is used in [Sty98], but the main idea is similar to (4.4).

4.4.2 Joint Density GMM

Kain and Macon [Kai98] proposed to model the joint density of the source and the target features with a GMM. The source features are augmented with the corresponding target features as $\mathbf{z}_n = [\mathbf{x}_n^{\mathrm{T}}, \mathbf{y}_n^{\mathrm{T}}]^{\mathrm{T}}$, and $\mathbf{z}_n$ is modeled by a GMM as

$$p(\mathbf{z}_n) = \sum_{m=1}^{M} \boldsymbol{\alpha}_m \mathcal{N}(\mathbf{z}_n; \boldsymbol{\mu}_m^{(z)}; \boldsymbol{\Sigma}_m^{(z)}) \tag{4.11}$$

where

$$\boldsymbol{\mu}_m^{(z)} = \begin{bmatrix} \boldsymbol{\mu}_m^{(x)} \\ \boldsymbol{\mu}_m^{(y)} \end{bmatrix} \tag{4.12}$$

$$\boldsymbol{\Sigma}_m^{(z)} = \begin{bmatrix} \boldsymbol{\Sigma}_m^{(xx)} & \boldsymbol{\Sigma}_m^{(xy)} \\ \boldsymbol{\Sigma}_m^{(yx)} & \boldsymbol{\Sigma}_m^{(yy)} \end{bmatrix} \tag{4.13}$$

When partitioning the joint density into $p(\mathbf{z}_n) = p(\mathbf{y}_n|\mathbf{x}_n)p(\mathbf{x}_n)$, both densities $p(\mathbf{y}_n|\mathbf{x}_n)$ and $p(\mathbf{x}_n)$ are also multivariate Gaussian [Mar79]. This also applies to finite Gaussian mixtures. The joint density can be written as

$$p(\mathbf{z}_n) = \sum_{m=1}^{M} \omega_{m,n} \mathcal{N}(\mathbf{y}_n|\mathbf{x}_n; \mathbf{E}_{m,n}^{(y)}, \mathbf{D}_m^{(y)}) \mathcal{N}(\mathbf{x}_n; \boldsymbol{\mu}_m^{(x)}; \boldsymbol{\Sigma}_m^{(xx)}) \tag{4.14}$$

where $\mathbf{E}_{m,n}^{(y)}$ and $\mathbf{D}_m^{(y)}$ are conditional mean and covariance matrix, respectively and are given by [Mar79]

$$\mathbf{E}_{m,n}^{(y)} = \boldsymbol{\mu}_m^{(y)} + \boldsymbol{\Sigma}_m^{(yx)} \boldsymbol{\Sigma}_m^{(yy)^{-1}} (\mathbf{x}_n - \boldsymbol{\mu}_m^{(x)}) \tag{4.15}$$

$$\mathbf{D}_m^{(y)} = \boldsymbol{\Sigma}_m^{(yy)} - \boldsymbol{\Sigma}_m^{(yx)} (\boldsymbol{\Sigma}_m^{(xx)})^{-1} \boldsymbol{\Sigma}_m^{(xy)} \tag{4.16}$$

The converted target $\hat{\mathbf{y}}_n$ can be obtained based on the minimum mean squared error (MMSE) solution [Kai98] that is the same as the mean vector of the conditional distribution $p(\mathbf{y}_n|\mathbf{x}_n)$, i.e.,

$$\hat{\mathbf{y}}_n = \sum_{m=1}^{M} \omega_{m,n} \mathbf{E}_{m,n}^{(y)} = \sum_{m=1}^{M} \omega_{m,n} [\boldsymbol{\mu}_m^{(y)} + \boldsymbol{\Sigma}_m^{(yx)} (\boldsymbol{\Sigma}_m^{(xx)})^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_m^{(x)})] \tag{4.17}$$

Assuming diagonal covariance matrices $\mathbf{\Sigma}_m^{(xx)}$, $\mathbf{\Sigma}_m^{(xy)}$, $\mathbf{\Sigma}_m^{(yx)}$, and $\mathbf{\Sigma}_m^{(yy)}$ in (4.13), all feature dimensions are transformed independently from each other.

The joint density model has been more popular than the source GMM. The selection of the covariance type has a major impact on the resulting conversion function. In the literature, both diagonal and full covariance matrix GMMs are used. Full covariance matrices are slightly more common, but for maximum likelihood GMM approach (Section 4.5.1) diagonal covariance matrices are typically used. The problem of setting the covariance type is discussed in Section 4.5.3.

4.5 Problems of GMM-Based Conversion and Improvements Proposed in the Literature

The control of model complexity is a crucial issue when learning a model from data. There is a trade-off between two objectives: the generalization of the model on unseen data and fidelity. This trade-off problem, also referred to as bias-variance dilemma [Gem92], is common for all model fitting tasks, such as GMM-based VC. Simple models tend to have a low variance but a high bias and they are subject to oversmoothing. On the other hand, the use of complex models may result in overfitting meaning that the resulting model does not generalize well for unseen testing data, i.e. has too high variance.

In addition to oversmoothing and overfitting, a major problem in conventional GMM-based conversion is that it ignores the temporal correlation of speech features. The time-independent mapping problem is also common for other techniques such as the most mapping codebook approaches.

GMM-based VC, especially the joint density model (Section 4.4.2), has been a dominating technique despite of these problems. Many studies aim at improving the core GMM techniques and/or use them as a benchmark method. In this section, the problems are reviewed together with improvements proposed to overcome them.

4.5.1 Time-Independent Mapping

Speech features exhibit strong temporal correlation, but in conventional GMM-based conversion, features in each frame are transformed independently from the neighboring frames. This can lead to discontinuities in feature trajectories and thus perceptual speech quality degradation.

An alternative to MMSE solution (4.17) is to use a maximum likelihood (ML) criterion [Tod07a]. The MMSE solution ignores the conditional covariance matrices (4.16). If conversion is carried out on a frame-by-frame basis, the MMSE and ML results are identical, but in [Tod07a] the ML estimation of the converted *trajectory* is proposed. The static source and target vectors in the joint density

model are augmented with dynamics, i.e. the first-order deltas. The resulting variables are

$$\mathbf{X}_n = [\mathbf{x}_n^{\mathrm{T}}, \Delta\mathbf{x}_n^{\mathrm{T}}]^{\mathrm{T}} \tag{4.18}$$

$$\mathbf{Y}_n = [\mathbf{y}_n^{\mathrm{T}}, \Delta\mathbf{y}_n^{\mathrm{T}}]^{\mathrm{T}} \tag{4.19}$$

where a standard configuration for the dynamic features is

$$\Delta\mathbf{x}_n = -0.5\mathbf{x}_{n-} + 0.5\mathbf{x}_{n+} \tag{4.20}$$

$$\Delta\mathbf{y}_n = -0.5\mathbf{y}_{n-} + 0.5\mathbf{y}_{n+} \tag{4.21}$$

A GMM is estimated over the joint density of $\mathbf{Z}_n = [\mathbf{X}_n^{\mathrm{T}}, \mathbf{Y}_n^{\mathrm{T}}]^{\mathrm{T}}$ similarly to the case of not using the dynamic features. The parameter set of the GMM is denoted by $\boldsymbol{\lambda}^{(Z)}$, which consists of weights, mean vectors, and the covariance matrices for each Gaussian component.

The converted static vectors are obtained separately for each utterance by maximizing the following

$$\hat{\mathbf{y}} = \arg\max_{\mathbf{y}} p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\lambda}^{(\mathrm{Z})}) \tag{4.22}$$

Since the explicit relationship between the static and dynamic features is known, Equation (4.22) is maximized subject to $\mathbf{Y} = \mathbf{W}\mathbf{y}$. The matrix $\mathbf{W}$ for transforming the static feature sequence into static and dynamic features contains delta coefficient weights and zeros as

$$\mathbf{W} = [\mathbf{W}_1, \mathbf{W}_2, \ldots, \mathbf{W}_t, \ldots, \mathbf{W}_L]^{\mathrm{T}} \tag{4.23}$$

$$\mathbf{W}_t = [\mathbf{w}_t^{(0)}, \mathbf{w}_t^{(1)}] \tag{4.24}$$

$$\mathbf{w}_t^{(0)} = [\overset{1^{st}}{\mathbf{0}_{D\mathrm{x}D}}, \quad \ldots, \quad \overset{(t-1)^{th}}{\mathbf{0}_{D\mathrm{x}D}}, \quad \overset{t^{th}}{\mathbf{I}_{D\mathrm{x}D}}, \quad \overset{(t+1)^{th}}{\mathbf{0}_{D\mathrm{x}D}}, \quad \ldots, \quad \overset{L^{th}}{\mathbf{0}_{D\mathrm{x}D}}]^{\mathrm{T}} \tag{4.25}$$

$$\mathbf{w}_t^{(1)} = [\overset{1st}{\mathbf{0}_{D\mathrm{x}D}}, \quad \ldots, \quad \overset{(t-1)^{th}}{-0.5\mathbf{I}_{D\mathrm{x}D}}, \quad \overset{t^{th}}{\mathbf{0}_{D\mathrm{x}D}}, \quad \overset{(t+1)^{th}}{0.5\mathbf{I}_{D\mathrm{x}D}}, \quad \ldots, \quad \overset{L^{th}}{\mathbf{0}_{D\mathrm{x}D}}]^{\mathrm{T}} \tag{4.26}$$

where L is the number of frames in the utterance, and $\mathbf{0}_{D\mathrm{x}D}$ and $\mathbf{I}_{D\mathrm{x}D}$ are a zero and an identity matrix of size $D\mathrm{x}D$, respectively. The size of $\mathbf{W}$ is $2DL\mathrm{x}DL$.

The converted static sequence (4.22) can be obtained by defining an auxiliary function and using the EM algorithm for maximization or using a suboptimum mixture component sequence. In this thesis, only the suboptimum sequence solution is reviewed, since it is computationally feasible and according to [Tod07a], gives similar results to the EM algorithm. For detailed derivation of both solutions, see [Tod07a].

A suboptimum sequence is determined by

$$\hat{\mathbf{m}} = \arg\max_{\mathbf{m}} p(\mathbf{m}|\mathbf{X}, \boldsymbol{\lambda}^{(\mathrm{Z})}) \tag{4.27}$$

Using the suboptimum sequence $\hat{\mathbf{m}} = [\hat{m}_1, \hat{m}_2, \ldots, \hat{m}_T]$ the static feature sequence is given by

$$\hat{\mathbf{y}} = [\mathbf{W}^{\mathrm{T}}(\mathbf{D}^{(Y)}_{\hat{\mathbf{m}}})^{-1}\mathbf{W}]^{-1}\mathbf{W}^{\mathrm{T}}(\mathbf{D}^{(Y)}_{\hat{\mathbf{m}}})^{-1}\mathbf{E}^{(Y)}_{\hat{\mathbf{m}}} \tag{4.28}$$

where

$$\mathbf{E}^{(Y)}_{\hat{\mathbf{m}}} = [\mathbf{E}^{(Y)}_{\hat{m}_1,1}, \mathbf{E}^{(Y)}_{\hat{m}_2,2}, \ldots, \mathbf{E}^{(Y)}_{\hat{m}_L,L}] \tag{4.29}$$

$$\mathbf{D}^{(Y)^{-1}}_{\hat{\mathbf{m}}} = \mathrm{diag}[(\mathbf{D}^{(Y)}_{\hat{m}_1})^{-1}, (\mathbf{D}^{(Y)}_{\hat{m}_2})^{-1}, \ldots, (\mathbf{D}^{(Y)}_{\hat{m}_L})^{-1}] \tag{4.30}$$

The approach is similar to the parameter generation algorithm in HMM-based speech synthesis [Tok00], but now the means and covariances are obtained from the conditional distributions. The parameter generation approach was exploited in text-independent VC using state mapping codebooks [Zha08]. The ML estimation of the converted trajectory can be seen as a form of a Kalman smoother [Tod07a]. It bears some similarity to [P5] where we proposed a post-processing technique that balances between frame-by-frame conversion error and temporal continuity through the minimization of a cost function. The post-processing technique in [P5] is transparent to the conversion algorithm operating at frame-by-frame. It is described in Section 4.7.

A simplified and heuristic alternative to introduce correlation between frames is to smooth the generated parameters by low-pass filtering each feature sequence after conducting the conversion [Che03]. In [P6] the posterior probabilities (4.5) were smoothed by a low-pass filter before applying the mapping function. Applying filtering on posterior probabilities instead of features as in [Che03] ensures that smoothing does not affect the intra-frame correlations between different features and avoids extending the oversmoothing problem. A more sophisticated way to introduce correlation is to augment the current source frame data with the previous and next frame data as suggested in [P7]. It is described in the context of kernel transformed features in Section 6.1.3.

4.5.2 Oversmoothing

Oversmoothing results in muffled-sounding speech. It occurs both in the frequency- and the time-domain. In frequency-domain, the fine details of the spectrum are lost and formants become broadened. Post-filtering [Koi95, Hua01] can be used to emphasize the formants. Combining the frequency warped source spectrum with the GMM-based converted spectrum [Tod01] provides a way to preserve more spectral details.

In time-domain, the converted feature trajectory has much less variation than the original target feature trajectory. According to [Che03], oversmoothing occurs because the term $\mathbf{\Sigma}^{(yx)}_m(\mathbf{\Sigma}^{(xx)}_m)^{-1}$ in (4.17) becomes close to zero and thus the converted target becomes as a weighted sum of the target mean vectors. In speaker recognition, it is common to adapt only the means of a GMM [Rey00].

In [Che03], the source GMM was built from a larger data set and only the means were adapted for the target using maximum a posteriori (MAP) estimation.

The oversmoothing effect is especially severe in the ML parameter generation approach (Section 4.5.1). The global variance was taken into account together with the spectral trajectory estimation in [Tod07a]. The use of global variance allows more precise control of individual spectral features compared to standard postfiltering. Global variance has been successfully used for both VC [Tod07a] and HMM-based speech synthesis [Tod05] with cepstral features such as MCCs. On the other hand, the use of global variance did not improve the performance when using LSFs as the spectral features [Lin10]. Alternatively, global variance can be taken into account already in the training phase [Ben11]; the error is minimized between the original and the converted target features under the constraint that the global variance of the converted features should match to the variance of the original target features. The use of global variance degrades the objective results but improves subjective quality [Tod07a, Ben11].

4.5.3 Overfitting

Overfitting can be caused by two factors: first, the GMM may be overfitted to the training set. Second, when a separate mapping function is estimated (Section 4.4.1), it may become overfitted.

In particular, a GMM with full covariance matrices is difficult to estimate and is subject to overfitting [Mes07]. With unconstrained (full) covariance matrices, the number of parameters that needs to be estimated grows quadratically with the input dimensionality D. The total number is $M(1+D+0.5(D^2+D))$. Considering for example 24-dimensional source and target feature vectors and a joint density GMM model (D=48) and eight Gaussian components, the number of parameters is 9800. To reduce the number of parameters, a mixture of factor analyzers was applied to GMM-based conversion in [Uto06].

Figure 4.2 illustrates the concept of overfitting. A joint density GMM with full covariance matrices is used in conversion. The experimental setup (database and parameterizations) is the same as in Chapter 7 and not described here, since the aim is only to illustrate the overfitting effect. As it can be seen in Figure 4.2, the MCD decreases as a function of model complexity (number of Gaussians) when converting the same data (dashed line) that was used in the GMM training. However, for the unseen testing data (solid line) the MCD reaches the minimum at six Gaussians and then starts to increase.

In the joint density model, all the statistical information of the data is assumed to be stored in joint density function and no auxiliary conversion functions are used. Considering the problem with full covariance matrices, diagonal covariance matrices $\mathbf{\Sigma}_m^{(xx)}$, $\mathbf{\Sigma}_m^{(xy)}$, $\mathbf{\Sigma}_m^{(yx)}$, and $\mathbf{\Sigma}_m^{(yy)}$ can be used but a high number of mixtures is needed for accurate parameter modeling.

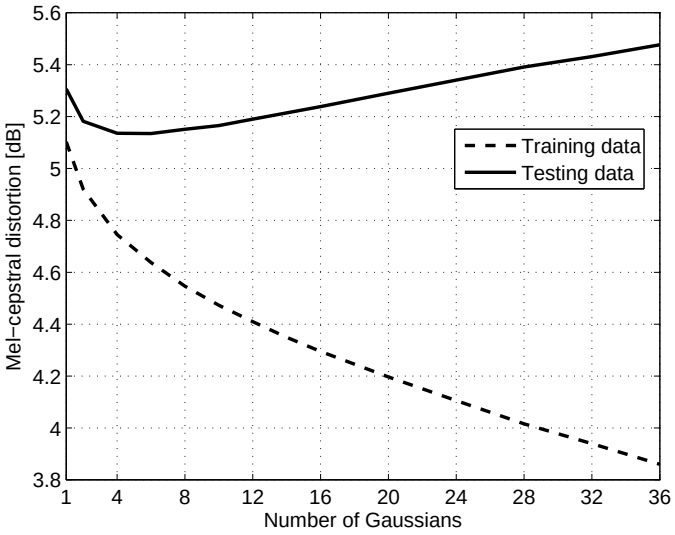


Figure 4.2: An illustration of overfitting: the Mel-cepstral distortion for the training and testing data as a function of model complexity (number of Gaussians).

The use of diagonal covariance matrices results in converting each feature dimension separately. This can be problematic since no direct correspondence between d^{th} feature of the source and d^{th} feature of the target may exist. This is investigated in Section 5.1 in the case of MCCs. LSFs, on the other hand, add another degree of complexity due to intra-frame correlation. An approach for mapping them with very little training data was proposed in [Hel08]. For each target LSF a separate GMM was built based on using only the source and target LSFs that correlated the most with the current target LSF. In [P6], PLS regression was combined with source GMM-based conversion in order to cope with a limited amount of data and still avoid restrictions related to the conversion function. The technique is described in Chapter 5.3.

4.6 Non-linear Techniques

Artificial neural networks (ANNs) offer a powerful tool for modeling non-linear relationships between the source and the target features, but they have not been widely studied in the context of VC. In [Nar95], a mapping function between first three formants is learned using ANNs. Desai et al. [Des10] exploited ANNs for converting features commonly used in many VC studies. They concluded that ANN gave similar or even better results than the state-of-the-art GMM-based conversion approach described in [Tod07a].

In [P7], non-linearities in the data were captured using a kernel transformation as a pre-processing step and applying a linear technique (PLS regression) to

extract the mapping function between the kernel-transformed source features and the original target features. This technique is described in Chapter 6. It bears some similarity to support vector regression (SVR) that was recently applied to voice conversion [Son11] and their difference is discussed in Section 6.6. ANNs, SVR, and the technique proposed in [P7] are able to model non-linear relations between the source and the target features but the last two require less tuning and are less prone to overfitting than ANNs.

4.7 Feature Sequence Optimization Algorithm Using Sequential Monte Carlo Methods

An algorithm for finding an optimal speech feature sequence that balances between frame-based conversion and temporal continuity was proposed in [P5]. The sequence optimization is carried out as a post-processing step after converting the source features in an utterance with an arbitrary frame-by-frame mapping procedure. The aim is to find a feature sequence that minimizes a cost function representing a trade-off between frame-based conversion and continuity.

The optimization problem is transformed into a discrete-time dynamical system in a state-space form [Míg12]. The model is matched with the cost function in the sense that the MAP estimate of the system state is a global minimizer of the cost. Sequential Monte Carlo methodology (particle filtering) is used to obtain a discretization of the state-space and the MAP estimate is approximated by searching over the discretized space. The best path is searched using the Viterbi algorithm as described in [God01].

The cost function is defined

$$C_t(s_{1:t}) = \sum_{i=1}^{t} \alpha_i |s_i - \hat{y}_i|^p + \sum_{i=2}^{t} \gamma_i |s_i - \rho_i(s_{i-1})|^q \tag{4.31}$$

where $s_{1:t}$ is the optimal speech sequence to be estimated, $\hat{y}_i$ is a converted speech feature in frame i resulting from a frame-based mapping algorithm, α_i and γ_i are scale factors to control the trade-off between the quality of the frame-by-frame conversion and continuity, respectively, ρ_i is a prediction function that yields the expected value of s_i from s_{i-1} and $p, q > 0$. The cost function (4.31) is defined for each feature dimension separately but vectors of features can be handled in a similar way.

The cost function (4.31) can be written in a recursive form

$$C_t(s_{1:t}) = C_{t-1}(s_{1:t-1}) + \alpha_t |s_t - \hat{y}_t|^p + \gamma_t |s_t - \rho_t(s_{t-1})|^q \tag{4.32}$$

A posterior probability density function (PDF) can be associated to (4.32) by means of exponential transformation that is of form

$$\begin{aligned}\pi\left(s_{0:t}|\hat{y}_{1:t}\right) &\propto \exp\{-C_t(s_{0:t})\} \\ &= \exp\{-C_{t-1}(s_{0:t-1}\}\exp\{-\alpha_t|s_t-\hat{y}_t|^p\}\times \\ &\quad \exp\{-\gamma_t|s_t-\rho_t(s_{t-1})|^q\}\end{aligned}$$

where $\propto$ denotes for proportionality.

The posterior PDF can be decomposed into

$$\pi(s_{1:t}|\hat{y}_{1:t}) = \pi(s_1)\pi(\hat{y}_1|s_1)\prod_{i=2}^{t}\pi(\hat{y}_i|s_i)\pi(s_i|s_{i-1}) \tag{4.33}$$

where the likelihood and transition density are

$$\pi(\hat{y}_i|s_i) \propto \exp\{-\alpha_i|s_i-\hat{y}_i|^p\} \tag{4.34}$$

$$\pi(s_i|s_{i-1}) \propto \exp\{-\gamma_i|s_i-\rho_i(s_{i-1})|^q\} \tag{4.35}$$

and the prior PDF $\pi(s_1)$ is assumed to be uniformly distributed since its value does not affect the cost of the PDF. Note that the proportionality constants in (4.34) and (4.35) should be independent of both s_i and s_{i-1}. This can be easily ensured in most cases of practical interest [Míg12].

To find the MAP estimate, the following procedure was carried out in [P5]:

1. A standard bootstrap filter [Gor93] is used to obtain a sequence of particles of size LxN_p where L is the number of frames in the utterance ($t = L$ in (4.31)) and N_p is the number of particles. The resulting random grid approximates the state-space model.

2. A Viterbi algorithm is used to find the optimal sequence of particles (size Lx1) that maximizes the posterior PDF (4.33).

In the experiments of [P5], the dynamics model $\rho_t(s_{t-1})$ was obtained by adding an offset term to s_{t-1}. The offset term was extracted by copying the average source dynamics from a few consecutive frames around frame t. The frame-based conversion algorithm was a codebook containing PLS regression matrices for different classes. The sounds were clustered with k-means algorithm. The error variance of the state transition process (i.e. $0.5\gamma^{-1}$) was determined for each cluster and the error variance of the state transition process (i.e. $0.5\alpha^{-1}$) was determined globally.

In subjective evaluation, the proposed smoothing technique was preferred in 86% of the cases when compared to using no smoothing and in 74% of the cases when compared to the use of linear multivariate regression [P5]. Also objective

decrease in the MCD was obtained when using smoothing. However, the configuration was very simple to fully benefit from the proposed approach. In Chapter 7, the offset term in the dynamics model is predicted from the source features and the error variances are determined experimentally using cross-validation.

In the experiments of [P5] and Chapter 7, it would have been possible to use a Kalman smoother. For a Kalman smoother, the dynamics model should be linear and completely known, and only Euclidean norm ($p, q = 2$) can be used. Nevertheless, there is a lot of freedom and flexibility to choose the cost function in the proposed approach.

4.8 Transformation of Excitation

A residual signal is obtained after suppressing the vocal tract information from the speech signal, for example by LPC inverse filtering. The residual is important for achieving good quality. It is possible to operate with the original source residuals left over from spectral envelope extraction. Copying the source residuals for the target results in voice quality that sounds neither like the source nor the target but rather a third speaker [Kai01]. The same techniques that are used for spectral mapping can be exploited for converting the excitation [Ars99, Kai01]. The residual can be parameterized into e.g. LSFs similarly to spectral envelope.

Duxans et al. [Dux06a] categorized residual transformation techniques to be based on either prediction or conversion. In residual prediction, a source residual is modified to match the target residual and in residual conversion, the residual is estimated based on the resulting converted spectral features. Comparison between different techniques showed that predicted residuals resulted in better quality than the converted ones [Dux06a].

Usually a simplified version of the residual is used, since the residual already contains something unpredictable left over from the more predictable part (spectral envelope). The simplest speech codecs only indicate a frame to be voiced or unvoiced, i.e. to contain an F_0 value or not. Some VC systems even leave unvoiced sounds unchanged [Err10b] but many unvoiced sounds do have some vocal tract coloring. The conversion of unvoiced sounds is important especially in cross-gender conversion [Ye06]. Typically voicing decisions are copied from the source and F_0 is transformed for the voiced frames. An example of voicing prediction is given in [Yut09], where spectral features were modeled together with F_0 with multi-space probability distribution for getting improved prediction for F_0 and voicing decision. In [P7], we proposed a voicing prediction technique that is described in Section 6.4.

The STRAIGHT mixed excitation was converted in [Oht06] using the ML parameter generation described in Section 4.5.1. In [Oht06], aperiodicity and spectrum were converted independently from each other, but in [P7], we showed that the use of source spectral features together with the source aperiodicity

features decreased the objective prediction error. Both [Oht06] and [P7] used the aperiodicity averaged over five frequency bands, but the error measures and the scalings were different, so the results cannot be directly compared.

4.9 Prosody Conversion

The most representative parameters of prosody are pitch and duration. Prosody is affected by syntactic, semantic, and pragmatic content of the sentences as well as the speaker's voice characteristics, mental state, etc. This makes prosody conversion challenging. Moreover, the amount of training data is limited and hence sophisticated prosody models cannot be built.

Usually the subjects in a VC listening test do not personally know the target speakers. This may be one reason why most of the approaches focus on segmental spectral envelope and only scale the F_0 level, that cannot be considered true prosody conversion.

In most studies, speaking rate differences are not modeled or a global speaking rate manipulation factor is used. This does not account for local differences resulting from different factors such as accent or dialect. Local speaking rate conversion was obtained by using different scalings for different phonetic codewords in [Ars99]. Syllable-level duration scaling was used in [P2].

4.9.1 Transformation of F_0

The most common way to convert F_0 is a simple scaling. For obtaining the target F_0 value $f_0^{(y)}$, the following transformation is applied to source F_0 value $f_0^{(x)}$

$$f_0^{(y)} = \mu_{f_0}^{(y)} + \frac{\sigma_{f_0}^{(y)}}{\sigma_{f_0}^{(x)}}(f_0^{(x)} - \mu_{f_0}^{(x)}) \tag{4.36}$$

where $\mu_{f_0}^{(x)}$, $\sigma_{f_0}^{(x)}$, $\mu_{f_0}^{(y)}$, and $\sigma_{f_0}^{(y)}$ represent the mean and standard deviation of the F_0 values for the source and the target, respectively. This is referred to as MV (mean-variance) scaling method.

The conversion is usually performed in logarithmic domain to enable better match with human perception. The mean and variance are calculated in logarithmic domain and F_0 is converted as

$$\log f_0^{(y)} = \mu_{\log f_0}^{(y)} + \frac{\sigma_{\log f_0}^{(y)}}{\sigma_{\log f_0}^{(x)}}(\log f_0^{(x)} - \mu_{\log f_0}^{(x)}) \tag{4.37}$$

A benefit of the MV scaling approach is that parallel data is not required. More sophisticated F_0 transformation techniques for non-parallel data have been proposed in [Lol08, Wu10]. In [Lol08], syllable-level F_0 and duration information

were obtained for the target using maximum likelihood linear regression. Wu et al. [Wu10] proposed a frame-level F_0 conversion approach where non-parallel F_0 training data is first aligned, a GMM is estimated on the aligned data, and F_0 histogram equalization is applied to reduce the oversmoothing problem.

Approaches based on parallel data include an utterance level codebook [Cha98], GMM-based F_0 prediction [Nur06], and a codebook based on weighted average of segmental pitch contours [Tur03]. In [Gil03], separate MV scalings (4.36) were estimated for hand-marked accents, sentence-initial and sentence-final F_0 values.

4.9.2 Proposed F_0 Conversion Technique and Results

Prosody is a suprasegmental phenomenon. However, there is not much data available for training prosody conversion models in VC. A sensible choice is thus to model prosody at syllable level, since syllable (or mora in some languages) is usually considered as the smallest prosodic unit in a language [Jun05]. In [P2], a codebook was generated from syllable-aligned F_0 contours of the source and the target. Due to different lengths of syllables, syllable-level F_0 contours from both speakers were transformed into DCT coefficients and the first eight coefficients (excluding the bias term) were retained. In addition to DCT contours, syllable durations and simple linguistic information such as lexical stress and syllable position in the word were stored.

After collecting the DCT contours together with linguistic and durational data, a classification and regression tree (CART) was formed [P2]. The CART was used to help the selection process after candidate contour pre-selection.

The training of the CART goes as follows: a distance matrix $\mathbf{A}$ between all N source DCT vectors is calculated resulting in a matrix of size N^2. A similar distance matrix $\mathbf{B}$ is calculated between the target DCT vectors. For the j^{th} source DCT vector, the closest source DCT vectors are searched from $a_{j,k}$ where $k = 1, 2, \cdots, N, k \neq j$ and the indices of the closest vectors become potential candidates. For each potential candidate, the corresponding target distance is obtained. Based on the target distance $b_{j,k}$, an entry k is considered either as a "possibly" optimal (small $b_{j,k}$), a "neutral" or a "non-optimal" (large $b_{j,k}$) candidate as an substitute for the entry j. The "possibly optimal" and "non-optimal" candidates are used for forming the training data for CART, where "possibly optimal" denotes for one class and "non-optimal" for another class. In such a manner, the training data is collected for CART forming.

The predictors for the tree are formed for the optimal and non-optimal candidates. Their linguistic information is compared against the linguistic information of the entry j, resulting in a binary vector where a zero means that there was a match in the corresponding feature and the value one means that the features were not the same. In addition, the absolute differences in syllable durations and in voiced segment durations are computed and stored.

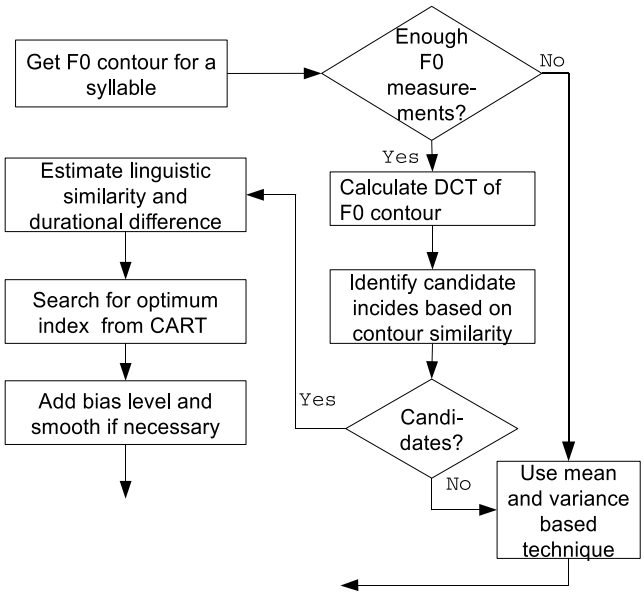


Figure 4.3: The F_0 conversion process for a source F_0 contour.

The conversion phase is depicted in Figure 4.3. The aim is to pre-select good candidates from the codebook based on the distance between the source contour under conversion and the source contours in the codebook. If there are not initially enough F_0 measurements in the syllable or no candidates are found in the pre-selection phase, MV scaling (4.36) is used for the current syllable. Otherwise; the CART is used to predict which target contour could be the best choice (high probability of being "possibly optimal") based on the difference in linguistic and durational information. IDCT is applied to the corresponding target DCT contour of the chosen index and a bias level is added resulting from MV scaling (4.36). Smoothing is used if the boundary between two syllables is initially continuous.

The performance of the proposed F_0 conversion was evaluated against GMM-based F_0 conversion [P2]. The source database was a part of a database recorded for text-to-speech purposes and the target database was spoken in a more vivid and expressive manner. 95 parallel sentences were used for training and 25 for testing. The proposed method was rated in 67.0% of the cases to provide more similar prosody to the target whereas the score for baseline was only 10.3%. Otherwise they were rated equal. Similarly, the proposed approach was found to provide less robotic speech quality common in vocoded speech [P2].

4.10 Summary

GMM-based conversion, frequency warping, codebooks, and non-linear techniques such as ANNs are most popular mapping techniques for spectral conversion. GMM-based conversion, as well as many other techniques, suffer from overtraining, oversmoothing and time-independent mapping. A vast amount of studies have concentrated on solving these problems. For example time-independent mapping can cause discontinuities and one solution is to utilize a post-processing technique [P5]. The post-processing technique optimizes a global cost function that balances between parameter continuity and conversion accuracy.

An important cue of identity is prosody. A prosody conversion algorithm selects the best matching target syllable F_0 contour from a syllable contour codebook [P2]. Prosody is likely to play more important role when listeners know the speaker.

Chapter 5

Hybrid GMM and PLS Regression Voice Conversion Algorithm

The drawbacks of GMM-based conversion, oversmoothing, overtraining, and time-independent mapping, and some solution proposals were reviewed in Section 4.4. In this chapter, solutions for the overfitting and time-independent mapping problem are proposed based on [P6].

5.1 Motivation

In the joint density GMM approach (Section 4.4.2), the choice of the covariance matrix has a direct impact on the resulting conversion function. For reference, it is examined here how the corresponding elements of the source and target spectral vectors correlate with each other. The evaluations are done for 24 cross-gender and 24 intra-gender conversion pairs and the data is collected from 50 sentences for each pair. MCCs (24^{th} order) are extracted from the spectra given by STRAIGHT and aligned using DTW. An absolute correlation coefficient between all MCC dimensions (energy term not included) of the source and the target is calculated. The percentage of the d^{th} source MCC having the highest absolute correlation coefficient with the d^{th} MCC of the target is examined. The results are shown in Figure 5.1 with red and blue line for cross-gender and intra-gender case, respectively. It can be seen that especially in cross-gender conversion, the MCC element correspondence is decreased gradually after the first few MCCs.

In GMM-based conversion, the conversion result is based on several local conversion functions. Hence, in addition to global correlation statistics, k-means clustering is carried out for each conversion pair. The number of classes is set to 16 and the percentage of highest correlation coefficients is determined for each class similarly to the global case. The results are shown in Figure 5.1 with red and blue dashed line for cross-gender and intra-gender case, respectively. It can

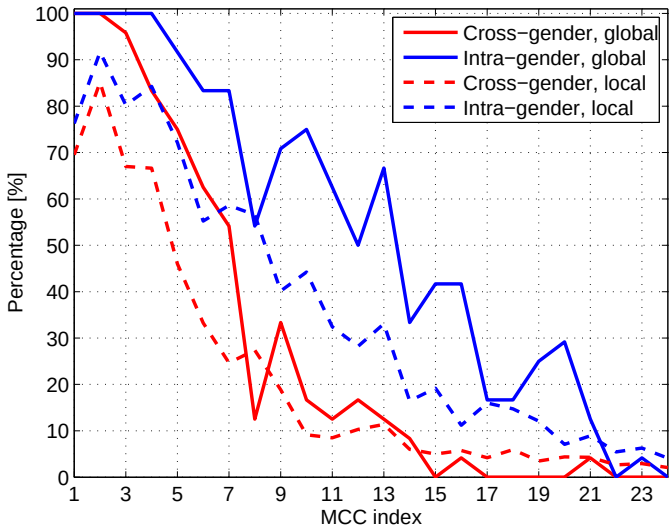


Figure 5.1: The percentage of cases where the d^{th} MCC of the source has the highest absolute correlation coefficient with the d^{th} MCC of the target compared to other MCCs.

be concluded that a specific source MCC may not be the most informative for predicting the corresponding target MCC neither at local level.

Ideally the GMM-based model should be able to model the correlation between different feature dimensions but avoid the use of full covariance matrices in the joint density model or full-rank transforms in the source GMM approach. The joint density model has been more popular than the source GMM approach, but they give similar results. Now the source GMM approach is considered. In the case of diagonal covariance matrices, the conversion function was defined separately for each feature dimension in [Sty98], but this is not necessary. Here diagonal covariance matrices are used but the conversion function is estimated jointly for all the dimensions. However, the size of the transform matrix $\boldsymbol{\beta}$ in (4.6) is DxDM, where D is the number of features and M is the number of Gaussians. The transform matrix $\boldsymbol{\beta}$ is of full rank.

To overcome the assumption of variable independence in the diagonal-covariance joint density model and the overfitting problem in the full least squares solution, PLS regression [dJ93] is used to extract the transform matrix $\boldsymbol{\beta}$ for the source GMM approach. It is described in Section 5.3.

5.2 Partial Least Squares Regression

The goal of regression is to predict the behavior of variables $\mathbf{Y}$ from the observed variables $\mathbf{X}$ (matrix of predictors). The regression model is defined as

$$\mathbf{Y} = \boldsymbol{\beta}\mathbf{X} + \boldsymbol{\epsilon} \tag{5.1}$$

where the size of $\mathbf{X}$ is IxN and the size of $\mathbf{Y}$ is JxN, and the residual term $\boldsymbol{\epsilon}$ is of size JxN. Equation (5.1) assumes that both $\mathbf{X}$ and $\mathbf{Y}$ are mean-centered meaning that the empirical means are subtracted prior to processing, and afterwards added to the regression results.

If variables in $\mathbf{X}$ are correlated, the error variance of the ordinary least squares (OLS) solution for $\boldsymbol{\beta}$ (4.6) can become high. This results in degraded prediction ability over new data. There exists a number of biased estimators dealing with problem of multicollinearity instead of the unbiased OLS estimator. PLS regression is one of them.

PLS is a family of regression-based methods for modeling and analyzing relationships between observed variables by means of latent variables. PLS has been especially popular in chemometrics [Bre07], but also in many other areas such as social science [Hul99], econometrics [Boo80], bioinformatics [Ngu02], marketing [Vin10], medicine [Kri11], and recently in speaker recognition [Sri11]. In chemometrics, PLS is used as a regression and prediction tool whereas in e.g. marketing and social science, PLS is used as an approach for structural equation modeling (called PLS path modeling) for explaining causal relations [Vin10]. In this thesis, PLS is considered from a regression and prediction aspect.

The aim of PLS regression is to find new factors, called latent variables or components, that are a linear combination of the original $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_N$. PLS regression is similar to principal component regression (PCR), where principal components of $\mathbf{X}$ are used as regressors on $\mathbf{Y}$. The main difference is that in PCR, the principal components are determined by $\mathbf{X}$ only whereas in PLS, the aim is to extract components that capture most of the covariance between $\mathbf{X}$ and $\mathbf{Y}$. If the number of components is equal to the number of predictors, PLS regression performs similarly to standard multivariate regression but without the problem of matrix inverses. However, the optimal number of latent variables is usually lower. The optimal number of latent variables can be chosen by cross-validation described in Section 6.3.

The idea of PLS is to decompose both $\mathbf{X}$ and $\mathbf{Y}$ as a product of scores and loadings as

$$\mathbf{X} = \mathbf{QR} + \mathbf{E} \tag{5.2}$$

$$\mathbf{Y} = \mathbf{PR} + \mathbf{F} \tag{5.3}$$

where $\mathbf{R}$ is a HxN matrix of score vectors, $\mathbf{Q}$ and $\mathbf{P}$ are loading matrices of size IxH and JxH, respectively, H is the number of latent components, and $\mathbf{E}$

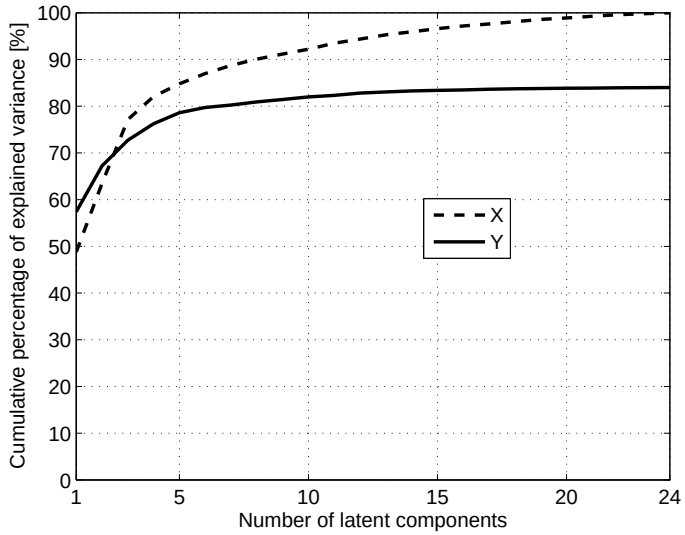


Figure 5.2: The cumulative variance percentage explained by each latent component for predictors ($\mathbf{X}$) and responses ($\mathbf{Y}$).

(size IxN) and $\mathbf{F}$ (size JxN) are matrix residuals. Note that the bias terms are removed before decomposition. The regression matrix $\boldsymbol{\beta}$ is obtained as

$$\boldsymbol{\beta} = \mathbf{R}\mathbf{Q}^{\mathrm{T}} \tag{5.4}$$

Two of the most widely used techniques for obtaining $\mathbf{R}$, $\mathbf{P}$, and $\mathbf{Q}$ are NIPALS (non-linear iterative PLS) and SIMPLS (simple PLS) [dJ93]. The SIMPLS algorithm is used in this thesis due to its computational efficiency. The algorithm is given in [P6, P7].

Consider the following conversion example with a training data of only one sentence. The speech features are 24^{th} MCCs (energy term not included) extracted from STRAIGHT analysis/synthesis framework. The data is aligned with DTW and the number of frames after the alignment is 574.

The cumulative percentage of variance explained by each PLS latent vector is shown in Figure 5.2 for both $\mathbf{X}$ and $\mathbf{Y}$ where $\mathbf{X}$ and $\mathbf{Y}$ include the aligned source and target MCC vectors, respectively. The first 3 latent vectors explain 77.3% of variance in $\mathbf{Y}$. With 13 and 24 components the percentage is 83.0% and 84.0%, respectively, meaning that not much improvement is achieved after 13 components.

A PLS regression model is calculated for different amount of latent components varying from 1 to 24. The obtained regression models are tested on 49-sentence testing data. Figure 5.3 shows the MCD (3.4) as a function of the number of latent components. As it can be seen in Figure 5.3, the optimal number of latent components is 7 (MCD 5.76 dB). If a full transform is used (24 latent components), the MCD is 6.02 dB.

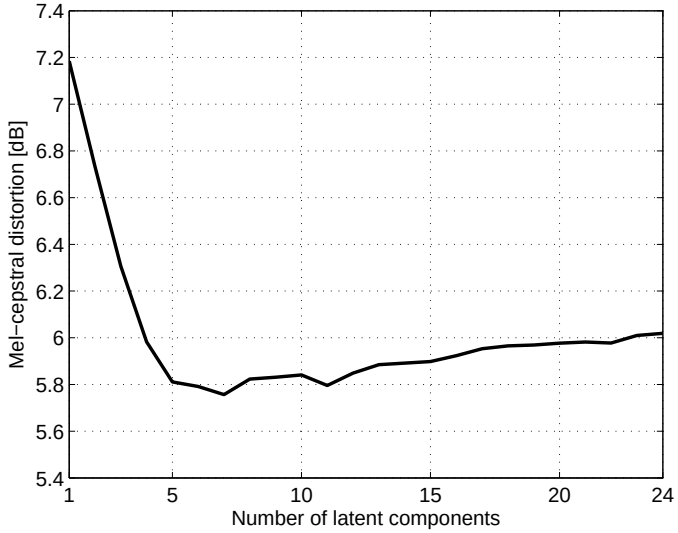


Figure 5.3: The Mel-cepstral distortion on the testing data as a function of the number of latent components.

5.3 Combining GMM and PLS

When PLS is applied to the original source and target features, a global linear regression matrix is obtained. If there is enough training data (more than two sentences), the benefits of applying PLS on the original data are lost. A full rank regression matrix can be used since for example MCCs do not contain multicollinearity.

PLS was used as a regression technique in GMM-based conversion [P6] and the procedure goes as follows. First a source GMM (diagonal covariance matrices) is trained. Alternatively, a joint density GMM can be estimated to guide a more judicious allocation of mixtures. After training the GMM, the posterior probabilities (4.5) are obtained. The predictor matrix $\mathbf{X}$ is the same as in (4.9) but repeated here for clarity

$$\mathbf{X}_n = \left[\omega_{1,n}\mathbf{x}_n^{\mathrm{T}}, \omega_{2,n}\mathbf{x}_n^{\mathrm{T}}, \ldots, \omega_{M,n}\mathbf{x}_n^{\mathrm{T}}\right]^{\mathrm{T}} \tag{5.5}$$

and the output is the original target vectors

$$\mathbf{Y}_n = \mathbf{y}_n \tag{5.6}$$

The transform $\boldsymbol{\beta}$ is now found with PLS regression (5.4). The bias terms are again removed from both $\mathbf{X}_n$ and $\mathbf{Y}_n$ before estimating the regression matrix. The training phase is depicted in Figure 5.4.

Given the testing data, a similar matrix to (5.5) is formed based on posterior probabilities. By multiplying the matrix with the PLS regression matrix $\boldsymbol{\beta}$, the

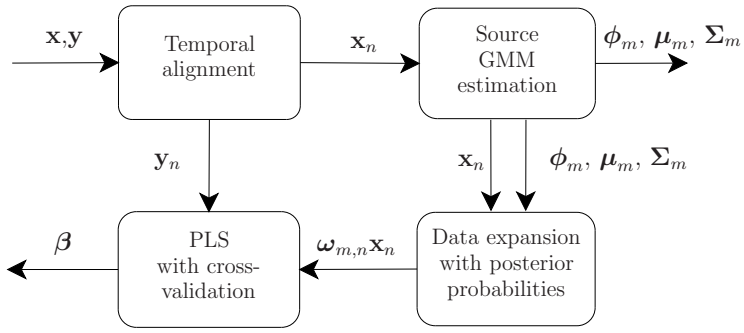


Figure 5.4: Overview of the training procedure using GMM and PLS.

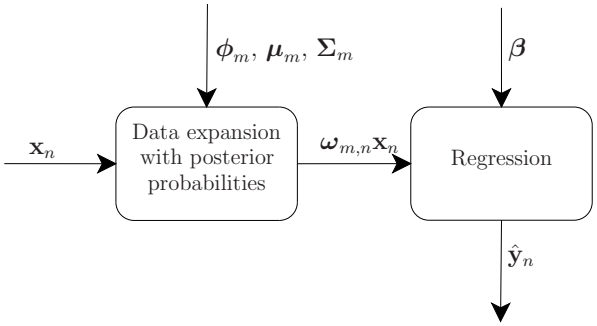


Figure 5.5: Overview of the conversion/testing procedure using GMM and PLS.

predicted target $\hat{\mathbf{y}}$ is obtained. Bias terms are then added to the result. The conversion (testing) phase is depicted in Figure 5.5.

Consider the same example as in the previous section, but now the training and testing data consists of 10 and 30 sentences, respectively. A joint density GMM with full covariance matrices and a source GMM with PLS regressions as described in this section were formed for a different number of Gaussians. Figure 5.6 shows the MCD for the testing data as a function of number of Gaussians with the two GMM approaches. It can be seen that the error of the conventional GMM-based technique is increased steeply when increasing the amount of Gaussians after the optimal number (2). For the proposed technique, the error is only slightly increased when increasing the amount of Gaussians after the optimal number (7). Thus the selection of the number of Gaussians is not extremely crucial for preventing the overfitting when using the proposed technique. Moreover, the proposed technique is able to obtain lower MCD.

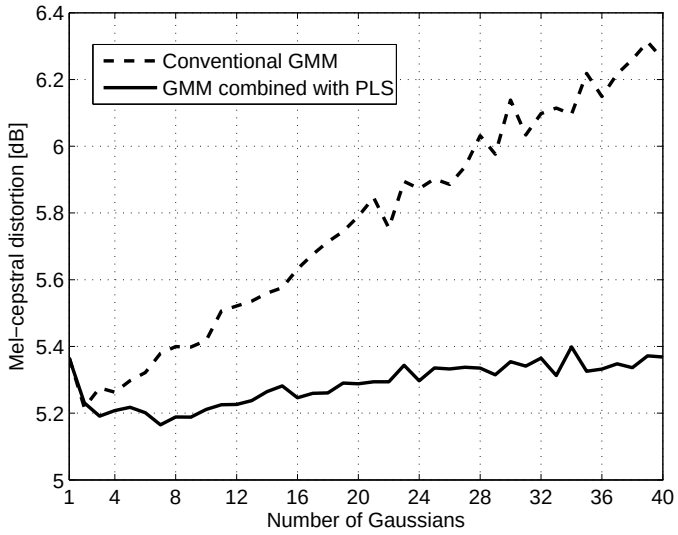


Figure 5.6: The Mel-cepstral distortion as a function of model complexity (number of Gaussians) using a conventional full covariance joint density GMM (dashed line) and a GMM with PLS regression (solid line).

5.4 Soft Alignment Assumption and Posterior Probability Smoothing

Although overfitting can be effectively avoided when using PLS regression to extract the transformation matrix in GMM-based conversion, the problem of temporal continuity still exists. Until now, no explanation has been given on why conventional GMM-based conversion does not work well, since it is supposed to divide the acoustic space into overlapping classes, i.e. make a soft classification. In [P6], this assumption is revised and a crucial conclusion is drawn that in most of the frames, only one Gaussian is active meaning that its posterior probability is close to one and for other Gaussians close to zero. When the highest posterior probability is changed, the change is very rapid. This makes the conventional GMM-based approach to shift from a soft acoustic classification method to a hard classification method. It is thus susceptible to discontinuities similarly to the codebook-based methods.

The problem is illustrated in Figure 5.7 that depicts the posterior probabilities for the frames in one sentence. A 32-mixture joint density GMM (diagonal cross-covariance matrices) is trained for 24^{th} order MCCs extracted from 20 parallel sentences. The posterior probabilities are now extracted from the source features only, but the sentence in Figure 5.7 was included in the training data. As it can be seen, there is usually a single Gaussian dominating in a frame and when the

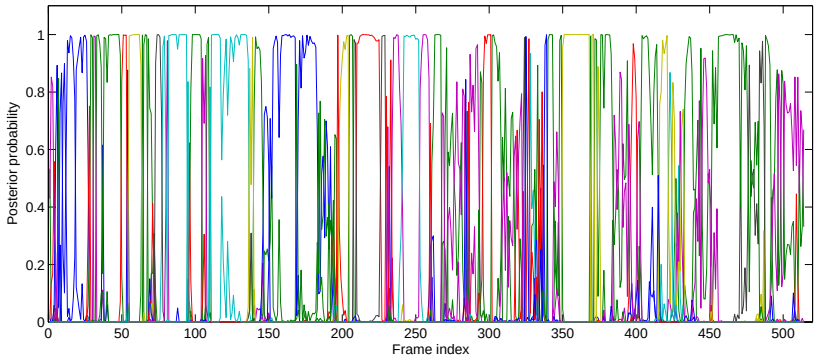


Figure 5.7: Posterior probabilities of a GMM (32 mixture components) for a sentence. Different colors describe different mixture components, but they are only used for illustration purposes.

dominant component is changed, the transition is not smooth. The frame shift is 5 ms.

To overcome the problem, posterior probability smoothing was proposed in [P6]. The posterior probabilities $\omega_{i,n}$ were smoothed in the conversion phase by a low-pass filter before applying the regression matrix and they are normalized to one within a frame. In [P6], the low-pass filter was a finite impulse response filter of order 10 with a cut-off frequency 0.1 relative to the sampling frequency (the number of frames in a second). The specific selection of the cut-off frequency is not expected to be highly crucial. Note that smoothing at posterior probability level does not result in oversmoothing as such.

5.5 Simulation Results

In [P6], the spectral mapping performance of the proposed technique (referred to as *GMM-PLS*) was compared against a baseline technique. The baseline technique was a conventional joint density GMM-based conversion [Kai98]. For the baseline (referred to as *GMM-D*), all covariance matrices, $\mathbf{\Sigma}_m^{(xx)}$, $\mathbf{\Sigma}_m^{(xy)}$, $\mathbf{\Sigma}_m^{(yy)}$, and $\mathbf{\Sigma}_m^{(yy)}$, were diagonal and the number of Gaussians was 16. For the proposed technique, the number of Gaussians for the source GMM was 8. The number of training sentences was 10. In addition, the performance of linear multivariate regression (referred to as *LMR*) was evaluated in the objective experiments. In this case, *LMR* corresponds to using a joint density model with a single Gaussian and a full covariance matrix.

Four speaker pairs, a female-to-female (F-F), a male-to-male (M-M), a female-to-male (F-M), and a male-to-female (M-F) from CMU ARCTIC database [Kom03] were considered in conversion [P6]. STRAIGHT A/S system was used to extract the spectral envelope and F_0. Spectral envelope was parameterized in 24^{th} or-

Table 5.1: The Mel-cepstral distortion in dB using linear multivariate regression (*LMR*), a joint density GMM (*GMM-D*) and the proposed technique (*GMM-PLS*).

	LMR	*GMM-D*	*GMM-PLS*
M-F	5.76	5.79	5.61
F-M	5.69	5.69	5.54
M-M	6.00	5.94	5.83
F-F	5.31	5.23	5.22
Average	5.69	5.66	**5.55**

der MCCs. For both baseline and the proposed technique, F_0 was transformed according to (4.37) and voicing values were copied from the source.

In the objective evaluation, posterior probability smoothing was not used. The spectral conversion results are shown in Table 5.1. As it can be seen, *GMM-PLS* resulted in lower MCD than the other techniques except for the F-F case where it performed similarly to *GMM-D*. *LMR* was able to perform as well or even better than *GMM-D* in cross-gender conversion (M-F and F-M). This may result from the fact that *GMM-D* assumes that target features in each dimension can be obtained by using the corresponding source feature dimension only. This assumption is likely to be more erroneous in the case of cross-gender conversion as it was depicted in Figure 5.1. On the other hand, *LMR* is too simple to fully exploit the training data. The use of *GMM-PLS* provides a way to use a more complicated model but still without the problem of overfitting.

The subjective quality was evaluated using a preference test [P6]. The quality of the proposed technique (*GMM-PLS*) with posterior probability smoothing was compared against the quality of the baseline technique with and without posterior probability smoothing. In addition, a similar comparison test was conducted concerning the identity but only for the baseline technique with posterior probability smoothing. The results are summarized in Table 5.2. It can be seen that there was a clear preference for the proposed technique over the baseline with and without using posterior probability smoothing. In cross-gender and M-M conversion, the proposed technique achieved higher preference scores for both identity and quality [P6]. For the F-F conversion case, the smoothed baseline and the proposed technique were preferred equally according to 95% confidence intervals [P6]. Thus, the results were in line with the objective measures.

5.6 Summary

GMM-based voice conversion is prone to overfitting. The overfitting problem can be alleviated by using diagonal covariance matrices instead of full matrices. It, however, restricts the conversion function. Overfitting can be effectively avoided

Table 5.2: The results from a preference test on quality and identity when comparing the proposed *GMM-PLS* technique with posterior probability smoothing against a baseline technique with and without posterior probability smoothing.

Technique in comparison	Test for	Prefer proposed technique (*GMM-PLS*) [%]
Baseline without smoothing	Quality	84.2
Baseline with smoothing	Quality	67.0
Baseline with smoothing	Identity	62.8

by replacing the ordinary least squares estimation with PLS regression [P6]. The problem of determining a suitable amount of Gaussians is also alleviated with this technique. There is usually a single Gaussian component that dominates in each frame that makes GMM-based approaches susceptible to discontinuities. In order to improve continuity between frames, posterior probabilities can be smoothed.

Chapter 6

Dynamic Kernel PLS Regression Based Voice Conversion

The *GMM-PLS* technique described in the previous chapter effectively prevents overfitting. A GMM was used to extract the probabilistic cluster memberships in the data, but the memberships were found to be hard rather than soft. Thus the technique as well as other GMM-based techniques are not able to approximate non-linear functions using multiple local linear transforms. In addition, the temporal continuity should be taken into account in a consistent way in both training and conversion instead of smoothing used at the post-processing phase as in [P6].

A variety of techniques have been proposed to extend the linear PLS algorithm into a non-linear algorithm, for example by modeling the relationship between latent variables with a neural network [Qin96]. In this chapter, an efficient and straightforward technique for non-linear mapping is proposed and the same model is extended to take into account the temporal continuity [P7]. The main idea is to extent a linear regression method to a non-linear method by carrying out a kernel transformation as a pre-processing step [Emb05]. After the data has been transformed, linear PLS regression is applied on kernel transformed data. Non-linear aspects of the problem can be captured in the kernel but the simplicity of linear regression techniques is retained. Tuning of the algorithm is easy; the only variable that needs to be optimized excluding the kernel parameters, is the number of latent vectors for PLS regression. To take the dependencies between consecutive frames into account, the kernel transformed source data was augmented with the kernel transformed source data from previous and next frames.

6.1 Dynamic Kernel PLS Technique

6.1.1 Kernel Transformation

A kernel is a matrix containing similarity measures for a dataset. The similarity measures are obtained between the data itself or with other data. In this thesis, the kernel matrix $\mathbf{K}$ is formed using the source features $\mathbf{x}_n$ and a set of reference vectors $\mathbf{c}_j$

$$\mathbf{K} = \begin{bmatrix} k_{11} & k_{12} & \dots & k_{1N} \\ k_{21} & k_{22} & \dots & k_{2N} \\ \vdots & \vdots & \dots & \vdots \\ k_{C1} & k_{C2} & \dots & k_{CN} \end{bmatrix} \tag{6.1}$$

where k_{jn} are entries of a kernel matrix and C is the number of reference vectors. In this thesis, a Gaussian kernel is used and it is defined as

$$k_{jn} = e^{\frac{-\left\|\mathbf{x}_n - \mathbf{c}_j\right\|^2}{\phi}} \tag{6.2}$$

where ϕ is a scaling parameter. The selection of ϕ is crucial for pattern classification approaches [Wan09], but for prediction, there is usually a relatively broad range for ϕ where stable quality is obtained [Emb05].

6.1.2 Applying PLS on Kernel-Transformed Data

The kernel-transformed source data is multicollinear and the regression problem cannot be solved with standard linear regression. PLS regression offers a good solution for handling multicollinearity. The approach used in this thesis is referred to as kernel PLS (KPLS). An alternative is to directly kernelize the PLS algorithm [Ros02] but the approach is more complicated and limited to square kernels.

Before carrying out PLS regression, the kernel matrix must be centered *both* row- and column-wise. Centering in the kernel space is not as obvious as in the original feature space, since the mean cannot be computed directly [Sch98]. A similar strategy is used as for kernel principal component analysis in [Sch98], but kernel centering is done in a computationally more efficient way. It goes as follows [Emb05]: for centering a training kernel, the average of each row is subtracted and the averages are stored for later use. Then the average of each column is subtracted from the obtained row-centered training kernel. The centered kernel is denoted by $\tilde{\mathbf{K}} = [\tilde{\mathbf{k}}_1, \tilde{\mathbf{k}}_2, \dots, \tilde{\mathbf{k}}_n, \dots, \tilde{\mathbf{k}}_N]$ where $\tilde{\mathbf{k}}_n$ is the n^{th} column of the centered kernel matrix similarly to non-centered kernel matrix (6.1).

For the test data, the kernel is formed using the testing data and the reference vectors, and the following centering operations are applied: the stored row-averages from the training kernel are subtracted from the testing kernel. Then the average of each column is calculated and subtracted from the obtained row-centered testing kernel.

6.1.3 Incorporating Dynamics

The problem of time-independent mapping applies to KPLS as well, although it is a global model and does not have the hard clustering problem of the GMM. By simply augmenting the original source data with previous and next frame data, dynamic relations can be modeled [Ric88, Qin96]. In addition to improving perceptual quality with smoother transitions from frame to frame, neighboring frames offer information that enables building better conversion models.

Taking the previous and the next frame into account, the predictor variables become

$$\mathbf{X}_n = [\mathbf{x}_{n-}^{\mathrm{T}}, \mathbf{x}_n^{\mathrm{T}}, \mathbf{x}_{n+}^{\mathrm{T}}]^{\mathrm{T}} \tag{6.3}$$

or when the centered kernel vectors are used

$$\mathbf{X}_n = [\tilde{\mathbf{k}}_{n-}^{\mathrm{T}}, \tilde{\mathbf{k}}_n^{\mathrm{T}}, \tilde{\mathbf{k}}_{n+}^{\mathrm{T}}]^{\mathrm{T}} \tag{6.4}$$

The approaches using (6.3) and (6.4) are called *DPLS* and *DKPLS*, respectively.

6.1.4 Reference Vectors

Each entry of the kernel matrix (6.1) is a similarity measure between an original source vector and a reference vector (6.2). In the benchmark studies of [Ben03], all original predictor vectors $\mathbf{x}_n$ were used as reference vectors. In VC applications, N becomes rather high and using all original vectors as reference vectors is not computationally efficient.

In [P7], k-means algorithm was used to cluster the original source vectors into C classes and the cluster centers acted as reference vectors. The same reference vectors were used in constructing $\mathbf{k}_n$, $\mathbf{k}_{n-}$, and $\mathbf{k}_{n+}$. About 100-200 reference vectors were enough for 20-sentence training data ($\sim$11000 vectors) [P7].

6.2 Example Using Kernel PLS

The benefit of using *KPLS* (no consecutive frame information used) is illustrated in the following example. Consider the same conversion pair as in Section 5.2 with the same training and testing data. All 574 source MCC vectors act as reference vectors resulting in a square kernel matrix, i.e. C=N=574 in (6.1). The scaling term ϕ in (6.2) is set to 10. Here the suitable range for ϕ is obtained by summing the standard deviations of each feature together.

Figure 6.1 shows the MCD on the testing data as a function of the number of latent components used in PLS regression. The highest possible number of latent components is 573 but for illustration purposes, only 200 first components are shown. The result for using all 573 components (corresponding to the solution obtained by using a pseudoinverse) is the same as for 200 components, 5.72 dB. By restricting the amount of latent components, overfitting is effectively

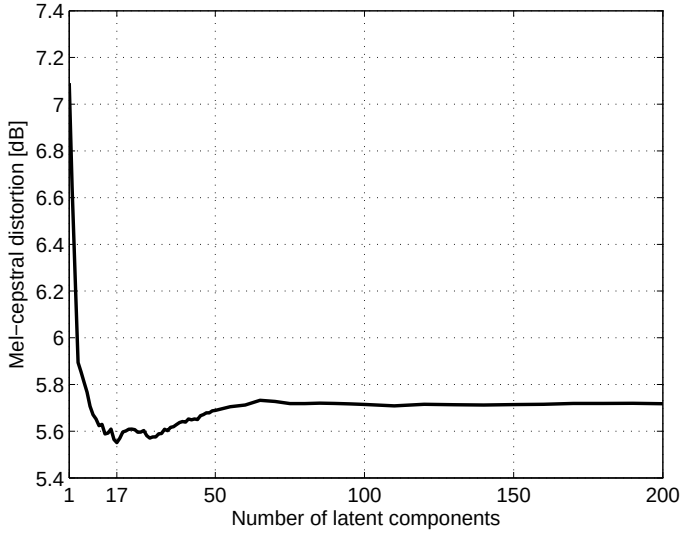


Figure 6.1: The Mel-cepstral distortion on the testing data as a function of number of latent components. The prediction model was built on kernel-transformed data extracted from one sentence.

prevented. The optimal amount of PLS components is 17 with the MCD of 5.55 dB. Comparing to Section 5.2 where PLS regression was applied to the original source data, the resulting values were 7 and 5.76 dB. Hence, the use of kernel transformed source features improves the performance.

6.3 Cross-Validation of Speech Features

Cross-validation (CV) is a statistical method for model evaluation. The data is divided into two parts: one is used to train a model and the other is used to validate the model. The optimal number of latent components for PLS can be chosen by CV as it is done in this thesis. Regression models are built on the training data using a various amount of latent component. The number of latent components used to build the regression model that gave the minimum error on the test data is selected for building the final model from all training data.

The temporal correlation of speech features must be taken into account. If samples collected from a set of sentences are divided randomly into a training and a testing set, CV may give a too optimistic error estimate with a high number of latent vectors. The prediction for $\mathbf{y}_n$ is more accurate, if $\mathbf{y}_{n-}$ and $\mathbf{y}_{n+}$ that are temporally correlated with $\mathbf{y}_n$, have been available for constructing the regression matrix. The resulting model may still not generalize well for the new data.

In [P7], we proposed to divide the data into sequential CV sets. For example in the case of 10-fold CV of 20-sentence training data, sentences 1 and 2 are

left out in the first round and the regression matrix is computed using sentences 3-20 and so on. The proposed CV scheme is also important for GMM-based conversion, since the optimal number of Gaussians can be chosen by CV [Shi10].

The CV issue was assessed in the context of spectral features in [P7]. The MCD for the real testing data was almost 0.2 dB lower when extracting the optimal number of latent components from the proposed sequential CV scheme instead of using random CV.

6.4 Aperiodicity and Voicing Prediction

For a high-quality VC system, also the excitation parameters should be converted. The aperiodicity map of STRAIGHT averaged over five frequency bands (Figure 3.3 in Section 3.1) was used as an excitation parameter in [P7]. Usually excitation and spectral envelope are assumed to be uncorrelated, but in practice this is not completely true. In [Sil11], the BAPs were predicted from the spectral features of the same speaker in HMM-based speech synthesis. In [P7] a prediction model for converting the BAPs was constructed with DKPLS using the source BAP and MCC values. Separate kernel matrices were formed for different features.

Voicing decision prediction was proposed in [P7]. Conventionally the voicing decisions are copied from the source. Random voicing errors can occur in feature extraction and by voicing prediction the appearance of such errors can be alleviated. The idea is similar to aperiodicity prediction. Voicing prediction utilized source voicing decisions, kernel transformed source MCC and BAP features and they were augmented with the data from the consecutive frames [P7].

The predictor vectors $\mathbf{X}_n^{\mathrm{ap}}$ and $\mathbf{X}_n^{\mathrm{v}}$ for aperiodicity and voicing features, respectively, are

$$\mathbf{X}_n^{\mathrm{ap}} = \begin{bmatrix} \tilde{\mathbf{k}}_{n-}^{\mathrm{sp}} \\ \tilde{\mathbf{k}}_{n}^{\mathrm{sp}} \\ \tilde{\mathbf{k}}_{n+}^{\mathrm{sp}} \\ \tilde{\mathbf{k}}_{n-}^{\mathrm{ap}} \\ \tilde{\mathbf{k}}_{n}^{\mathrm{ap}} \\ \tilde{\mathbf{k}}_{n+}^{\mathrm{ap}} \end{bmatrix} \quad \mathbf{X}_n^{\mathrm{v}} = \begin{bmatrix} \tilde{\mathbf{k}}_{n-}^{\mathrm{sp}} \\ \tilde{\mathbf{k}}_{n}^{\mathrm{sp}} \\ \tilde{\mathbf{k}}_{n+}^{\mathrm{sp}} \\ \tilde{\mathbf{k}}_{n-}^{\mathrm{ap}} \\ \tilde{\mathbf{k}}_{n}^{\mathrm{ap}} \\ \tilde{\mathbf{k}}_{n+}^{\mathrm{ap}} \\ v_{n-} \\ v_{n} \\ v_{n+} \end{bmatrix} \tag{6.5}$$

where $\tilde{\mathbf{k}}_n^{sp}$ and $\tilde{\mathbf{k}}_n^{ap}$ denote the centered kernel vectors extracted from the spectral and aperiodicity source vectors, respectively, and v_n is a binary source voicing value (either 0 or 1).

Table 6.1: Description of the VOICES database.

VOICES	Speakers	Males	Females	Sentences
Count	12	7	5	50

6.5 Simulation Results

In this section, the most important results from [P7] are summarized. The VOICES database [Kai06] summarized in Table 6.1 was used in the evaluations. STRAIGHT was used as an analysis/synthesis framework and the parameterizations were made according to Table 3.1.

6.5.1 Objective Mapping Performance

Eight M-M, eight F-F, eight M-F, and eight F-M conversion pairs were used in objective evaluation. The objective evaluation of spectral, aperiodicity and voicing mapping performance was averaged over these 32 conversion pairs.

In kernel matrix calculation (6.1), the scaling parameter ϕ was set to 10 and 30 for MCCs and BAPs, respectively. The number of reference vectors was 200 for both MCC and BAP kernel matrices.

Spectral mapping performance was evaluated for a variety of methods. The MCD obtained by the most important methods are summarized in Table 6.2 for the case of 20-sentence training data. In addition to *DKPLS* and *DKPLS* without dynamics (*KPLS*), Table 6.2 shows the MCD for

- *GMM-D*: Standard joint density GMM technique with diagonal cross-covariance matrices [Kai98]
- *MLGMM-D*: The ML parameter generation (a state-of-the-art GMM technique) with diagonal cross-covariance matrices [Tod07a]
- *GMM-PLS*: The technique combining GMM (diagonal covariance matrices) and PLS (Section 5.3)

The optimal number of Gaussians was chosen from a set M={1,2,4,8,16,32,64,128, 256} based on testing data.

The results in Table 6.2 are in line with the results presented in [Tod07a] and [P6], that compared *GMM-D* against *MLGMM-D* and *GMM-PLS*, respectively. The databases used in [Tod07a] and in [P6] were different, but the performance ordering of the techniques was similar to Table 6.2: both *MLGMM-D* and *GMM-PLS* outperformed *GMM-D*. However, Table 6.2 shows that *DKPLS* obtained the lowest MCD. Comparison between *KPLS* and *DKPLS* implies that the use of

Table 6.2: The Mel-cepstral distortion in dB from 32 conversion pairs using 20 training sentences.

Technique	Number of Gaussians	MCD
GMM-D	64	5.29
MLGMM-D	128	5.24
GMM-PLS	8	5.20
KPLS	(-)	5.15
DKPLS (proposed)	(-)	**5.10**

neighboring frame information improves the prediction although the errors were calculated on a frame-by-frame basis.

The BAPs were predicted with and without MCCs. Different techniques were examined. All of them gave a major improvement to the case of using the original source BAPs. The use of MCCs played a more important role than the use of kernel transformation: the average RMSE was smaller when using MCCs without kernel transformation compared to the use of kernel transformation without MCCs [P7]. Nevertheless, the combined use of kernel matrices and MCCs (6.5) resulted in the lowest RMSE. At the first frequency band (0–1 kHz), a decrease of 7.8% in the RMSE was obtained when comparing to a joint density GMM without MCCs. The lower the band, the bigger was the difference between these techniques. At the last frequency band (6–8 kHz) the performance was equal. The last band typically contains more non-predictable (noiselike) information. The more structured sound information especially present in voiced sounds is at lower bands.

Using the proposed voicing prediction (6.5) versus copying the source voicing decisions decreased the voicing decision error by almost 20% for 20-sentence training data [P7].

6.5.2 Subjective Quality Evaluation

Three different systems were evaluated in a MOS test [P7]. Table 6.3 summarizes the three systems and their conversion strategy. If *DKPLS* was used for aperiodicity and voicing prediction, the predictor vectors (6.5) are used. Otherwise; a joint density GMM with full covariance matrices (*GMM-F*) was used for aperiodicity prediction without any auxiliary information from spectrum. The term "Copied" in Table 6.3 means that the binary voicing decisions are copied from the source. The term "Log MV" denotes that F_0 is converted using the MV scaling in a logarithmic scale (4.37).

Table 6.3: Systems evaluated in a MOS test and their conversion strategy for spectrum, aperiodicity, and voicing.

	Spectrum	Aperiodicity	Voicing	F_0
Baseline	*MLGMM-D*	*GMM-F*	Copied	Log MV
Proposed - spectral only	*DKPLS*	*GMM-F*	Copied	Log MV
Proposed	*DKPLS*	*DKPLS*	*DKPLS*	Log MV

Table 6.4: Mean opinion scores with 95% confidence intervals obtained by three different systems.

System	MOS
Baseline	2.21±0.05
Proposed - spectral only	3.28±0.06
Proposed	3.51±0.06

The number of training sentences was 20. Eight randomly selected test sentences from eight different speaker pairs from four categories (M-M, M-F, F-F, F-M) were generated by the three systems of Table 6.3.

The results from the MOS test are shown in Table 6.4. As it can be seen, there is a major difference between the *Proposed - spectral only* and the *Baseline* system. This indicates that the spectral features play an important role. However, the *Proposed* system was rated the highest and the difference between the *Proposed* and the *Proposed - spectral only* system were statistically significant in terms of 95% confidence intervals on the means. The confidence intervals were obtained by assuming a normal distribution with the mean given in Table 6.4 and unknown variance. Detailed description on calculating the confidence intervals can be found in any introductory statistics textbook, e.g. [Joh09]. The results in Table 6.4 imply that further improvements can be achieved with better aperiodicity and voicing prediction techniques.

6.5.3 Identity Evaluation

The success of identity conversion was evaluated both objectively and subjectively, i.e. conducting identification with a speaker recognition system and an ABX test.

A simple speaker identification system was built and used similarly as described in Section 3.3.2. Samples generated by two systems, *Baseline* and *Proposed* (Table 6.3), were recognized by the identification system. MFCCs and

Table 6.5: Average θ_{st} values for the converted sentences specified into inter- and intra-gender conversion pairs, and into male-to-male and female-to-female transformations.

	All	Inter-gender	Intra-gender	Male-to-male	Female-to-female
Proposed	+4.46	+5.35	+3.56	+2.80	+4.33
Baseline	+3.83	+4.60	+3.05	+2.41	+3.69
No conversion	–4.22	–5.06	–3.39	–2.92	–3.86

their deltas were extracted from the samples. The original source and target files were analyzed and synthesized with STRAIGHT before MFCC extraction. The number of sentences for training the speaker identification models was twenty and the same amount of sentences was used to train a VC system. For the speaker identification system, 128 Gaussians and diagonal covariance matrices were used in training the GMM for each speaker. The number of EM iterations was 50 and the results were averaged from five different trials. Twenty converted sentences from 32 speaker pairs were recognized by the identification system.

The identification was performed either using the source and the target models only or the models of all twelve speakers in the database. The proposed system achieved a recognition rate of 99.9% and the baseline 99.6% when only the target and the source models were considered. Using the models of all twelve speakers, the recognition rate was 99.3% and 98.8% for the proposed system and the baseline system, respectively. Thus, both systems succeeded well in spoofing the identification system. Mouchtaris et al. [Mou06] used the same database in VC evaluations. Despite of proposing an algorithm for non-parallel VC, they also gave identification results on the case of having parallel data available. Their recognition result for using all twelve speaker models was about 97%.

In addition to binary recognition results, the performance measure θ_{st} (3.5) was calculated. The averaged results for both systems are shown in Table 6.5. The results are separated for inter- and intra-gender conversion pairs as well as for male-to-male and female-to-female conversion pairs. The higher the θ_{st}, the more successful is the identity transformation. A negative θ_{st} indicates that the sample is closer to the source. In all cases, the proposed system obtained higher θ_{st} compared to the baseline system indicating a better identity conversion performance.

In a subjective speaker identity test, listeners identified intra-gender conversion pairs. Only the proposed system was evaluated and the listeners were not given any training examples. A listener was asked which one of the samples, A or B, sounds more like the person in sample X. A and B contained samples from

source and target speaker, and X was a converted sentence. The sentences were the same for all A, B, and X, since prosodic differences were not obvious due to mimic-style utterances included in the VOICES database [Kai06].

The recognition rate of the desired target speakers was 77.1% but there were significant differences between M-M and F-F conversion pairs [P7]. In F-F conversion, the rate was 90.2% whereas for M-M conversion only 64.0%. This is in line with the objective results of Table 6.5: on average, the male speakers were initially (before conversion) much closer to each other than the female speakers, and after conversion, they still remained closer to each other. Furthermore, one M-M speaker pair obtained a recognition rate of only 23.5% indicating failure of conversion. The objective results did not give any explanation for this. Excluding the specific pair, the overall recognition rate was 80.7%.

Schmidt-Nielsen and Crystal [SN00] compared the speaker verification performance of human listeners and computer algorithms. They concluded that there was a major difference between individual listeners' performance but generally human performance was rather robust to degradation. In [P7], the individual listener recognition rates varied between 0.5 and 0.9. However, one must note that in voice conversion the sample is not originally spoken by the desired speaker but modified to sound like him/her. The data in [P7] was clean speech that probably made the automatic identification to perform so well. Furthermore, listeners typically use also speaking habits as cues for speaker identification [SN00] but speaking style was not transformed in [P7].

6.6 Discussion

The support vector regression approach [Son11] mentioned in Section 4.6 bears some similarity to the technique presented in this chapter. In SVR, the learning is carried out in high-dimensional space and the model only depends on a subset of the training data (support vectors). In SVR training, the aim is to minimize the generalized error bound, that is a combination of training error and a regularization term that controls the complexity. In PLS regression, the selection of the number of latent variables constitutes the regularization.

A major benefit in PLS is that the only choice that needs to be made is the number of latent components. It is a discrete variable that is easy to optimize. The performance of applying PLS on kernel-transformed data in several benchmark classification and regression tasks was equal or even better than other kernel-based support vector approaches but tuning is much easier [Ben03].

The handling of dynamics is different in this thesis. Song et al. [Son11] used the first- and second-order dynamics together with original source features. In conversion, they had to use adaptive median filtering to smooth the converted spectral trajectories. The modeling they used does not solve the temporal correlation problem, since the first- and second-order dynamics become different for

each frame, unlike when using information from the neighboring frames directly. The use of first- and second-order dynamics were not taken explicitly into account such as for example in the ML parameter generation approach [Tod07a].

6.7 Summary

DKPLS is a mapping technique that allows non-linear conversion and improves temporal continuity [P7]. The source features are kernel transformed as a pre-processing step and the mapping function between kernel-transformed source features and original target features is estimated with standard linear PLS regression. Moreover, temporal continuity is taken explicitly into account by concatenating the kernel vectors from adjacent frames. Various subjective and objective experiments confirmed the effectiveness of the technique. *DKPLS* requires very little tuning. Sequential CV is recommended instead of random CV for choosing the number of PLS components with temporally continuous features.

Chapter 7

Spectral Mapping Performance Comparison

To make a comprehensive evaluation of the techniques in [P5, P6, P7] and the most widely used GMM approaches, spectral mapping performance results which have not been published in the author's publications, are reported in this chapter.

All 12 speakers from VOICES database (Table 6.1) serve as a source speaker four times and the target speakers are randomly chosen twice from a set of male speakers and twice from a set of female speakers resulting in 48 conversion pairs. The number of training sentences is either the first five or the first twenty sentences of the database. The remaining 30 sentences are used for testing. The sentences are aligned with DTW. The STRAIGHT spectral envelope is parameterized into 24^{th} order MCCs.

7.1 Mapping Techniques in Comparison

GMM-based techniques used in the evaluation are summarized in Table 7.1. The optimal number of Gaussians is chosen from a set M={1,2,4,8,16,32,64,128,256} based on the testing data.

The predictor vectors $\mathbf{X}_n$ for techniques using PLS regression differ depending on the approach but the original target vectors are always used as output variables ($\mathbf{Y}_n = \mathbf{y}_n$). The predictor vectors for each PLS-based approach is summarized below with additional settings:

- *PLS*: $\mathbf{X}_n = \mathbf{x}_n$
- *DPLS*: $\mathbf{X}_n = [\mathbf{x}_{n-}^{\mathrm{T}}, \mathbf{x}_n^{\mathrm{T}}, \mathbf{x}_{n+}^{\mathrm{T}}]^{\mathrm{T}}$
- *GMM-PLS*: $\mathbf{X}_n = [\omega_{1,n}\mathbf{x}_n^{\mathrm{T}}, \omega_{2,n}\mathbf{x}_n^{\mathrm{T}}, \ldots, \omega_{M,n}\mathbf{x}_n^{\mathrm{T}}]^{\mathrm{T}}$ and M=4 for five training sentences and $M = 8$ for twenty training sentences

Table 7.1: GMM-based approaches used in the evaluation. The numbers denote for the number of Gaussians for five (5) and twenty (20) training sentences.

	Covariance type	Number of Gaussians
GMM-D [Kai98]	Diagonal	16, 64
GMM-F [Kai98]	Full	2, 4
MLGMM-D [Tod07a]	Diagonal	32, 128

- *GMM-DPLS*:

$$\mathbf{X}_n = [\omega_{1,n-}\mathbf{x}_{n-}^{\mathrm{T}}, \omega_{2,n-}\mathbf{x}_{n-}^{\mathrm{T}}, \ldots, \omega_{M,n-}\mathbf{x}_{n-}^{\mathrm{T}}, \omega_{1,n}\mathbf{x}_{n}^{\mathrm{T}}, \omega_{2,n}\mathbf{x}_{n}^{\mathrm{T}}, \ldots, \omega_{M,n}\mathbf{x}_{n}^{\mathrm{T}}, \omega_{1,n+}\mathbf{x}_{n+}^{\mathrm{T}}, \omega_{2,n+}\mathbf{x}_{n+}^{\mathrm{T}}, \ldots, \omega_{M,n+}\mathbf{x}_{n+}^{\mathrm{T}}]^{\mathrm{T}}$$

  and M=4 for five training sentences and M=8 for twenty training sentences

- *KPLS*: $\mathbf{X}_n = \tilde{\mathbf{k}}_n$ and ϕ=10, C=200

- *DKPLS*: $\mathbf{X}_n = [\tilde{\mathbf{k}}_{n-}^{\mathrm{T}}, \tilde{\mathbf{k}}_{n}^{\mathrm{T}}, \tilde{\mathbf{k}}_{n+}^{\mathrm{T}}]$ and ϕ=10, C=200

In addition, a speech sequence optimization approach using particle filtering (referred to as *PF-PLS*) [P5] is evaluated. The number of particles is set to 150. For static feature conversion, k-means algorithm is used for clustering $\mathbf{x}_n$ augmented with $\mathbf{y}_n$ into K classes and a PLS regression model for each class is built. Class-dependent error variances are determined using CV. The number of clusters is five and ten for five and twenty training sentences, respectively. It was chosen from a set K= {1,2,5,10,15,20} based on the test data. The model for dynamics is an offset term that is now predicted from the source features as

$$\rho_t(s_{t-1}) = s_{t-1} + \boldsymbol{\beta} \begin{bmatrix} \mathbf{x}_t \\ \mathbf{x}_t - \mathbf{x}_{t-1} \end{bmatrix} \tag{7.1}$$

where the transform $\boldsymbol{\beta}$ is learned between the source data and target dynamics using PLS regression. For each target MCC dimension, a separate model $\rho_t(s_{t-1})$ is formed but all static and dynamic features of the source are used in the prediction. The error variance of $\rho_t(s_{t-1})$ is frame-independent as in [P5].

7.2 Simulation Results

Table 7.2 shows the averaged MCD for 5- and 20-sentence training data. As it can be seen, all the proposed techniques (*PF-PLS*, *GMM-PLS*, *GMM-DPLS*, *KPLS* and *DKPLS*) result in lower MCD values compared to GMM-based techniques.

Table 7.2: The Mel-cepstral distortion in dB for different techniques averaged from 48 speaker pairs using five (5) and twenty (20) training sentences. For each technique, the 95% confidence interval is the MCD average plus or minus 0.003 dB. Before conversion the average MCD was 7.89 dB.

Training sentences	5	20
PLS	5.57	5.30
DPLS	5.53	5.24
GMM-D	5.59	5.20
GMM-F	5.55	5.14
MLGMM-D	5.52	5.15
PF-PLS	5.51	5.09
GMM-PLS	5.50	5.10
GMM-DPLS	5.47	5.06
KPLS	5.41	5.04
DKPLS	**5.37**	**5.00**

With five sentences, simple techniques (*PLS* and *DPLS*) perform reasonably well but with twenty sentences, they are too simple to fully exploit the training data. In both training data cases, the lowest MCD is obtained with *DKPLS*. The margin of errors (95% confidence intervals) for each technique is relatively small, ranging from 0.0030 to 0.0033. The confidence intervals are very small compared to the MCD averages indicating that the results are statistically significant.

Speaker-pair specific MCD results using *GMM-F*, *ML-GMM*, and *DKPLS* are shown in Figure 7.1 and Figure 7.2 for 5 and 20 training sentences, respectively. It can be seen that the performance order is not dependent on a particular speaker pair; *DKPLS* performs the best regardless of the speaker pair or the amount of training sentences. Due to limited space, Figure 7.1 and Figure 7.2 do not contain confidence intervals. At speaker level, *DKPLS* was compared to *GMM-F* and *ML-GMM* in terms of statistical significance of the MCD result. The smaller MCD of *DKPLS* was found to be statistically significant for all 48 speaker pairs when comparing to *ML-GMM* or to *GMM-F* in the case of 5 training sentences. With 20 training sentences, there was only one speaker pair where the difference was not statistically significant.

In addition, an average variance ratio (VR) is calculated. VR has been used to assess the oversmoothing effect in [Tod07a, Ben11, God11]. However, it should be noted that the VR is somewhat misleading, since undesired abrupt changes in the converted trajectory make it higher. The VR is calculated for each MCC by dividing the variance of the converted target MCCs by the variance of the original

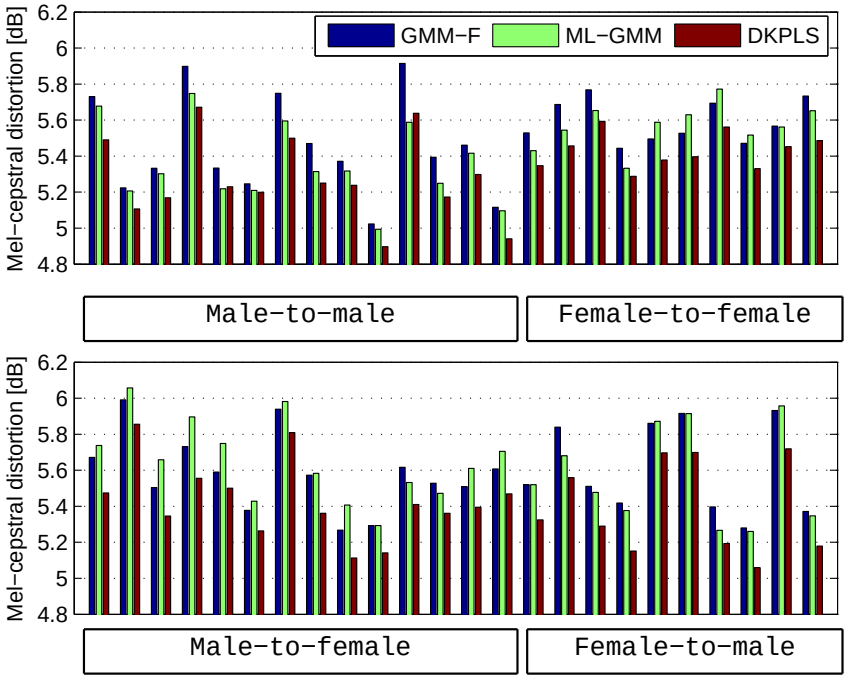


Figure 7.1: The Mel-cepstral distortion for different speaker pairs with 5 training sentences using *GMM-F*, *ML-GMM*, and *DKPLS*.

target MCCs in a sentence. The VRs are averaged over all testing sentences and speaker pairs. The VRs for different MCC dimensions are shown in Figure 7.3 for *GMM-D*, *MLGMM-D* and *DKPLS* using 20 training sentences. The VR of *MLGMM-D* is the smallest indicating that it suffers the most from oversmoothing. *DKPLS* succeeds in obtaining both high VRs and the smallest MCD (Table 7.2).

The mean of VRs from each MCC was used as an objective measure in [God11, Ben11]. It can be observed in Figure 7.3 that the last MCCs have the lowest VR. However, the last MCCs are not as important as the first ones. The last MCCs had initially smaller variance and thus were given less weight in the mapping function estimation. Hence, they should not be given too much weight when assessing the oversmoothing effect either.

Figure 7.4 shows an example of the converted and the original target trajectory for the third MCC. MCCs were converted using *GMM-D*, *MLGMM-D*, and *DKPLS*. There are no major jumps in the *DKPLS* trajectory but *GMM-D* sometimes suffers from those. The trajectory of *MLGMM-D* is at times too smooth and does not capture big variations. Further, small sawtooth-like fluctuation is observed in the trajectory. The problem was reported by Chen et al. [Che10] and they suggested to avoid the problem using acceleration parameters or alternative delta window coefficients.

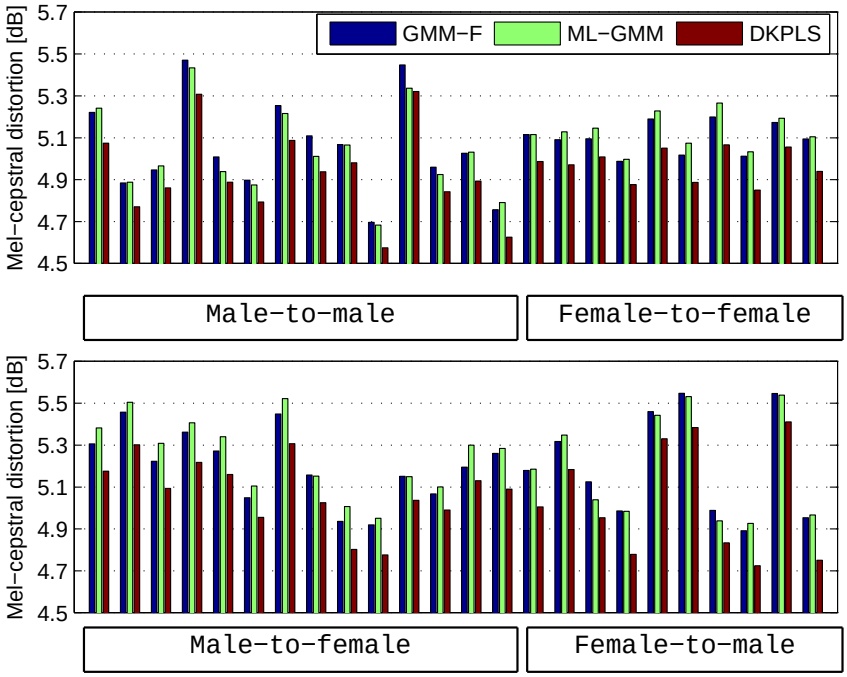


Figure 7.2: The Mel-cepstral distortion for different speaker pairs with 20 training sentences using *GMM-F*, *ML-GMM*, and *DKPLS*.

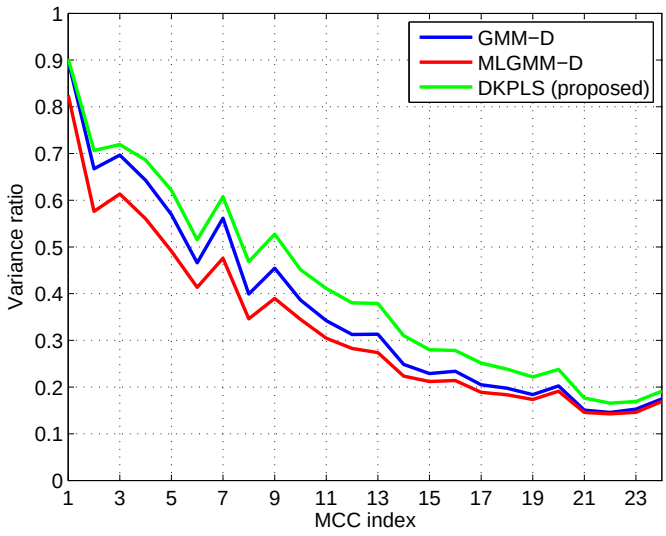
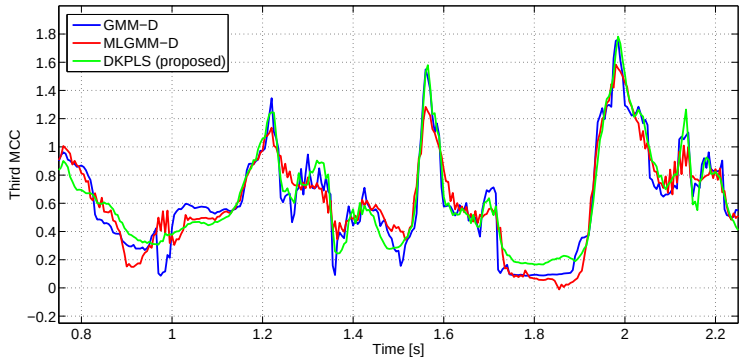
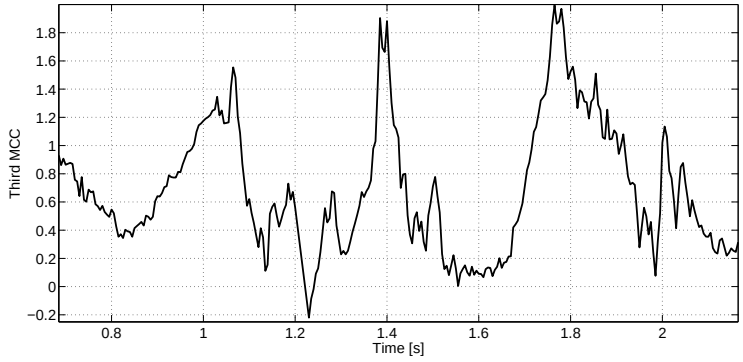


Figure 7.3: The variance ratio between the converted target and the original target for each MCC.



(a)



(b)

Figure 7.4: An example of (a) converted and (b) original target trajectories for 3^{rd} MCC corresponding to the same phoneme sequence. Note that phoneme durations and timings are different.

7.3 Summary

Spectral conversion techniques proposed in this thesis as well as three reference techniques were compared using the same dataset and the same A/S system and parameterization. Objective spectral conversion performance was evaluated for 48 speaker pairs. All proposed techniques obtained lower cepstral distortion value than the reference techniques for both 5- and 20-sentence training data cases. *DKPLS* described in Chapter 6 performed the best.

Chapter 8

Conclusions, Discussion and Future Work

This thesis has presented several mapping techniques for voice conversion. GMM-based conversion has been a dominating technique for a long time despite its problems. This thesis has considered the problems and proposed a solution that combines a GMM with PLS regression (*GMM-PLS*). In addition, a new technique (*DKPLS*) has been proposed. Both of them are based on transforming the original source features into another domain either with the help of data division given by a GMM (*GMM-PLS*) or with a kernel transformation (*DKPLS*). Then a regression matrix is estimated. This is accomplished by PLS regression in order to avoid overfitting. The use of the proposed sequential cross-validation gives realistic estimates of the performance over unseen speech data. Both the techniques require very little tuning and are most effective when there is a relatively small amount of training vectors available. How useful they are for a large amount of training data requires further investigation.

The temporal continuity of speech features is important. An algorithm to balance between frame-based conversion error and temporal continuity through optimization of a cost function was proposed. More research is needed for tuning the cost function to fully exploit the technique. In addition, posterior probability smoothing was proposed for GMM-based conversion. To improve temporal continuity more explicitly and elaborately, the source data from the current frame was augmented with the data from the previous and the next frames.

The proposed techniques and well-known GMM-based benchmark methods were compared. All the proposed methods were found to decrease the objective spectral distortion value compared to the GMM-based approaches. A large variety of speaker pairs was used in this thesis in order to provide more reliable results. It was shown that *DKPLS* performed the best in terms of objective measures.

To make a complete voice conversion system, also excitation needs to be converted. The BAPs of the target were predicted from both source BAPs and MCCs using *DKPLS*. The binary voicing decision is usually copied from the

source, but in this thesis, it was predicted with *DKPLS* by using source voicing decisions, BAPs and MCCs. Improved prediction of BAPs and voicing decisions gave a statistically significant improvement in the subjective quality. This thesis also proposed a prosody conversion algorithm operating at syllable level. It was found to outperform a conventional F_0 conversion algorithm. Prosody conversion is likely to be more important when the target speakers are familiar to the listeners.

The effect of different A/S framework and parameterization has not been considered in the literature, although they play a major role in the speech quality. It is likely that the converted samples using a different A/S framework differ from each other more than the samples converted by different statistical mapping techniques. The artifacts in the converted speech may result from the failure of the A/S framework. Furthermore, the specific selection of the features representing the spectral envelope has not been addressed. A potentially important future direction is enhanced parameterization. For example MCCs may not give enough information of inter-speaker vocal tract correlation. In addition, since the third and the fourth formant are the most important vocal tract features for speaker individuality, their frequency region could be given more weight.

The techniques described in this thesis are based on the assumption that parallel databases are available. When using parallel data, the alignment procedure is important and should not be ignored but it is enough to carry out a simple DTW algorithm. Text-independent VC is a topic of interest since in real-life situations parallel corpora is not easy to obtain. Nevertheless, most non-parallel approaches treat the data alignment as a separate step before estimating the conversion function for segmentally aligned source and target features. The techniques described in this thesis are thus applicable after solving the alignment problem. The ideal goal would be to exploit target speech data from various environments and for example collect training data when a person is speaking on a cellular phone [Hel10]. No effort from the user would be required. However, the development of better and better VC technologies also poses a threat for applications relying on speaker verification systems. It is important to study this topic in the future.

The lack of benchmark databases and techniques is problematic. Most studies use the conventional joint density GMM approach as a baseline technique, which cannot be considered as a state-of-the-art technique due to many problems attached to it. It can be concluded that other benchmarks techniques should be used or at least use PLS regression in GMM-based conversion together with posterior probability smoothing or source data augmentation from consecutive frames.

Finally, the Blizzard Challenge [Bli] is a yearly organized "competition" for comparing different speech synthesizers based on the same data. This kind of evaluation campaign would be beneficial for voice conversion to compare different techniques and parameterizations.

References

[Abe88] M. Abe, S. Nakamura, K. Shikano, and H. Kuwabara, "Voice conversion through vector quantization," in *Proc. of the ICASSP*, New York, USA, Apr. 1988, pp. 655–658.

[Ami07] K. Amino, T. Arai, and T. Sugawara, "Effects of the phonological contents on perceptual speaker identification," in *Proc. of Speaker Classification*, 2007, pp. 83–92.

[Ars99] L. Arslan, "Speaker transformation algorithm using segmental codebooks (STASC)," *Speech Communication*, vol. 28, no. 3, pp. 211–226, Jul. 1999.

[Atk97] I. Atkinson, S. Yeldner, and A. Kondoz, "High quality split band LPC vocoder operating at low bit rates," in *Proc. of the ICASSP*, vol. 2, Munich, Germany, Apr. 1997, pp. 1559–1562.

[Ben03] K. Bennett and M. Embrechts, "An optimization perspective on kernel partial least squares regression," in *Advances in Learning Theory: Methods, Models and Applications*, 2003, pp. 227–250.

[Ben08] J. Benesty, M. M. Sondhi, and Y. Huang, Eds., *Springer Handbook of Speech Processing*. Berlin: Springer, 2008.

[Ben11] H. Benisty and D. Malah, "Voice conversion using GMM with enhanced global variance," in *Proc. of the Interspeech*, Florence, Italy, Aug. 2011, pp. 669–672.

[Bli] Blizzard, "The Blizzard challenge," http://festvox.org/blizzard/.

[Boo80] F. L. Bookstein, "Data analysis by partial least squares," in *Evaluation of Econometric Models*. Academic Press, 1980, pp. 75–90.

[Bre07] R. Brereton, *Applied chemometrics for scientists*. West Sussex, England: John Wiley & Sons, 2007.

[Cha98] D. Chapell and J. Hansen, "Speaker-specific pitch contour modelling and modification," in *Proc. of the ICASSP*, Seattle, USA, May 1998, pp. 885–888.

[Che03] Y. Chen, M. Chu, E. Chang, J. Liu, and R. Liu, "Voice conversion with smoothed GMM and MAP adaptation," in *Proc. of the Eurospeech*, Geneva, Switzerland, Sep. 2003, pp. 2413–2416.

[Che10] Y.-N. Chen, Z.-J. Yan, and F. Soong, "A perceptual study of acceleration parameters in HMM-based TTS," in *Proc. of the Interspeech*, Makuhari, Japan, Sep. 2010, pp. 426–429.

[Dav80] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no. 4, pp. 357–366, Aug. 1980.

[Dem77] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 39, no. 1, pp. 1–38, 1977.

[Des10] S. Desai, A. Black, B. Yegnanarayana, and K. Prahallad, "Spectral mapping using artificial neural networks for voice conversion," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 5, pp. 954–964, Jul. 2010.

[dJ93] S. de Jong, "SIMPLS: An alternative approach to partial least squares regression," *Chemometrics and Intelligent Laboratory Systems*, vol. 18, no. 3, pp. 251–263, Mar. 1993.

[Dut07] T. Dutoit, A. Holzapfel, M. Jottrand, A. Moinet, J. Perez, and Y. Stylianou, "Towards a voice conversion system based on frame selection," in *Proc. of the ICASSP*, vol. 4, Honolulu, Hawaii, May 2007, pp. 513–516.

[Dux06a] H. Duxans and A. Bonafonte, "Residual conversion versus prediction on voice morphing systems," in *Proc. of the ICASSP*, Toulouse, France, May 2006.

[Dux06b] H. Duxans, D. Erro, J. Perez, F. Diego, A. Bonafonte, and A. Moreno, "Voice conversion of non-aligned data using unit selection," in *Proc. of the TC-STAR Workshop on Speech-to-Speech Translation*, Barcelona, Spain, Jun. 2006, pp. 237–242.

[Eat94] J. Eatock and J. Mason, "A quantitative assessment of the relative speaker discriminating properties of phonemes," in *Proc. of the ICASSP*, vol. I, Adelaide, Australia, Apr. 1994, pp. 133–136.

[Emb05] M. J. Embrechts and B. Szymanski, *Computationally Intelligent Hybrid Systems*. Wiley, 2005, ch. Introduction to Scientific Data Mining: Direct Kernel Methods & Applications, pp. 317–365.

[Err10a] 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*, vol. 18, no. 5, pp. 944–953, Jul. 2010.

[Err10b] D. Erro, A. Moreno, and A. Bonafonte, "Voice conversion based on weighted frequency warping," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 5, pp. 922–931, Jul. 2010.

[Esl11] M. Eslami, H. Sheikhzadeh, and A. Sayadiyan, "Quality improvement of voice conversion systems based on trellis structured vector quantization," in *Proc. of the Interspeech*, Florence, Italy, Aug. 2011, pp. 665–668.

[Far10] M. Farrus, M. Wagner, D. Erro, and J. Hernando, "Automatic speaker recognition as a measurement of voice imitation and conversion," *International Journal of Speech Language and the Law*, vol. 17, no. 1, 2010.

[Gem92] S. Geman, E. Bienenstock, and R. Doursat, "Neural networks and the bias/variance dilemma," *Neural Communication*, no. 4, pp. 1–58, Jan. 1992.

[Gil03] B. Gillet and S. King, "Transforming F0 contours," in *Proc. of the Eurospeech*, Geneva, Switzerland, Sep. 2003, pp. 101–104.

[God01] S. Godsill, A. Doucet, and M. West, "Maximum a posteriori sequence estimation using Monte Carlo particle filters," *Annals of the Institute of Statistical Mathematics*, vol. 53, no. 1, pp. 82–96, Mar. 2001.

[God11] E. Godoy, O. Rosec, and T. Chonavel, "Spectral envelope transformation using DFW and amplitude scaling for voice conversion with parallel or nonparallel corpora," in *Proc. of the Interspeech*, Florence, Italy, Aug. 2011, pp. 673–676.

[Gor93] N. Gordon, D. Salmond, and A. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *Radar and Signal Processing*, vol. 140, no. 2, pp. 107–113, Apr. 1993.

[Gu06] L. Gu, Y. Gao, F.-H. Liu, and M. Picheny, "Concept-based speech-to-speech translation using maximum entropy models for statistical natural concept generation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 2, pp. 377–392, Mar. 2006.

[Hel08] E. Helander, J. Nurminen, and M. Gabbouj, "LSF mapping for voice conversion with very small training sets," in *Proc. of the ICASSP*, Las Vegas, USA, May 2008, pp. 4669–4672.

[Hel10] E. Helander, J. Nurminen, V. Popa, and J. Tian, "Voice conversion training and data collection, US patent 7813924," Oct. 2010.

[Hel11] E. Helander and J. Nurminen, "Prosody conversion, US patent 7996222," Aug. 2011.

[Hua01] X. Huang, A. Acero, and H.-W. Hon, *Spoken Language Processing: A Guide to Theory, Algorithm and System Development*. Prentice Hall PTR, May 2001.

[Hul99] J. Hulland, "Use of partial least squares (PLS) in strategic management research: a review of four recent studies," *Strategic Management Journal*, vol. 20, no. 2, pp. 195–204, Feb. 1999.

[Hun96] A. J. Hunt and A. W. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," in *Proc. of the ICASSP*, Atlanta, USA, May 1996, pp. 373–376.

[Joh09] R. Johnson and G. Bhattacharyya, *Statistics: Principles and Methods*. John Wiley & Sons, 2009.

[Jun05] S.-A. Jun, *Prosodic typology: the phonology of intonation and phrasing*. Oxford University Press, 2005.

[Kai98] A. Kain and M. W. Macon, "Spectral voice conversion for text-to-speech synthesis," in *Proc. of the ICASSP*, vol. I, Seattle, USA, May 1998, pp. 285–288.

[Kai01] A. Kain and M. Macon, "Design and evaluation of a voice conversion algorithm based on spectral envelope mapping and residual prediction," in *Proc. of the ICASSP*, Salt Lake City, USA, May 2001, pp. 813–816.

[Kai06] A. Kain, *CLSU: Voices*. Linguistic Data Consortium, Philadelphia, USA, 2006.

[Kaw99a] H. Kawahara, I. Masuda-Katsuse, and A. de Cheveignè, "Restructuring speech representations using a pitch-adaptive time-frequency smoothing and a instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds," *Speech Communication*, vol. 27, pp. 187–207, Apr. 1999.

[Kaw99b] H. Kawanami, Y. Iwami, T. Toda, H. Saruwatari, and K. Shikano, "GMM-based voice conversion applied to emotional speech synthesis," *IEEE Transactions on Speech and Audio Processing*, vol. 7, pp. 2401–2404, Jan. 1999.

[Kaw06] H. Kawahara, "STRAIGHT, exploitation of the other aspect of VOCODER: Perceptually isomorphic decomposition of speech sounds," *Acoustical Science & Technology*, vol. 27, no. 6, pp. 349–353, Nov. 2006.

[Kit07] T. Kitamura and M. Akagi, "Speaker individualities in speech spectral envelopes and fundamental frequency contours," in *Speaker Classification II*, ser. Lecture Notes in Computer Science, C. Möller, Ed. Springer Berlin, Heidelberg, 2007, vol. 4441, pp. 157–176.

[Koi95] K. Koishida, K. Tokuda, T. Kobayashi, and S. Imai, "CELP coding based on Mel-cepstral analysis," in *Proc. of the ICASSP*, vol. 1, Detroit, USA, May 1995, pp. 33–36.

[Kom03] J. Kominek and A. Black, "CMU ARCTIC databases for speech synthesis," Carnegie Mellon University, Tech. Rep., 2003.

[Kon04] A. M. Kondoz, *Digital speech coding for low bit rate communication systems*. England: Wiley and Sons, 2004.

[Kre11] J. Kreiman and D. Sidtis, *Foundations of Voice Studies: An Interdisciplinary Approach to Voice Production and Perception*. West Sussex, England: Wiley-Blackwell, 2011.

[Kri11] A. Krishnan, L. J. Williams, A. R. McIntosh, and H. Abdi, "Partial least squares (PLS) methods for neuroimaging: A tutorial and review," *Neuroimage*, vol. 56, no. 2, pp. 455–475, May 2011.

[Kuh00] R. Kuhn, J. Janqua, P. Nguyen, and N. Niedzielski, "Rapid speaker adaptation in eigenvoice space," *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 6, pp. 695–707, Nov. 2000.

[Kuw95] H. Kuwabara and Y. Sagisaka, "Acoustic characteristics of speaker individuality: Control and conversion," *Speech Communication*, vol. 16, no. 2, pp. 165–173, Feb. 1995.

[Lav01] Y. Lavner, J. Rosenhouse, and I. Gath, "The prototype model in speaker identification by human listeners," *International Journal of Speech Technology*, vol. 4, pp. 63–74, Mar. 2001.

[Lin10] Z.-H. Ling, Y. Hu, and L.-R. Dai, "Global variance modeling on the log power spectrum of LSPs for HMM-based speech synthesis," in *Proc. of the Interspeech*, Makuhari, Japan, Sep. 2010, pp. 825–828.

[Lol08] D. Lolive, N. Barbot, and O. Boeffard, "Pitch and duration transformation with non-parallel data," in *Proc. of Speech Prosody*, Campinas, Brazil, May 2008, pp. 111–114.

[Mar79] K. Mardia, J. Kent, and J. Bibby, *Multivariate analysis*, ser. Probability and mathematical statistics. Acad. Press, 1979.

[McA90] R. McAulay and T. Quatieri, "Pitch estimation and voicing detection based on a sinusoidal speech model," in *Proc. of the ICASSP*, Albuquerque, USA, Apr. 1990, pp. 249–252.

[Mes07] L. Mesbashi, V. Barreaud, and O. Boeffard, "Comparing GMM-based speech transformation systems," in *Proc. of the Interspeech*, Antwerp, Belgium, Aug. 2007, pp. 1989–1992.

[Míg12] J. Míguez, D. Crisan, and P. Djurić, "On the convergence of two sequential Monte Carlo methods for maximum a posteriori sequence estimation and stochastic global optimization," 2012, to appear in Statistics and Computing.

[Mou06] A. Mouchtaris, J. Van der Spiegel, and P. Mueller, "Nonparallel training for voice conversion based on a parameter adaptation approach," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 3, pp. 952–963, May 2006.

[Mou07] A. Mouchtaris, J. Van der Spiegel, P. Mueller, and P. Tsakalides, "A spectral conversion approach to single-channel speech enhancement," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 4, pp. 1180–1193, May 2007.

[Mül07] M. Müller, "Dynamic time warping," in *Information retrieval for music and motion*. Springer, 2007, pp. 69–84.

[Nar95] 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, Feb. 1995.

[Ngu02] D. Nguyen and D. Rocke, "Tumor classification by partial least squares using microarray gene expression data," *Bioinformatics*, vol. 18, no. 1, pp. 39–50, Jan. 2002.

[Nur06] J. Nurminen, V. Popa, J. Tian, Y. Tang, and I. Kiss, "A parametric approach for voice conversion," in *Proc. of the TC-STAR Workshop on Speech-to-Speech Translation*, Barcelona, Spain, Jun. 2006, pp. 225–229.

[Oht06] Y. Ohtani, T. Toda, H. Saruwatari, and K. Shikano, "Maximum likelihood voice conversion based on GMM with STRAIGHT mixed excitation," in *Proc. of the Interspeech*, Pittsburgh, USA, Sep. 2006, pp. 2266–2269.

[Oht10] Y. Ohtani, T. Toda, H. Saruwatari, and K. Shikano, "Non-parallel training for many-to-many eigenvoice conversion," in *Proc. of the ICASSP*, Dallas, USA, Mar. 2010, pp. 4822–4825.

[Pal93] K. Paliwal and B. Atal, "Efficient vector quantization of LPC parameters at 24 bits/frame," *IEEE Transactions on Speech and Audio Processing*, vol. 1, no. 1, pp. 3–14, Jan. 1993.

[Par00] K.-Y. Park and H. S. Kim, "Narrowband to wideband conversion of speech using GMM based transformation," in *Proc. of the ICASSP*, vol. 3, Istanbul, Turkey, Jun. 2000, pp. 1843–1846.

[Pop12] V. Popa, H. Silén, J. Nurminen, and M. Gabbouj, "Local linear transformation for voice conversion," in *Proc. of the ICASSP*, Kyoto, Japan, Mar. 2012, pp. 4517–4520.

[Pri06] A. Pribilova and J. Pribil, "Non-linear frequency scale mapping for voice conversion in text-to-speech system with cepstral description," *Speech Communication*, vol. 48, no. 12, pp. 1691–1703, Dec. 2006.

[Qin96] S. J. Qin and T. J. McAvoy, "Nonlinear FIR modeling via a neural net PLS approach," *Computers & Chemical Engineering*, vol. 20, no. 2, pp. 147–159, Feb. 1996.

[Rab93] L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition*. New Jersey, USA: Prentice Hall Signal Processing Series, Prentice-Hall Inc., 1993.

[Ren04] D. Rentzos, S. Vaseghi, Y. Q., and C.-H. Ho, "Voice conversion through transformation of spectral and intonation features," in *Proc. of the ICASSP*, vol. 1, Montreal, Canada, May 2004, pp. 21–24.

[Rey95] D. Reynolds and R. Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 1, pp. 72–83, Jan. 1995.

[Rey00] D. A. Reynolds, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, vol. 10, no. 1-3, pp. 19–41, Jan. 2000.

[Ric88] N. L. Ricker, "The use of biased least-squares estimators for parameters in discrete-time pulse-response models," *Industrial & Engineering Chemistry Research*, vol. 27, pp. 343–350, Feb. 1988.

[Ros02] R. Rosipal and L. Trejo, "Kernel partial least squares regression in reproducing kernel Hilbert space," *Journal of Machine Learning Research*, vol. 2, pp. 97–123, Dec. 2002.

[Sch98] B. Schölkopf, A. J. Smola, and K.-R. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Computation*, vol. 10, no. 5, pp. 1299–1319, Jul. 1998.

[Shi10] T. Shinozaki, S. Furui, and T. Kawahara, "Gaussian mixture optimization based on efficient cross-validation," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 3, pp. 540–547, Jun. 2010.

[Shu06] Z. Shuang, R. Bakis, and Y. Qin, "Voice conversion based on mapping formants," in *Proc. of the TC-STAR Workshop on Speech-to-Speech Translation*, Barcelona, Spain, Jun. 2006, pp. 219–223.

[Shu08] Z. Shuang, F. Meng, and Y. Qin, "Voice conversion by combining frequency warping with unit selection," in *Proc. of the ICASSP*, Las Vegas, USA, Apr. 2008, pp. 4661–4664.

[Sil09] H. Silén, E. Helander, J. Nurminen, and M. Gabbouj, "Parameterization of vocal fry in HMM-based speech synthesis," in *Proc. of the Interspeech*, Brighton, UK, Sep. 2009, pp. 1775–1778.

[Sil11] H. Silén, E. Helander, and M. Gabbouj, "Prediction of voice aperiodicity based on spectral representations in HMM-based speech synthesis," in *Proc. of the Interspeech*, Florence, Italy, Aug. 2011, pp. 105–108.

[SN00] A. Schmidt-Nielsen and T. H. Crystal, "Speaker verification by human listeners: Experiments comparing human and machine performance using the NIST 1998 speaker evaluation data," *Digital Signal Processing*, vol. 10, no. 1-3, pp. 249–266, Jan. 2000.

[Son98] G. Sonntag and T. Portele, "PURR - a method for prosody evaluation and investigation," *Journal of Computer and Language*, vol. 12, no. 4, pp. 437–451, Oct. 1998.

[Son11] P. Song, Y. Bao, L. Zhao, and C. Zou, "Voice conversion using support vector regression," *Electronics Letters*, vol. 47, no. 18, pp. 1045–1046, Sep. 2011.

[SPT] SPTK, "Speech signal processing toolkit (SPTK) version 3.4.1," http://sp-tk.sourceforge.net/.

[Sri11] B. V. Srinivasan, D. Garcia-Romero, D. N. Zotkin, and R. Duraiswami, "Kernal partial least squares for speaker recognition," in *Proc. of the Interspeech*, Florence, Italy, Aug. 2011, pp. 493–496.

[Sty98] Y. Stylianou, O. Cappe, and E. Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Transactions on Speech and Audio Processing*, vol. 6, no. 2, pp. 131–142, Mar. 1998.

[Sty05] Y. Stylianou, "Modeling speech based on harmonic plus noise models," in *Nonlinear Speech Modeling and Applications*, ser. Lecture Notes in Computer Science, G. Chollet, A. Esposito, M. Faundez-Zanuy, and M. Marinaro, Eds. Springer Berlin / Heidelberg, 2005, vol. 3445, pp. 244–260.

[Sün03] D. Sündermann and H. Ney, "VTLN-based crosslanguage voice conversion," in *Proc. of the ASRU*, 2003, pp. 676–681.

[Sün06a] D. Sündermann, H. Höge, A. Bonafonte, H. Ney, A. Black, and S. Narayanan, "Text-independent voice conversion based on unit selection," in *Proc. of the ICASSP*, vol. 1, Toulouse, France, May 2006, pp. 81–84.

[Sün06b] D. Sündermann, H. Höge, A. Bonafonte, H. Ney, and J. Hirschberg, "Text-independent cross-language voice conversion," in *Proc. of the Interspeech*, Pittsburgh, USA, Sep. 2006, pp. 2262–2265.

[Tao10] J. Tao, M. Zhang, J. Nurminen, J. Tian, and X. Wang, "Supervisory data alignment for text-independent voice conversion," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 5, pp. 932–943, Jul. 2010.

[Tod01] T. Toda, H. Saruwatari, and K. Shikano, "Voice conversion algorithm based on Gaussian mixture model with dynamic frequency warping of STRAIGHT spectrum," in *Proc. of the ICASSP*, Salt Lake City, USA, May 2001, pp. 841–844.

[Tod05] T. Toda and K. Tokuda, "Speech parameter generation algorithm considering global variance for HMM-based speech synthesis," in *Proc. of the Interspeech*, Lisbon, Portugal, Sep. 2005, pp. 2801–2804.

[Tod07a] T. Toda, A. Black, and K. Tokuda, "Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 8, pp. 2222–2235, Nov. 2007.

[Tod07b] T. Toda, Y. Ohtani, and K. Shikano, "One-to-many and many-to-one voice conversion based on eigenvoices," in *Proc. of the ICASSP*, vol. 4, Honolulu, Hawaii, Apr. 2007, pp. 1249–1252.

[Tod08] T. Toda, A. W. Black, and K. Tokuda, "Statistical mapping between articulatory movements and acoustic spectrum using a Gaussian mixture model," *Speech Communication*, vol. 50, no. 3, pp. 215–227, Mar. 2008.

[Tod09] T. Toda, K. Nakamura, H. Sekimoto, and K. Shikano, "Voice conversion for various types of body transmitted speech," in *Proc. of the ICASSP*, Taipei, Taiwan, Apr. 2009, pp. 3601–3604.

[Tok94] K. Tokuda, T. Kobayashi, T. Masuko, and S. Imai, "Mel-generalized cepstral analysis - a unified approach to speech spectral estimation," in *Proc. of the ICSLP*, Yokohama, Japan, Sep. 1994, pp. 1043–1046.

[Tok95] K. Tokuda, T. Kobayashi, and S. Imai, "Adaptive cepstral analysis of speech," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 6, pp. 481–489, Nov. 1995.

[Tok00] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," in *Proc. of the ICASSP*, Istanbul, Turkey, Jun. 2000, pp. 1315–1318.

[Tok02] K. Tokuda, H. Zen, and A. W. Black, "An HMM-based speech synthesis system applied to English," in *Proc. of the IEEE Workshop on Speech Synthesis*, Santa Monica, USA, Sep. 2002, pp. 227–230.

[Tot08] A. Toth and A. W. Black, "Incorporating durational modification in voice transformation," in *Proc. of the Interspeech*, Brisbane, Australia, Sep. 2008, pp. 1088–1091.

[Tur03] O. Turk and L. Arslan, "Voice conversion methods for vocal tract and pitch contour modification," in *Proc. of the Eurospeech*, Geneva, Switzerland, Sep. 2003, pp. 2845–2848.

[Tur06] O. Turk and L. Arslan, "Robust processing techniques for voice conversion," *Computer Speech and Language*, vol. 4(20), pp. 441–467, Oct. 2006.

[Uto06] Y. Uto, Y. Nankaku, T. Toda, A. Lee, and K. Tokuda, "Voice conversion based on mixtures of factor analyzers," in *Proc. of the Interspeech*, Pittsburgh, USA, Sep. 2006, pp. 2278–2281.

[Val92] H. Valbret, E. Moulines, and J. Tubach, "Voice transformation using PSOLA technique," in *Proc. of the ICASSP*, vol. 1, San Francisco, USA, Mar. 1992, pp. 145–148.

[Vin10] V. E. Vinzi, W. W. Chin, J. Henseler, and H. Wang, *Handbook of Partial Least Squares: Concepts, Methods and Applications*, 1st ed. Springer Publishing Company, Incorporated, 2010.

[Wan09] J. Wang, H. Lu, K. Plataniotis, and J. Lu, "Gaussian kernel optimization for pattern classification," *Pattern Recognition*, vol. 42, no. 7, pp. 1237–1247, Jul. 2009.

[Whi76] G. White and R. Neely, "Speech recognition experiments with linear prediction, bandpass filtering, and dynamic programming," *IEEE Transactions on Speech and Signal Processing*, vol. 24, no. 2, pp. 183–188, Apr. 1976.

[Wik11] Wikipedia, "Speech production— Wikipedia, the free encyclopedia," 2011, [accessed 10-Dec.-2011]. [Online]. Available: http://en.wikipedia.org/wiki/Speech_production

[Wu10] Z.-Z. Wu, T. Kinnunen, E. S. Chng, and H. Li, "Text-independent F0 transformation with non-parallel data for voice conversion," in *Proc. of the Interspeech*, Makuhari, Japan, Sep. 2010, pp. 1732–1735.

[Yam09a] J. Yamagishi, T. Kobayashi, Y. Nakano, K. Ogata, and J. Isogai, "Analysis of speaker adaptation algorithms for HMM-based speech synthesis and a constrained SMAPLR adaptation algorithm," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 1, pp. 66–83, 2009.

[Yam09b] J. Yamagishi, M. Lincoln, S. King, J. Dines, M. Gibson, J. Tian, and Y. Guan, "Analysis of unsupervised and noise-robust speaker-adaptive HMM-based speech synthesis systems toward a unified ASR and TTS framework," in *Proc. of the Blizzard Challenge*, 2009.

[Yam10] J. Yamagishi, B. Usabaev, S. King, O. Watts, J. Dines, J. Tian, Y. Guan, R. Hu, K. Oura, Y.-J. Wu, K. Tokuda, R. Karhila, and

M. Kurimo, "Thousands of voices for HMM-based speech synthesis-analysis and application of TTS systems built on various ASR corpora," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 5, pp. 984–1004, Jul. 2010.

[Ye06] H. Ye and S. Young, "Quality-enhanced voice morphing using maximum likelihood transformations," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1301–1312, Jul. 2006.

[Yut09] K. Yutani, Y. Uto, Y. Nankaku, A. Lee, and K. Tokuda, "Voice conversion based on simultaneous modeling of spectrum and F0," in *Proc. of the ICASSP*, Taipei, Taiwan, May 2009, pp. 3897–3900.

[Zet04] E. Zetterholm, M. Blomberg, and D. Elenius, "A comparison between human perception and a speaker verification system score of a voice imitation," in *Proc. of the Australian International Conference on Speech Science and Technology*, Sydney, Australia, Dec. 2004, pp. 393–397.

[Zha08] M. Zhang, J. Tao, J. Tian, and X. Wang, "Text-independent voice conversion based on state mapped codebook," in *Proc. of the ICASSP*, Las Vegas, USA, Apr. 2008, pp. 4605–4608.

Total number of entries is 124.

Publication P1

Elina E. Helander and Jani Nurminen, On the importance of pure prosody in the perception of speaker identity. In *Proceedings of the 8th Annual Conference of the International Speech Communication Association, Interspeech*, Antwerp, Belgium, Aug. 2007, pp. 2665–2668.

On the Importance of Pure Prosody in the Perception of Speaker Identity

Elina E. Helander[1], *Jani Nurminen*[2]

[1]Institute of Signal Processing, Tampere University of Technology, Finland
[2]Nokia Technology Platforms, Tampere, Finland

elina.helander@tut.fi, jani.k.nurminen@nokia.com

Abstract

Many of the current techniques and systems that deal with speaker identity do not regard detailed prosody as a crucial source of speaker-dependent information. The reasoning behind this relates to the common assumption that the F0 level and the spectral data carry all or almost all of the speaker-dependent information. But is this assumption really valid? We have investigated the importance of prosodic information in the perception of speaker identity by conducting a test where the listeners tried to identify people they know after hearing only delexicalized pure prosody signals. The findings presented in this paper show that even a very rough prosodic representation consisting only of a single sinusoid can contain information on speaker identity, giving motivation for the development and wider usage of techniques that better exploit the prosodic aspects.

Index Terms: prosody, speaker identity

1. Introduction

Voice conversion techniques aim at converting a speech signal to sound as if it was uttered by another speaker. This research topic has gained a lot of interest during recent years, e.g. [1], [2], [3], [4]. Most voice conversion systems published in the literature neglect detailed prosody modeling and concentrate mainly on spectral conversion. Fundamental frequency (F0) contours are usually modified in a very simple manner using e.g. Gaussian mixture modeling (GMM) based conversion or a simple mapping based on the mean F0 values and the variances of F0 [1], [3], [4]. The lack of wider interest towards more sophisticated prosody conversion techniques could originate from the fact that researchers have typically focused on the problem of changing some stranger's voice into the voice of another stranger. In this kind of cases the detailed conversion of prosodic features may not be highly crucial. However, when thinking about the potential applications of voice conversion, it seems quite likely that the users would be interested in applications where they can convert some source voice into the voice of a person they know (friends, family, celebrities, etc.). In this scenario, the user is quite familiar with the person's speaking style and our assumption is that the user might be quite likely to detect peculiarities in intonation patterns or tempo if prosody is not properly converted.

We have recently introduced a novel approach for converting prosody in voice conversion [5]. This new approach was evaluated in a listening test in connection with a complete voice conversion system [6] and it was found to clearly outperform the conventional GMM based F0 conversion scheme. In fact, the results presented in [5] indicated that the more accurate modeling of prosody increased the performance of voice conversion in two ways: the speaking style in the converted samples was found to be closer to the target, and the converted speech was found to be less robotic or monotonic. In the listening test, the target speaker was allowed to speak more freely than the source speaker from the prosodic point of view. While this is well in line with the potential use case where a text-to-speech (TTS) system acts as the source, it may also raise an important question: could the observed enhancement be caused by the increased expressiveness in the converted speech? Furthermore, can we be sure that the prosody contains person-dependent information that should be converted, other than the F0 level and the rough speaking style that could also be approximated using the variance of F0? The listeners judging the success of prosody conversion in [5] were not previously familiar with the target speaker and thus they evaluated more the speaking style than the real identity.

To further support our claims on the importance of prosody from the viewpoint of speaker identity, we show in this paper that people can identify themselves or persons they know on the basis of pure prosodic stimuli, although the average F0 levels of the different speakers are very close to each other. The pure prosody signals are obtained by estimating F0, duration and intensity information from the original signals and by creating a signal containing a single sinusoid whose frequency follows the F0 contour and whose amplitude follows the intensity contour of the original speech signal. This simplified pure prosody signal generation approach is very close to that proposed in [7]. The results, together with the earlier results presented in [5], give a clear indication that the prosodic aspects would deserve more attention in the voice conversion related research and development work.

This paper is organized as follows. In Section 2, some issues related to prosody and speaker identity are discussed. Section 3 describes the listening tests and the results obtained in the test. The discussions and the conclusions presented in Section 4 and Section 5, respectively, conclude this study.

2. On prosody and speaker identity

Prosody is a supra-segmental phenomenon that is not conveyed through a single phonetic segment but through larger units like words, sentences, utterances or even paragraphs. A vast amount of research has been devoted to prosody in relation to text-to-speech systems during the previous decades, especially related to intonation that is considered perhaps the most important aspect of prosody. Intonation can have linguistic, paralinguistic or extralinguistic functions [8]. Linguistic functions comprise of the morphological and lexical levels of phrase, as well as of discourse and dialogue levels. Emotions and mental states are categorized as paralinguistic. Extralinguistic factors like age and sex are personal and physical characteristics. These different linguistic sources of information are resolved in the ear via perceptual processing. Based on the formant frequencies and

the pitch, the listener usually forms an estimate if the speaker is a male, female or a child, even if he/she does not know the speaker.

2.1. Prosody and speaker identity in the literature

It is widely accepted that prosody plays an important role in naturalness of speech. In text-to-speech systems, the aim is to generate a "good" and acceptable prosody. However, in natural speech there exists a great deal of inter-speaker and intra-speaker variations in prosody [9]. An interesting experiment was made in [10] where five speakers of the same sex and age were presented to familiar listeners in different forms of natural and synthetic stimuli. It was shown that the speakers were highly identifiable on the basis of their fundamental frequency characteristic even when the spectrum was generated artificially. A slightly similar experiment was carried out in [11] where the spectral information related to speaker individuality was hidden using the average envelopes of all speakers. The pitch contours were shown to make a difference from the viewpoint of speaker individuality.

Many voice conversion systems transform the F0 contours through simple scaling between target's and source's mean F0 value and the standard deviation of F0. Some experiments with more detailed pitch conversion have been reported [2],[12],[13] but compared to the total number of voice conversion papers published so far, the topic of more detailed F0 modeling in voice conversion has not received much attention.

In addition to F0 contours, another important aspect of prosody are the durational issues: the durations of segments and pauses that are usually referred to as speech tempo. In voice conversion, the durations are usually modeled by simple scaling of the target's and source's utterance lengths. However, it was shown in [14] that there are inter-speaker differences between sound classes. This will not favor uniform speaking rate modifications. Rate modifications, global or local, are not entirely linear due to the properties of human speech production, but affect vocalic segments more than consonantal segments [9]. In [4], the durational modifications are separated into three distinct classes, silent, unvoiced, and voiced. It was claimed that more detailed modeling, i.e. phoneme level, would be too inaccurate but in [1] context-dependent duration difference modeling between speakers in phone or triphone level was suggested.

When looking outside the topic of voice conversion, most speaker recognition systems also ignore the prosodic features and concentrate only on the short-term spectral features. However, the addition of short-term [15] or long-term [16] prosodic information has been shown to improve the performance. Related to speaker identification, an interesting finding was made in [17] where an imitator was asked to mimic two subjects. Human listeners did not recognize the difference between the imitator and the real speaker nearly as well as a speaker verification system that did not use prosodic information. From the viewpoint of voice conversion, this supports the assumption that it could be possible to convince a human listener with even a slightly imperfect spectral conversion as long as the prosody is converted more accurately.

2.2. Some experimental findings

We have studied the prosodic features in the speech of different speakers and found that there are clear speaker-dependencies in speech signals that should be modeled for example in voice conversion. An example case including F0 contours from two speakers uttering the same sentence is shown in Figure 1. The

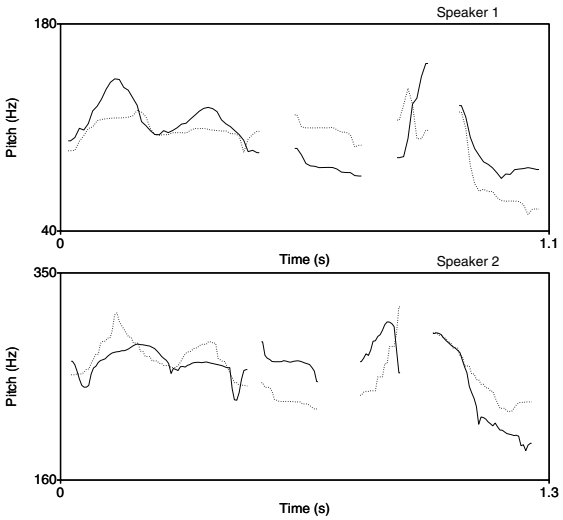


Figure 1: *Partial F0 contours of two speakers (solid lines) from a Finnish sentence "Meiän äiti ei ikinä tekis mitään sellaista" (Our mother would never do anything like that) and the transformed F0 contours (dashed lines)*

straight lines represent the measured F0 contours and the dashed lines indicate the transformed F0 with the second speaker's mean and variance applied to the first signal and vice versa. In the transformed signals, the different durations were taken into account by applying dynamic time warping to the speech signals. It can be observed that there are similarities due to the same context but some clear speaker-specific differences can also be found. Furthermore, it is also possible to see that the mean and variance based approach does not achieve very good results.

Further experimental findings were presented in [5]. In that paper, we proposed a new prosody conversion scheme that uses a syllable-based prosodic codebook. The selection from the codebook employs not only the source contour but also linguistic information and segmental durations with the aid of a trained classification and regression tree. The new method was included as a part of the voice conversion system originally presented in [6] and it was evaluated in a listening test. The results clearly indicated that more detailed prosody conversion enhanced the quality of the converted samples when compared to the conventional GMM based approach: it increased the naturalness (or made the output sound less robotic or monotonous) and the identity mapping was improved through the better modeling of the person-dependent speaking style.

Even after these findings and good results, one important question remains. Does our prosody conversion approach improve the results through better prosody modeling in general or does the prosody really contain some person-dependent features that can be modeled in voice conversion? The rest of this paper demonstrates through a listening test that the prosody indeed contains person-dependent information.

3. Listening tests

To get more evidence on the speaker-dependencies in prosody, a listening test involving pure prosodic signals was carried out. This approach was chosen to be able to really focus on the

prosodic aspects and to avoid any influence from other features in speech signals or from the voice conversion system.

3.1. Test arrangement

Several neutrally spoken sentences from 16 different native Finnish speakers were recorded in a quiet room. The recorded signals were analyzed automatically to obtain F0 and energy contours and these contours were used in the generation of pure prosody signals. In these signals, the voiced regions contain a single sinusoid whose frequency follows the F0 contour and whose amplitude is in line with the energy contour, while the unvoiced and silent regions are represented as silence. An example including a speech signal and the corresponding pure prosody signal is shown in Figure 2.

In total 14 native Finnish listeners participated in the test. Each listener heard sinewave signals generated from speech spoken by people she/he knows (possibly including her/himself) in random order and was asked to identify the speaker. For some of the sinewave signals, the listener was shown the textual form of the corresponding sentence while for other sinewaves no supplementary information was given. Before starting the actual test, each listener was given a chance to get familiar with the sinewave representation. In this training phase, each listener was able to listen to both the original recorded form and the corresponding sinewave form of some example sentences as many times as she/he wanted. One sentence from each speaker was included in this training material.

Many of the listeners also served as a speaker and vice versa since we also wanted to examine how well people recognize themselves from the sinewave based prosodic representation. The speakers were grouped into three categories based on the gender and age: in total there were 6 female, 8 male and 2 child speakers. The speakers were also further divided into groups of people who knew each other. Each listener heard only samples from her/himself or from speakers that she/he knows from some context: either as a family member or a close relative, as a friend, or as a colleague. Each listener was explicitly given the list of speakers that could appear in the samples. If we assume that the listener can always recognize the correct category (female, male or child voice), there were always either two or three alternatives to choose from. This assumption was found valid with only a couple of exceptions that were most likely caused by unfamiliarity with the sinewave representation. The two or three alternatives were chosen using two criteria: the listener has to know the speakers and their F0 levels were very close to each other. This applied in all the cases except for the children.

3.2. Results

Table 1 illustrates the overall result obtained from the whole test. The notations "From 2" and "From 3" denote the number of alternatives the listeners can choose from provided that the answer falls in the correct category (male, female, or child). As can be seen from the average identification rates obtained from a total of 524 answers, it can be concluded that even the very rough representation of pure prosody including only a single sinusoid contains speaker-dependent information. Even though the sinewave representation was a rather odd and unfamiliar representation for the listeners, it still helped significantly in the speaker identification. For comparison, the table also shows the average rate that would be achieved by guessing (with the assumption that the listener can always recognize the speaker category correctly). The difference between the test result and

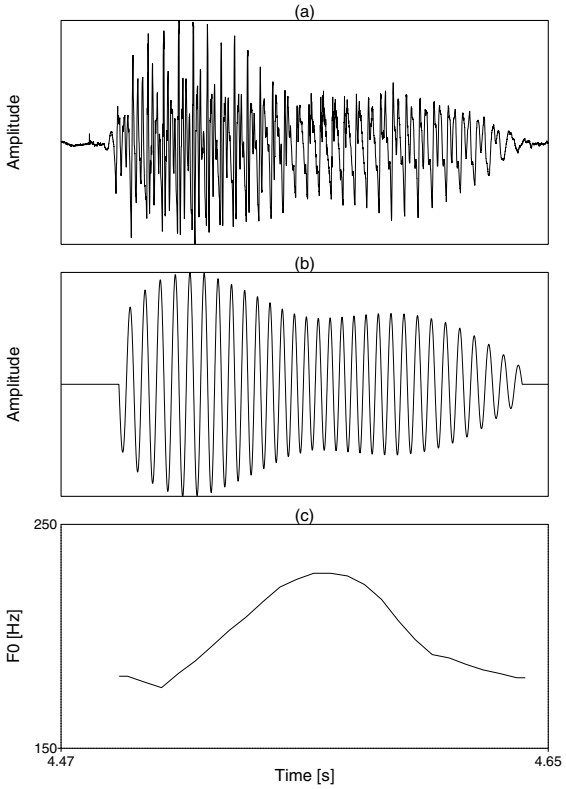


Figure 2: *(a) a speech segment, (b) the corresponding prosodic signal, and (c) common F0 contour for both (a) and (b).*

Table 1: Overall results

	From 2	From 3
No. of sentences	284	240
Total recognition	70.4 %	47.9 %
Lover conf. interval	64.7 %	41.4 %
"Guessing rate"	50.0 %	33.3 %
Upper conf. interval on guessing	56.0 %	39.7 %

the average guessing rate is statistically significant based on the fact that the upper bound of the confidence interval in the case of guessing is lower than the lower confidence interval of the test results. We also observed that there was no difference on how listeners recognized themselves compared to others.

The potential influence of having the textual version of the sinewave signal available is analyzed in Table 2. Indeed, the identification rate seems to slightly improve when the listener knows the sentence that is spoken but this result is not statistically significant considering the confidence intervals.

4. Discussion

Although the results are favorable as such, some further issues should be pointed out. First, the language used in the experiments was Finnish that is regarded as a rather monotonic language in terms of F0 contours. This was especially true in the case of some male speakers. Additional results using some other language could reveal even more evident findings. Moreover, we also observed that the presence of the microphone and

Table 2: The effect of having the text available

	From 2	From 3
No. of sentences without text	209	105
Recog. without text	66 %	47 %
Upper conf. interval	72 %	57 %
No. of sentences with text	75	135
Recog. with text	81 %	49 %
Lower conf. interval	71 %	40 %

the usage of pre-defined sentences did not help in capturing natural prosodies that would make the identification easier.

It was also found out that special attention should be paid to the planning of the text materials when studying the prosodic effects. This also applies more widely e.g. to voice conversion since the planning of a limited number of sentences that could capture the spectral, the prosodic and the speaking style related individualities poses a real challenge. Voice conversion prompt sheet planning is not covered well in the literature.

We believe that the results were affected by the fact that pure prosodic signals were strange and unfamiliar to the listeners. Some people adapted to listening to the sinewave representations faster than others but further training might have helped all the listeners. In fact, many listeners commented that more practice with the sinewave signals could have improved their performance substantially. The listeners found dealing with pure prosody signals more comfortable and easier when they had the opportunity to see the texts. However, the general recognition rate was not significantly improved, as shown in Table 2.

Finally, it should also be noted that the F0 levels of the speakers in the same category (male, female or child voice) were often very close to each other. The listeners had to identify speakers whose F0 levels were as little as 3 Hz apart.

5. Conclusions

In this study, we have discussed the importance of prosody in the perception of speaker identity. We conducted a listening test in which 14 listeners were asked to identify speakers based on simplified prosodic signals. Each utterance consisted only of a single sinewave following the F0 and energy contours of the corresponding recorded sentence. It was shown that it is possible to identify familiar people on the basis of pure prosody. From the viewpoint of voice conversion, the result, together with the results we have presented earlier in [5], highlights the potential advantage that may be gained in voice conversion through proper treatment of prosody. This will presumably be especially true in relation to commercial voice conversion applications since the consumers are likely to be interested in having a voice converted to the voice of a person whose prosodic peculiarities they know very well.

6. Acknowledgements

This work was partially supported by the Academy of Finland, project No. 5213462 (Finnish Centre of Excellence program 2006-2011). Moreover, this work has partially been funded by the European Union under the integrated project TC-STAR - Technology and Corpora for Speech-to-Speech Translation - (IST-2002-FP6-506738, http://www.tc-star.org).

7. References

[1] L. Arslan, "Speaker transformation algorithm using segmental codebooks (STASC)," *Speech Commucation*, vol. 28.

[2] D. Rentzos, S. Vaseghi, Y. Q., and C.-H. Ho, "Voice conversion through transformation of spectral and intonation features," in *Proc. of ICASSP*, vol. I, 2004, pp. 21–24.

[3] Y. Stylianou, O. Cappe, and E. Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Trans. on Speech and Audio Processing*, vol. 6(2), pp. 131–142, March 1998.

[4] A. Verma and A. Kumar, "Voice fonts for individuality representation and transformation," *TSLP*, vol. 2(1), pp. 1–19, February 2005.

[5] E. Helander and J. Nurminen, "A novel method for prosody prediction in voice conversion," in *Proc. of ICASSP 2007, to appear*.

[6] J. Nurminen, V. Popa, J. Tian, and Y. Tang, "A parametric approach for voice conversion," in *Proc. of TC-STAR Workshop on Speech-to-Speech Translation*, 2006, pp. 225–229.

[7] G. Sonntag and T. Portele, "PURR - a method for prosody evaluation and investigation," *J. of Computer and Language*, vol. 12(4), pp. 437–451, 1998.

[8] A. Botinis, B. Ganstrom, and B. Möbius, "Developments and paradigms of intonation research," *Speech Communication*, vol. 33, pp. 263–296, 2001.

[9] E. Keller, *Fundamentals of Speech Synthesis and Speech Recognition*. Chichester, United Kingdom: John Wiley and Sons Ltd, 1994.

[10] E. Abberton and A. Fourcin, "Intonation and speaker identification," *Language and Speech*, vol. 21(4), pp. 305–318, 1978.

[11] T. Ienaga and M. Akagi, "Speaker individuality in fundamental frequency contours and its control," *J. Acoust. Soc. Jpn.(E)*, vol. 18(2), pp. 73–80, 1997.

[12] D. Chappell and J. Hansen, "Speaker-specific pitch modeling and modification," in *Proc. of ICASSP*, vol. II, 1998, pp. 885–888.

[13] B. Gillet and S. King, "Transforming f0 contours," in *Eurospeech*, Geneve, September 2003, pp. 101–104.

[14] C. Shih, W. Gu, and J. van Santen, "Efficient adaptation of tts duration model to new speakers," in *SSW3-1998*, vol. II, 1998, pp. 105–110.

[15] M. Carey, E. Parris, H. Lloyd-Thomas, , and S. Bennett, "Robust prosodic features for speaker identification," in *Proc. of ICSLP*, vol. III, 1996, pp. 1800–1803.

[16] A. Adami, R. Mihaescu, D. Reynolds, and J. Godfrey, "Modeling prosodic dynamics for speaker recognition," in *Proc. of ICASSP*, vol. 4, 2003, pp. 788–791.

[17] E. Zetterholm, M. Blomberg, and D. Elenius, "A comparison between human perception and a speaker verification system score of a voice imitation," in *Proc. of SST2004*, 2004.

Publication P2

Elina E. Helander and Jani Nurminen, A novel method for prosody prediction in voice conversion. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, Honolulu, Hawaii, USA, Apr. 2007, pp. IV-509–IV-512.

Publication P3

Elina Helander, Jan Schwarz, Jani Nurminen, Hanna Silén, and Moncef Gabbouj, On the impact of alignment on voice conversion performance. In *Proceedings of the 9th Annual Conference of the International Speech Communication Association, Interspeech*, Brisbane, Australia, Sep. 2008, pp. 1453–1456.

interspeech 2008 incorporating SST 08

On the impact of alignment on voice conversion performance

Elina Helander[1], *Jan Schwarz*[2], *Jani Nurminen*[3], *Hanna Silen*[1], *Moncef Gabbouj*[1]

[1]Department of Signal Processing, Tampere University of Technology, Finland
[2]Institute for Circuit and System Theory, Christian-Albrechts University of Kiel, Germany
[3]Nokia Devices R&D, Tampere, Finland

elina.helander@tut.fi, js@tf.uni-kiel.de, jani.k.nurminen@nokia.com, hanna.silen@tut.fi

Abstract

Most of the current voice conversion systems model the joint density of source and target features using a Gaussian mixture model. An inherent property of this approach is that the source and target features have to be properly aligned for the training. It is intuitively clear that the accuracy of the alignment has some effect on the conversion quality but this issue has not been thoroughly studied in the literature. Examples of alignment techniques include the usage of a speech recognizer with forced alignment or dynamic time warping (DTW). In this paper, we study the effect of alignment on voice conversion quality through extensive experiments and discuss issues that should be considered. The main outcome of the study is that alignment clearly matters but with simple voice activity detection, DTW and some constraints we can achieve the same quality as with hand-marked labels.

Index Terms: voice conversion, alignment, DTW

1. Introduction

The aim in voice conversion (VC) is to convert speech from one speaker (source speaker) to sound like the speech of another particular speaker (target speaker). VC consists of two phases: training and conversion. Training usually relies on parallel data from the source and target speakers, although some approaches for non-parallel VC data alignment have also been proposed. An interesting framework for voice conversion is offered by voice adaptation with hidden Markov model (HMM) based speech synthesizer [1] that does not require parallel sentences for training. Many VC systems are based on applying a conversion scheme directly to the source speech or its parametric representation. The most popular conversion scheme is to use a Gaussian mixture model (GMM) to model the joint density of aligned source and target features [2]. Thus, before training the GMM it is necessary to align the training data, i.e. to find a corresponding target frame for each source frame.

The alignment process has been studied little in the literature. Nevertheless, there are several techniques that can be used for carrying out the alignment. The simplest alignment technique is linear interpolation that works under the assumption that speaking rate variation is only global, not local. Non-linear warping can be obtained using dynamic time warping (DTW) that finds an optimal path through a difference matrix computed between the source and target features. It is also possible to use a speech recognizer with forced alignment. Compared to this solution, DTW has the advantage that alignment can be done without knowing the content of the sentence. Moreover, there is no need to have a speech recognizer available.

In this study, we consider the conventional GMM based voice conversion that uses parallel sentences for training data generation, and we analyze how the alignment process affects the conversion result. We carry out experiments and point out alignment related aspects that may improve or degrade the conversion performance. The results support our hypothesis that it is possible to affect the conversion quality through alignment. We also show that a reasonably simple alignment procedure can be used for obtaining a quality level similar to the level that can be obtained using manually annotated labels.

This paper is organized as follows. Section 2 describes DTW in general whereas the usage of DTW in VC is discussed in Section 3. Experimental results related to alignment accuracy are presented in Section 4. Section 5 presents voice conversion experiments with different alignments and provides analysis of the results. Section 6 concludes the study.

2. Dynamic time warping in speech alignment

The objective of DTW is to find an optimal alignment between speech patterns $\mathcal{X}$ and $\mathcal{Y}$. Speech patterns $\mathcal{X}$ and $\mathcal{Y}$ are represented by short-time feature vector sequences. The feature vectors typically relate to the corresponding speech spectra. The overall distortion $d(\mathcal{X}, \mathcal{Y})$ is a sum of the local distances $d(i_x, i_y)$ computed over the path. The optimal alignment basically minimizes the overall distortion with some constraints.

Constraints on the warping function are required in order to provide a meaningful alignment. Constraints also save computational resources. But the use of strict constraints can introduce problems if the "correct" path cannot fit into the allowed area. Examples of typical warping constraints include [3]

- *Endpoint constraints* define that the alignment starts at the first frame pair $d(1,1)$ and the stops at $d(N,M)$ where N and M are the number of source and target frames, respectively.
- *Monotonicity constraints* do not allow the warping path to have a negative slope.
- *Local constraints* define the set of allowed predecessors and transitions to the current node.
- *Global constraints* define the region of nodes that are searched for the optimal path.

3. DTW in voice conversion

Usually the training data in VC is not as corrupted as the signals that speech recognition systems have to deal with. Thus, voice conversion is a relatively easy use case for DTW. Moreover, there is no single "correct" time alignment between the

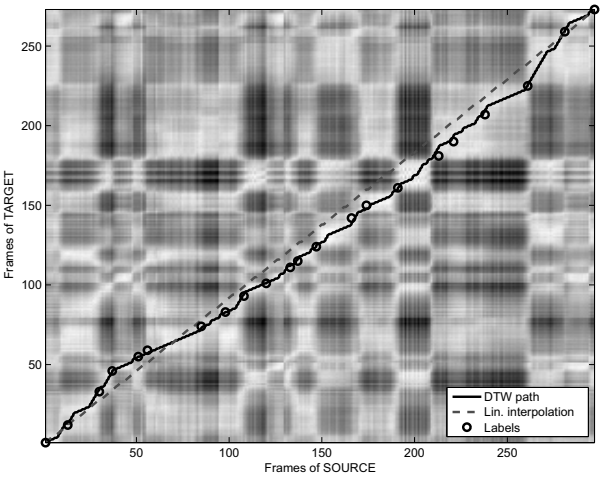


Figure 1: *Example of distance matrix, labels (circle), DTW (solid) and linear interpolation (dashed) result.*

sentences from two different speakers and thus there may be several acceptable alignments. However, there are still some aspects that should be considered when implementing DTW for voice conversion purposes.

3.1. Alignment features

In alignment, parallel speech waveforms are converted into sequences of features that can be compared with each other. The alignment searches for the source-target feature path that minimizes the overall distortion. The closer the speaker characteristics are to each other, the better the features are likely to be aligned. It is somewhat paradoxical that in VC we try to capture the differences between the source and the target speaker characteristics but at the same time the alignment process tries to minimize the difference. To minimize the effect of this paradox, the features used in the alignment should be as speaker-independent as possible.

In speech recognition, Mel-frequency cepstral coefficients (MFCCs) are commonly used features. Thus, they should also be suitable for the alignment. Linear prediction related features such as LSFs are more complicated to use since the k^{th} LSF from the source may not correspond to the k^{th} LSF of the target.

3.2. Problems with DTW in the context of VC

There are some potential problems when DTW is used to align signals for VC. These inherent problems that are discussed later in this section can be dealt with by additional constraints or by removing ”bad” frame pairs. Although VC systems need to be able to cope with small amounts of training data and an increase in the training set size is likely to enhance the conversion quality, single frame pairs can be safely removed from the training data when necessary. Removing badly aligned frames may even increase the resulting speech quality.

3.2.1. Silent segments

When recording a sentence, some silence is usually included before and after the meaningful speech content. Silence segments represent non-interesting information and can sometimes confuse the performance of DTW depending on the constraints. Exploitation of end points is highly crucial for good DTW performance. However, unreliable estimation of end points is also a problem, if strict end point constraints are used.

The most severe problems arise when there is silence at the beginning and/or at the end of a speech pattern $\mathcal{X}$ but no silence at the beginning or the end of a speech pattern $\mathcal{Y}$. If the silent parts are not removed before the alignment and the end point constraints are strict, the alignment will go wrong, most likely for the whole sentence. Even if the alignment could seek its way into the ”correct” path at some phase, the alignment will still produce silence-speech frame pairs that should preferably be removed from the training data.

3.2.2. Global optimization

DTW provides a globally optimal alignment through the source-target difference matrix. However, this does not mean that each frame pair would represent a decent feature pair for GMM training. For example, DTW can handle short silence segments between words even if the silence is present only either in the source or the target speech. Nevertheless, it is questionable if we should use that kind of pair for training where the source part is silence and the target part is speech or vice versa. Moreover, the local performance needs to be sacrificed for the global optimization, i.e. some clearly voiced frames become paired with some clearly unvoiced frames. Including such data for GMM training may not be meaningful.

3.2.3. One-to-many and many-to-one mappings

The purpose of DTW is to generate a non-linear warping function of feature sequences along the time axis. This means a target frame may become mapped to more than one source frame. Also, a source frame can have more than one target frame mapped into it. This results in one-to-many and many-to-one mappings. Such data is ambiguous for the GMM. In general, the main problem of GMM in VC is oversmoothing and if one-to-many or many-to-one mappings occur systematically (e.g. in the case where the speaking rates of source and target differ significantly) the oversmoothing may become worse.

4. Experiments on alignment accuracy

4.1. Database and alignment features

The database consists of a set of the *Berlin sentences* taken from the German speech database *The Kiel Corpus of Read Speech* (KCoRS) [4] sampled at 16 kHz. All the sentences have been manually labeled by experts and in the alignment experiments we assume these labels to be precise.

For the alignment, 13 MFCCs at 5 ms steps with an analysis window of 25 ms were extracted. The first MFCC was omitted and the other 12 MFCCs were normalized to zero mean.

4.2. Schemes included in comparison

There are many alternatives concerning the different constraints for guiding the optimal path search with DTW. In addition, the selection of the alignment features can affect the result as discussed in Sec. 3.1. However, according to our experiments, the use of different local constraints (I, II, V and Itakura) [3] did not make much difference on the alignment performance. Furthermore, we compared the use of only static features (12 MFCCs) with the use of both static and dynamic features (12 MFCCs, their delta and delta-delta coefficients). Incorporating dynamic information had only minor effects on the results. Thus, local constraint type II and 12 MFCCs without dynamic information

are used.

Two different methods for the alignment of two utterances were compared, namely linear interpolation and DTW algorithms. Linear interpolation is a basic method that can be used to lengthen or shorten any sequence. In Figure 1, an example of linear interpolation is shown as well as a non-linear mapping given by DTW. The alignment given by DTW varies depending on the selection of the constraints and the assumptions described in Sec. 2.

The problem of having silent frames at the beginning and end of sentences was discussed in 3.2.1. We used a simple voice activity detection (VAD) technique based on a heuristic energy threshold to find the silent frames at the beginning and at the end of the sentences. This type of processing could not remove all the breathing effects that sometimes appear at the beginning or the end of the sentences for example with speaker *k65*. However, we wanted to examine whether this had an effect on the voice conversion quality.

4.3. Results on alignment accuracy

As a preliminary step for assessing the alignment in VC, we measured the misalignment given by linear interpolation and different DTW approaches with respect to the manual labels. The alignment analysis comprised 100 sentences spoken by 5 different speakers (2 female (*k04*, *k06*) and 3 male (*k05*, *k61*, *k65*) speakers). Table 1 lists the results using different approaches to align two utterances. It shows the mean misalignment in ms and the percentages of misalignments greater than 20 ms, 50 ms and 100 ms. Linear interpolation (D_0) was tested against different DTW approaches that differed firstly in terms of the use of global constraints (GC) and forced end-point constraints (EC). In Table 1 *n* means that the particular constraint was not used while *y* means it was in use. Regardless of the use of EC, the DTW algorithm always assumed that the first feature vector of the source and the target form a pair. In addition to the two constraints, three types of DTW schemes were included in the test. The first one (D_1) had no silence removal. The second one (D_2) removed silent frames before path calculation from the beginning and the end using a simple VAD based on an energy threshold. Finally, the third case (D_3) used the starting and ending points given by the manually annotated labels. The results are commented together with VC results in Sec. 5.4

Table 1: *Misalignment caused by linear interpolation and different DTW approaches.*

Algorithm			Mean	Misalignment [%]		
	GC	EC	[ms]	$>$ 20ms	$>$ 50ms	$>$ 100ms
D_0	–	–	132.4	91.7	79.9	56.4
D_1	y	y	11.0	11.1	3.6	1.7
D_1	n	n	63.1	12.1	5.7	4.8
D_1	n	y	11.0	11.1	3.6	1.7
D_2	y	y	19.4	19.3	10.9	5.4
D_2	n	n	107.4	30.2	24.3	19.8
D_2	n	y	19.4	19.3	10.9	5.4
D_3	y	y	7.2	5.0	1.0	0.6
D_3	n	n	7.2	5.1	0.9	0.5
D_3	n	y	7.2	5.0	1.0	0.6

5. Experiments on voice conversion performance

5.1. Voice conversion framework

In the analysis and synthesis, a VC framework similar to the one presented in [5] is used but now for wideband signals (sampling rate is 16 kHz). The alignment was performed using several different alignment schemes and separate GMMs were trained using the different alignments. 12 MFCCs were applied as alignment features. For GMM training the corresponding 16-dimensional LSF vectors computed from the source and the target signals were used. The joint density of the aligned source and target LSF vectors was modeled with a GMM as explained in [2]. In conversion, all the other speech parameters (voicing information and harmonic amplitudes for the residual spectrum, pitch and energy) were handled in an identical way. In addition to the LSF modification, pitch level adjustment and residual spectrum resampling was carried out.

The resulting voice conversion quality was evaluated for 9 different alignment schemes, as summarized in 2. All 9 alignment schemes were evaluated using objective metrics. The main techniques (*gmm1*, *gmm2*, and *gmm3*) were also evaluated in a listening test. Global constraints were not used and bad data removal was used only with *gmm2*. Cases *gmm8* and *gmm9* correspond to a fictional situation where the source had silence removed from the beginning and the end while the target did not (*gmm8*) and vice versa (*gmm9*).

Speaker pairs *k04–k05* (female-male) and *k61–k05* (male-male) were used in the evaluation. 70 sentences were used in training of the GMM models.

Table 2: *Alignment schemes tested with voice conversion.*

gmm1	DTW goes through manual labels (”ideal” case)
gmm2	DTW + simple VAD, forced end, data removal
gmm3	Linear interpolation, endpoints from manual labels
gmm4	Linear interpolation, endpoints with simple VAD
gmm5	Same as *gmm2* but no data removal
gmm6	DTW + simple VAD, no forced end
gmm7	DTW + no VAD, forced end
gmm8	DTW + silence removed from source, forced end
gmm9	DTW + silence removed from target, forced end

5.2. Listening test results

The VC performance achieved using the alignment given by DTW and linear interpolation was evaluated in a listening test. 17 native German listeners were asked to judge the quality of the transformed voice by doing a comparison category rating (CCR). The listeners were asked to compare the voice conversion quality of ten sentence pairs not included in the training set from two speaker pairs in two different comparisons. DTW with simple VAD and forced endpoint (*gmm2*) was compared against the ”ideal” case that utilizes the manually annotated labels (*gmm1*). In addition, *gmm2* was compared with linear interpolation (*gmm3*). Additional data removal was also applied when training *gmm2*: unvoiced-voiced pairs were discarded as well as pairs where at least one of the frames had an energy level less than 10% of the mean energy. The results for the preference test are shown in Table 3.

Table 3: *Results of the CCR test.*

	gmm1 – gmm2			*gmm2 – gmm3*		
	gmm1 better	*gmm2* better	identical	*gmm2* better	*gmm3* better	identical
k04k05	10.0%	4.1%	85.9%	94.1%	1.2%	4.7%
k65k05	7.7%	4.1%	88.2%	98.8%	0.6%	0.6%
total	8.8%	4.1%	87.1%	96.5%	0.9%	2.6%

5.3. Objective voice conversion results

We also evaluated the voice conversion quality by measuring spectral distortion (SD) between the converted target and real target features. The alignment that goes through real labels was assumed to be the correct one. We compared the LSFs of converted target and real target features using mean spectral distortion. The average SD in 30 test sentences not included in the training set was calculated at two different bands (125-3100 Hz and 0-8000Hz). The results are shown in Table 4. The number of GMM mixtures was 8 in all cases except with *gmm7* (4 mixtures). The selection of the number of mixtures was optimized by selecting the number of mixtures resulting in the lowest SD.

5.4. Analysis of results

The results for subjective and objective voice conversion quality are very consistent with each other. The listeners could not observe a clear difference between the samples generated using the GMM based on the ideal alignment (*gmm1*) and the GMM with a simple VAD and data removal (*gmm2*). Although *gmm1* was preferred more often, the difference was not significant according to a two-sample t-test (p=0.08 for *k04–k05* and p=0.40 for *k65–k05*). Thus, the objective results indicate that *gmm1* and *gmm2* have roughly equal performance.

There was a clear difference between the performance of *gmm2* and *gmm3*. It is interesting to note that according to Table 4, linear interpolation (*gmm3*) with *k04–k05* seems to be slightly more successful than with *k61–k05* and this can be also concluded from the objective results. This may be explained with the fact that speaker *k65* had a quite unique speaking style compared to *k04* and *k05* and thus the global speaking rate assumption was far from valid. Results in Table 1 also confirm that errors can be rather high. If simple VAD was used instead of correct starting and ending points for linear interpolation (*gmm4*), the quality was degraded.

The use of data removal in *gmm2* increased the quality over the case where no data removal was employed (*gmm5*). The use of a simple VAD did not automatically remove all silence-speech pairs, and removing them improved the quality slightly, at least based on the objective results. Also some voiced-unvoiced and silence-silence pairs were removed with *gmm2*.

The performance of *gmm8* and *gmm9* was very poor. This indicates that silent frames can be problematic for DTW and this will degrade the performance significantly if they are not taken into account. In contrast, the performance of *gmm7* was rather successful. In the training, some problems with covariance matrices occurred. This is due to the high number of silence-silence pairs in the training. Nevertheless, the performance of DTW in that case seemed to be successful as could be predicted also from the results shown in Table 1. There was silence with both speakers and no strict constraints were given which enabled DTW to find a decent path. However, in practice we should at least verify the existence of silence for each source-target sentence (compare to the cases *gmm8* and *gmm9*). On the contrary, using VAD without forcing the endpoint performed poorly (*gmm6*). This is also indicated by the results in Table 1.

The spectral distance between the original source and the target was also shown in the last row of Table 4. Since the target is a male, it was expected that source *k65* (male) was closer to the target than source *k04* (female). However, this applied only to the frequency band 125-3200 Hz and not for the whole speech band.

The conversion error can also be expressed as mean-squared error normalized to the difference between the source and the target. For the ideal case (*gmm1*) this conversion error was 0.35 and 0.46, for *gmm2* 0.36 and 0.47 and for *gmm3* 0.46 and 0.80, for the speaker pairs *k04–k05* and *k65–k05*, respectively.

It should also be noted that the database had its impact on the results. The database used in the experiments did not contain very long sentences or noise but there were some breathing effects that can affect the results. Finally, it should be noted that the alignment results are speaker-pair specific.

Table 4: *Results for mean spectral distortion (dB).*

	k04–k05 125-3100Hz	k04–k05 0-8kHz	k65–k05 125-3100Hz	k65–k05 0-8kHz
gmm1	4.57	4.63	4.61	4.60
gmm2	4.65	4.69	4.66	4.65
gmm3	5.40	5.31	6.39	6.14
gmm4	6.34	6.15	7.22	6.80
gmm5	4.77	4.79	4.74	4.73
gmm6	5.50	5.35	4.81	4.79
gmm7	4.87	4.92	4.89	4.86
gmm8	11.38	10.39	12.38	11.25
gmm9	6.94	6.46	6.13	5.87
source	8.17	7.25	6.60	7.11

6. Conclusions

The experiments presented in this paper have verified that the quality of GMM based voice conversion can be significantly enhanced by improving the alignment. However, the results also indicate that a combination of DTW and a simple VAD can be used for successful alignment in most cases. It also seems to be beneficial to remove inappropriate data: frame pairs containing clearly non-matching data and silence-silence pairs should be removed from the training data. On a higher level, the main conclusion that can be drawn based on our study is that while the main challenges in voice conversion are elsewhere, alignment is still an important piece of the puzzle that should be taken into account in the development of voice conversion systems.

7. References

[1] J. Yamagishi, K. Ogata, Y. Nakano, J. Isogai, and T. Kobayashi, "HSMM-based model adaptation algorithms for average-voice-based speech synthesis," in *ICASSP*, vol. I, 2006, pp. 77–80.

[2] A. Kain and M. W. Macon, "Spectral voice conversion for text-to-speech synthesis," in *ICASSP*, vol. 1, 1998, pp. 285–288.

[3] L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition*. Prentice Hall Signal Processing Series, Prentice-Hall Inc., 1993.

[4] K. Kohler, "The Kiel Corpus of Read Speech," Institute of Phonetics and digital speech processing at the Christian-Albrechts University of Kiel, Kiel, Germany, 1994.

[5] J. Nurminen, V. Popa, J. Tian, and Y. Tang, "A parametric approach for voice conversion," in *Proc. of TC-STAR Workshop on Speech-to-Speech Translation*, 2006, pp. 225–229.

Publication P4

Elina Helander, Jani Nurminen, and Moncef Gabbouj, Analysis of LSF frame selection in voice conversion. In *Proceedings of the 12th International Conference on Speech and Computer, SPECOM*, Moscow, Russia, Oct. 2007, pp. 651–656.

Analysis of LSF frame selection in voice conversion

Elina Helander[1], Jani Nurminen[2], Moncef Gabbouj[1]

[1]Institute of Signal Processing, Tampere University of Technology, Finland

[2]Nokia Technology Platforms, Tampere, Finland

Abstract

In practical applications of voice conversion, it is necessary to be able to cope with small amounts of speaker-specific training data. Consequently, most of the proposed voice conversion algorithms are based on probabilistic conversion functions. Recently, however, there has been increased interest in unit selection based approaches for voice conversion. It is evident that typical training sets are too small for enabling meaningful selection of large units such as diphones. But would it be possible to use smaller segments like frames for high quality results provided that the selection is handled very well? In this paper, we analyze the performance of the frame selection approach in ideal conditions. In the experiments, line spectral frequencies of test sentences are replaced with the best matches from different training sets. The results show that perceptually transparent quality cannot be achieved with realistic database sizes.

1. Introduction

In unit selection speech synthesis [1], speech is produced by selecting segments from a recorded database and by concatenating them together. The database is large, typically consisting of several hours of speech, sometimes even tens of hours for providing an optimal unit sequence. The most popular unit sizes used in the selection are diphones and triphones. *Voice conversion* (VC) provides means for generating new text-to-speech (TTS) voices in a fast and easy manner using only small training sets. Voice conversion (or voice morphing) has inspired many researchers during the last two decades. The aim in VC is to convert speech from one speaker (source speaker) to sound like the speech from another particular speaker (target speaker).

Most voice conversion systems proposed in the literature are based on applying a conversion scheme directly to the source speech or its parametric representation. Typical examples of conversion schemes include Gaussian mixture model (GMM) based conversion [2] and the use of codebooks [3, 4]. Another approach for voice conversion is the parametric adaptation in a hidden Markov model (HMM) based speech synthesis framework [5]. All of the approaches share the same fundamental requirement: they have to be able to cope with small amounts of speaker-specific training data. Due to this requirement, the unit selection idea cannot be directly used with conventional unit sizes in voice conversion because there simply is not enough data to select from.

Speaker identity can be partially characterized using formant positions and bandwidths. Since it is very hard to handle the estimation of formants in a reliable and robust manner, the features most often used for conversion in VC systems are the line spectral frequencies (LSFs). LSFs are features that are derived through linear prediction (LP) where speech is modeled using a filter given by the LP coefficients and a residual. In most VC studies, the residuals are left unconverted, but there are strong arguments for converting residuals and some techniques have been proposed for this task for example in [6]. LSFs have also been used widely in speech coding, where typically large amounts of data from various speakers and languages is used for the training of LSF quantizers to obtain a good representation of the LSF space of all speakers. In speaker identification, high order LSFs have been reported to perform well as speaker identification features [7] and they have been used in many related studies (e.g. [8]).

Although LSFs seem to carry a lot of speech identity information, only a few personalized speech coding approaches have been proposed ([9, 10]).

The ultimate goal in voice conversion is to convert the speaker identity as accurately as possible while maintaining high speech quality. However, these requirements have been found to be somewhat contradictory in practice; better identity conversion usually requires more signal modifications that may cause more distortions. The main problem of the current VC techniques is that they are not very successful in changing the identity. Good results are mainly obtained because of forced ABX tests; the speech sample may sound more like target speech than source speech but it does not mean that it would ultimately sound like speech of the target speaker. All of the current techniques, including the GMM based conversion and the use of codebooks, have inherent drawbacks from this point of view.

Recently, Dutoit et al [11] proposed to first use a conventional GMM based approach to convert source LSFs to target LSFs and then search from the target speech database for the closest match to the converted LSFs in order to obtain more "realistic" target LSFs. The idea is attractive but can it help in achieving high quality conversion? In this study, we analyze if it is possible to select LSFs from a target database of a realistic size in such a manner that the quality of the converted speech would be very high or even indistinguishable from the target speech. The results of our experiments reveal how accurately LSFs could be chosen provided that the conversion is successful. Multiple speakers, different test sentence sets and different sizes of target databases are examined and the results are presented in the light of quality criteria used widely in speech coding.

This paper is organized as follows. In Chapter 2, the basic properties of LSFs and the related distance metrics and quality criteria are discussed. The experiments and results demonstrating the idealized frame selection performance are described in Chapter 3. Chapter 4 provides a short discussion on the results and Chapter 5 concludes the study.

2. Linear prediction and line spectral frequencies

Linear prediction is one of the basic techniques used in speech processing. This source-filter model can be used for separating a speech signal into linear prediction coefficients that model the vocal tract contribution and into an excitation signal. More precisely, the excitation signal, also referred to as the residual signal, can be obtained through LP analysis filtering,

$$r(t) = x(t) - \sum_{k=1}^{m} a_k x(t-k), \tag{1}$$

where $x(t)$ is the input speech signal and m is the order of the analysis filter $A(z)$. The linear prediction coefficients $\{a_k\}$ are usually estimated in a frame-wise manner using either the autocorrelation or covariance methods. The autocorrelation method is widely used because it always ensures that the resulting filters are stable.

For further processing, the linear prediction coefficients are often converted into the line spectral frequency representation. The fully reversible conversion can be carried out by first calculating the roots of the polynomials

$$\begin{aligned} P(z) &= A(z) + z^{-(m+1)} A(z^{-1}), \\ Q(z) &= A(z) - z^{-(m+1)} A(z^{-1}). \end{aligned} \tag{2}$$

Then, the LSF representation is formed simply by the angular positions $\{\omega_k\}$ of the complex roots in ascending order. The LSF representation is favored in different areas of speech processing for many reasons. For example, this representation offers advantageous properties from the viewpoint of quantization, interpolation and other processing, and it can guarantee filter stability.

The LSF representation has also been widely used in voice conversion. In selection based voice conversion, some distance measure is needed. The distance between two LSF vectors can be computed e.g. using weighted squared error with a diagonal weighting matrix,

$$d(\boldsymbol{\omega},\hat{\boldsymbol{\omega}}) = (\boldsymbol{\omega}-\hat{\boldsymbol{\omega}})^{\mathrm{T}}\,\mathbf{W}(\boldsymbol{\omega}-\hat{\boldsymbol{\omega}}) = \sum_{k=1}^{m} w_k(\omega_k-\hat{\omega}_k)^2 \ . \tag{3}$$

The weights can be used for approximating the properties of human hearing. We use the weights given in [13] defined by

$$w_k = c_k \left| H\left(e^{j2\pi f_k/f_s}\right)\right|^{0.6}, \tag{4}$$

where f_k denotes the frequency of the kth LSF element, f_s is the sampling frequency, and $H(z)$ denotes the synthesis filter $H(z) = 1/A(z)$. Furthermore, when dealing with 10-dimensional LSF vectors at a sampling frequency of 8 kHz, c_k is set to one for all k except for $c_9 = 0.64$ and $c_{10} = 0.16$, as proposed in [13].

In addition to the weighted squared error distance, another useful and popular metric for measuring the distance between two LP spectra is spectral distortion (SD). It is defined in dB as

$$SD = \sqrt{\frac{1}{f_{u-}f_l}\int_{f_l}^{f_u}\left(20\log_{10}\frac{\left|H\left(e^{j2\pi f/f_s}\right)\right|}{\left|\hat{H}\left(e^{j2\pi f/f_s}\right)\right|}\right)^2 df} \ , \tag{5}$$

where f_l and f_u denote the lower and upper frequency limits of the integration. A convenient property of this measure is the fact that there are generally accepted SD based criteria for perceptual spectral transparency, i.e. criteria that guarantee that two spectra are indistinguishable through listening. In [13], it was concluded that transparency is achieved if the following three criteria are met: 1) average SD is less than 1 dB, 2) there are no outlier frames having SD above 4 dB, and 3) less than 2% of frames have SD in the range from 2 to 4 dB.

3. Experimental results

To study the performance level achievable in voice conversion using the frame-based selection approach, we carried out experiments in idealized conditions. The main idea in these experiments was to focus only on the frame selection by making the assumption that other parts of voice conversion would perform perfectly. In practice, we achieved this perfect conversion using recorded sentences from the target speaker as "converted test sentences". Frame-based selection was then applied on these recorded test sentences by replacing the LSF vectors in the test sentences with the best matches found in a selection database. The selection database was formed using uncompressed LSF vectors estimated from the speech of the target speaker. We experimented with various selection database sizes but different sentences were always used in testing and training, making the experiment realistic apart from the above-mentioned assumption of idealized conditions. Thus, the results achieved in these experiments demonstrate the upper bound for the performance of frame-based selection in voice conversion.

The experiments were carried out using the publicly available CMU Arctic database [12], a database of 1132 utterances spoken by 7 different speakers, 2 female and 2 male American English speakers, 1 Canadian English male, 1 Scottish English male, and 1 male speaker with Indian accent. The waveforms in the databases were downsampled to 8 kHz and 10th order LP analysis was performed at 10-ms intervals with overlapping 25-ms analysis frames, using the analysis module of the voice conversion system presented in [14]. Each analysis frame was windowed using a Hamming window and the LP coefficients were computed using the autocorrelation method.

Each speaker served as a reference speaker (speaker in test sentences) and as a selection database speaker for him/herself. In addition, each speaker was also used as a database speaker for the other speakers for comparison purposes. The number of sentences in the selection databases was varied (5, 10, 20, 50 and 100) by including new sentences in such a way that larger sets always also contained the sentences included in the smaller sets. All 80 reference sentences and the database sentence sets were selected randomly but they were kept the same for all speaker combinations.

The new LSFs replacing the LSFs in the original reference sentences were selected from the selection database using the weighted squared error distortion in Eq. (3) together with the weighting in Eq. (4). This scheme was used to obtain a reasonable computational complexity. The final results were evaluated using the spectral distortion formula in Eq. (5) since it provides the best comparison capabilities. Frames classified as silence were not included in the results. The average spectral distortion, measured in the range from 0 to 3.2 kHz, and the percentage of 2 and 4 dB outliers were calculated for two different categories: i) the reference speaker is the same as the database speaker (7 cases) and ii) the reference speaker is different than the database speaker (42 cases).

The mean SD averaged over all speakers is shown in Figure 1 for categories i) (solid line) and ii) (dash-dotted line). The best and worst results in category i) are also shown (dashed lines). The dotted line represents the mean values of each reference speaker's best results when selecting from another speaker's database, i.e. the result with another speaker's database that gave the lowest average spectral distortion values for the reference speaker. The mean percentage of 2 dB and 4 dB outliers is shown in Figure 2 and Figure 3, respectively.

As can be seen from Figure 1, the best matching LSFs were on average far away from ideal transparent quality. There are large differences between the speakers but even the best results were not very good. The low number of 4 dB outliers with larger databases is encouraging, but the requirement of having less than 2% of 2 dB outliers is far from being fulfilled. As expected, using other speaker's database was not as successful as using the speaker's own database, indicating that there are strong speaker-dependencies in LSFs. An interesting observation not directly visible in the figures was that the best results with other speaker's LSFs were always achieved when the LSFs were selected from a speaker with a matching gender. This is in line with the fact that the formant frequencies of female speakers are generally higher than the formant frequencies of male speakers due to the shorter vocal tract.

We also examined whether the quality would be much better if the number of sentences in the database was significantly increased. A set of 250 sentences resulted in an average SD of 1.3 dB for category i) with 7% and 0.1% of 2 dB and 4 dB outliers, respectively. The best result among the speakers was 1.15 dB. In addition, we tested if the usage of the whole Arctic database (1132 sentences minus one reference sentence) as the selection database could result in low spectral distortion. The mean of averaged SD was 1.09 dB for all speakers and the best speaker obtained an average SD of 0.97 dB, measured using 20 different reference sentences. There were 2.2 % of 2 dB outliers and no 4 dB outliers. Using the whole database offers almost transparent quality. For the best other speaker, the average SD was 1.45 and the percentage of 2 dB outliers about 14%. Nevertheless, the database of this size would not be suitable for practical voice conversion.

4. Discussion

LSF selection from a single frame does not seem to provide very high spectral quality if the size of the database is realistic from the viewpoint of practical applications. In [11], the authors do mention that there is a relatively large non-parallel database available – which means in their case over 12 minutes of data. It can be considered as a very large database for voice conversion. This would equal to almost 250 sentences if a sentence is on average 3 seconds long. The results presented in this paper show that transparent quality cannot be achieved even with this kind of relatively large database in idealized conditions.

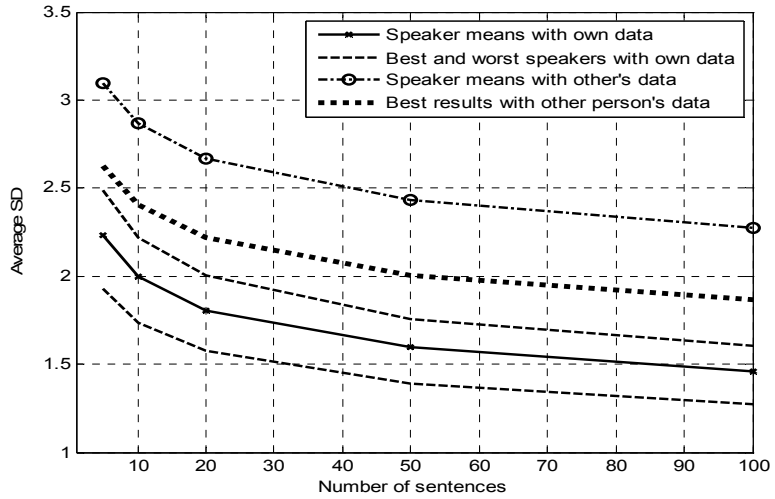


Figure 1. Spectral distortions of LSF databases gathered from the same speaker or from other speakers.

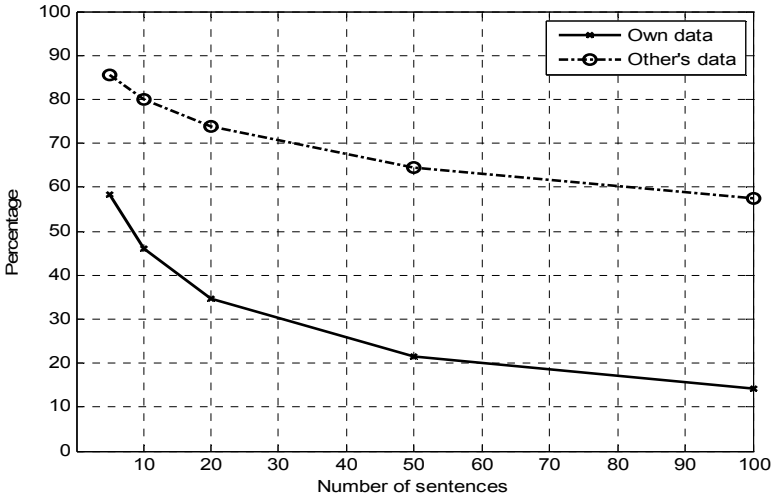


Figure 2. The mean percentage of 2 dB outliers for all speakers

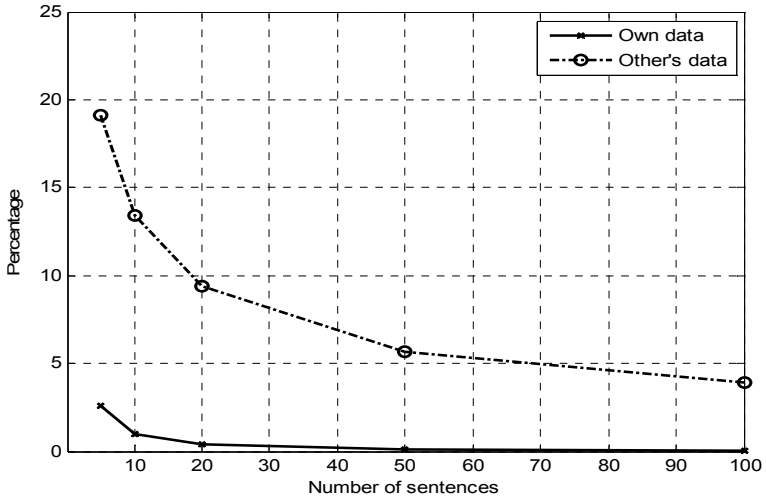


Figure 3. The mean percentage of 4 dB outliers for all speakers.

Even if there would be enough sentences to fulfill the requirement of transparent or otherwise very high quality, there is no target signal available during the conversion and thus the selections must be based on the source speaker's sentence. This moves the realistically achievable quality even further away from the transparent level. Moreover, we have only considered LSFs in this study. In reality, there is also a need for transforming the residual. Residual selection techniques have been proposed to be based on the LSF vector and its corresponding residual. In [6], residual selection was analyzed and it was found that the selection of an optimal LSF sequence similarly as in unit selection can be more preferable than direct selection without considering neighboring frames. Nevertheless, the residual selection was ultimately based on the converted LSF vector, and it is reasonable to assume that residual selection will be even more challenging than LSF selection.

5. Conclusions

In this paper, we analyzed whether it is possible to select LSF vectors from a small database with very high quality in the scope of voice conversion. The CMU Arctic database with 7 speakers was used to test if a small set of sentences could act as an effective selection database in a voice conversion. We found that small database sizes commonly used in voice conversion are not adequate for representing the LSF space of a speaker and the achievable quality is far from transparent quality even in ideal conditions.

References

1. *A. Black, A. Hunt.* Unit selection in a concatenative speech synthesis system using a large speech database. In Proc. of ICASSP, pp. 373-376, 1996.

2. *Y. Stylianou, O. Cappe, E. Moulines*. Continuous probabilistic transform for voice conversion. IEEE Trans. on Speech and Audio Procdessing, vol. 6(2), pp. 131-142, March 1998.

3. *M. Abe, S. Nakamura, K. Shikano, H. Kuwabara.* Voice conversion through vector quantization. In Proc. of ICASSP, pp. 565-568, 1988.

4. *O. Turk, L. M. Arslan.* Robust processing techniques for voice conversion. Computer Speech and Language, vol. 4(20), pp. 441-487, October 2006.

5. J. Yamagishi, K. Ogata, Y. Nakano, J. Isogai, T. Kobayashi. HSMM-based model adaptation algorithms for average voice-based speech synthesis. In Proc. of ICASSP, vol. I., pp. 77-80, 2006.

6. D. Sündermann, H. Höge, A. Bonafonte, H. Ney, A. Black. Residual prediction based on unit selection. In Proc. of ASRU, pp. 369-374, 2005.

7. D. Reynolds. Experimental evaluation of features for robust speaker identification, IEEE Trans. on Speech and Audio Processing, Vol. 2, no. 4, pp. 639-643, October 1994.

8. T. Kinnunen, E. Karpov, P. Fränti. Real-time speaker identification and verification, IEEE Trans. on Audio, Speech and Language Processing, vol. 14, no. 1, pp. 277-288, January 2006.

9. *W. Jia, W.-Y. Chan*. An experimental assessment of personal speech coding. Speech Communication, vol. 30, no. 1, pp. 1–8, 2000.

10. *C.-H, Lee, S.-K. Jung, H.-G. Kang.* Applying a speaker-dependent speech compression technique to concatenative TTS synthesizers. IEEE Trans. on Audio, Speech and Language Processing, vol. 15, no. 2, pp. 632-640, February 2007.

11. *T. Dutoit, A. Holzapfel, M. Jottrand, A. Moinet, J. Pérez, Y. Stylianou.* Towards a voice conversion system based on frame selection, in Proc. of ICASSP, vol. 4, pp. 513-516, 2007.

12. *J. Kominek, A. Black*. CMU Arctic databases for speech synthesis version 0.95. Technical report, Carnegie Mellon University, 2003.

13. *K. Paliwal, B. Atal.* Efficient vector quantization of LPC parameters at 24 bits/frame. IEEE Trans on Speech and Audio Processing, Vol. 1, no. 1, pp. 3-14, January 1993.

14. *J. Nurminen, V. Popa, J. Tian, Y. Tang, I. Kiss.* A parametric approach for voice conversion. In Proc. Workshop on Speech-To-Speech Translation, pp. 225-229, 2006.

Publication P5

Elina Helander, Hanna Silén, Joaquin Míguez, and Moncef Gabbouj, Maximum a posteriori voice conversion using sequential Monte Carlo methods. In *Proceedings of the 10th Annual Conference of the International Speech Communication Association, Interspeech*, Makuhari, Chiba, Japan, Sep. 2010, pp. 1716–1719.

Maximum a posteriori voice conversion using sequential Monte Carlo methods

Elina Helander[1], *Hanna Silén*[1], *Joaquin Míguez*[2], *Moncef Gabbouj*[1]

[1]Department of Signal Processing, Tampere University of Technology, Finland
[2]Departamento de Teoria de la Senal y Comunicaciones, Universidad Carlos III de Madrid, Spain
elina.helander@tut.fi, hanna.silen@tut.fi, joaquin.miguez@uc3m.es, moncef.gabbouj@tut.fi

Abstract

Many voice conversion algorithms are based on frame-wise mapping from source features into target features. This ignores the inherent temporal continuity that is present in speech and can degrade the subjective quality. In this paper, we propose to optimize the speech feature sequence after a frame-based conversion algorithm has been applied. In particular, we select the sequence of speech features through the minimization of a cost function that involves both the conversion error and the smoothness of the sequence. The estimation problem is solved using sequential Monte Carlo methods. Both subjective and objective results show the effectiveness of the method.

Index Terms: voice conversion, maximum a posteriori, Viterbi algorithm, smoothing, particle filter

1. Introduction

Voice conversion aims at modifying speech spoken by one speaker (*source*) to give an impression that it was spoken by another specific speaker (*target*). The voice conversion process consists of two phases: training and conversion. In training, a mapping model from source features to target features is created based on training data from both speakers. In the conversion phase, any unknown utterance from the source speaker can be converted to sound like the target speaker. Speech is usually parameterized into excitation and spectral envelope for enabling modifications and mapping functions on feature level.

Many voice conversion methods proposed in the literature are based on frame-wise mapping of features. They do not take into account the inherent correlation between consecutive frames. In the case of multiple conversion functions, e.g. multivariate regression in a codebook based mapping [1], discontinuities may occur in the boundaries of a conversion function change. This can decrease the subjective quality. A widely used approach, Gaussian mixture model (GMM) based mapping (e.g. [2]) is assumed to make transitions smoother. However, there is usually only a single GMM component that dominates in each frame and the transitions from one frame do not take place smoothly [3] leading to discontinuities. An approach for solving the time-independency problem was proposed in [4] through the introduction of maximum likelihood estimation of spectral parameter trajectory based on GMMs. In [5], the converted features were low-pass filtered after conducting the transformation. Alternatively, the GMM posterior probabilities can be smoothed before making the conversion [3].

In this paper, we introduce an approach that tries to balance between the quality of the frame-based converted speech and the natural continuity of the target speech features. In particular, we propose to select the sequence of speech features through the minimization of a cost function that involves both the conversion error and the smoothness of the sequence. We apply the method for spectral features but it can be used with any type of continuous features.

The proposed sequence optimization is carried out as a post-processing step that follows a frame-based standard algorithm. Here, we obtain conversion functions which are learned from the data using partial least-squares regression models with cross-validation, but the methodology introduced in this paper is by no means restricted to this specific technique. Indeed, any frame-by-frame conversion procedure can be applied prior to the optimization.

Assuming that the different spectral features are independent, we can improve each sequence of converted features individually. To be specific, the aim is to find a spectral feature sequence that minimizes a cost function representing a trade-off between frame-based conversion and continuity. This is typically a high-dimensional optimization problem that often cannot be solved with classical techniques. Our approach is taken from [6] and involves the transformation of the cost minimization problem into one of maximum a posteriori (MAP) estimation in a dynamical system. The estimation problem, in turn, can be solved using sequential Monte Carlo methods.

The rest of the paper is organized as follows. Section 2 describes an algorithm for constructing the local regression matrices for frame-based mapping. Section 3 is the core of the paper and describes the class of cost functions we consider, how the optimization problem is transformed into one of estimation and the particle methods applied to approximate the optimal sequences of features. Some results are shown in Section 4, including both objective and subjective experiments to compare the proposed method with two techniques that carry out frame-based conversion using regression matrices but no post-processing. Finally, Section 5 is devoted to the conclusions.

2. Cluster Partial Least Squares Regression

To avoid the problem of discontinuity, a global transformation function can be used for mapping aligned source speech features s_i^{src} into target speech features s_i^{tgt}. However, this may be ineffective. E.g., the optimal mapping functions for vowels, where the effect of the vocal tract is clearly visible, are rather different from the optimal mapping functions for fricatives.

We propose to use partial least squares (PLS) regression to build local models for different clusters. The idea of PLS is to create orthogonal score vectors (latent components) by maximizing the covariance between predictors and responses. The only thing that needs to be chosen is the number of latent components. If it is set to the number of predictors, PLS is the same as multivariate regression. Otherwise the regression matrix is of lower rank. There are many ways to extract the latent vectors, e.g. the SIMPLS algorithm [7] that we use. The optimal number of components can be chosen using cross-validation.

The algorithm for cluster partial least squares (CPLS) can be outlined as follows.

1. Initialize cluster memberships randomly or with k-means for each datapoint for clusters G_i, where $i = 1, 2 \ldots, K$.
2. Build local models β_i for each cluster G_i using PLS regression with h_i latent components chosen from cross-validation results.
3. Calculate the prediction error for each local model and update the memberships as

$$s_t^{src} \in G_i \quad \text{if} \quad \|s_t^{tgt} - \beta_i s_t^{src}\|^2 < \|s_t^{tgt} - \beta_j s_t^{src}\|^2 \tag{1}$$

   for all $j = 1, 2, \ldots K$.
4. Calculate new cluster centers for each cluster G_i

$$\mu_i = \frac{1}{|G_i|} \sum_{s_t^{src} \in G_i} s_t^{src} \tag{2}$$

5. Assign memberships according to the cluster centers as

$$s_t^{src} \in G_i \quad \text{if} \quad \|s_t^{src} - \mu_i\|^2 < \|s_t^{src} - \mu_j\|^2 \tag{3}$$

   for all $j = 1, 2, \ldots K$. The objective function to minimize is

$$J = \sum_{i=1}^{K} \sum_{s_t^{src} \in G_i} \|s_t^{tgt} - \beta_i s_t^{src}\|^2 \tag{4}$$

6. Repeat steps 2-5 until there is only a small decrease in the objective function or the maximum number of iterations is achieved.

3. Post-processing of converted speech

3.1. Optimization criterion

After converting the source speech sequence by frame-by-frame transformation techniques e.g. as described in Section 2, there is no guarantee of temporal continuity of the features. Temporal continuity is inherent in speech and our goal is to balance between the quality of frame-based mapping and the continuity of the features. To jointly handle these two figures of merit, we build a cost function for a target spectral feature sequence $s_{1:T} = \{s_1, \ldots, s_T\}$ (note that we drop the *tgt* superscript in the sequel), where the integer T is possibly large but finite. The function up to time t has the form

$$C_t(s_{1:t}) = \sum_{i=1}^{t} \alpha |s_i - y_i|^p + \sum_{i=2}^{t} \gamma |s_i - \rho_i(s_{i-1})|^q, \tag{5}$$

where y_i is the converted speech for the i-th frame, α and γ are scale factors chosen to trade off between the quality of the frame-by-frame conversion (first term) and continuity (second term), ρ_i is a prediction function that yields the expected value of s_i from s_{i-1} and $p, q > 0$. Note that in this paper we focus on the post-processing of each feature sequence separately, but vectors of features can also be handled jointly in a similar way.

The post-processing of the converted frames $y_{1:T}$ is performed by choosing a sequence $\hat{s}_{1:T}$ that minimizes the proposed cost, i.e.,

$$\hat{s}_{1:T} \in \arg\min_{s_{1:T}} C_T(s_{1:T}). \tag{6}$$

3.2. MAP estimation

Following [6] we transform the problem (6) into one of MAP estimation in a dynamical system matched to the cost C_T. Specifically, consider the sequence of probability density functions (pdf's)

$$\pi_t(s_{1:t}|y_{1:t}) \propto \exp\{-C_t(s_{1:t})\}, \quad t = 1, 2, ..., T, \tag{7}$$

where we are assuming the integrability of every $\exp\{-C_t(s_{1:t})\}$. Obviously, the minimization of C_t is equivalent to the maximization of π_t, i.e.,

$$\arg\min_{s_{1:t}} C_t(s_{1:t}) = \arg\max_{s_{1:t}} \pi_t(s_{1:t}|y_{1:t}).$$

Moreover, since the cost function is additive,

$$C_t(s_{1:t}) = C_{t-1}(s_{1:t-1}) + \alpha|y_t - s_t|^p + \gamma|s_t - \rho_t(s_{t-1})|^q, \tag{8}$$

it is straightforward to obtain a recursive decomposition of π_t, namely

$$\pi_t(s_{1:t}|y_{1:t}) \propto \pi_{t-1}(s_{1:t-1}|y_{1:t-1})\lambda_t(y_t|s_t)\tau_t(s_t|s_{t-1}), \tag{9}$$

where $\lambda_t(y_t|s_t) \propto \exp\{-\alpha|y_t - s_t|^p\}$ plays the role of the *likelihood* of s_t given the converted frame feature y_t and $\tau_t(s_t|s_{t-1}) \propto \exp\{-\gamma|s_t - \rho_t(s_{t-1})|^q\}$, $t \geq 2$, is a transition density that determines the dynamics (time evolution) of the sequence s_t. The pair λ_t, τ_t, together with a uniform density $s_1 \sim U(S_1)$ in some suitable interval S_1 describes the dynamic state-space model

$$s_t \sim \tau_t(s_t|s_{t-1}), \quad y_t \sim \lambda_t(y_t|s_t), \quad t \geq 2 \tag{10}$$

which has

$$\pi_t(s_{1:t}|y_{1:t}) \propto \lambda_1(y_1|s_1)\tau_1(s_1) \prod_{k=2}^{t} \lambda_k(y_k|s_k)\tau_k(s_k|s_{k-1}) \tag{11}$$

as a posterior pdf of $s_{1:t}$ given $y_{1:t}$.

Therefore, the desired sequence of features $\hat{s}_{1:T} \in \arg\max_{s_{1:T}} \pi_T(s_{1:T}|y_{1:T})$ is actually a MAP point-estimate for the system (11).

3.3. Implementation by particle approximations

As shown in [6], $\hat{s}_{1:T}$ can be approximated using sequential Monte Carlo methods, either with a straightforward extension of the standard particle filtering algorithm or with a combination of the latter with the Viterbi algorithm, as originally proposed in [8]. In both cases, almost sure convergence of the approximation is guaranteed.

3.4. Setting of parameters

The sequence $s_{1:t}$ in (5) is the target spectral feature sequence that we want to estimate. The error variance of the state transition process (i.e., $0.5\gamma^{-1}$) is determined from the data. In addition, the state-space model is able to take into account the errors made at the conversion ("measurement") phase. The variance of the conversion error can be determined for each cluster k separately and change the α in (5) to be dependent on the current frame. In this paper, we used a common α for all the data and determined the conversion error variance (i.e., $0.5\alpha^{-1}$) jointly for all clusters. The conversion error can be calculated using PLS with cross-validation resulting more realistic values.

The prediction function ρ_i in (5) describes the desired change between frames. It is not known for the target speaker. There are two main options. One is to predict it for the target by building a model. In [4], target dynamics are modeled together with statistical features in GMM-based conversion. Our approach is not restricted to GMMs that pose challenges related to covariance matrices. Nevertheless, the models may be rather difficult to obtain from a small amount of data, though, and they tend to become rather averaged. Alternatively, we can assume that the spectral feature dynamics are somewhat speaker-independent and copy the dynamics from the source features. In order to avoid too detailed dynamics, we decided to model $\rho_i(s_{i-1})$ as s_{i-1} added to an offset term. The offset term is taken from the average dynamics of the source speech features s_i^{src} from P previous and P next frames of a current frame. Hence, the prediction functions become

$$\rho_i(s_{i-1}) = s_{i-1} + \frac{1}{2P+1} \sum_{i=t-P}^{t+P} (s_i^{src} - s_{i-1}^{src}) \qquad (12)$$

4. Experiments and results

Both objective and subjective results were carried out to evaluate the performance of the proposed method. We conducted tests for two speaker pairs: male-to-female (M-F), and female-to-female (F-F). The target speaker was the same in both.

The analysis-synthesis system STRAIGHT [9] was used for extracting F_0, aperiodicity and the spectral envelope at 5 ms steps. The spectral envelope was represented with 24-order Mel-cepstrum coefficients (MCCs) resulting in 25 cepstral parameters. The first term describing the energy was not used and in the sample generation it was copied from the source. Aperiodicity was calculated at five bands and it was converted using a single regression matrix calculated with PLS regression where both source MCCs and 5-band aperiodicity served as predictors for 5-band aperiodicity of the target. F_0 was converted by transforming the mean and variance in a logarithmic scale and voicing decisions were copied from the source speaker.

4.1. Objective results

For each speaker pair, conversion functions were built based on 30 sentences and 10 sentences were left for testing. MCCs were aligned with dynamic time warping and some data pairs were omitted based on heuristic energy threshold since they were thought to be silent frames. The objective results were run four times where each time 30 sentences were used for training and 10 for testing. Each of the 40 sentences served once as a testing sentence. For the test data, the post-processing procedure described in Section 3 was conducted for the whole sentence but the objective results were calculated based on data that had gone through a similar selection and alignment process as the training data.

We compared the mapping algorithm described in Section 2 with and without MAP sequence estimation, referred to as *MAPCPLS* and *CPLS*, respectively. The results are compared to the usage of one global regression matrix (referred to as *GRM*) and to the usage of full rank matrices (referred to as *CRM*) instead of PLS for each cluster. The amount of training data used in each training set was on average 16000 frames.

First we evaluated the performance without post-processing. The effect of the number of clusters on spectral distortion averaged for the two speaker pairs is shown in Figure 1 for *CPLS* (solid line), *CRM* (dashdotted line), and *GRM* (dashed line). It seems that a rather low number of clusters is a best choice. This can be explained by the fact that the data is multidimensional and the cluster centers may become overlapping in the case of many clusters. If the number of clusters is high, there are big differences between the *CPLS* method and the *CRM* method. For a low number of clusters this effect is not clearly visible.

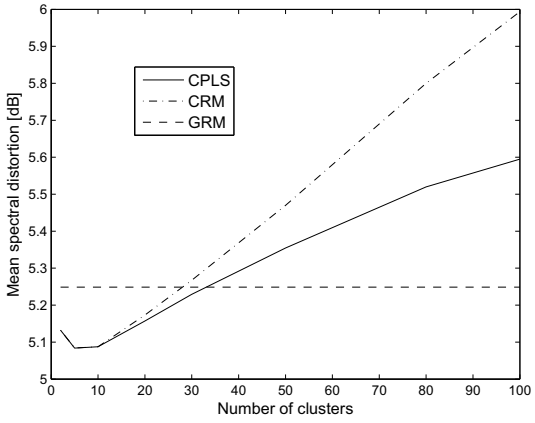


Figure 1: *Mean spectral distortion using a different number of clusters.*

Table 1: Mean spectral distortion in dB using 5, 10, 20, and 30 clusters.

	5	10	20	30
GRM	5.25 dB	5.25 dB	5.25 dB	5.25 dB
CPLS	5.08 dB	5.08 dB	5.16 dB	5.22 dB
MAPCPLS	5.01 dB	4.99 dB	5.01 dB	5.04 dB

We evaluated how the number of particles affects the error when using the *MAPCPLS* method. The *MAPPLS* method was implemented using the particle filter combined with the Viterbi algorithm [8]. The mean squared error (MSE) results for post-processing the 1st, 4th, and 10th MCC are given in Figure 2 in solid, dashed and dotted line, respectively. The MSE is calculated relative to the MSE in the case where post-processing is not used (i.e. *CPLS*) shown as dashdotted line in Figure 2. The results are obtained from the F-F case. As can be seen, the first MCC achieved the highest relative improvement using post-processing. Improvement is achieved also with other MCCs, but differences are smaller. The computational complexity of the Viterbi algorithm for one sentence of T frames, N particles and M MCCs is $O(N^2TM)$. Figure 2 indicated that after having 100 or more particles, the result is not improved anymore. We can also use a different amount of particles for different MCCs depending on their importance to decrease the computational complexity. In the experiment, we used Euclidean norm in the cost function ($p, q = 2$) and P in (12) was set to 2. This configuration was also used in the rest of the experiments.

Table 1 gives the results using 5, 10, 20, and 30 clusters. The number of particles in this case was 100. Note that the results are frame-based errors. Nevertheless, *MAPCPLS* obtains the lowest average spectral distortion and can even compensate well for the error when the number of clusters was too high (20 or 30 clusters) for the *CPLS* method.

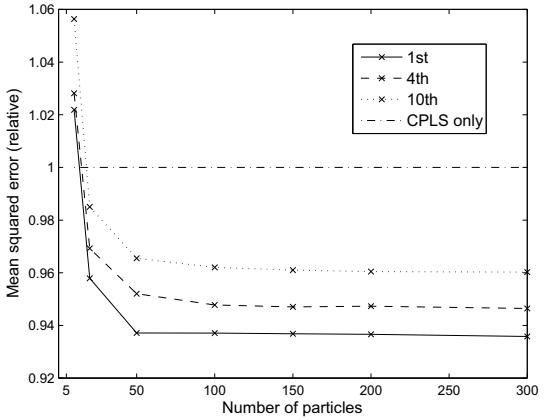


Figure 2: *The effect of the number of particles on mean squared error for the 1st, 4th, and 10th MCC relative to the mean squared error of not using post-processing.*

Table 2: Quality preference percentages from a listening test with 95% confidence intervals.

Quality	Methods	MAPCPLS preferred
M-F	MAPCPLS vs. CPLS	82.5±8.5%
M-M	MAPCPLS vs. CPLS	90.0±6.7%
In total	MAPCPLS vs. CPLS	86.3±5.4%
M-F	MAPCPLS vs. GRM	77.5±9.4%
M-M	MAPCPLS vs. GRM	70.0±10.3%
In total	MAPCPLS vs. GRM	73.8±6.9%

4.2. Subjective results

The objective quality is usually measured in frame-wise measures and does not indicate continuity between frames. Although the objective results are promising, it is difficult to say what kind of changes make the speech sound natural and what kind of changes annoy.

We conducted a listening test to compare the quality of the methods. The test consisted of quality comparisons between the *CPLS* and the *MAPCPLS* method as well as between the *GRM* and the *MAPCPLS* method. Aperiodicity and f_0 were transformed similarly in both sentences. Postfiltering of MCCs was not employed. The number of clusters was 10. A few examples are provided in `http://www.cs.tut.fi/sgn/arg/IS2010_VC/map.html`.

Ten subjects participated in the listening test and compared in total 32 randomly chosen sentence pairs in terms of voice conversion quality. The results are shown in Table 2. According to the results, *MAPCPLS* was preferred the most. It was clearly better than the *CPLS* where no post-processing was conducted. The difference was not that clear when comparing it to *GRM* that preserves the continuity rather well. In many papers the quality of a new voice conversion algorithm is compared against the conventional GMM methods. We suggest to compare against the *GRM* method, since it is likely that *GRM* most often beats the conventional GMM methods that do not preserve continuity [3].

5. Discussion and conclusions

We have proposed a novel post-processing method for voice conversion features. It is based on minimizing a cost function that tries to balance between frame-by-frame mapping and temporal continuity. The mapping method used in the paper was based on locally built partial least squares regression matrices. Nevertheless, the proposed post-processing procedure can be used in conventional GMM-based conversion or with codebooks as well.

The cost function provides flexibility on balancing between the converted sequence and the natural evolvement of target parameters. Further investigation on the scaling factors as well as for the continuity model in the cost function can lead to better performance. In addition to having cluster-specific conversion error variance, the data could be divided into clusters in terms of temporal predictability. In the experiments, we used Euclidean norm and a linear model for continuity. In this case, it would have been possible to use Kalman smoothing. However, we chose the more general particle filtering approach to account for the possibility of setting different norms and non-linear continuity model as well as the potentiality to extend the problem into multiple dimensions.

A similar cost optimization procedure can be used in unit selection speech synthesis to reduce the mismatch between the boundary of consecutive units. The cost function is flexible and more emphasis could be given on the observations in the middle of a unit whereas near boundaries more weight could be given to the continuity.

6. Acknowledgements

This work was supported by the Academy of Finland, (application number 129657, Finnish Programme for Centres of Excellence in Research 2006- 2011). J. M. acknowledges the support of the Ministry of Science and Innovation of Spain (Program Consolider-Ingenio 2010 CSD2008-2010 COMONSENS and project DEIPRO TEC2009-14504-C02-01).

7. References

[1] H. Valbret, E. Moulines, and J. Tubach, "Voice transformation using PSOLA technique," in *Proc. of ICASSP*, 1992, pp. 145–148.

[2] Y. Stylianou, O. Cappe, and E. Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Trans. on Speech and Audio Processing*, vol. 6(2), pp. 131–142, March 1998.

[3] E. Helander, T. Virtanen, J. Nurminen, and M. Gabbouj, "Voice conversion using partial least squares regression," *IEEE Trans. on Speech and Audio Processing*, vol. 18(5), pp. 912–921, July 2010.

[4] T. Toda, A. Black, and K. Tokuda, "Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory," *IEEE Trans. on Speech and Audio Processing*, vol. 15(8), pp. 2222–2235, Nov. 2007.

[5] Y. Chen, M. Chu, E. Chang, J. Liu, and R. Liu, "Voice conversion with smoothed GMM and MAP adaptation," in *Proc. of EUROSPEECH*, 2003, pp. 2413–2416.

[6] J. Míguez, D. Crisan, and P. Djurić, "Sequential Monte Carlo methods for the optimization of a general class of objective functions," *SIAM Journal on Optimization*, 2010, submitted.

[7] S. de Jong, "SIMPLS: An alternative approach to partial least squares regression," *Chemometrics and Intelligent Laboratory Systems*, vol. 18, no. 3, pp. 251–263, March 1993.

[8] S. Godsill, A. Doucet, and M. West, "Maximum a posteriori sequence estimation using Monte Carlo particle filters," *Annals of the Institute of Statistical Mathematics*, vol. 53, 2001.

[9] H. Kawahara, I. Masuda-Katsuse, and A. de Cheveigné, "Restructuring speech representations using a pitch-adaptive time-frequency smoothing and a instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds," *Speech Communication*, vol. 27, pp. 187–207, 1999.

Publication P6

Elina Helander, Tuomas Virtanen, Jani Nurminen, and Moncef Gabbouj, Voice conversion using partial least squares regression. *IEEE Transactions on Audio, Speech, and Language Processing*, Special Section on Voice Transformation, vol. 18, no. 5, pp. 912–921, Jul. 2010.

Publication P7

Elina Helander, Hanna Silén, Tuomas Virtanen, and Moncef Gabbouj, Voice conversion using dynamic kernel partial least squares regression. *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 3, pp. 806–817, Mar. 2012.

Tampereen teknillinen yliopisto
PL 527
33101 Tampere

Tampere University of Technology
P.O.B. 527
FI-33101 Tampere, Finland

ISBN 978-952-15-2842-2
ISSN 1459-2045