Computer Science Department
University of Crete

Institute of Computer Science
FO.R.T.H.

On Speaker Interpolation and Speech Conversion for parallel corpora.

MSc Thesis

Giorgios Grekas

Heraklion
November 2010

Department of Computer Science
Faculty of science and engineering
University of Crete

On Speaker Interpolation and Speech Conversion for parallel corpora.

Submitted to the Department of Computer Science in partial fulfillment of the requirements for the degree of Master of Science

November 1, 2010

© 2010 University of Crete & ICS-FO.R.T.H. All rights reserved.

Author:

Giorgos Grekas
Department of Computer Science

Committee:

Supervisor

Yannis Stylianou
Associate Professor

Member

Athanasios Mouchtaris
Assistant Professor

Member

Gerasimos Potamianos
Director of Ressearch (Rank A)

Accepted by:

Chairman of the
Graduate Studies Committee

Angelos Bilas
Professor

Heraklion, November 2010

Abstract

In daily speech the linguistic information plays a major role in the communication between people. However, voice quality and individuality are important in speech recognition and understanding. For instance, it is exceptionally significant to understand and discriminate between two or more speakers in a radio or a television program. Voice individuality, apart from providing the aforementioned advantages in communication, enriches our daily life with variety.

For a number of modern applications it is important to create and maintain data bases for different speakers, for example, in gaming, in text-to-speech synthesis and in cartoon movies. This may be time consuming and expensive, depending on the requirements of the application. Speaker interpolation (SI) is the process of producing an intermediate voice between two or more speakers, while voice conversion (VC) is the technique of processing the voice of one person, namely the source speaker, such that his/her voice resembles the voice of another person, namely the target speaker. Moreover, the converted or interpolated speech should sound natural and intelligible.

Despite the extended research in VC, high-quality voice conversion has not been achieved yeet. A number of reasons explain this current shortcoming, with the main ones being a) the oversmoothing effect by using of statistical modeling b) inaccurate estimation of the speaker-depended features and c)the inadequacy of the used synthesis methods. Voice conversion methods are based on spectral envelope information, which represents the vocal tract, since it has an important role on speech individuality. In conventional VC the excitation signal of the source speaker is extracted first by inverse filtering. Then this excitation signal is filtered from the vocal tract of the target speaker. In speech interpolation the excitation signal is filtered from an interpolated vocal tract of the given speakers.

The scope of this thesis is to deal with this research gap and achieve high quality speech interpolation and voice conversion of parallel corpora using accurate methods for spectral envelope estimation (true envelope), time and frequency alignment (piecewise linear time and frequency warping), and speech synthesis (interpolated lattice filter or overlap and add). With the use of precise methods in each processing step it was expected to reduce the artifacts currently met in voice conversion. In speech interpolation the produced vocal tract is not just an interpolation between the given speakers, but the vocal tract length can be altered, producing a broad range of voices. Hence, given a limited data base a substantially larger one that contains individual speakers for every use can be created.

Περίληψη

Στον καθημερινό λόγο οι γλωσσικές πληροφορίες διαδραματίζουν έναν πρωτεύοντα ρόλο στην επικοινωνία μεταξύ των ανθρώπων. Ωστόσο, η ποιότητα και η ατομικότητα του λόγου είναι επίσης σημαντικά για την αναγνώριση και την κατανόησή του. Για παράδειγμα, η κατανόηση και η διάκριση μεταξύ δύο ή περισσότερων ομιλητών σε ένα ραδιοφωνικό ή τηλεοπτικό πρόγραμμα είναι εξέχουσας σπουδαιότητας. Η ατομικότητα του λόγου, εκτός από τα πλεονεκτήματα της επικοινωνίας, επιπλέον προσθέτει ποικιλία στην καθημερινή μας ζωή.

Πολλές σύγχρονες εφαρμογές απαιτούν τη δημιουργία και διατήρηση βάσεων δεδομένων για διαφορετικούς ομιλητές, όπως π.χ. βιντεοπαιχνίδια, σύνθεση φωνής απο κείμενο, ταινίες, κ.α. Αυτό μπορεί να καταστεί χρονοβόρο και ακριβό, ανάλογα με τις απαιτήσεις της εφαρμογής. Η παρεμβολή ομιλητή (σπεακερ ιντερπολατιον, ΣΙ) είναι η διαδικασία παραγωγής μίας ενδιάμεσης φωνής μεταξύ δύο ή περισσότερων ομιλητών. Η μετατροπή φωνής (οιςε ςονερσιον, "") είναι η τεχνική επεξεργασίας της φωνής ενός ατόμου (αρχικός ομιλητής), ώστε να ακούγεται σαν τη φωνή ενός άλλου ατόμου (τελικός ομιλητής). Η παρεμβαλλόμενη ή μετατρεπόμενη ομιλία θα πρέπει να ακούγεται φυσική και κατανοητή.

Παρά την εκτεταμένη έρευνα στη μετατροπή φωνής, δεν έχει έως τώρα επιτευχθεί η παραγωγή φωνής υψηλής ποιότητας. Ένας αριθμός αιτίων εξηγεί την αδυναμία αυτή, που συνοψίζονται στις α) υπερομαλοποίηση λόγω της χρήσης στατιστικών μοντέλων, β) απουσία ακρίβειας στην εκτίμηση των χαρακτηριστικών της φωνής που εξαρτώνται από τον ομιλητή, και γ) ανεπάρκεια των χρησιμοποιούμενων μεθόδων σύνθεσης. Οι μέθοδοι μετατροπής φωνής βασίζονται σε πληροφορίες της φασματικής περιβάλλουσα, που αντιπροσωπεύει την στοματική κοιλότητα, γιατί διαδραματίζει ένα σημαντικό ρόλο στη διαμόρφωση της ατομικότητας της φωνής. Στη συνήθη μετατροπή φωνής αρχικά εξάγεται το σήμα διέγερσης του ομιλητή-πηγής μέσω αντίστροφου φιλτραρίσματος. Κατόπιν το σήμα ερεθισμού φιλτράρεται από τη στοματική κοιλότητα του ομιλητή-στόχου. Στην παρεμβολή λόγου το σήμα ερεθισμού φιλτράρεται από μία παρεμβαλλόμενη στοματική κοιλότητα των δεδομένων ομιλητών.

Ο στόχος της παρούσας μελέτης είναι να αντιμετωπίσει το ερευνητικό κενό και να επιτύχει υψηλής ποιότητας παρεμβολή ομιλίας και μετατροπή φωνής παράλληλων κειμένων χρησιμοποιώντας ακριβείς μεθόδους εκτίμησης της φασματικής περιβάλλουσας (με την μέθοδο σωστή περιβάλλουσα), στρέβλωση χρόνου και συχνότητας (με τμηματική γραμμική στρέβλωση χρόνου και συχνότητας), και σύνθεση ομιλίας. Με τη χρήση μεθόδων ακριβείας σε κάθε στάδιο επεξεργασίας αναμένονταν η μείωση των αλλοιώσεων που παρουσιάζονται έως τώρα στη μετατροπή φωνής. Στην παρεμβολή ομιλίας η παραγόμενη στοματική κοιλότητα δεν αποτελεί απλά μία παρεμβολή μεταξύ των υπαρχόντων ομιλητών, αλλά το μήκος της κοιλότητας μπορεί να αλλαχθεί, παράγοντας ένα μεγάλο εύρος φωνών. Συνεπώς, έχον-

τας μία περιορισμένη βάση δεδομένων μπορούμε να δημιουργήσουμε μία πολύ πλουσιότερη, η οποία να περιλαμβάνει ατομικές φωνές για κάθε χρήση.

Acknowledgements

Firstly, I would like to thank my advisor teacher Dr Yiannis Stylianou for his guidance, and especially for kindly providing the working space and equipment, essential for the completion of this work. I am also in dept to Dr Yiannis Agiomyrgiannakis for his crucial advice and contribution, particularly during the final months of this journey.

I am grateful to Dr Anastasia Manola, not only for her psychological support throughout these years, but also for her recommendations and corrections on writing up this dissertation.

I owe a big thank you to Dr Giorgio Tsagarakis, for proofreading and advising on the structure of parts of the document. Lastly, I would like to thank my friends and the guys in the lab for all the good and the difficult times we've been through together during the completion of this course.

Contents

List of Figures

List of Tables

List of Symbols and Abbreviations

Abbreviation	Description
AR	Auto-regressive
ARMA	Autoregressive Moving Average
DAP	Discrete All-pole Modelling
DFT	Discrete Fourier Transform
DFW	Dynamic Frequency Warping
DTW	Dynamic Time Warping
FFT	Fast Fourier Transform
FIR	finite-duration impulse response
GMM	Gaussian Mixture Model
IIR	infinite-duration impulse response
IS	Itakuta-Saito distance
LP	Linear Prediction
LPC	Linear Prediction Coding
LSF	Line Spectral Frequencies
MA	Moving Average
PTW	Point-wise Time Warping
SC	Speech Conversion
SI	Speaker Interpolation
STW	Segment-wise Time Warping
TE	True Envelope
VQ	Vector Quantization

Chapter 1

Introduction

1.1 Importance of speaker interpolation and speaker conversion.

Speech is the ordinary way for people to communicate. In daily speech the linguistic information plays a major role in the communication between people. Yet, voice quality and individuality are essential in speech recognition and understanding. For instance, it is exceptionally important to understand and discriminate between two or more speakers in a radio or a television program. Voice individuality, apart from the advantage of communication enriches our daily life with variety.

For a number of modern applications it is important to create and maintain data bases for different speakers. In video games, it is a common thing to see many characters speaking. In cartoons, for each character's voice a different actor his/her voice is needed. In mobile telephony bigger databases of speakers are necessary. Speaker recording requires a large number of people to utter sentences, a procedure deemed costly and time consuming. The method of speaker interpolation is used to offer a helping hand in similar applications.

There are also applications were the individual voice characteristics are useful, because the identification of the speaker is demanded. In voice mailboxes, incoming voicemail is reproduced with the voice identity of the user. In interpreted telephony two speakers of different language communicate, first recognizing the sentence uttered by the speaker, and then translating and composing it with his/her own voice identity. Hence the result is to hear his/her voice in a language he/she might not speak.

In human-machine communication systems speech is focused upon as a medium for such communication. The former are classified in the category text to speech synthesis [20] [38]. Nevertheless they are also useful in method enabling a patient

without a larynx or with a non-functional larynx to produce voice or speech [4]. The method of speech conversion deals with this kind of problems.

1.2 Problem definition

The present thesis aspires to tackle two issues, namely Speaker-Interpolation and Voice Conversion, using the fact that spectral envelope information, which represents the vocal tract, plays an important role in speech individuality [22]. In other words it is the feature that colors voice.

Speaker interpolation (SI) is the process of producing an intermediate voice between two or more speakers. Provided the shape of the glottis, we seek to filter it through an interpolated vocal tract in order to produce an interpolated speaker. The interpolated vocal tract can be a linear combination of the vocal tracts of the speakers to be interpolated.

Speech conversion (SC) is the technique of processing the speech of one person, the source speaker, to sound like the speech of another person, the target speaker. Briefly, in SC the excitation signal of the source speaker is first extracted by inverse filtering. Then, this signal is filtered through the vocal tract of the target speaker, while the excitation can be modified to obtain a conversion perceptually more precise.

In either interpolation or conversion, speech should sound natural and intelligible. Unfortunately high-quality voice conversion has not yet been achieved [38]. Usually speech conversion suffers from artifacts which affect perception naturalness. The aim of the present piece of work is to improve the quality of the produced voice by reducing the hitherto present artifacts.

1.3 motivation

Up to this point there have been numerous attempts to convert speech. These past efforts have relied on a wide variety of methods, such as modeling the glottal source [5], changing the formant frequencies of the target speaker [26], using vector quantization over the aligned spectrum [2], transforming formants via neural networks (in this method the implicit formant transformation is captured by a neural network)[28], only to name a few. The most popular one is the continuous probabilistic transform for voice conversion, with a Gaussian mixture model (GMM) that realizes continuous mapping based on soft mapping [36]. While the mapping is effective, the performance of the conversion is still inadequate. In recent years there have been some improvements proposed for this method [37] [39].

Despite extended research in speech conversion, high-quality speech conversion has not yet been achieved. A number of reasons may explain this current

shortcoming, with the main ones being a) the oversmoothing effect by use of statistical modeling b) lack o precision of the speaker-depended features estimation and c) inadequacy of the used synthesis methods [39], d) possibly reasons that have not been identified as yet.

The objective of this thesis is to deal with this research gap and achieve high quality voice conversion of parallel corpora using accurate methods a) for spectral envelope True Envelope is applied, which is an improved spectral model for a precise representation of the estimation of the speakers' features, b) time and frequency alignment is achieved by piecewise linear time and piecewise linear frequency warping, which compared with dynamic time warping yields better results, c) in synthesis interpolated lattice filter or overlap and add methods is used and is expected to achieve a high quality speech conversion. By the use of parallel corpora the mapping between two speakers features is optimized. In this way a reduction in the sources of the artifacts is achieved, thus facilitating their elimination.

In addition, speech interpolation between two, three or more speakers is proposed in order to achieve high quality speech interpolation. The idea is to extract the glottal signal of a speaker, and use this signal as the input of an interpolated vocal tract of the given speakers. The output will be the interpolated signal. The produced vocal tract is not just an interpolation between the given speakers, but the vocal tract length can be altered, producing a broad range of voices. Hence, given a limited database we can create a substantially larger one that contains many individual speakers.

1.4 Structure of the thesis

The thesis is organized as follows: In Chapter 2, is described methods for spectral envelope estimation, their advantages and disadvantages. In Chapter 3, methods are interpreted for the treatment of spectral envelopes, time and frequency warping algorithms, lattice filtering, and the combination of techniques in order to attain speech conversion and interpolation. Finally, in Chapter 4, we conclude this work and we propose future research directions.

Chapter 2

Spectral Envelope Estimation

2.1 Autoregressive (AR) Method

The Autoregressive (**AR**) [11] [32] or Linear Predictive (**LP**) is one of the earlier developed methods of digital signal processing. Originally, it was utilized on speech transmission and compression. An **AR** process represents a signal $s(n)$ designated by a linear combination of the p preceding values, where p is referred to as the order of the model, plus white noise $v(n)$:

$$s(n) = v(n) + \sum_{k=1}^{p} a_k s(n-k), \tag{2.1}$$

where $a_1, a_2, \cdots, a_p$ are constant called the **AR parameters** or **linear prediction coefficients**. The term autoregressive comes from the fact than in (2.1) the variable $s(n)$ is regressed on previous values of itself; hence the term"autoregressive".

We can rewrite equation (2.1) in the following form:

$$\sum_{k=0}^{M} a_k^* s(n-k) = v(n), \text{ where } a_0^* = 1, a_k^* = -a_k, \text{ for } k \geq 1. \tag{2.2}$$

The left-hand side of (2.2) is a convolution of the input signal and the sequence parameters a_k. Taking the z-transform of equation (2.2) yields the following result:

$$H_a(z)S(z) = V(z) \tag{2.3}$$

where

$$H_a(z) = \sum_{n=0}^{M} a_n^* z^{-n} \tag{2.4}$$

and $S(z), V(z)$ are the z-transform of $s(n)$ and $v(n)$ respectively. We can employ equation 2.3 for two alternative purposes:

1. Given the AR process $s(n)$, we can use equation 2.3 and filter $s(n)$ in order to extract $v(n)$. Filter $H_a(z)$ is an all-zero filter that analyses the input signal. This procedure is known as inverse filtering and is a *finite-duration impulse response* (**IIR**) filter.

2. On the other hand given the white noise $v(n)$ as input, $s(n)$ will become the filter output. In this case the filter in z-transform can be interpreted as $1/H_a(z)$, which is an all-pole filter and represents a process generator. This is an *infinite-duration impulse response* (**IIR**) filter.

2.1.1 Spectral Envelope Estimation from Linear Prediction

Linear prediction (LP), which is well known as linear prediction coding (LPC), has some special properties that make this method suitable for spectral envelope estimation (vocal tract). In section 2.1 equation (2.3) can be viewed as the z-transform of speech sound $S(z)$, vocal tract $H(z) = A/H_a(z)$, A is the gain of the filter which is an all-pole, and of prediction error $E(z)$, respectively. The goal here is to estimate the filter coefficients a_k for specific order p and the gain A to have a representation of vocal tract (transfer function):

$$H(z) = \frac{A}{H_a(z)} \tag{2.5}$$

Denoted by

$$\hat{s}(n) = \sum_{k=1}^{p} a_k s(n-k), \tag{2.6}$$

the approximated value $\hat{s}(n)$ of $s(n)$ is calculated as a linear combination of p $s(n)$ preceding values. The minimization of the prediction error is wanted which is:

$$e(n) = \hat{s}(n) - s(n)$$

In **LP** the coefficients a_k are computed for each frame in a way that the expected value of the squared prediction error

$$E\big[e^2(n)\big] = E\big[\big(\hat{s}[n] - s[n]\big)^2\big] \tag{2.7}$$

is minimized. We arrive at the optimizing solution by taking derivatives with respect to a_k,

$$\frac{\partial E\big[e^2(n)\big]}{\partial a_k} = 0,\ k = 1, 2, \cdots, p. \tag{2.8}$$

Assuming that input $s(n)$ is a stationary process equation, we get from (2.8) a set of p simultaneous equations, with $r(0), r(1), \cdots, r(p)$ being the known quantities and $a_1, a_2, \cdots, a_p$ being the unknown quantities, and produce the following linear system:

$$\mathbf{Ra = r} \tag{2.9}$$

where R is the autocorrelation matrix:

$$R = \begin{bmatrix} r(0) & r(1) & \cdots & r(p-1) \\ r`(1) & r(0) & \cdots & r(p-2) \\ \vdots & \vdots & \ddots & \vdots \\ r`(p-1) & r`(p-2) & \cdots & r(0) \end{bmatrix},$$

$r(k)$ is the autocorrelation function of the input signal, **a** is the vector containing the **LP coefficients**:

$$\mathbf{a} = [a_1, a_2, \cdots, a_p]`T$$

and **r** is the vector

$$\mathbf{r} = [r(1), r(2), \cdots, r(p)]^T.$$

Equation (2.9) is known as the Yule-Walker equations for an autoregressive model.

Although equation (2.9) can be solved using the inverse matrix $\mathbf{R^{-1}}$, $\mathbf{a} = \mathbf{R^{-1}r}$, the Toeplitz form of **R** leads to a more efficient approach. Levinson algorithm is an efficient solution of system (2.9) and is presented briefly in table 2.1.

Initialization: $E_0 = r(0)$
For $1 \leq l \leq p$
$k_l = \frac{1}{E_{l-1}}\left[r(l) - \sum_{j=1}^{l-1} a_j^{l-1} r(l-j)\right]$
$a_l^j = k_l$
$a_j^l = a_j^{l-1} - k_l a_{l-j}^{l-1},\ 1 \leq j \leq l-1$
$E_l = E_{l-1}(1 - k_l^2)$

Table 2.1: Levinson-Durbin algorithm

To compute gain A, the equation (2.5) is transformed in time domain:

$$h(n) = \sum_{k=1}^{p} a_k h(n-k) + Ad(n) \tag{2.10}$$

multiplying equation (2.10) by $h(n)$ and sum over all n the following equations produced, with r_h to be the autocorrelation function of $h(n)$:

$$r_h(0) = \sum_{k=1}^{p} a_k r_h(k) + A^2. \tag{2.11}$$

An approach to compute gain A is to postulate that the energy in the all-pole impulse response $h(n)$ equals the energy in the measurement signal $s(n)$, then $r_h(0) = r(0)$, reminding that $r(n)$ is the autocorrelation function of the signal $s(n)$. Using the property of autocorrelation matching [31], which states that:

$$r_h(k) = r(k), \quad \text{for } |k| \leq p, \tag{2.12}$$

we conclude that by (2.11) and (2.12) the gain can be computed as:

$$A = \sqrt{r(0) - \sum_{k=1}^{p} a_k r(k)} \tag{2.13}$$

Using the normal equation solution it is easy to show that the corresponding minimum mean-squared prediction error is given by

$$E[e(n)] = r(0) - \sum_{k=1}^{p} a_k r(k).$$

Thus from the previous and (2.13) equation yields that $A^2 = E[e(n)]$. Therefore, postulating that the energy in the all-pole impulse response $h(n)$ equals the energy in the measurement signal $s(n)$ brings the squared gain equal to the minimum mean-squared error.

When the residual signal $e[n]$ is minimized, the analysis filter with transfer function given by:

$$\frac{1}{H(z)} = \frac{1 - \sum_{i=1}^{p} a_i z^{-1}}{A}$$

will attempt to achieve a maximally flat spectrum of the input signal. The synthesis filter is the inverse of the analysis filter, and thus the frequencies that have been attenuated by the analysis filter are amplified. The transfer function of the synthesis filter is given by:

$$H(z) = \frac{A}{H_a(z)} = \frac{A}{1 - \sum_{i=1}^{p} a_i z^{-1}}$$

As the analysis filter strives to flatten the spectrum, its inverse filter will describe the **spectral envelope** of the signal.

An intermediate set of parameters can be attained from **LPC-coefficients**, namely the **reflection coefficients** k_i, in Levinson-Durbin algorithm table 2.1, corresponding to the reflection of acoustic waves at the boundaries between successive sections of an acoustic tube. The use of **reflection coefficients** result to some advantages in synthesis, and can be interpolated without affecting the stability of the resulting synthesis filter.

Disadvantages of LP method

One disadvantage of the LP method is the chosen model order. The expected value of the squared prediction error (2.7) is minimized optimum when the **LP-coefficients** order is equal to AR model order, while in practice the real order of the system is unknown.

Another disadvantage is in the case in which the measured signal is deterministic, for example in a speech signal which the excitation pitch period is small,

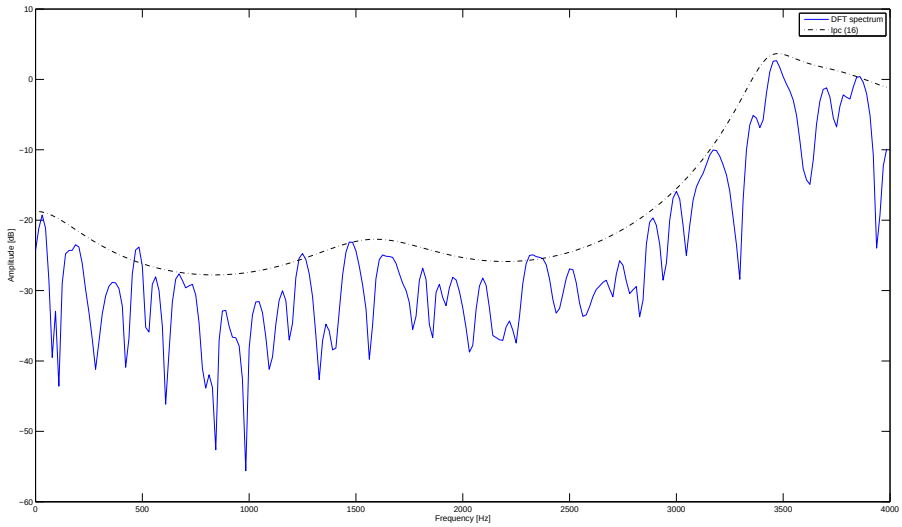


Figure 2.1: The LPC spectral envelope of a stochastic signal (unvoiced speech).

then it can be shown that the autocorrelation function of the speech signal r, is a severely aliased version of the autocorrelation function r_h of the transfer function of the system. This means that the estimated transfer function $\hat{H}$ is only mathematically obliged to match the real transfer function H as a few under-sampled points in the frequency domain.

To have a sense of the last disadvantage the estimated spectral envelope will descend down to the level of residual noise in the gap between two harmonic partials, where the space between partials is large (small pitch period).

Frequently the LP model exhibit sharp peaks near pitch harmonics with large powers, featuring an unnatural vocal contour that underestimates the formant bandwidth, Figure 2.2. This limitation is dealt with with a regularization technique [27] [21] as depicted in Figure 2.3.

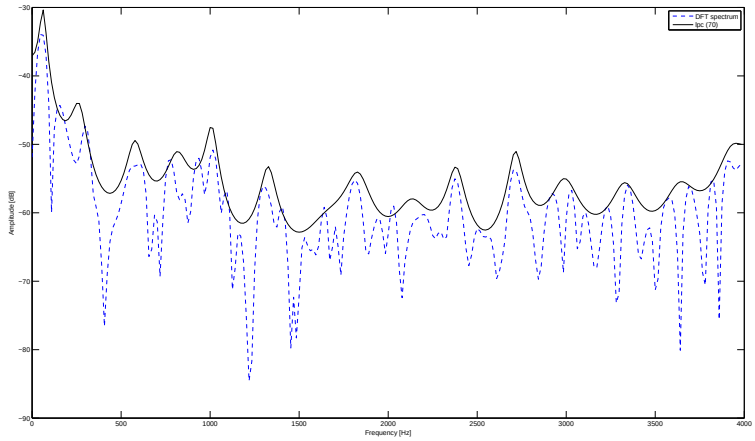


Figure 2.2: The LP spectral envelope of a voiced speech signal, with lpc order 70.

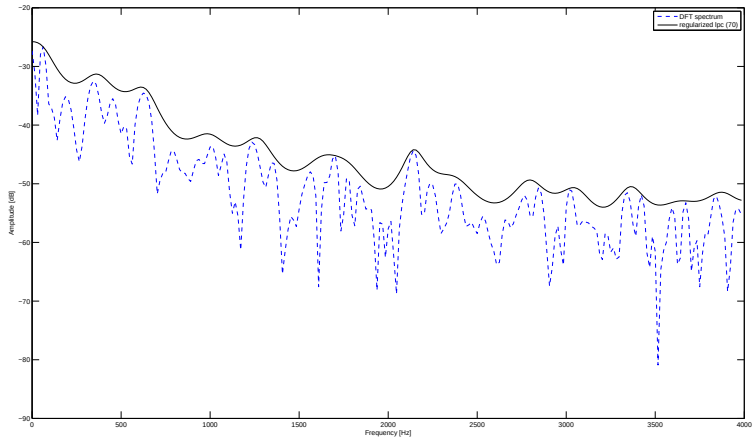


Figure 2.3: The regularized LP spectral envelope of a voiced speech signal, with lpc order 70.

2.2 Discrete all-pole modelling

The idea of **Discrete all-pole modelling (DAP)** [7] is to overcome limitations of the LP method. To be more specific the spectral envelope of deterministic sounds is wanted to be estimated. As presented in section 2.1.1 in **LP** method the first p coefficients of the autocorrelation function r_h of the impulse response of the **AR** model are not equal to the first p coefficients of the autocorrelation function r of the signal, but r is an aliased version of r_h, especially for deterministic sounds with high fundamental frequency f_0. While for stochastic sounds **LP** gives a satisfactory spectral envelope estimation.

Note, **LP** optimization criterion (2.7) is the minimization of the squared prediction error's expected value. However, there are many optimization criteria for the solution of (2.7). All of them can be expressed by the ratio between the signal power spectrum $|S(\omega)|^2$ and the model power spectrum $|\hat{S}(\omega)|^2 = \frac{G^2}{|A(\omega)|}$.

In **LP** the minimization error of (2.7) can be expressed in Itakuta-Saito (IS) distance measure [32][24]:

$$E = \frac{1}{2\pi}\int_{-\pi}^{\pi}\left(\frac{|S(\omega)|^2}{|\hat{S}(\omega)|^2} - \ln\Big[\frac{|S(\omega)|^2}{|\hat{S}(\omega)|^2}\Big] - 1\right)d\omega \tag{2.14}$$

With respect to deterministic signals the contribution of **DAP** is to minimize equation (2.14) over discrete frequencies [43]. These discrete frequencies will be the harmonics, as we are interested in estimated spectral envelope passing through spectrum peaks. Hence, we use a a discrete version of IS distance measure to produce an all-pole power spectrum $\hat{S}(\omega)$:

$$E = \frac{1}{M}\sum_{n=1}^{M}\left(\frac{|S(\omega_n)|^2}{|\hat{S}(\omega_n)|^2} - \ln\Big[\frac{|S(\omega_n)|^2}{|\hat{S}(\omega_n)|^2}\Big] - 1\right)d\omega \tag{2.15}$$

where w_n for $n = 1, 2, \cdots, M$ are the harmonic frequencies. Minimizing the above criterion, i.e. IS distance, we create an all-pole model differing form the conventional LP, whose distance is minimized (2.14) in continuous frequency axis $(-\pi, \pi]$ and is influenced by the energy of frequencies located between two neighboring harmonics.

As in equation (2.8) to find the optimum solution we will set

$$\frac{\partial E}{\partial a_k} = 0, \qquad \text{for } k = 0, 1, \cdots, p \tag{2.16}$$

and gives:

$$\sum_{i=0}^{p} \frac{a_i}{L} \left(\sum_{n=1}^{M}\Big[|S(\omega_n)|^2 - |\hat{S}(\omega_n)|^2\Big]e^{j\omega_n k}\right) = 0 \Rightarrow \tag{2.17}$$

$$\sum_{i=0}^{p} a_i \Big(r[k-i] - \hat{r}[k-i]\Big) = 0 \tag{2.18}$$

where $r, \hat{r}$ are the autocorrelation functions of signal and transfer function respectively. The method suggested by [7] is an iterative procedure that tackles this set of non-linear equations by writing:

$$\hat{h}[-k] = \sum_{i=0}^{p} a_i \hat{r}[k-i] : k = 0, 1, \cdots, p \tag{2.19}$$

$$\sum_{i=0}^{p} a_i r[k-i] = \hat{h}[-k] : k = 0, 1, \cdots, p \tag{2.20}$$

At a first step we perform the standard **LP** analysis to obtain a set of $\hat{a_i}$ coefficients and assign $a_i = \hat{a_i}$. We can then compute $\hat{r}$ from equations (2.17) and (2.18) and a first estimation of $\hat{h}$, from (2.19), is evaluated. Equation (2.20) is then solved to extract a new set of coefficients a_i, which will generally be closer to the required ones than the set of $\hat{a}_i$. To start a new iteration set:

$$\hat{a}_i = (1-\rho)\hat{a}_i + \rho a_i, \text{ for } i = 0, 1, \cdots, p, \tag{2.21}$$

where ρ is a "damping" factor, $\rho \in (0, 1)$. We can re-compute equation (2.19) for $a_i = \hat{a}_i$. The process goes on until E is significantly reduced.

In this way for deterministic signals aliasing is minimized with the iterative process presented above. While with stochastic signals the conventional LP method is used.

Disavantages of All-Pole model

The main disadvantages of All-Pole modeling for spectral envelope estimation are a)there are some problems with filter stability, b)harmonic-peaks tracking is needed [41], c) the optimal order is difficult to be determined.

2.3 Cepstrum Spectral Envelope

The cepstrum is a speech analysis method that relies on a spectral representation of the signal. This method takes advantage of the fact that in the speech production model [31] [19], speech is made up by an excitation sequence $e(n)$ produced by the glottis, together with the impulse response of the vocal tract $h(n)$:

$$s(n) = e(n) * h(n).$$

.

Removing one of two signals is easier when the signals are combined in a linear manner. This can be achieved employing the log magnitude spectra. In frequency domain, convolution becomes the product of the pertinent Fourier transform:

$$X(\omega) = E(\omega) \cdot H(\omega) \tag{2.22}$$

Utilizing the logarithm of the absolute value of the Fourier transform, we convert the multiplication of equation Eq (2.22) to an addition:

$$\log|X(\omega)| = \log|E(\omega)| + \log|H(\omega)|$$

Following to this, the application of a Fourier transform to the logarithm of the magnitude spectrum leads to the extraction of the frequency distribution of the fluctuations in the curve of the spectrum c. This distribution is referred to as the **cepstrum**:

$$c = F^{-1}\big(\log|X(\omega)|\big) = F^{-1}\big(\log|S(\omega)|\big) + F^{-1}\big(\log|H(\omega)|\big)$$

Under the reasonable assumption that the source spectrum has only rapid fluctuations, its contribution to c will be concentrated in tis higher regions, while the contribution of H will be the slow fluctuations in the spectrum of X, and will therefore be concentrated only in the lower part of c. Thus, separation of the two components becomes trivial: Only the first p of the cepstral coefficients are kept, where p is called the **order** of the cepstrum*. This computation can been seen in Figure 2.4.

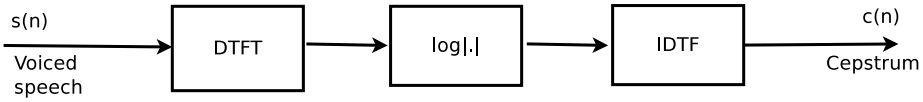


Figure 2.4: Computation of the Cepstrum.

To finally obtain the spectral envelope from the cepstral coefficients, the frequencies f_i are defined at which the values of the envelope is to be obtained. Usually n equidistant frequencies are calculated up to the Nyquist frequency $f_s/2$:

$$f_i = i\frac{f_s/2}{n}, i = 1, ..., n$$

Then, after passing to angular frequencies:

$$\omega_i = f_i\frac{2\pi}{f_s}$$

the envelope value u_i for frequency f_i is

$$u_i = \exp\Big(\sum_{j=1}^{p} c_j \cos(j\omega_i)\Big) \tag{2.23}$$

so equation 2.23 becomes

$$\mathbf{u} = \exp(\mathbf{Mc})$$

* The cepstrum methob belongs to the class of **homomorphic deconvolution methods**

where

$$\mathbf{M} = \begin{bmatrix} 1 & \cos(2\pi f_1) & \cos(2\pi f_1 2) & \cdots & \cos(2\pi f_1 p) \\ \vdots & \vdots & & \vdots & \\ 1 & \cos(2\pi f_n) & \cos(2\pi f_n 2) & \cdots & \cos(2\pi f_n p) \end{bmatrix}. \tag{2.24}$$

Disadvantages of cepstrum spectral envelope

Unfortunately, low pass filtering create an envelope following the mean of the spectrum and not as desired the contour of the spectral peaks. Another disadvantage, similar to LP, happens when high-pitched harmonic are analyzed, where in two neighboring spectral peaks the estimated curve will follow down to the residual noise level in the gap between partials.

2.4 Discrete Cepstrum Spectral Envelope

The method presented in the previous displays a number of disadvantages. For example adequate number of cepstrum is necessary in order to generate accurate envelope. Moreover when the partials are located far apart, the envelope estimated employing cepstrum descends down in to the space. The same limitation is presented in the **LP** method and is dealt by **DAP** method by accounting for the harmonics for the minimization of the error criterion. To overcome the same limitation we can use the discrete cepstrum method [19] [18], which also overcomes the problem by considering the harmonics, however for a least squares error criterion.

Cepstrum is computed using the spectral representation of the signal with points spaced equally, on the frequency axis. On the other hand the method of discrete cepstrum is computed by distinct points in the frequency-amplitude plane.

The latter is more preferable as some spectral peaks of a sound are not required to be regularly spaced in frequency in a sinusoidal model.

Given a set of L values of harmonic amplitudes a_k measured at the normalized harmonic frequencies of the fundamental frequency f_0, $f_k = k\frac{f_0}{F_s}$ where F_s is the sampling frequency, the log-amplitude envelope $A_c(f)$ can be evaluated by the real cepstrum parameters c_i:

$$A_c(f) = c_0 + 2\sum_{i=1}^{p} cos(2\pi f i) \tag{2.25}$$

where $\mathbf{c} = [c_0, \cdots, c_p]^T$ are the real cepstrum coefficients, and p is the order of the cepstrum.

The optimum discrete cepstrum coefficients are calculated by minimizing the squared error in the log-amplitude domain as shown below:

$$\epsilon = \sum_{k=1}^{L} ||20 \log_{10} a_k - A_c(f_k)||^2, \tag{2.26}$$

The least-squares solution is

$$\mathbf{c} = \left(\mathbf{M}^T\mathbf{M}\right)^{-1}\mathbf{M}^T\mathbf{a} \tag{2.27}$$

where $\mathbf{M}$ is defined as:

$$\mathbf{M} = \begin{bmatrix} 1 & \cos(2\pi f_1) & \cos(2\pi f_1 2) & \cdots & \cos(2\pi f_1 p) \\ \vdots & \vdots & & \vdots & \\ 1 & \cos(2\pi f_n) & \cos(2\pi f_n 2) & \cdots & \cos(2\pi f_n p) \end{bmatrix}. \tag{2.28}$$

Note the difference of $\mathbf{M}$ in (2.24) with $\mathbf{M}$ in (2.28). In the first case f_i are points spaced regularly, $f_i = i\frac{Fs}{2n}$, in the frequency axis. While in the second case $f_i = i\frac{f_0}{F_s}$.

A problem with the solution given by (2.27) is that the matrix $\mathbf{M^T M}$ is often poorly conditioned, meaning that small differences in data can produce large differences in the result. Hence, the estimated spectral envelope may fall far from the true one. These problems frequently appear when:

1. The are large frequency regions with no frequency point specified in them.
2. A number of frequency points are closely located in frequency, but differ substantially in magnitude.
3. The number of cepstrum coefficients reaches the number of frequency points.

The above issues can be solved using a regularization technique [29] that achieves an improved spectral envelope estimation by the regularized discrete cepstrum coefficients.

2.5 True Envelope Estimator

True envelope (**TE**) is a method that was developed in 1979 in Japan [17]. Unluckily, the method failed to gain recognition among researchers due to its introductory paper being published in Japanese. It is a cepstrum-based estimator that employs iteratively cepstral smoothing, aiming to link the prominent peaks of the log spectrum with a smooth and steady curve. The problem of spectral envelope estimation is viewed as a band-limited extrapolation problem. The well known Papoulis-Gerchberg algorithm [30] [15] [8] for restoring lost samples is used. In the following section a shortly presentation of Papoulis-Gerchberg algorithm is illustrated.

2.5.1 Papoulis-Gerchberg Algorithm

The formal Papoulis-Gerchberg algorithm, for a band-limited signal $s(n)$ with known values at $n \in A$, performs the following steps:

1. $f(n)$ initialization:
$$f(n) = \begin{cases} s(n) & \text{if } n \in A \\ 0 & \text{if } n \notin A \end{cases} \tag{2.29}$$

2. Calculate the Fourier transform of $f(n)$:
$$F(k) = \frac{1}{N} \sum_{n=0}^{N-1} f(n) e^{-j2\pi kn/N} \tag{2.30}$$

3. Use the band-limited property:
$$F(k) = F(k)P_\sigma(k),\ P_\sigma(k) = \begin{cases} 1 & |k| \leq \sigma \\ 0 & |k| \geq \sigma \end{cases} \tag{2.31}$$
where σ is the bandwidth of the original signal, $s(n)$.

4. Perform IDFT:
$$f_\sigma(n) = \sum_{k=0}^{N-1} F(k) e^{j2\pi kn/N} \tag{2.32}$$

5. Update $f(n)$ and an estimation of lost samples retrieval is computed:
$$f(n) = \begin{cases} s(n) & \text{if } n \in A \\ f_\sigma(n) & \text{if } n \notin A \end{cases} \tag{2.33}$$

Repeat steps (2)-(5) until convergence.

It is easy to prove [15] that the Mean Square Error

$$MSE = \frac{1}{N} \sum_{n=0}^{N-1} |f(n) - s(n)|$$

is reduced in each iteration of Papoulis-Gerchberg algorithm, but in [8] is indicated that there is a class of problems, like low-pass signals with contiguous set of nonzero harmonics, where convergence is very slow. However the algorithms conversion rate usually is slow.

Despite that, the conversion rate of the algorithm is generally low. Efforts for a more effective approach of the algorithm have been made by [15], where it transforms the initial signal to a new one, matching the border conditions in a way to approximate a band-limited signal.

In this section it was presented a useful algorithm for restoring lost samples. The following section will focus on showing that by using some properties of the algorithm we can have an efficient estimation of spectral envelope known as **True Envelope**.

2.5.2 True Envelope estimation

In all spectrum envelope estimations it is desirable to link the outstanding peaks in a smooth way. In this method, the knowledge of Papoulis-Gerchberg algorithm, presented in the previous section, provides a tool for a smooth curve passing through the prominent peaks of the amplitude spectrum.

It is known that the prominent spectral peaks contain the information about the spectral envelope. Thus, having a sub-sampled spectral envelope, we can reach an approximation by the Papoulis-Gerchberg algorithm, only in the case the spectral envelope is band limited.

TE is a cepstral-based method, thus for an input signal frame $s(n)$ the log-spectrum is computed:

$$A(k) = log(|S(k)|) \tag{2.34}$$

where $S(k)$ is the K-point DFT of the signal frame, $s(n)$.

The algorithm for the TE estimation follows the next steps:

1. $C(k)$ initialization:
$$C(k) = A(k) \tag{2.35}$$

2. Calculate the Fourier transform of $C(k)$:
$$c(i) = \frac{1}{K}\sum_{n=0}^{K-1} C(k)e^{-j2\pi ik/N} \tag{2.36}$$
where c is the discrete cepstrum coefficient introduced in section **2.4**.

3. Use the band-limited property:
$$c(i) = c(i)p_\sigma(i),\ p_\sigma(i) = \begin{cases} 1 & |i| \leq \sigma \\ 0 & |i| \geq \sigma \end{cases} \tag{2.37}$$
where σ is the bandwidth of the original signal, $C(k)$.

4. Perform IDFT:
$$C_\sigma(k) = \sum_{i=0}^{I-1} c(i)e^{j2\pi ik/N} \tag{2.38}$$

5. Update $C(k)$:
$$C(k) = \max(A(k), C_\sigma(k)) \tag{2.39}$$

6. Go back to step (2) until:
$$|er - er_{old}| < \epsilon \tag{2.40}$$

where ϵ is sufficiently small quantity compare to the desired accuracy, er is the mean square error in the desired points:

$$er = \frac{1}{M} \sum_{i=0}^{M-1} |C(k_i) - A(k_i)|^2,$$

of the current estimation, k_i is a desired point and M the total number of these points, while er_{old} is the mean square error of the previous iteration.

At this point it is important to highlight a number of remarks for the algorithm presented above. In the first step the log Spectrum is assigned to variable C, is reminded the fact that the perceived loudness of human listener is approximately logarithmic with the signal amplitude, in order to perform cepstral smoothing in the following steps, (2)-(4). Second step provides us discrete cepstrum coefficients, for this reason TE is a cepstral-based method for the spectral envelope representation.

In the third step it is important to pay attention to the selection of the appropriate cut-off frequency during the estimation of TE. For harmonic signals with fundamental frequency f_0 and sample rate F_s it is easy to show that the optimal cut-off frequency (**Nyquist frequency** [23]) is:

$$\sigma = \frac{F_s}{2F_0},$$

Consequently the optimal cepstral order p, for harmonic sounds is $\frac{F_s}{2F_0}$, while in general the optimal order and cut-off frequency is is

$$\sigma = p = \frac{Fs}{2\delta_F}$$

where δ_F is the largest distance between two neighboring spectral peaks.

The execution of steps four and five results in filling the valleys between two neighboring spectral peaks with the mean spectrum. Additionally, it increases the estimated envelope that tends to cover the spectral peaks.

The final step (6), is a termination criterion which determines the accuracy of estimated envelope, producing a trade-off between accuracy and the number of iterations. More specifically, as the number of iterations increases the estimated envelope will also increase until it covers all the spectral peaks.

Following the above steps we achieve a good estimation of the spectral envelope (Figure 2.5), from which we can obviously extract directly the discrete cepstrum coefficients, since this is a cepstrum-based method. However, having a good representation of the spectral envelope it would be possible to estimate other types of coefficients too, such as the LPC ones. The next section describes how to estimate the LP coefficients, called TE-LPC.

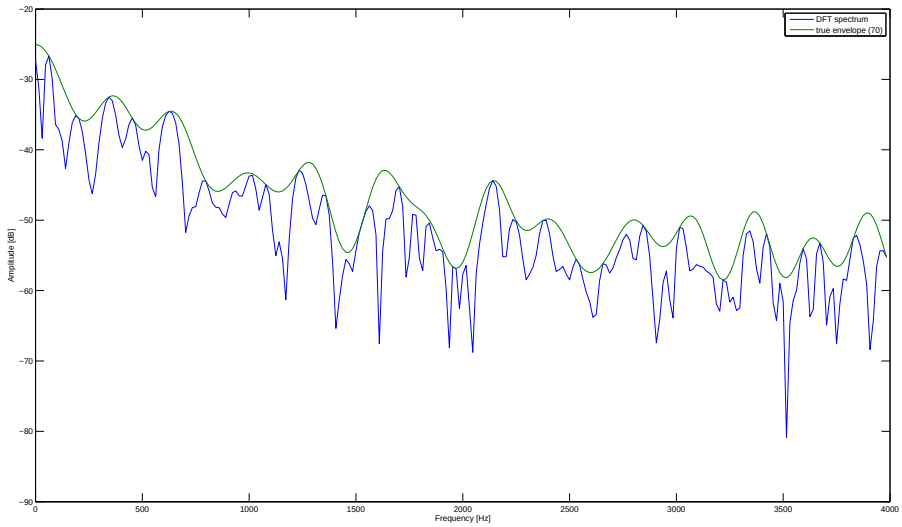


Figure 2.5: The TE spectral envelope estimation for a male speaker.

2.5.3 True envelope LPC modeling

The type of the limitations shown in the spectral modeling of voiced speech by the standard LPC technique shows that LPC performance is restricted by the procedure itself and is also influenced by the local characteristics of the signal.

More particularly, in LPC method it is assumed that the autocorrelation function r of the measured signal equals the autocorrelation function r_h of the transfer function. Nevertheless, as we have seen in harmonic signals, r is a severally aliased version of r_h. In this case, the fitting of the spectral envelope is not as close as possible, but it is rather closer to the original spectrum. To deal with these shortcomings the proposition of [12] is followed.

Initially it is used an optimal band limited interpolation to interpolate the spectrum as described in chapter 2.5.2. Subsequently is imposed a high order all-pole model such that the Line Spectral Frequency (LSF) representation of the spectral envelope is still attainable. Further to this improvement it is demonstrated [40] that the high order LPC model will gain a superior representation of the narrow formants of the spectrum, which are generally too broad following the band limited interpolation.

Having a good estimation of the spectral envelope H_{TE} we can compute the autocorrelation function r_h, h is the impulse response of the transfer function H, directly from the estimated spectral envelope. The Wiener–Khinchin theorem relates the autocorrelation function to the power spectral density via the Fourier

transform:

$$r_h(i) = \frac{1}{K}\sum_{k=1}^{K} H_{TE}(\omega_k)^2 e^{j\omega_k i}, \tag{2.41}$$

where H_{TE} is the estimated K-point spectral envelope. Calculating r_h from (2.41), LP can be computed from the well-known Yule-Walker equations:

$$R = \begin{bmatrix} r_h(0) & r_h(1) & \cdots & r_h(p-1) \\ r_h(1) & r_h(0) & \cdots & r_h(p-2) \\ \vdots & \vdots & \ddots & \vdots \\ r_h(p-1) & r_h(p-2) & \cdots & r_h(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} r_h(1) \\ r_h(2) \\ \vdots \\ r_h(p) \end{bmatrix} \tag{2.42}$$

and the **TE-LPC** coefficients are obtained from:

$$\mathbf{a} = \mathbf{R^{-1}r_h} \tag{2.43}$$

It should be noted that the above estimation does not minimize the expected value of the standard squared prediction error as in the conventional LP estimator (equation 2.7). Nevertheless, this estimation yields (generates) a curve which is closer to the spectral envelope. In Picture 2.6 below is depicted the comparison between the LP and the TE-LP method.

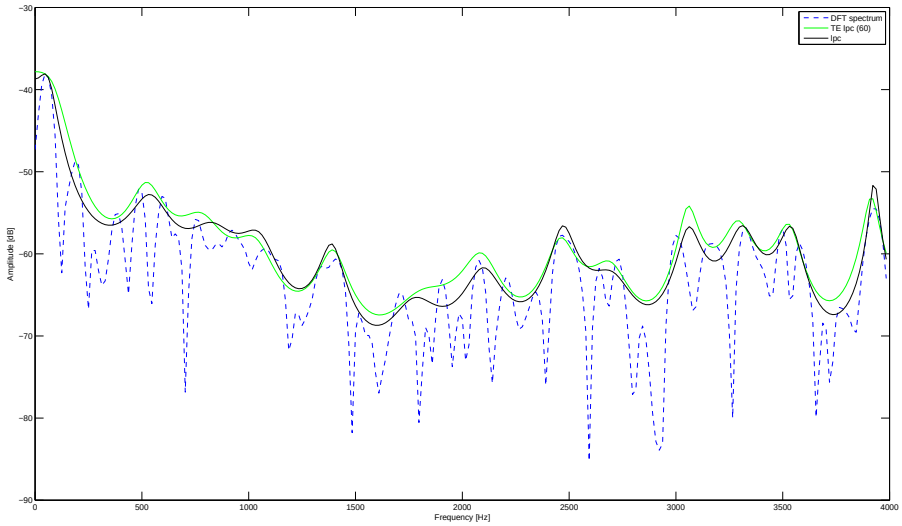


Figure 2.6: Example of LPC and TE-LPC spectral fitting (model order =60).

However this method can be improved with respect to execution complexity, by using a proposal of Robel and Rodet in [1] and by taking advantage of zeros appearing in Fourier transform for a pruned fast fourier transform (FFT) implementation [14] [9].

Chapter 3

Speaker Conversion and Interpolation

3.1 Spectral Matching

3.1.1 Dynamic Time Warping

Dynamic time warping (DTW) is a time alignment algorithm launched in the 60's [19] [35]. In the past it has been extensively used in speech recognitions, but in the present its application has been extended in various areas (e.g. online signature matching, gesture recognition, computer animation, surveillance, etc). It is an algorithm popular amongst researchers for being efficient as the time-series similarity measure, that minimizes the effects of shifting and distortion in time. This is accomplished by warping the time axis of time series in order to detect shapes with different phases (figure 3.1). Given two time series $A = \{a_1, a_2, \cdots, a_N\}$, and $B = \{b_1, b_2, \cdots, b_M\}$, with $N, M \in \mathbb{N}$, DTW will find the optimal solution for the sequence alignment in the $O(MN)$ time.

Let sequence A, B take values form the feature space Φ. In order to compare them a local distance function, often called cost function, is needed to define the measure function:

$$d : \Phi \times \Phi \to \mathbb{R} \geqslant 0. \tag{3.1}$$

Intuitively, d has a small value when sequences are similar and a large one if they are different. The optimal alignment of the sequences becomes the task of arranging all sequence points by minimizing the cost function.

The algorithm starts with the distance matrix $D \in \mathbb{R}^{N \times M}$ representing all pairwise distances between A and B. This matrix is labeled the local cost matrix

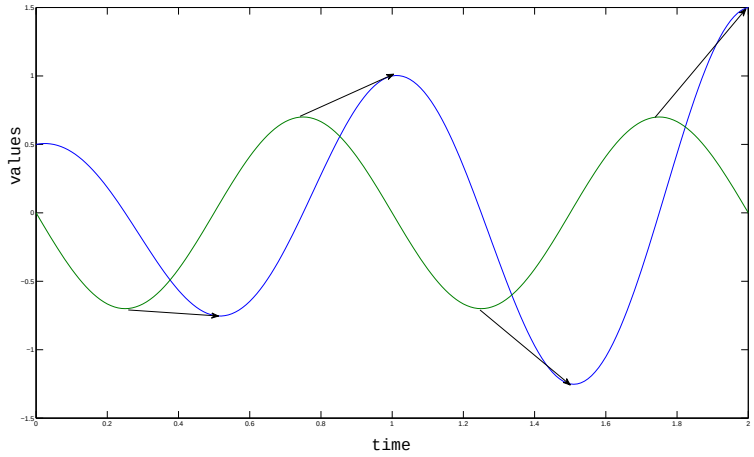


Figure 3.1: Raw time series, arrows show the desirable points of alignment.

for the alignment of two sequences A and B:

$$D \in \mathbb{R}^{N \times M} : d_{i,j} = \|a_i - b_j\|, i \in \{1, 2, \cdots, N\}, j \in \{1, 2, \cdots, M\}. \tag{3.2}$$

After the computation of the distance matrix, the algorithm finds the alignment path which runs through the low-cost areas, "valeys" on the cost matrix Figure 3.2. This alignment path defines the correspondence of an element $a_i \in A$ to $b_i \in B$ ($a_1 \to b_1$), with the restriction to assign first and last elements $a_1 \to b_1$ and $a_n \to b_n$ respectively, known as boundary conditions.

Formally speaking, the alignment path built by DTW is a sequence of points $p = p_1, p_2, \cdots, p_K$ with $p_k = (i_k, j_k) \in [1, N] \times [1, M] \bigcap \mathbb{N}$ for $k = 1, 2, \cdots, K$ which must satisfy the following criteria:

1. **Boundary condition:** $p_1 = (1, 1)$ and $p_k = (N, M)$. The starting and ending points of the warping path must be the first and the last points of aligned sequences.

2. **Monotonicity:** The path should be *monotonic*. This means that:

$$i_{k-1} \leq i_k \text{ and } j_{k-1} \leq j_k \tag{3.3}$$

3. **Global Path Constraints:** Is the amount of the allowable compression or expansion in time axis from the warping function:

$$| j_k - i_k | \leq W, \tag{3.4}$$

where W is called the "window width".

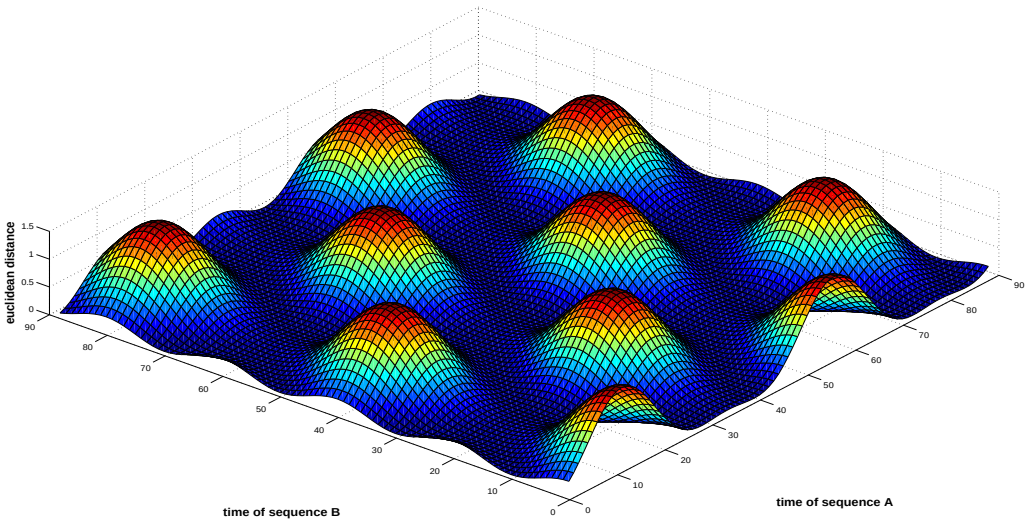


Figure 3.2: Cost matrix containing all pairwise distances.

4. **Local Path Constraints**: Given a node $p_k = (i_k, j_k)$ on the cost matrix, the legal set of predecessor nodes is specified by the Local Path Constraints. Several types of local path constraints in term of predecessor nodes have been studied by various researchers. for example:

$$p_{k-1} = \begin{cases} (i_k - 1, j_k) \\ (i_k, j_k - 1) \\ (i_k - 1, j_k - 1) \end{cases} \tag{3.5}$$

constraints the predecessor node in the overall path to take one value only of equation (3.5).

The **cost function** associated with a warping path computed with respect to the local cost matrix will be:

$$d_p = \sum_{l=1}^{L} d(i_l, j_l) \tag{3.6}$$

where $i_l \subseteq \{1, 2, \cdots, N\}$ and $j_l \subseteq \{1, 2, \cdots, M\}$.

The warping path with the minimum cost function is called **optimal warping path** and is denoted by P^*. To find the optimal warping path we calculate all the possible warping paths between time series A and B. However the computational cost grows with exponential order. DTW overcomes this difficulty via Dynamic Programming resulting in an algorithm with $O(MN)$ complexity.

In speech signal processing the utility of DTW as described above is straight forward. The position of time series A, B can be taken by cepstrum coefficients,

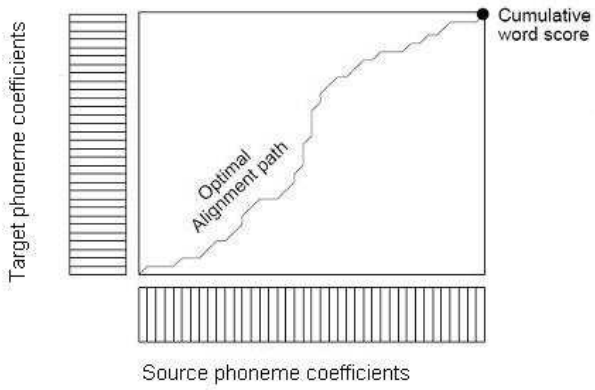


Figure 3.3: Coefficient alignment in a phoneme between two speakers.

mel-cepstrum or LPC coefficients. Note that for LPC the distance measure changes from Euclidean to Itakura-Saito. Figure 3.3 illustrates an example coefficient alignment of two speakers in the same phoneme.

3.1.2 Dynamic Frequency Warping

In DTW technique two different time sequences of the same shape with different phase are aligned. It is obvious that the DTW provides reliable correspondences for sequences having similar shape and different phase. In speech signals DTW achieves an accurate comparison when the vocal tract lengths are close, male to male or female to female. When speakers of different gender are aligned with DTW, the comparison is not accurate because the frequency characteristics of the features are not taken into account. We will attempt to resolve this problem using an alternative method called Dynamic Frequency Warping (DFW), which is used to normalize the frequency scale [10] [16].

DFW aims at getting an optimal non-linear warping function of the frequency axis to normalize the signal characteristics in the frequency domain. For speech signal DFW is more closely related the acoustic theory of speech production, because it uses frequency specific properties of the speech signals. Therefore, in this subsection will be dedicated in describing this method specifically for speech signals.

To begin with, we will briefly explain the computation of the DFW path on a pair of log-magnitude spectrum. Next, we will explain how to find a transformation for a given acoustic class.

Let $S^r(k)$ and $S^t(k)$ denote the reference log-amplitude spectrum and the target log-amplitude spectrum respectively, with K is the bins of Fourier transform. The algorithm starts with the distance matrix $D \in \mathbb{R}^{N \times M}$ representing all pairwise distances between $S^r(.)$ and $S^t(.)$. This matrix is labeled the local cost matrix for the alignment of two spectrum S^r and S^t:

$$D \in \mathbb{R}^{K \times K} : \ d(i,j) =\mid S^r(i) - S^t(j) \mid, \text{ for } i,j \in \{1,2,\cdots,K\}. \tag{3.7}$$

Each element of cost matrix can be considered as a two-dimensional matching space. The alignment path built by DFW is a sequence of points $p = p_1, p_2, \cdots, p_K$ with $p_k = (i_k, j_k)$ for $i_k, j_k, k = 1, 2, \cdots, K$. The algorithm for the case of a pair of log-amplitude cepstrum is the same to the DTW algorithm but with some different constraints:

1. **Boundary condition:** $p_1 = (1,1)$ and $p_k = (K, M)$, $M \leq K$. Only the starting points of the warping path must be the first points of aligned sequences.

2. **Monotonicity:** The path should be *monotonic*. This means that:

$$i_{k-1} \leq i_k \text{ and } j_{k-1} \leq j_k \tag{3.8}$$

3. **Global Path Constraints:** Is the amount of the allowable compression or expansion in time axis from the warping function:

$$\mid j_k - i_k \mid \leq W, \tag{3.9}$$

and W is called the "window width". Here we can use stricter slope constraints for low frequencies than for high ones, becaues in vowel formant frequencies for different speakers the variability of the two formants is low.

4. **Local Path Constraints**: Given a point $p_k = (i_k, j_k)$ on the cost matrix the legal set of predecessor nodes is specified by the Local Path Constraints, a legal constraint that it often used is:

$$p_k = \begin{cases} (i_k - 1, j_k) \\ (i_k, j_k - 1) \\ (i_k - 1, j_k - 1) \end{cases} \tag{3.10}$$

As in DTW, the optimum path is:

$$P^* = \arg\min_p \sum_{l=1}^{K} d(i_l, j_l) \tag{3.11}$$

We have shown how to apply DFW for a pair of log-amplitude spectrum. Next we will focus in the procedure which find DFW for an utterance pronounced by two speakers.

Initially we extract a set of coefficients representing the sequence for each speaker. These can be cepstrum coefficients, LPC etc. Subsequently, we perform DTW to provide the correspondences of the coefficients. Each pair of corresponding coefficients leads to a respective pair of spectral envelopes. The computation of the optimal DFW path takes place for the whole utterance. Each pair of reference and target spectral envelope will be accounted for in the frequency normalized distance. Consequently, DFW procedure will find the path P^* that minimizes the new global distance:

$$D = \min_{p} \sum_{n=1}^{N} \sum_{k=1}^{K} d_n(i_k, j_k) \tag{3.12}$$

where N is the number of analysis frames and d_n calculates the distance in the frame n.

Note that the optimum path P^* is global, meaning that in all spectral envelope pairs the warping frequency function is the same.

With this method we achieve a normalization of vocal tract length for the two speakers. Resulting a more accurate comparison of the extracted features, especially when DFW applied in voices of different gender.

3.1.3 Segment-wise time warping

As discussed in subsection 3.1.1, the technique of searching for similar patterns among time series data can be applied to many different areas. Time warping distance is a similarity measure that is derived from and related to the area of speech recognition. When the time warping technique was introduced in the area of time series searching, it was applied using DTW [6]. With DTW some portion of the time is dynamically warped to minimize the effects of distortion in the time domain.

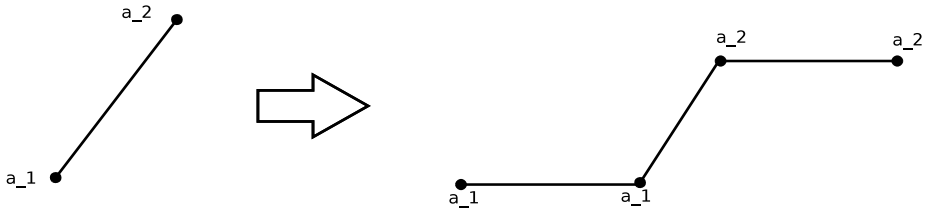


Figure 3.4: Distortion of time series using point-wise time warping.

For example, given two time series $A = \{a_1, a_2\}$, $B = \{b_1, b_2, b_3, b_4\}$, a_1 can be matched with $\{b_1, b_2\}$ and a_2 with $\{b_3, b_4\}$. The warping in this example means repetition, i.e. a_1 and a_2 are repeated two times, figure 3.4 . This time distortion is called point-wise warping (PTW).

An alternative method was proposed called segment-wise time warping (STW) [25]. This time warping method based on segments. For the previous example the stretched segment will be:

$$s_i = a1 + \frac{i}{4+1}(a2 - a1),\ 0 \leq i \leq 4+1, \tag{3.13}$$

and is depicted in figure 3.5.

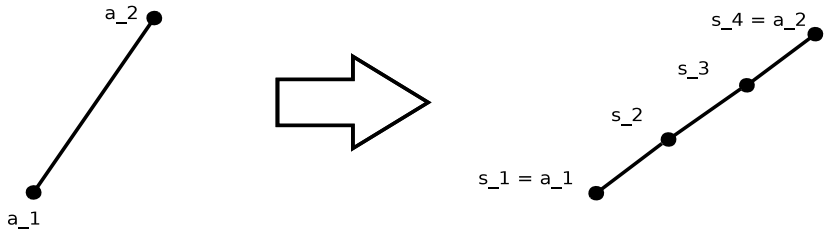


Figure 3.5: Distortion of time series using segment-wise time warping, s_i is given from equation 3.13.

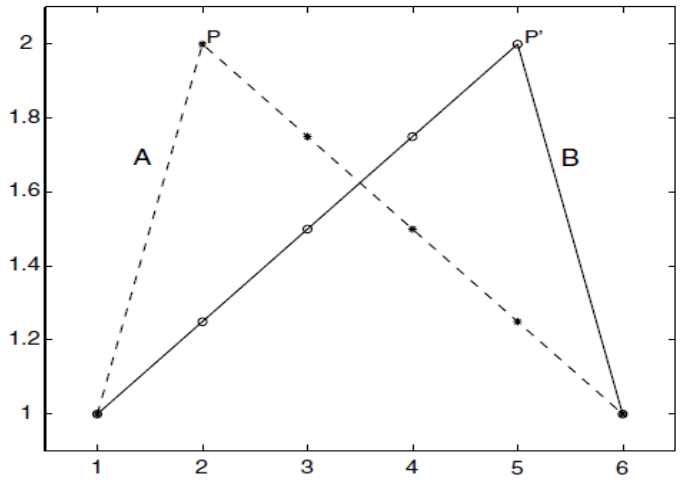


Figure 3.6: Time series A and B (taken by Zhou and Wong [25])

.

Zhou and Wong [25] give an example of two sequences $A = (1, 2, 1.75, 1.5, 1.25, 1)$ and $B = (1, 1.25, 1.5, 1.75, 2, 1)$ Figure 3.6. The correspondences between A and B using DTW is shown in Figure 3.7I and the time warped sequences in 3.7II. Similarly the correspondences between A and B using STW is shown in Figure 3.7I and the time warped sequences in 3.7II.

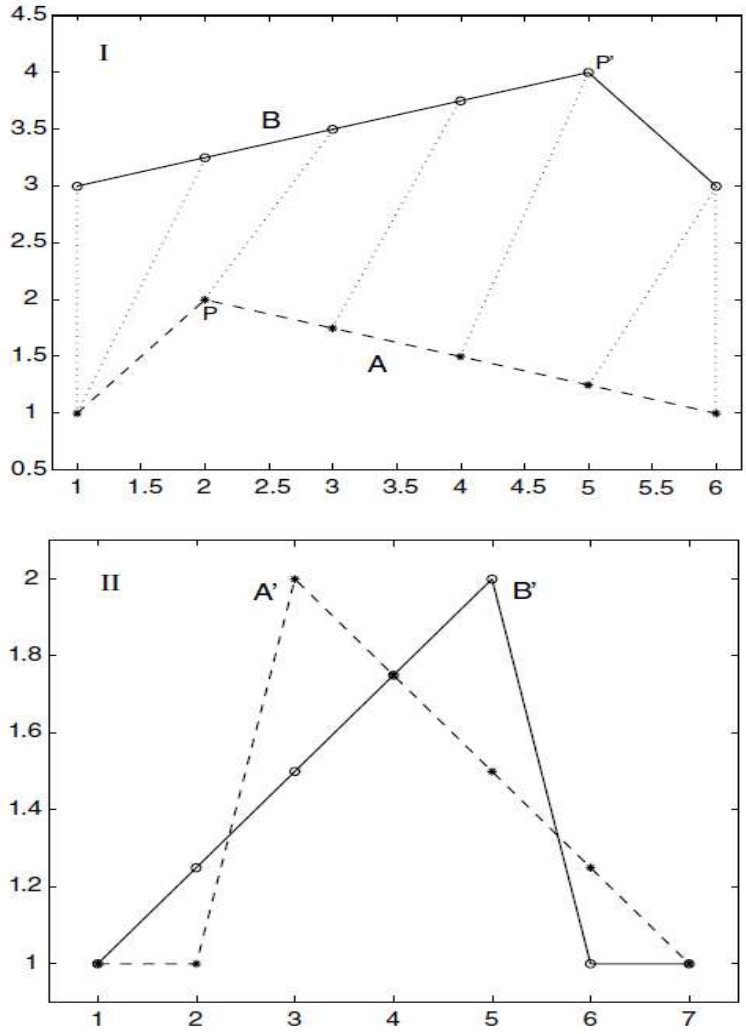


Figure 3.7: (I) The correspondences between A and B using point-wise warping, (II) the time warped A and B (taken by Zhou and Wong [25]).

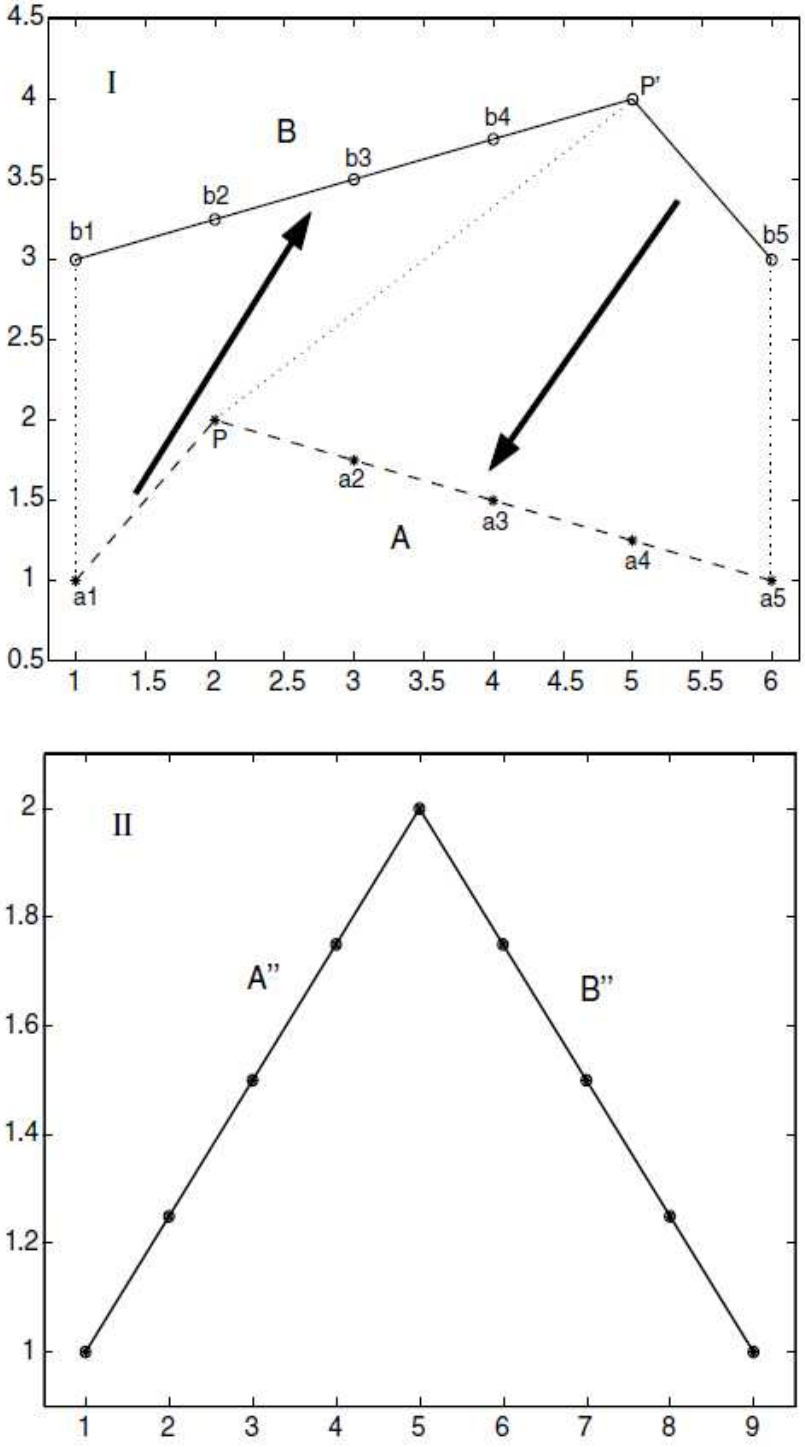


Figure 3.8: (I) The correspondences between A and B using segment-wise warping, (II) the time warped A and B (taken by Zhou and Wong [25]).

We have given a general idea of STW method, however it is not in the purpose of this paper to present it in more detail.

Piecewise linear time warping for speech signals

The time series warping distance was derived from the time warping in speech processing. Now we will use the segment-wise time warping (STW) that was created for time series, to obtain a similar method for speech signals.

Given two time series the reference $A = \{a_1, a_2, \cdots, a_N\}$, and the target $B = \{b_1, b_2, \cdots, b_M\}$, with $N, M \in \mathbb{N}$, we will find the optimal solution for a piecewise linear time warping of the target axis.

The algorithm begins with the transformation of the reference axis from $\{1, 2, \cdots, N\}$ to $[0, 1]$, dividing interval $[0, 1]$ to three equal segments:$(0, \frac{1}{3})$, $(\frac{1}{3}, \frac{2}{3})$, $(\frac{2}{3}, 1)$, as depicted in Figure 3.9a.

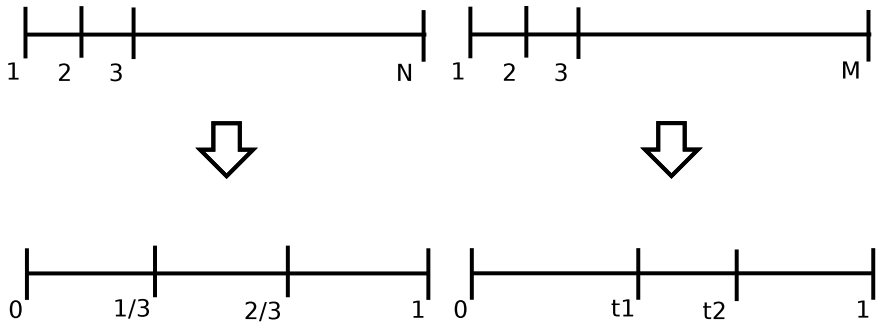
Following, is the transformation of the target axis from $\{1, 2, \cdots, N\}$ to $[0, 1]$, dividing the interval $[0, 1]$ to three segments:$(0, t_1)$, (t_1, t_2), $(t_2, 1)$ as depicted in Figure 3.9b. Segment wise warping is then performed in each interval:

$$(0, t_1) \rightarrow \left(0, \frac{1}{3}\right) \tag{3.14}$$

$$(t_1, t_2) \rightarrow \left(\frac{1}{3}, \frac{2}{3}\right) \tag{3.15}$$

$$(t_2, 1) \rightarrow \left(\frac{2}{3}, 1\right) \tag{3.16}$$

In this way a warped axis for the whole sequence is obtained. We will symbolize this warping function $g(.)$, figure 3.10.



(a) Reference axis normalization and segmentation in three equal segments.

(b) Target axis normalization segmentation in three segments, $0 < t_1 < t_2 < 1$.

Figure 3.9: Transformation and segmentation of reference (a) and target (b) axis.

Let sequence A, B take values from the feature space Φ. The local distance

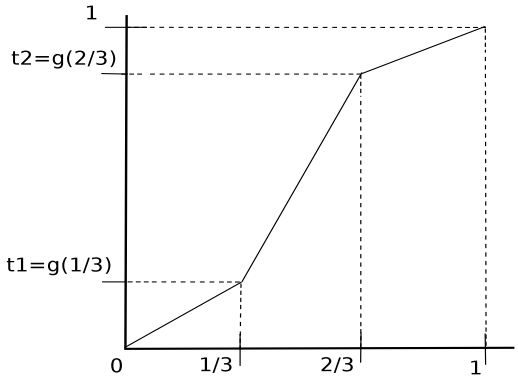


Figure 3.10: Piecewise linear time warping function.

function is used in order to define the measure function:

$$d(i,j) = \|a_i - b_j\|, \quad i \in \{1,2,\cdots,N\}, \quad j \in \{1,2,\cdots,M\} \qquad (3.17)$$
$$d : \Phi \times \Phi \to \mathbb{R} \geqslant 0.$$

then the warping error will be:

$$\epsilon = \sum_{k=1}^{N} d(k, g(k)) \qquad (3.18)$$

The optimal t_1, t_2 will be obtained by the minimization of ϵ:

$$\{t_1^{opt}, t_2^{opt}\} = \arg \min_{t_1 \in (0,1)} \left\{ \arg \min_{t_2 \in (t_1,1)} \epsilon \right\} \qquad (3.19)$$

To clarify things an example is given for a phoneme represented by cepstrum coefficients. Let the reference axis $(0, 1/3)$ contain the cepstrum coefficient vectors $\{\vec{a}_1, \vec{a}_2, \vec{a}_3, \vec{a}_4\}$, and the target axis $(0, t_1)$ contain the cepstrum coefficient vectors $\{\vec{b}_1, \vec{b}_2\}$, $\vec{a}_i, \vec{b}_i \in \mathbb{R}^p$ (p is the order of cepstrum coefficients). Then the target axis is warped from $(0, t_1)$ to $(0, 1/3)$ and the new target cepstrum vectors are obtained $\{\vec{s}_1, \vec{s}_2, \vec{s}_3, \vec{s}_4\}$ from the relation:

$$\vec{s}_i = (i-1)\frac{\vec{b}_2 - \vec{b}_1}{3} + \vec{b}_1, \quad 1 \leq i \leq 4. \qquad (3.20)$$

The number of the warped vectors is the same with the number of the reference vectors in order to compare them. Thus the local error for the first pair of segments, $(0, 1/3)$ with $(0, t_1)$, will be:

$$\epsilon_{local} = \sum_{k=1}^{4} \|\vec{a}_k - \vec{s}_k\|_2 \qquad (3.21)$$

The same procedure is performed for the rest two pairs, $(1/3, 2/3)$ with (t_1, t_2) and $(2/3, 1)$ with $(t_2, 1)$, and the total error ϵ is calculated as the sum of local errors. Then we choose the t_1 and t_2 that minimize the total error ϵ and the optimum warping path is obtained, which we remind that is a piecewise linear path, as in figure 3.10.

Usually when we compare the phonemes of two speakers we find that their time duration differs. When using DTW the consequence of this is the repetition of some moments during warping. With the use of the above method we avoid these repetitions by extracting a new set for which the number of features equals with that of the reference set. This method can also be used as an alternative to DFW [13]. The only difference is the warping of the local path. The results were improved compare to both DTW and DFW.

3.2 Lattice filtering

The lattice filter is extensively used in digital speech processing and in the implementation of adaptive filters. It is a preferred form of realization over other **FIR** or **IIR** filter structures because in speech analysis and synthesis the small number of coefficients allows a large number of formants to modeled in real time [31][42][3].

3.2.1 The FIR lattice filter

Let $s(n)$ be the measured signal, which is an AR process. Consider the transfer function:

$$A^i(z) = 1 - \sum_{k=1}^{i} a_k^i z^{-k} \tag{3.22}$$

which is the ith-order prediction error filter, section 2.1.1. Taking the inverse z-transform we obtain:

$$e_i(n) = s(n) - \sum_{k=1}^{i} a_k^i s(n-k) = s(n) - \hat{s}(n) \tag{3.23}$$

where $e_i(n)$ is the forward prediction error sequence for the ith-order prediction error (inverse) filter. In the z-domain the forward prediction error is given by:

$$E_i(z) = A^i(z)S(z) \tag{3.24}$$

Consider now the backward prediction error sequence as:

$$b_i(n) = s(n-i) - \sum_{k=1}^{i} a_k^i s(n-i+k), \tag{3.25}$$

and b_i represents the difference between the $s(n-i)$, the value of the input function i samples ago, and a linear combination of the following i samples of $s(n-i)$. One way to think about $b_i(n)$ is what we would have obtained if we calculated $e_i(n)$ but with the input function presented in time-reversed order. The z-transform of $b_i(n)$ can be found as follow:

$$\begin{aligned} B_i(z) &= z^{-i}S(z) - \sum_{k=1}^{i} a_k^i z^{-i} S(z) \\ &= z^{-i}S(z)\big[1 - \sum_{k=1}^{i} a_k^i z^k\big] \\ &= z^{-i}A^i(z^{-1})S(z) \end{aligned} \tag{3.26}$$

With the above time and frequency expressions for the forward and backward prediction and using the third step of the Levinson algorithm, table 2.1 i.e., $a_j^i = a_j^{i-1} - k_i a_{i-j}^{i-1}$, $1 \le j \le i$, we can arrive at the following relations:

$$e_i(n) = e_{i-1}(n) - k_i b_{i-1}(n-1) \tag{3.27}$$

$$b_i(n) = b_{i-1}(n-1) - k_i e_{i-1}(n) \tag{3.28}$$

These equations can be depicted by a flow graph which we refer to as a *lattice filter* structure (Figure 3.11). The output of the pth stage of the lattice equals the forward prediction error for the Nth-order predictor, i.e, $e_N(n) = e(n)$ which equals the output of $A(z)$.

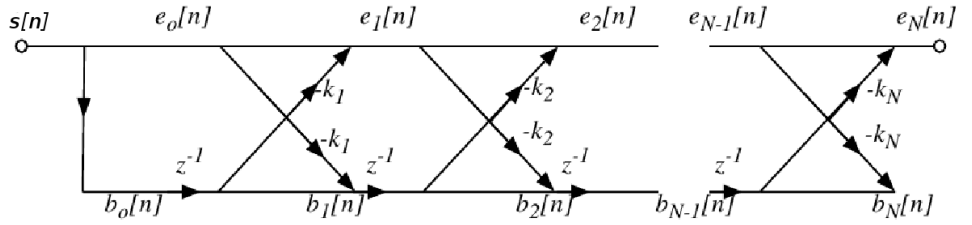


Figure 3.11: Flow graph of the lattice FIR filter.

3.2.2 The IIR lattice filter

As noted above, we have developed an all-zero lattice filter in the previous section with transfer function

$$A(z) = 1 - \sum_{k=1}^{N} a_k^N z^{-k} \tag{3.29}$$

In figure 3.11 the input is $s(n)$, measured signal, and the output is $e(n)$, the prediction error.. Keeping the same filter structure but interchanging the input

and output, the following transfer function is obtained

$$\frac{1}{A(z)} = \frac{1}{1 - \sum_{k=1}^{N} a_k^N z^{-k}} \tag{3.30}$$

which is an all-pole transfer function.

Recall that the original definitions of the stages if the FIR lattice filter are

$$e_i(n) = e_{i-1}(n) - k_i b_{i-1}(n-1), \tag{3.31}$$

$$b_i(n) = b_{i-1}(n-1) - k_i e_{i-1}(n), \tag{3.32}$$

and equation (3.31) can be rewritten as

$$e_{i-1}(n) = e_i(n) + k_i b_{i-1}(n-1). \tag{3.33}$$

From equations (3.32), (3.33) is obtained the lattice structure of figure (3.12).

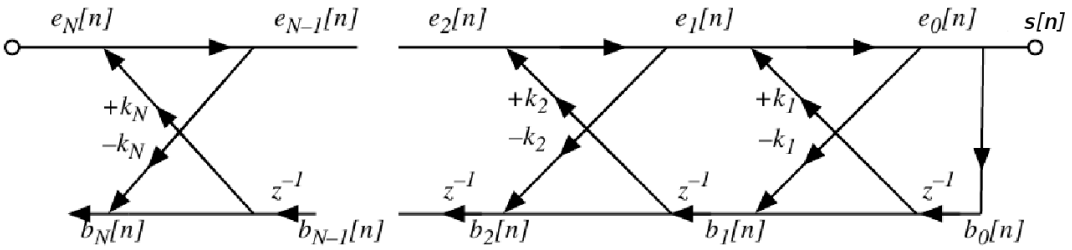


Figure 3.12: Flow graph of the lattice IIR filter.

Note that $e_N(n)$ is now the input and that $e_0(n)$ is the output. This filter have the transfer function:

$$H(z) = \frac{1}{\sum_{k=1}^{N} a_k^N z^{-k}} \tag{3.34}$$

where the LPC parameters a_i are related to the reflection coefficients k_i according to the usual Levinson-Durbin relationship. Since the filter is IIR with feedback loops, it does have the potential to be unstable. However, it is guaranteed to remain stable because

$$| k_i | < 1 \quad \text{for all } i. \tag{3.35}$$

3.2.3 Interpolated Lattice Filter

The use of reflection coefficients results in some advantages in lattice filter interpolation, because they can be interpolated without affecting the stability of the interpolated lattice filter.

In fact when two or more lattice filter are interpolated, each $k_i^{'}$ of the resulting filter is an interpolation $k_i^1, k_i^2, \cdots, k_i^M$ coefficients, where m of k_i^m indicate reflection coefficient of filter m.

For example, the linear interpolation of two FIR lattice filters H_1, H_2, will have the reflection coefficients:

$$k_i = ak_i^1 + (1-a)k_i^2, \quad a \in [0,1] \tag{3.36}$$

The same can performed in IIR latice filter. In the case of interpolation between two ARMA lattice filter, the numerators and denominators are interpolated as above.

3.3 Speech Conversion

Given an utterance voiced by two speakers, i.e. the source and the target speaker, we wish to convert the voice of the source speaker in a way that it sounds like the voice of the other speaker. Spectral envelope information plays an important role in voice individuality [22]. Extracting the excitation signal of the source speaker and replacing the spectral envelopes of the source speaker to the corresponding of the target speaker we expect the conversion of the voice. The steps that need to be taken in order to achieve the conversion of speech are the analysis, spectral matching and conversion.

3.3.1 Analysis.

In analysis step the estimation of spectral envelopes is performed in order to extract speaker voice characteristics. To smoothly represent the alternations of the spectral envelope, a hamming window with length 30ms with step 5ms is used. In voiced speech portions the True Envelope (TE) method is used, because it is an efficient method for spectral envelope estimation, compared with the existing methods (section 2.5) [33]. TE is based on fundamental frequency F_0 and sampling rate F_s. The optimal order is $F_s/(2F_0)$ (section 2.5.2), so pitch estimation is performed before spectral envelope extraction. The basis of the TE algorithm is the consideration of spectral envelope as band-limited signal with lost samples. The known samples lie on the spectral peaks and the restoring Papoulis-Gerchberg algorithm is applied. Missing signal sampling regularly results in spectral peak at 0Hz and possibly $F_s/2$ being generally missing so that for harmonic excitation with fundamental frequency F_0 the maximal frequency difference between the supporting peaks will be $\Delta_F = 2F_0$. Thus to estimate the values at the frequencies 0 and $F_s/2$ the true envelope method is performed with order $F_s/(4F_0)$ first. In unvoiced portions LPC is an efficient method for spectral envelope estimation and it is being used. The order selection is chosen to $F_s/1000$ [19]. Table 3.1 presents a briefly description of the analysis procedure. We must

note here that F_0 varies in different frames and for different speakers, thus a mean value of the two speakers f_0 is taken which gives an estimated spectral envelope close to the optimum.

step 1:	pitch estimation of $s(n)$
step 2:	**if voiced frame:**
step 2$_\mathbf{a}$	TE method with order $\frac{F_s}{4F_0}$
step 2$_\mathbf{b}$	envelope estimation on 0Hz and $\frac{F_s}{2}$Hz
step 2$_\mathbf{c}$	TE method with order $\frac{F_s}{2F_0}$
	else:
step 2$_{\mathbf{a}'}$	LPC method is used with order $\frac{F_s}{1000}$

Table 3.1: Spectral envelope estimation for input signal $s(n)$.

3.3.2 Spectral matching.

In time and frequency warping the cepstrum coefficients are used. Cepstrum coefficients are suitable perceptual features in warping measurements [44]. Also they provide a better spectral alignment compared with the line spectral frequencies (lsf), in the sense of speech naturalness. The extraction of cepstrum coefficients from TE is straightforward, as the DFT of the estimated spectral envelope. For LPC the estimated spectral envelope is calculated from the log-amplitude of transfer function

$$H(\omega) = \frac{1}{1 - \sum_{k=1}^{p} a_k e^{-j\omega k}}, \tag{3.37}$$

where p is the model order. The cepstrum coefficients are computed via the log-amplitude DFT of the envelope.

Parallel corpora are processed, thus each corpus, of the source and the target speaker, is consisted from the same phonemes with the same order. The cepstrum coefficients of the corresponding phonemes are aligned using piecewise linear time warping, which compared with DTW, is superior for a more efficient alignment. After a first estimation of time warping between source and target speaker, piecewise frequency warping is performed for the normalization of the target vocal tract. Then again time warping between the normalized target cepstrum and original source cepstrum is performed and so on, until the warped time axis converges. The iterative procedure is shown in table 3.2. From the process explained above the enhancement of time alignment for the parallel corpora alignment is expected compared with a single DTW or piecewise linear warping execution.

```
   /*time warping of the target time axis*/
1  t' = timeWarping(C_s, C_t)
2  for a fixed number of maximum iterations do
     /*frequency warping of the target spectral envelope*/
3    C'_t = FrequencyWarping((C_s, C_t), t')
4    t'_new = timeWarping(C_s, C'_t)
5    if ||t' − t'_new||_2 < ε do
6      break;
7    end if
8  end for
```

Table 3.2: spectral matching of source C_s and target C_t cepstrum.

3.3.3 Conversion.

To this point we have the spectral envelope of the source and of the target, as well as their correspondences in time. Our purpose is to compute the LPC coefficients, and from LPC we could obtain reflection coefficients, for each spectral envelope. LPC or reflection coefficients are used depending on what king of synthesis is performed. For instance, using the interpolated ARMA lattice filter the reflections coefficients are needed. In voiced frames the TE-LPC coefficients (equation 2.43) are extracted for each spectral envelope of the source and the target individual, while in unvoiced frames the LPC coefficients are known for both speakers. Frequently the LP model exhibit sharp peaks near pitch harmonics with large powers. A regularization technique is applied to obtain smoother contoured all-pole spectral envelopes. Given the source signal as input in the inverse filter results in the excitation signal as output. In the position of the source's spectral envelopes used for inverse filtering we place the respective envelopes of the target speaker. Then we filter the excitation signal with the spectral envelopes of the target represented by the LPC coefficients. Both inverse filtering and filtering can be conducted with the use of the ARMA interpolated lattice filter, or with the use of a MA and AR filter with the overlap and add (OLA) method. The speech conversion procedure is depicted in figure 3.13.

3.4 Speaker Interpolation

Given the voices of two or more speakers, the source speakers, the task of speaker interpolation (IS) is to interpolate the voice characteristics of the given voices, in order to obtain an intermediate speaker. The new speaker's voice can be calculated as the linear combination between the voice characteristics of the source speakers. To be more specific, the voice characteristics represent the speaker

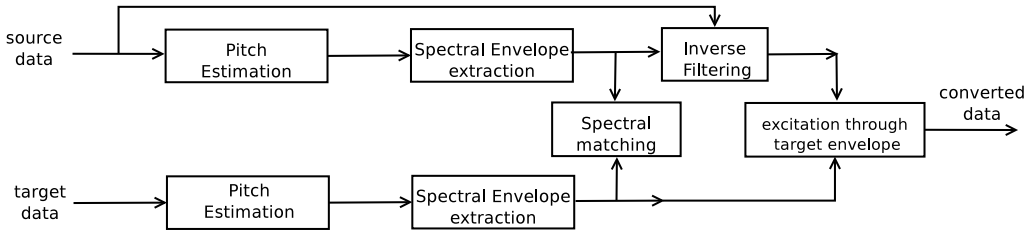


Figure 3.13: Block diagram of the conversion procedure.

identity and the appropriate voice feature that represents speaker identity is the spectral envelope.

In this section the source speakers for parallel corpora is considered. The steps for the proposed SI are:

1. **Analysis:** The extraction of spectral envelope for each speaker is performed. The analysis procedure is exactly the same as in voice conversion (subsection 3.3.1), using TE method in voiced frames and LPC in unvoiced (table 3.1).

2. **Spectral matching:** The alignment in speech conversion is based upon the source speaker. In SI the time warping takes place in accordance with a reference speaker. For instance, interpolating three speakers, speaker 1, speaker 2 and speaker 3, one of them is chosen as the reference speaker, let be speaker 1 as reference. Then the times axis of speakers 2 and 3 are warped, each of them, to correspond to the time axis of reference speaker 1. Spectral matching is also identical with speech conversion (table 3.2).

3. **Interpolating Speakers:** The difference here compared to VC is that the excitation of the reference speaker is filtered from an interpolated spectral envelope. To further the range of the produced voices, the vocal tracts of the given speakers are normalized in accordance with the reference speaker, and we differentiate the length of the interpolated vocal tract to a factor λ. Thus, for speakers 1 and 2, the interpolated vocal tract is:

$$H(\omega) = (1-a)H_1(\lambda\omega) + aH_2(\lambda\omega), \quad a \in [0,1], \tag{3.38}$$

where H_1 is the spectral envelope of the reference speaker, H_2 is the normalized spectral envelope of speaker 2 with reference to speaker 1. The vocal tract length normalization is conducted through linear piecewise frequency warping algorithm [13].

The above procedure could be also applied for N speakers, and the interpolated vocal tract will be (figure 3.14):

$$H(\omega) = a_1 H_1(\lambda\omega) + a_2 H_2(\lambda\omega) + \cdots a_N H_N(\lambda\omega), \quad \text{with} \sum_{i=1}^{N} a_i = 1. \tag{3.39}$$

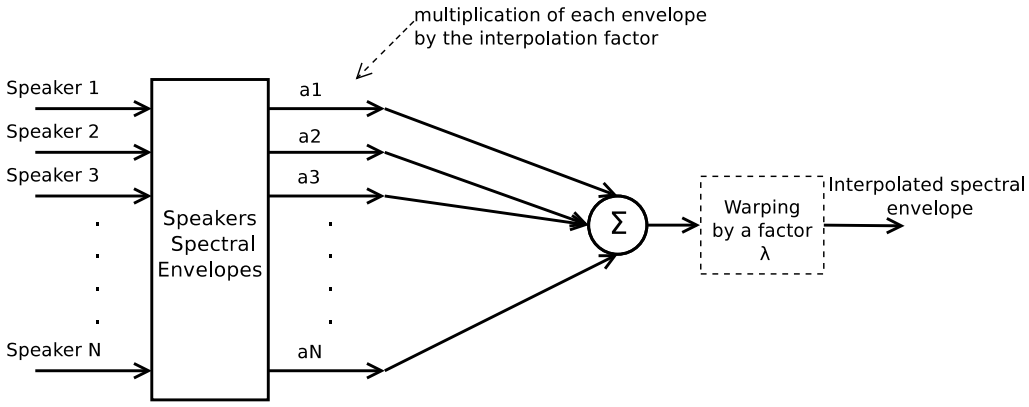


Figure 3.14: Spectral envelope interpolation.

Any subset of a given set of voices could be used applying the proposed method to produce a considerably larger new set that will contain high quality voices. The block diagram of two speakers interpolation is depicted in figure 3.15.In this thesis the interpolation was performed between two and three speakers. A subset of results is provided in the enclosed cd and in http://www.csd.uoc.gr/∼ ggrekas/SC_SI.html.

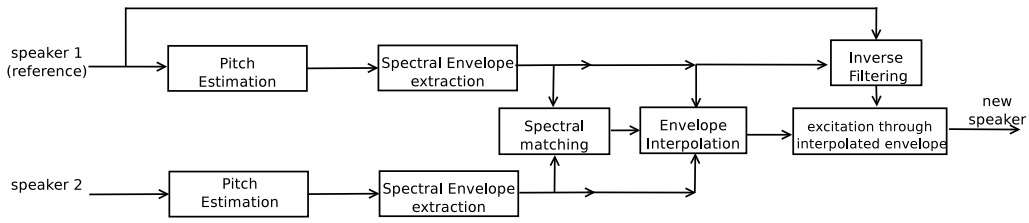


Figure 3.15: Block diagram of Speakers interpolation.

Chapter 4

Conclusion

To summarize, in the present thesis i worked on the speaker interpolation problem and on speech conversion. Parallel corpora were used to limit the artifacts of VQ, GMM or other conversion techniques, which generally are not methods of parallel corpora. Efficient techniques for spectral envelope extraction (TE) and time and frequency warping (i.e. piecewise linear and frequency warping) were used.

The usage of TE compared with the regularized LPC achieved a better representation of spectral envelope, particularly in high pitch voices. As a result, the produced SI and SC were substantially improved in quality, with the improvement being especially noticeable in high pitch voices. By applying piecewise linear time warping the interpolated and converted voices presented fewer artifacts in the areas ranging from voiced to unvoiced speech, or from unvoiced to voiced speech. In synthesis, the interpolated lattice filter resulted in high peaks in some areas of voiced to unvoiced portions, when the first formant amplitude, of the target speaker, was very high compared to the following formants. In OLA method these peaks were not present, and this happened because the lattice parameters vary with time and the unnormalized lattice filter may lose stability [34].

The advantages of this work were considerable in speaker interpolation. The linear interpolation of the spectral envelopes provided an intermediate speaker. The widening or the shrinking of spectral envelope was analogous to the change of vocal tract length, thus the resulting individual speaker has had an interpolated vocal tract with a desired length. Consequently, the number of the produced individual speakers is substantially larger compared to the initial set of speakers. In speech conversion the artifacts were reduced as expected, and they only appeared in a small set of speakers (10%), with most of them being detectable after careful listening.

Despite its contribution to the improvement of SI and SC, a number of limitations present should also be mentioned to be considered in future research. Firstly,

the proposed method is deemed unsuitable for use in real-time applications, due to the iterative nature of the procedures used in TE and in spectral matching. Secondly, the quality of the results could be further improved, especially from unvoiced-to-voiced portions or to voiced-to-unvoiced. One more limitation of this work is the speaker interpolation and conversion for non-parallel corpora, but it is suitable for a compare measure as a the target quality in interpolation or conversion.

Future work could involve many different sections of the method, such as improvements with respect to the complexity of the algorithm. For instance, in TE method pruned FFT could be used [14][9], where the non-zero elements in cepstral smoothing are less than 10% of the total elements, while methods to reduce TE iterations could be proposed [1][15]. A stable lattice filter can be used to improve the quality of the results [34]. The extension of this thesis to non-parallel corpora, like GMM based conversion, VQ, etc. Another improvement is a pitch synchronous analysis-synthesis processing of the above method, which gives more efficient representations of the acoustic feature in each portion. The improvements varies with the desired application and the accommodation of this technique must take into account the possible artifacts could occur, when high quality is demanded.

Bibliography

[1] X. Rodet A. Robel. Real time signal transposition with envelope preservation in the phase vocoder. *IEEE Transactions on signal processing*, 1994. [cited at p. 20, 42]

[2] Shikano K. Kuwabara H. Abe M., Nakamura S. Voice conversion through vector quantization. *ICASSP 1988*, 1:655 – 658, 1988. [cited at p. 2]

[3] Ronald W. Schafer Alan V. Oppenheim. *Discrete-time Signal Processing.* Prentice Hall, second edition, 1999. [cited at p. 32]

[4] Ning Bi and Yingyong Qi. Application of speech conversion to alaryngeal speech enhancement. *Speech and Audio Processing, IEEE Transactions on*, 5(2):97 –105, March 1997. [cited at p. 2]

[5] D. G. Childers. Glottal source modeling for voice conversion. *Speech Communication*, 16(2):127 – 138, 1995. Voice Conversion: State of the Art and Perspectives. [cited at p. 2]

[6] J. Clifford D.J Berndt. Using dynamic time warping to find patterns in time series. *AAAI Workshop in Knowledge Discovery in Databases*, 1994. [cited at p. 26]

[7] A. EI-Jaroudi and J. Makhoul. Discrete all pole modeling. *IEEE Transactions on Signal Processing*, 1991. [cited at p. 11, 12]

[8] Paulo Jorge S. G. Ferreira. Interpolation and the discrete papoulis-gerchberg algorithm. *IEEE Transactions on signal processing*, 1994. [cited at p. 15, 16]

[9] M. Franchetti, F.; Puschel. Generating high performance pruned fft implementations. *ICASSP*, 2009. [cited at p. 20, 42]

[10] E. Moulines H. Valbret and J.P. Tubach. Voice transformation using psola technique. *Speech Communication*, 1992. [cited at p. 24]

[11] Simon Haykin. *Adaptive filter theory.* Prentice Hall, 4th edition, 2002. [cited at p. 5]

[12] H. Fujisaki H. Hermansky and Y. Sato. Spectral envelope sampling and interpolation in linear predictive analysis of speech. *ICASSP*, 1984. [cited at p. 19]

[13] Yaqin Li Hongcai Feng, Cao Yuan. Speaker normalization method based on the piece-wise linear frequency warping. *In IEEE International Conference on E-Learning, E-Business, Enterprise Information Systems, and E-Government*, 2009. [cited at p. 32, 38]

[14] Chau-Yun Hsu and Wei-Mei Chen. Application of pruning fft technique to discrete extrapolation. *Int. J. Electronics*, 1993. [cited at p. 20, 42]

[15] Chau-Yun Hsu and Tsung-Ming Lo. Improved papoulis-gerchberg algorithm for restoring lost samples. *IEEE International Symposium on Signal Processing and Information Technology*, 2005. [cited at p. 15, 16, 42]

[16] Zhenhua Huang and Limin Hou. Speaker normalization using dynamic frequency warping. *Audio, Language and Image Processing, 2008. ICALIP 2008. International Conference on*, pages 1091 –1095, jul. 2008. [cited at p. 24]

[17] S. Imai and Y. Abe. Spectral envelope extraction by improved cepstral method. *Electron. and Commun. (in Japan)*, 1979. [cited at p. 15]

[18] Stylianou Ioannis. *Harmonic plus Noice Models for Speech, compined with Statistical Methods, for Speech and Speaker Modification*. PhD thesis, Ecole Nationale Superieure des Telecommunications, 1996. [cited at p. 14]

[19] John H. L. Hansen John R. Deller, John G. Proakis. *Discrete-time processing of speech signals*. MACMILLAN, 1993. [cited at p. 12, 14, 21, 35]

[20] A. Kain and M.W. Macon. Spectral voice conversion for text-to-speech synthesis. In *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*, volume 1, pages 285 –288 vol.1, May 1998. [cited at p. 1]

[21] L. A. Ekman W. B. Kleijn and M. N. Murthi. Spectral envelope estimation and regularization. *in Proc. IEEE Int. Conf. of Acoustics, Speech and signal Processing*, 2006. [cited at p. 9]

[22] Hisao Kuwabara and Tohru Takagi. Acoustic parameters of voice individuality and voice-quality control by analysis-synthesis method. *Speech Communication*, 10(5-6):491 – 495, 1991. Speaker Characterization in Speech Technology. [cited at p. 2, 35]

[23] B.P Lathi. *Signal Processing and Linear Systems*. Oxford University Press, 1998. [cited at p. 18]

[24] Carlo Magi. All-pole modelling of speech: Mathematical analysis compined with objective and subjective evaluation of seven selected methods. Master's thesis, Department of Electrical and Communications Engineering, 2005. [cited at p. 11]

[25] Man hon Wong Mi Zhou. A segment-wise time warping method for time scaling searching. *Information Sciences*, 2005. [cited at p. ix, 27, 28, 29]

[26] Hideyuki Mizuno and Masanobu Abe. Voice conversion algorithm based on piecewise linear conversion rules of formant frequency and spectrum tilt. *Speech Commun.*, 16(2):153–164, 1995. [cited at p. 2]

[27] M. N. Murthi and W. B. Kleijn. Regularized linear prediction all-pole models. *in Proc. IEEE Workshop on Speech Coding*, 2000. [cited at p. 9]

[28] M. Narendranath, Hema A. Murthy, S. Rajendran, and B. Yegnanarayana. Transformation of formants for voice conversion using artificial neural networks. *Speech Communication*, 16(2):207 – 216, 1995. Voice Conversion: State of the Art and Perspectives. [cited at p. 2]

[29] E. Moulines O. Cappe, J. Laronche. Regularized estimation of cepstrum envelope from discrete frequency points. *Proc. IEEE ASSP Workshop on Applications of Signal to Audio and Acoustics*, 1995. [cited at p. 15]

[30] A. Papoulis. A new algorithm spectral analysis and band limited extrapolation. *IEEE Transactions on Circuits and Systems*, 1975. [cited at p. 15]

[31] Thomas F. Quatiery. *Discrete-Time Speech Signal Processing, principles and practice*. Prentice Hall, 2002. [cited at p. 7, 12, 32]

[32] Axel Robel. Source filter modeling and spectral envelope estimation. *lecture on analysis, modeling and transformation of audio signals*, 2006. http://recherche.ircam.fr/equipes/analyse-synthese/roebel/amt_audiosignale/VL4.pdf. [cited at p. 5, 11]

[33] A. Röbel, F. Villavicencio, and X. Rodet. On cepstral and all-pole based spectral envelope modeling with unknown model order. *Pattern Recognition Letters*, 28:1343–1350, 08/2007 2007. [cited at p. 35]

[34] C. Schwarz and S. Dasgupta. A new normalized relatively stable lattice structure. *Signal Processing, IEEE Transactions on*, 49(4):738 –746, April 2001. [cited at p. 41, 42]

[35] Pavel Senin. Dynamic time warping algorithm review. Technical report, University of Hawaii at Manoa, Information and Computer Science Department, 2008. [cited at p. 21]

[36] Y. Stylianou, O. Cappe, and E. Moulines. Continuous probabilistic transform for voice conversion. *Speech and Audio Processing, IEEE Transactions on*, 6(2):131 –142, March 1998. [cited at p. 2]

[37] T. Toda, A.W. Black, and K. Tokuda. Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory. *Audio, Speech, and Language Processing, IEEE Transactions on*, 15(8):2222 –2235, 2007. [cited at p. 2]

[38] Tomoki Toda. *High-Quality and Flexible Speech Synthesis with Segment Selection and Voice Conversion*. PhD thesis, Department of Information Processing Graduate School of Information Science Nara Institute of Science and Technology, 2003. [cited at p. 1, 2]

[39] F. Villavicencio, A. Robel, and X. Rodet. Applying improved spectral modeling for high quality voice conversion. In *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pages 4285 –4288, 2009. [cited at p. 2, 3]

[40] Robel A. Rodet X. Villavicencio, F. Improving lpc spectral envelope extraction of voiced speech by true envelope estimation. *IEEE ICASSP 2006*, 2006. [cited at p. 19]

[41] Robel A. Rodet X. Villavicencio, F. Applying improved spectral modeling for high quality voice conversion. *IEEE ICASSP 2009*, 2009. [cited at p. 12]

[42] John G. Proakis Vinay K. Ingle. *Digital Signal Processing using Matlab.* Brooks/Cole, 2000. [cited at p. 32]

[43] Bo Wei and Jerry D. Gibson. Comparison of distance measures in discrete spectral modeling, 2000. [cited at p. 11]

[44] Wim Haes Xavier and Xavier Rodet. Discrete cepstrum coefficients as perceptual features, 2003. [cited at p. 36]