UNIVERSITAT POLITÈCNICA
DE CATALUNYA

Ph.D. Thesis

Voice Conversion applied to Text-to-Speech systems

Author: Helenca Duxans i Barrobés

Advisor: Dr. Antonio Bonafonte Cávez

Speech Processing Group
Department of Signal Theory and Communications
Universitat Politècnica de Catalunya

Barcelona, May 2006

A la meva mare,

Abstract

In this Ph.D. dissertation, the study and design of Voice Conversion (VC) systems are addressed. VC systems modify a speaker voice (*source speaker*) to be perceived as if another speaker (*target speaker*) had uttered it. This technology has mainly emerged as a novel technique to create new speakers for Text-To-Speech systems (TTS).

The goal of this thesis is to develop a VC system to work as a post-processing block for a TTS. VC systems applied to a TTS have mainly two particular characteristics: source data is unlimited, as any utterance can be generated by the TTS, and phonetic information is available forehand. Both characteristics will be explored in order to improve the performance of the state of the art VC systems.

Speaker voice individuality is the result of many acoustic and linguistic cues. In the current work, speaker individuality will be described only by segmental acoustic characteristics. In particular, vocal tract parameters and LP residual signal, as speech model parameters, will be converted by two different strategies. Conversion of prosodic features are out of the scope of the thesis.

State of the art vocal tract conversion systems are based on Gaussian Mixture Models (GMM) to model the speaker acoustic space and map the vocal tract parameters. Although systems based on GMMs are well suited to the vocal tract conversion task, they can not deal with source data without its corresponding parallel target data. Two approaches for the used of non-parallel source data in GMM systems are proposed: a modified EM algorithm with fixed covariance matrices, and a strategy to complete non-parallel data by including transformed vectors as parallel vectors.

In order to improve the vocal tract mapping performance of GMM systems, two novel vocal tract conversion systems are proposed in this dissertation. On one hand, Hidden Markov Models (HMM) systems include dynamic information in the acoustic model to better convert phoneme boundary frames. On the other hand, phonetic information is introduced in the mapping by Classification and Regression Trees (CART), where a decision tree is used to classify the acoustic parameters prior to the application of the mapping function. The hypothesis that phonetic data carries information that allows to better split the acoustic space according to the transformation error has been validate by objective and perceptual evaluations.

The study of LP residual modification systems begins with an analysis of previous published works, identifying two strategies to face the problem: a conversion of the source residuals by mapping functions and a prediction of converted residuals from the converted vocal tract parameters. After a comparative of both strategies, a Phonetic Residual Selection and Fixed Smoothing system is proposed for generating voiced converted LP residual signals. This system selects the most appropriated residual for the converted vocal tract parameters from a collection

of databases, organized by phonetic information, which entries are target LSF-residual pairs. Once the sequence of voiced residual frames has been selected, a fixed length smoothing is applied. Unvoiced residual frames are generated as white Gaussian noise. Although phonetic information reduces the computational load of the selection module, converted speech quality is higher when using only one database instead of a collection of databases.

The complete VC system proposed in this dissertation, which is based on CART vocal tract conversion and LP residual signal selection, has been submitted to a formal evaluation of the integrated European project Technology and Corpora for Speech-to-Speech Translation (TC-STAR). TC-STAR evaluations concluded that although the proposed VC system achieves a speaker personality conversion, the VC technology needs to improve the converted speech quality to be acceptable in a real application.

Resum

Aquesta tesi planteja l'estudi i el disseny de sistemes de Conversió de Veu (CV). Els sistemes de CV modifiquen la veu d'un locutor (*locutor origen*) de tal forma que sembli la veu d'un altre locutor determinat (*locutor destí*). Aquesta tecnologia s'ha ideat sobretot com a tècnica per a la creació de nous locutor en sistemes de conversió text a veu (CTV).

L'objectiu de la tesi és el desenvolupament d'un sistema de CV per ser aplicat a la sortida d'un sistema de CTV. Els sistemes de CV aplicats a CTV tenen dues característiques principals: la quantitat de dades del locutor origen no és limitada, ja que qualsevol locució pot ser generada pel CTV, i es disposa d'informació fonètica. S'han explorat ambdues característiques per tal de millorar el funcionament del sistemes de CV actuals.

La personalitat de la veu d'un locutor ve determinada per diversos factors acústics i lingüístics. En aquest treball, la personalitat de la veu s'ha descrit a partir de només característiques acústiques segmentals. En concret, s'han utilitzat dues estratègies diferents per a convertir els paràmetres de tracte vocal i la senyal residual estimats amb tècniques de predicció lineal.

Els sistemes actuals de conversió de tracte vocal es basen en Models de Mescles de Gaussianes (GMM) per modelar l'espai acústic dels locutors i transformar els paràmetres de tracte vocal. Una limitació dels sistemes basats en GMM és que no poden tractar amb dades del locutor origen que en la fase d'entrenament no tenen correspondència amb el locutor destí. S'han proposat dues aproximacions per utilitzar aquestes dades sense correspondència: una versió modificada de l'algoritme EM amb les matrius de covarianca fixes, i una estratègia que completa les dades sense correspondència amb els vector d'origen transformats.

Per tal de millorar els resultats dels sistemes GMM, s'han proposat dues tècniques noves en la conversió de tracte vocal. La primera tècnica ha consistit en introduir informació dinàmica al model acústic per mitjà d'un sistema basat en Models Ocults de Markov per intentar millorar la conversió dels trams de senyal corresponents a transicions fonètiques. La segona tècnica introdueix informació fonètica a la conversió mitjancant arbres de decisió (CART, Classification and Regression Tree). L'arbre de decisió classifica els paràmetres acústics prèviament a la aplicació de la funció de transformació. Avaluacions objectives i perceptuals han validat la hipòtesi que la informació fonètica permet dividir l'espai acústic d'una forma més adequada per a la tasca de CV.

L'estudi dels sistemes de modificació de senyal residual s'ha iniciat amb un anàlisis dels treballs prèviament publicats, identificant dues estratègies per resoldre el problema. Una estratègia consisteix en convertir el senyal residual d'origen utilitzant funcions de transformació. L'altra estratègia prediu el senyal residual desitjat a partir dels paràmetres de tracte vocal convertits.

Després de realitzar una comparativa de les dues estratègies, s'ha proposat un sistema de Selecció Residual Fonètica amb Suavitzat Fix per generar els senyals residuals sonors per a la veu convertida. Aquest sistema selecciona el senyal residual més apropiat pels paràmetres de tracte vocal convertits a partir d'un conjunt de bases de dades organitzades mitjancant informació fonètica. Les entrades de cada base de dades són parelles de vectors LSF i senyals residuals del locutor destí. Un cop la seqüència de senyals residuals sonors has estat seleccionada, s'ha aplicat un suavitzat de duració fixa. Els senyals residuals sords s'han generat a partir de soroll blanc Gaussià. Encara que la introducció d'informació fonètica per organitzar les bases de dades ha reduït la càrrega computacional del mòdul de selecció, la qualitat de la veu convertida és millor quan s'utilitza una sola base de dades sense informació fonètica.

El sistema de CV complert proposat en aquesta tesi, que està basat en un sistema CART de conversió de tracte vocal i en selecció de senyal residual, ha participat en una campanya d'avaluació pública del projecte europeu Technology and Corpora for Speech-to-Speech Translation (TC-STAR). Les conclusions d'aquesta avaluació han estat que el sistema de CV proposat ha reeixit en l'objectiu de convertir la personalitat de la veu, tot i que s'ha de millorar la qualitat dels sistemes de CV actuals per a ser acceptable en un aplicació real.

Acknowledgements

Agraïments:

Primer de tot voldria agrair l'ajuda del meu director de tesi A. Bonafonte en la realització d'aquest treball, sobretot en la última etapa quan he tingut menys disponibilitat horària. També vull agrair totes les converses tècniques (i no tècniques) dels doctorants del grup de processament del senyal, ja que m'han ajudat a entendre moltes coses (i a passar-m'ho bé!). Vull agrair especialment l'ajuda d'en J. Padrell, per guiar-me els primers anys de doctorat, i del F. Diehl, per haver revisat un capítol de la tesi.

També vull agrair la participació de tota la gent que, en un moment o altre durant aquests quatre anys, han "patit" els testos perceptuals. La meva colla estarà contenta que ja acabi.

I sobretot, agraeixo al Juanjo haver suportat els alts i baixos d'ànim que porta fer una tesi, i per tots els caps de setmana que s'ha quedat sol cuidant el Gerard.

Helenca Duxans

Esparreguera, maig 2006.

Contents

Notation

Boldface upper-case letters denote matrices and boldface lower-case letters denote column vectors.

$\mathbb{R}, \mathbb{C}$	The set of real and complex numbers, respectively.
$\mathbf{X}^*$	Complex conjugate of the matrix $\mathbf{X}$.
$\mathbf{X}^T$	Transpose of the matrix $\mathbf{X}$.
$\lvert\mathbf{X}\rvert$ or $\det(\mathbf{X})$	Determinant of the matrix $\mathbf{X}$.
$\max, \min$	Maximum and minimum.
$\operatorname{argmax}, \operatorname{argmin}$	Argument of the maximum and minimum.
floor	The largest (closest to positive infinity) value that is not greater than the argument and is equal to a mathematical integer.
$\lvert x\rvert$	Modulus of the complex scalar x.
$\lVert\mathbf{x}\rVert$	Euclidean norm of the vector $\mathbf{x}$: $\lVert\mathbf{x}\rVert = \sqrt{\mathbf{x}^H\mathbf{x}}$.
$\mathbf{X}^{-1}$	Inverse of the matrix $\mathbf{X}$.
$\mathbb{E}[\cdot]$	Mathematical expectation.
$\mathcal{N}(\mu, \mathbf{\Sigma})$	Real Gaussian vector distribution with mean μ and covariance matrix $\mathbf{\Sigma}$.
$\lvert\mathcal{A}\rvert$	Cardinality of the set $\mathcal{A}$, i.e., number of elements in $\mathcal{A}$.
$\exp(\cdot)$	Exponential.
$\log(\cdot)$	Natural logarithm.
$\mathrm{P}(\cdot)$	Probability.
$\mathrm{p}(\cdot)$	Probability density function.

Acronyms

ANN	Artificial Neural Networks.
CART	Classification and Regression Tree.
DFW	Dynamic Frequency Warping.
GMM	Gaussian Mixture Model.
HMM	Hidden Markov Model.
IHMD	Inverse Harmonic Mean Distance.
iid	Independent and identically distributed.
LAR	Log Area Ratio.
LMR	Linear Multivariate Regression.
LPC	Linear Prediction Coding.
LP-PSOLA	Linear Prediction Pitch Synchronous Overlap and Add.
LSF	Line Spectral Frequency.
MAP	Maximum a Posteriori.
MLLR	Maximum Likelihood Linear Regression.
MOS	Mean Opinion Score.
OLA	Overlap and Add.
pdf	Probability density function.
REP	Reduced Error Pruning.
RMS	Root Mean Square.
SMS	Short Message Service.
STASC	Speaker Transformation Algorithm using Segmental Codebooks.
STRAIGHT	Speech Transformation and Representation using Adaptative Interpolation of weiGTHed spectrum.
TC-STAR	Integrated European project Technology and Corpora for Speech-to-Speech Translation.
TD-PSOLA	Time Domain Pitch Synchronous Overlap and Add.
TTS	Text-to-Speech.
VC	Voice Conversion.
VFS	Vector Field Smoothing.

VQ Vector Quantization.

Chapter 1

Introduction

Speech is the most used and natural way for people to communicate. From the beginning of the man-machine interface research, speech has been one of the most desired mediums to interact with computers. Therefore, speech recognition and text-to-speech conversion have been studied to make the communication with machines more human likely.

In order to increase the naturalness of oral communications between humans and machines, all speech aspects must be involved. Speech does not only transmit ideas and concepts, but also carries information about the attitude, emotion and individuality of the speaker.

Speaker identity, the sound of a person's voice, is a key factor in oral communications. Speaker identity allows us to differentiate between speakers in a conference call, on a radio program, etc. from their voices. Moreover, most people have strong preferences for particular voices, to such a point that a computer interface may be rejected for its voice individuality. Recently, the Voice Conversion technology has emerge as a way to control the voice identity of any natural or synthetic utterances.

In this Ph.D. dissertation, the study and design of Voice Conversion systems are addressed. This introductory chapter defines the concept of Voice Conversion, and section 1.2 summarizes the main motivations and applications of this technology. Finally, section 1.3 presents the objective of the thesis and in section 1.4 there is an outline of the dissertation.

1.1 Voice Conversion: a Definition

Voice Conversion (VC) systems modify a speaker voice (*source speaker*) to be perceived as if another speaker (*target speaker*) had uttered it. Therefore, given two speakers, the goal of a VC system is to determine a transformation that makes the speech of the source speaker sounds as it were uttered by the target speaker.

Speech signal carries different types of information, such as: linguistic content, speaker identity and environmental noise. Different fields of speech technologies are focused on each one of these information types. The focus of VC systems is speaker identity.

VC systems must deal with two problems. The first problem is the identification of the speaker characteristics during the analysis phase to acquire speaker-dependent knowledge. The second problem is the replacement of the source characteristics for the target characteristics in the synthesis phase. Both operations must be performed independently of the message and the environment information.

1.2 Applications of Voice Conversion

VC has mainly emerged as a novel technique to create new speakers for Text-To-Speech systems. However, applications of VC systems can be found in several other fields, such as automatic Speech-to-Speech translation, education, medical aids and entertainment.

TTS customization. Text-To-Speech Synthesis (TTS) quality has increased by the employment of large corpora and unit-selection techniques. This type of systems, generally called corpus-based TTS, generate synthetic speech by selecting the most appropriated sequence of acoustic units from a speaker-dependent database and then apply a smoothing strategy to joint the selected units together. Corpus-based TTS can synthesize only speech having the specific style present in the corpus. Therefore, in order to synthesize other types of speech, e.g. emotional or expressive speech or speech of various speakers, representative speech samples should be recorded in advance. Moreover, the representative speech samples need to be large-sized corpora, in order to maintain the output quality.

Speech recording and data processing for a TTS is expensive and time consuming. VC may be a fast and a cheap way to build new voices for a TTS. Thanks to VC, it will be possible to read e-mails or SMS with their sender's voice, to assign our and our friends' voices to characters when playing on a computer game, or to apply different voices to different computer applications. VC may be also applied to emotional speech synthesis [Kaw03], as an aid to prosodic modifications on a neutral sentence.

Automatic Speech-to-Speech translation. The VC technology will be very useful in interpreted telephony, when the translation task requires speaker identification by listeners. For example, in a conference call with more than two participants it is very important to be able to differentiate speakers by their voices.

Education. When learning foreign languages, proper intonation of sentences and pronunciation of non-existing phonemes in the native language is one of the most difficult tasks for

students. VC may help to learn foreign languages [Mas01, Mas02], especially in pronunciation exercises, when students would listen to their own voices pronouncing foreign sounds properly.

Medical aids. Another application field of VC is speaking aids for people with speech impairments. Voice transformation systems may be used to improved the intelligibility of abnormal speech uttered by a speaker who has speech organ problems [Hos03]. On the other hand, VC may be also useful for designing hearing aids appropriated for specific hearing problems. For example, some impairments prevent people from hearing specific frequency range sounds. VC technology may morph speech signals to other frequency ranges in order to improve the recognition rate.

Entertainment. One of the most obvious applications of VC in the entertainment field is karaoke, where the singer is helped to success in every kind of songs. Other experiments have been done in film dubbing and looping [1] [Tur02]. By employing only several dubbers, voices of famous actresses/actors can be generated in any language, and new utterances of actresses/actors who are not alive can be synthesized. Another dubbing application may be to regenerate the voices of actresses/actors who have lost their voice characteristics due to old age. Integrated with 3-D facial animation techniques, VC can be employed to create virtual characters with a desired speaker identity for multiple applications, such as video games.

In dive practices, VC might be applied to enhance the helium speech signals of the submariners.

VC may be also applied to the most classical fields of speech technology, for example in very low bandwidth speech encoding, by transmitting the speech without speaker information and adding it at the decoding step. Moreover, acquiring a high level of knowledge about speaker individuality will help the success of other speech technologies, as speech or speaker recognition tasks.

1.3 Thesis Objectives

The goal of this thesis is to develop a novel Voice Conversion system to work as a post-processing block in a Text-to-Speech system. Two factors of the developed system will be evaluated: the voice individuality change and the converted speech quality.

The main motivation for developing a VC system applied to a TTS is to be able to generate several speaker's voices without producing and storing large databases.

[1] to loop: to replace undesired utterances with the desired ones.

In order to propose a novel approach, the state of the art of VC is studied. The main objectives of this study are: to identify the most relevant topics of the problem, in order to delimit our research, and to find out the limitations of the current approaches, in order to improve the actual performance of VC systems.

Moreover, the particular characteristics of VC systems applied to a TTS will be taken into account in the final approach. These characteristics may be summarized in the following two statements:

- There is no limitation about source data to train the conversion system. Language resources used for actual corpus-based TTS systems are very large, usually several hours. Furthermore, any utterance can be generated by the TTS with an acceptable quality.
- Phonetic information is available forehand, due to the TTS requirements.

Both characteristics will be studied and incorporated to the final approach, in order to improve the performance of the state of the art VC systems.

All the hypothesis are studied and evaluated for Spanish intra-gender and cross-gender conversions. Conversion of the prosody (pitch evolution, speech rate, speech energy ...) and high level features conversion (syntactic patterns, vocabulary, speech style ..) are out of the scope of this thesis. Only fundamental frequency modifications by simple mean and variance adjustments will be carried out.

1.4 Thesis Overview

This dissertation is organized as follows:

Chapter 1 introduces the concept of Voice Conversion and its applications.

Chapter 2 provides the theoretical background related to the Voice Conversion task and summarizes the most influencing works of Voice Conversion.

Chapter 3 presents the framework of this study. In particular, there is a description of the training and evaluation corpora and the speech analysis performed, followed by a description of several evaluation techniques applied to Voice Conversion systems.

Chapter 4 is focused on the vocal tract conversion. After a study of a baseline transformation system based on Gaussian Mixture Models, new approaches to the problem are proposed. Dynamic information is included into the baseline system by the use of Hidden Markov Models, and phonetic information is used by Decision Trees to transform the vocal tract parameters.

Chapter 5 deals with the LP residual modification problem. First, an extended revision on previous works is presented identifying two main strategies to face the problem. Once both strategies are compared, a novel approach that uses phonetic information in order to reduce the computational load of the state of the art residual modification systems is proposed.

Chapter 6 discuss alternatives to generate several speakers for a Text-To-Speech system with few training data and presents the obtained results of a complete Voice Conversion system in the evaluation campaign organized by the integrated European project Technology and Corpora for Speech to Speech Translation (TC-STAR).

Chapter 7 summarizes the main research contributions of this thesis dissertation. Furthermore, lines for future research that can be considered as extensions of the work developed in this dissertation are described.

Chapter 2

Theoretical Background

This chapter provides the theoretical background related to the Voice Conversion task and summarizes the most influencing works of Voice Conversion.

When facing a problem of transforming the identity of a source speaker voice into the identity of a target speaker voice, the first statement to settle down is what *voice identity* means and how it can be described in terms of features to be converted. To answer both questions, several aspects are reviewed. Section 2.1 describes the human speech production process, in order to identify the main characteristics of the speech signal and their relationship to the human physics. In section 2.2 there is a brief introduction to phonetics, providing the definitions that will be used in the following chapters. The voice identity study is completed in section 2.3, which is focus on the speaker discrimination topic.

To finish the theoretical background overview, section 2.4 presents a general state of the art Voice Conversion system architecture, including a description of several published methodologies to developed each one of the system components.

2.1 Human Speech Production

The human production of sounds can be described in terms of three components:

1. The source of an airflow.
2. The source of the sound.
3. The sound modification system.

These components correspond to the *lungs*, the *larynx* and the *vocal tract* respectively (see figure 2.1). The resulting sound is a longitudinal pressure wave formed of compressions and rarefactions of air molecules, in a direction parallel to that of the application of energy [Hua01].

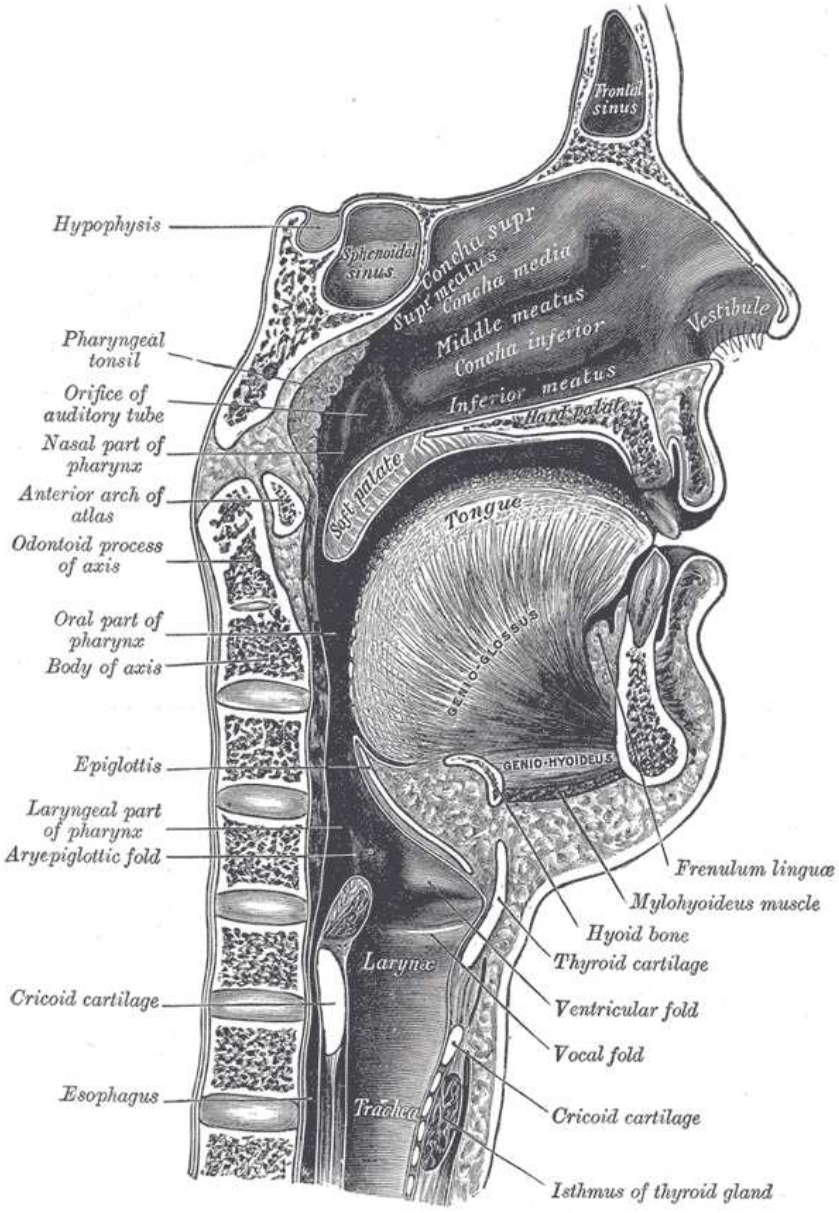


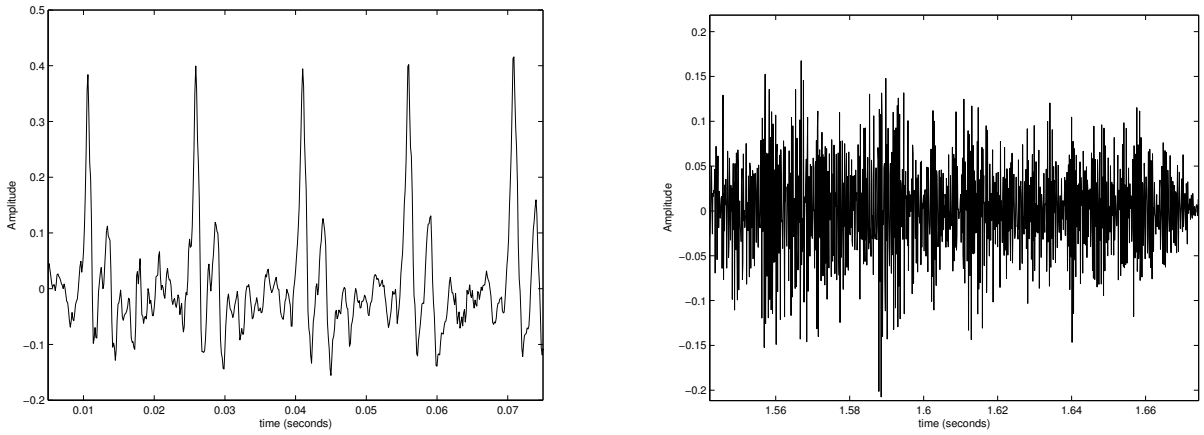
Figure 2.1: Representation of a sagittal section of the human speech production system, from Gray's Anatomy 1918 edition.

The production of a sound begins when the *lungs* push the air through the trachea into the larynx. In the *larynx* there are the vocal cords, also called vocal folds, which are composed of two in-foldings of mucous membrane stretched horizontally. According to how the vocal cords act during the phonation, two types of sources of the sound are distinguished:

Voiced During voiced phonation the vocal cords vibrate: from a close position to obstruct the airflow through the glottis[1], they are forced to open by an increase of the air pressure in the lungs, and to closed again as the airflow pass the cords.

Unvoiced During unvoiced phonation, the vocal cords remain open.

Figure 2.2 illustrates the properties of a voiced portion and an unvoiced portion of speech respectively. In the left figure, it is shown how the vibration of the vocal cords produce a periodicity on the speech signal. However, for unvoiced sounds as the one represented in the right figure, there is not such periodicity.



(a) (b)

Figure 2.2: Waveform of a portion of: a) a voiced phone b) an unvoiced phone.

A person's pitch is determined by the resonant frequency of the vocal cords. Altering the length and thickness of the vocal cords produces a change in the fundamental frequency of the sound.

The *vocal tract* is the area between the glottis and the lips. From the glottis, the air is pushed through the pharynx and then through the oral and/or nasal cavity. The vocal tract shape modifies the sounds produced by the vocal cords. The shape of the vocal tract is defined by the location and position of lips, jaw, tongue, and teeth acting as articulators. Different vocal tract

[1]The opening between the vocal cords.

shapes have different resonant frequencies, called formants, which determine different speech sounds.

2.2 A Brief Introduction to Phonetics

The speech production system, as described above, determines the inventory of sounds that can be produced by the human voice. Each one of the sounds of this inventory is called a phone. The most common discrete unit to describe phones is the phoneme. A phoneme is the smallest unit that distinguishes words and morphemes[2]. Therefore, changing a phoneme of a word to another phoneme produces a different word or a nonsense utterance, whereas changing a phone to another phone, when both belong to the same phoneme, produces the same word with an odd or an incomprehensible pronunciation. Phonemes are not physical segments themselves, but mental abstractions of them. Different acoustic realizations of a phoneme are called allophones. The acoustic characteristics of phonemes come from the vocal tract movement during their articulation.

Phonemes can be described according to:

Vowel/glide/consonant category During vowel articulation the tongue shape and its placement in the vocal cavity is not a major constriction for the airflow. In contrast, during consonant articulation there is a significant constriction or obstruction in the pharyngeal and/or oral cavities. Glides are similar to vowels, but shorter than true vowels and can not be stressed.

Point of articulation for consonants The point of articulation is the point of contact between an active articulator (typically some part of the tongue) and a passive articulator (typically some part of the roof of the mouth) during the obstruction in the vocal tract for a consonant phoneme. According to the point of articulation, consonants are divided in: *bilabial consonants*, articulated between the lips, *labiodental consonants*, articulated between the lower lip and the upper teeth, *dental consonants*, articulated between the front of the tongue and the top teeth, *velar consonants*, articulated between the back of the tongue and the soft palate (the velum), *alveolar consonants*, articulated between the front of the tongue and the ridge behind the gums (the alveolus), *palatal consonants*, articulated between the middle of the tongue and the hard palate, and *interdental consonants*, produced by placing the blade of the tongue against the upper incisors.

Manner of articulation for consonants The manner of articulation describes how speech organs involved in producing a sound make contact. For example, in *plosive consonants*

[2]A morpheme is the smallest language unit that carries a semantic interpretation.

there is a complete occlusion of both the oral and nasal cavities of the vocal tract, and therefore there is no airflow, whereas during *nasal consonants* articulation, there is a complete occlusion of the oral cavity and the airflow goes through the nose. A continuous frication (turbulent and noisy airflow) at the place of articulation is characteristic of *fricative consonants*, while in *approximant consonants* there is very little obstruction. *Lateral consonants* are a type of approximants pronounced with the side of the tongue. *Affricate consonants* begin like a plosive but evolve into a fricative. A *Flap consonant*, often called a tap, is a momentary closure of the oral cavity. And finally, *trill consonants* are those in which the articulator (usually the tip of the tongue) is held in place and the air stream causes it to vibrate.

Height for vowels and glides The articulatory features that distinguish different vowels are the height and the backness. Height refers to either the vertical position of the tongue relative to the roof of the mouth or the aperture of the jaw. Possible values of height are: open, mid-open, close and mid-close.

Backness for vowels and glides Backness refers to the horizontal tongue position during the articulation of a vowel relative to the back of the mouth. Possible values of backness are: front, center and back.

Voicing Voicing refers to the vocal cord vibration during the production of a sound. All the vowels are voiced, but there are voiced consonants as well as unvoiced consonants.

Tables 2.1 and 2.2 show an extended list of allophones and their characteristics that have been used in the current work.

SAMPA Code	**Group**	**Height**	**Backness**	**Voicing**
a	vowel	open	center	voiced
e	vowel	mid close	front	voiced
E	vowel	mid open	front	voiced
i	vowel	close	front	voiced
o	vowel	mid close	back	voiced
O	vowel	mid open	back	voiced
u	vowel	close	back	voiced
@	vowel	schwa	center	voiced
j	glide	close	front	voiced
w	glide	close	back	voiced
y	glide	close	front	voiced
uw	glide	close	back	voiced

Table 2.1: List of vocalic and glided allophones and their characteristics.

SAMPA Code	**Group**	**Point of articulation**	**Manner of articulation**	**Voicing**
p	consonant	bilabial	plosive	unvoiced
t	consonant	dental	plosive	unvoiced
k	consonant	velar	plosive	unvoiced
b	consonant	bilabial	plosive	voiced
d	consonant	dental	plosive	voiced
g	consonant	velar	plosive	voiced
B	consonant	bilabial	approximant	voiced
D	consonant	dental	approximant	voiced
G	consonant	velar	approximant	voiced
f	consonant	labiodental	fricative	unvoiced
s	consonant	alveolar	fricative	unvoiced
z	consonant	alveolar	fricative	voiced
S	consonant	palatal	fricative	unvoiced
Z	consonant	palatal	fricative	voiced
T	consonant	interdental	fricative	unvoiced
jj	consonant	palatal	fricative	voiced
x	consonant	velar	fricative	unvoiced
ts	consonant	alveolar	affricate	unvoiced
dz	consonant	alveolar	affricate	voiced
tS	consonant	palatal	affricate	unvoiced
dZ	consonant	palatal	affricate	voiced
l	consonant	alveolar	lateral	voiced
L	consonant	palatal	lateral	voiced
m	consonant	bilabial	nasal	voiced
n	consonant	alveolar	nasal	voiced
J	consonant	palatal	nasal	voiced
N	consonant	velar	nasal	voiced
rr	consonant	alveolar	trill	voiced
r	consonant	alveolar	tap	voiced
R	consonant	alveolar	tap	voiced

Table 2.2: List of consonantic allophones and their characteristics.

2.3 Speaker Discrimination

The acoustic speech signal carries three main types of information: the message of the utterance (*what* is said), the speaker identity (*who* says it) and some environment information (*where* it is said). Speaker characteristics refers to those factors in the spoken utterance which carry information about the speaker. Those factors are used by listeners to identify and discriminate the speaker of an utterance. In general, which those factor are and the degree of influence in the speaker identification task are not known. Studies in interspeaker variations and factors affecting voice quality have reveled that there are several parameters in the speech signal, both at the linguistic and at the acoustic level, which contribute to the interspeaker variability.

At the highest level voices are differentiated by the use of **linguistic cues** derived from the speech. These linguistic cues include the language of the speaker, the dialect, choice of lexical patterns, choice of syntactic constructs and the semantic context. The speaking style depends on socio/psychological factors of the speaker as: age group, social status, dialect, and the community that the speaker belongs. The characteristics of a speaker at the linguistic level are difficult to analyze and model.

There are other speaker dependent factors in a spoken utterance which can be measured or estimated directly from the acoustic waveform of speech. These factors form the acoustic level, which is divided into the segmental and suprasegmental levels.

The most important **segmental cue** is the timbre of the speaker's voice. When describing the human voice, people generally refer to the overall quality of a sound, its timbre, what the voice sounds like. The timbre enables the listener to differentiate between different speakers, despite the fact that they are uttering the same text. The timbre is a perceptual attribute, influenced by multiple factors. Acoustic descriptors of the timbre include the pitch, the glottal source spectrum and the vocal tract frequency response. These acoustic properties are conditioned by physical characteristics, such as dimensions of the vocal-tract and relative proportions between the supra-glottal cavities (laryngeal, oral, nasal ...). However, the speaker emotional state may modify the acoustic properties of a speaker.

There are many factors at the **suprasegmental level** which also affect the speaker's voice quality. Prosodic features such as pitch evolution, phoneme duration and intensity are unique for a given speaker. These features are dependent on the context and also on the language of the speaker.

In order to illustrate the differences between some of the segmental and suprasegmental cues from one speaker to another, figure 2.3 and figure 2.4 have been included. Both figures represent the waveform and the spectrogram of the Spanish sentence "¿nos podremos reincorporar Oscar y yo?" uttered by two different speakers. Differences on the spectrogram can be observed for

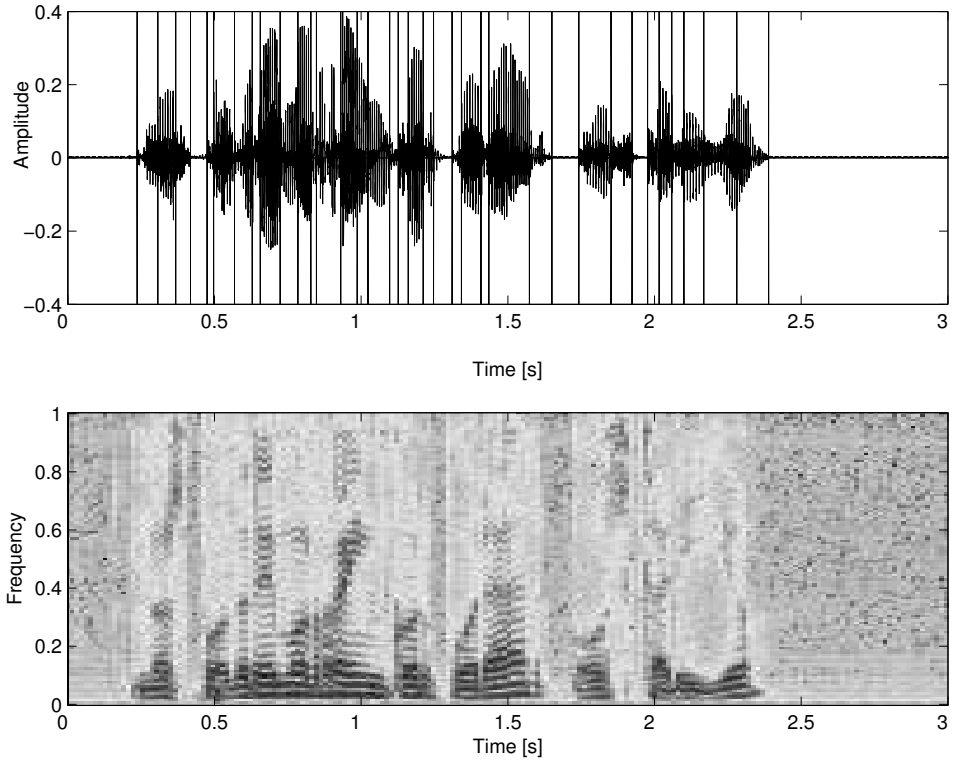


Figure 2.3: Waveform and spectrogram of the utterance "¿nos podremos reincorporar Oscar y yo?" for a male speaker, sampled at 16kHz. Vertical lines on the waveform represents phoneme boundaries.

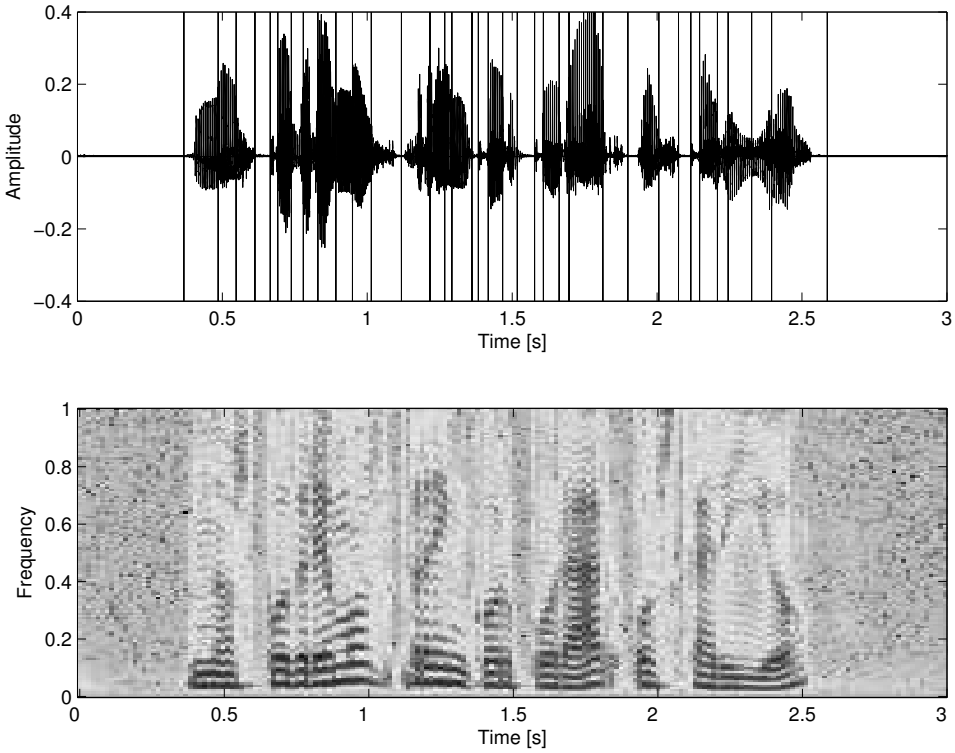


Figure 2.4: Waveform and spectrogram of the utterance "¿nos podremos reincorporar Oscar y yo?" for a female speaker, sampled at 16kHz. Vertical lines on the waveform represents phoneme boundaries.

the same phonemes: formant bandwidths of the female speaker are wider and formant locations higher than for the male speaker. Waveform representations illustrates the duration difference between the same phoneme of the male speaker and the female speaker. Moreover, it can be seen that the male speaker introduced a pause between the words "reincorporar" and "Oscar".

To sum up, there is not any single acoustic/linguistic parameter related to voice identity. Voice identity is a joint contribution of several factors, each factor with its own level of influence depending on the speaker of the utterance and the listener who is carrying out the discrimination. One of the main problems of VC is to find a way of representing the speaker individuality with a reduced number of parameters which are suitable to the conversion task.

2.4 State-of-the-art of Voice Conversion Systems

This section summarizes the most relevant published works in the VC field. First of all, the architecture of a generic VC system is described. Afterwards, several methodologies to developed each one of the system components are overviewed.

2.4.1 Voice Conversion System Architecture

A generic VC system operates in two different modes: the training mode and the transformation mode (see figure 2.5).

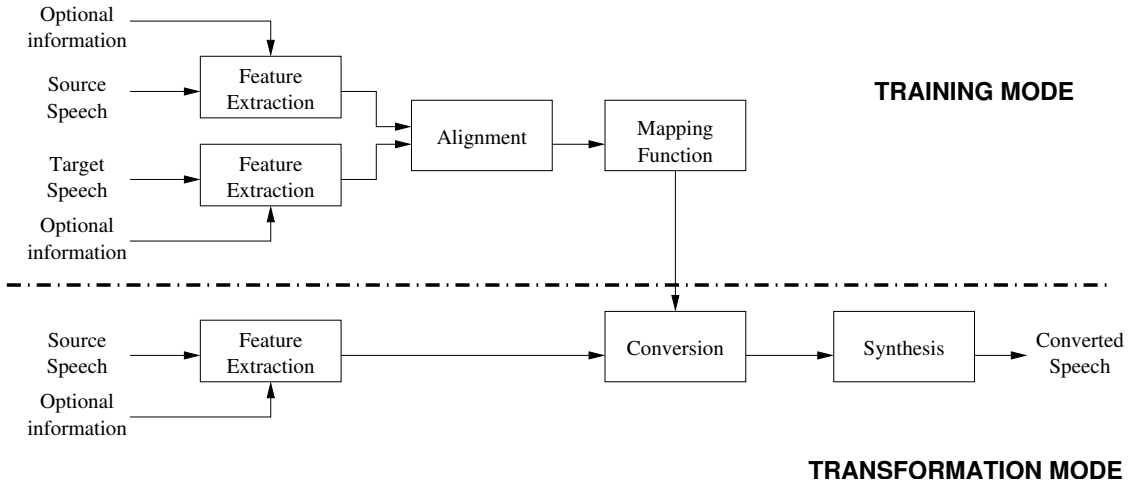


Figure 2.5: VC system block diagram.

In the training mode, speech samples and related information are analyzed for both source and target speakers and their characteristics are aligned. After learning the characteristics of each speaker, a conversion rule is estimated to map the source voice characteristics into those of

the target speaker. Therefore, for each new source-target speaker pair a training must be carried out. Four components must be highlighted in the training operating mode:

- The corpus (speech files and related information).
- The features used to model the speech and to be mapped.
- The alignment operation between source and target features.
- The mapping function from the source speaker to the target speaker.

During the transformation mode, the estimated conversion rule is applied to the original speech to create a converted speech that exhibits the target speaker's voice characteristics. Once the conversion function for a set of two speakers has been learned, any source utterance can be converted. One component must be highlighted in the transformation operating mode:

- The synthesis block, which includes the speech production and prosodic modification techniques.

In the following subsections there is a description of the state of the art alternatives for each one of the highlighted components.

2.4.2 Corpora for Voice Conversion Systems

Several strategies have been adopted to select a training and evaluation corpus for VC systems. In most of the published works, parts of already existing corpora, recorded for other purposes such as TTS, have been used. The amount of data goes from only one word to some minutes of speech. Speech data selected includes from vowels to continuous speech, and the number of speakers ranged from two to ten, including males and females. Sometimes, other kinds of information (phonetic transcription, laryngograph signal ...) were also used. Only four published corpora will be referenced, due to their special characteristics.

Kain et al. [Kai01a] recorded a corpus trying to minimize the prosodic differences between speakers. The recording procedure, which was designed in order to have a natural time alignment between speakers, was as follows. First a template speaker recorded a set of sentences, then five males and five females mimicked the timing and the accentuation pattern (not pitch nor voice quality) of the template. This corpus is useful when the focus of the conversion is voice quality, since speaker prosodic patterns become equal during the recordings. This corpus has mainly two advantages. In the system training step, errors in the time alignment path are reduced due to the time differences between source and target sentences are minimal. In the evaluation system step,

prosodic characteristics do not influence in the ratings, since they are the same for all speakers. Only the fundamental frequency must be adjusted between source and target speakers. The main drawback of mimicked corpora is that the recordings are not natural speech, in the sense that the speaker is asked to utter a set of sentences in a specific way.

Specific corpora have been used for VC in the context of translation, where two different languages are involved. Therefore, parallel corpus for source and target speakers are not possible. Mashimo et al. [Mas01] recorded two sets of bilingual utterances, one set in English and another in Japanese, from bilingual speakers. Sündermann et al. [Sün03b] recorded different corpora for each language and each speaker, in particular for a male English speaker and a female German speaker. This kind of corpus requires non-parallel alignment techniques to estimate the mapping functions.

In the framework of the European funded project TC-STAR (Technology and Corpora for Speech to Speech Translation), specifications on large, high quality TTS databases have been developed and data have been recorded for UK English, Spanish and Mandarin [Bon06]. In particular, a VC specific corpus has been designed for Speech-to-Speech translation task by translating a set of sentences taken from the European parliament. The corpus was recorded by bilingual speakers, containing one hour of read speech in each language (English/ Spanish, English/Mandarin). The sentences were recordered by the mimic strategy, for its application to intra-lingual voice conversion systems.

2.4.3 Speech Features used in Voice Conversion Systems

Speaker discrimination, by the human point of view, relays on two types of information: acoustic information and linguistic information. The state of the art of VC systems includes only acoustic features in the mapping between two speakers. By the moment, no linguistic level features has been studied in the VC field.

As in speech coding, speech recognition or speech synthesis, segmental feature extraction for VC begins by dividing the speech signal into frames, usually overlapping frames. Then, each frame is represented by acoustic features, reducing the dimensionality of the speech samples while at the same time highlighting some particular aspects of the signal.

In the bibliography, three main groups of approaches can be found according to the segmental acoustic features selected for the conversion:

Features related to a phonetic/acoustic model of the speech such as formant frequencies, formant bandwidths, and glottal flow parameters [Nar95, GA98, Mor03, Ren03].

LP related features LP features are based on the source-filter model for speech production. Usually, the polynomial coefficients of the all-pole filter are derived to other parameters with better interpolation properties, such as: LSF (line spectral frequencies) [Kai01b, Ars99], LAR (log area ratios) [Iwa95], reflexion coefficients [Ver96] or LPC cepstrum.

Features without assuming any speech model such as spectral lines [Sün03a] or mel frequency cepstrum [Mas97, Mas01].

Systems based on features related to a phonetic/acoustic model try to change voice characteristics modifying specific aspects of the signal. On one hand, as the phonetic/acoustic parameters are related to the speech physical production, it can be assumed that these systems will be able to find specific conversion functions and transform the signal with a lot of detail. On the other hand, such parameters are difficult to estimate reliably.

Systems that use LP related features or features without assuming any signal model employ techniques for global optimization and control, instead of modifying each phonetic/acoustic parameter separately. LP related features are used frequently in VC. Estimation techniques of LP coefficients are more robust than other parametric feature estimations. Moreover, LP separates the vocal tract contribution from the excitation contribution, what allows to convert each signal component in a different way. The main drawback of LP systems is the need of training two different mappings: one for the vocal tract parameters and another for the LP residual signal.

The decision about which segmental acoustic features are selected will influence the final performance of VC systems. Therefore, it is desired to select features not only that capture the speaker identity, but also low dimensional features with good interpolation properties. Moreover, it is a requirement to have available a speech production technique based on the selected features, since the converted features have to be transformed to speech.

2.4.4 Speech Alignment Techniques

In order to learn the relationship between source and target acoustic segmental features, some training correspondences are needed. Although most of the approaches use a training corpus containing the same set of sentences uttered by both speakers, the correspondence at the frame level must be found. Several strategies has been used in already published VC systems.

Manual alignment has been used in some parametric systems. For example, in [Miz95] formants were manually aligned to increase the accuracy of the system. Although this is a method with good accuracy, it is time consuming and it is not useful for automatic training applications.

Sentence HMM has been also used in VC systems to align acoustic parameters, in particular LSF vectors. To align with sentence HMM [Ars99], template sentences are uttered by both source and target speakers. For each sentence, cepstrum coefficients are extracted along with log-energy and zero-crossing for each analysis frame. Based on the parameter vector sequences, sentence HMMs are trained for each template sentence using data from the source speaker. The number of states for each sentence HMM is proportional to the duration of the utterance. Next, the best state sequence for each utterance is estimated using the Viterbi algorithm. The average LSF vector for each state is calculated for both source and target speakers using frame vectors corresponding to that state index. Finally, this average LSFs vectors for each sentence are the aligned source-target vectors.

Forced-alignment speech recognition was also employed [Ars99] to align LSF vectors. This method assumes that the orthographic transcription of the data to be aligned is available. First, the data is segmented automatically using force alignment to a phonetic translation of the orthographic transcription. Next, a LSF centroid is estimated for each phoneme for both source and target speakers by averaging across all the corresponding speech frames. Finally, a one to one mapping is established between source and target centroids.

Dynamic time warping (DTW) is probably the most frequently adopted technique. DTW consists in finding a time path to minimize the spectral distance between frames of source and target speakers. This technique has been applied to a whole sentence [Abe88] or inside a diphone [Kai01b], when data is phonetically labeled.

Source-target phonetic class mapping is described in [Sün03b]. This technique is applied when non-parallel source and target corpora are used to train the conversion system, as in VC applied to language translation systems. The outline of the source-target phonetic class mapping is as follows. First, a segmentation of source and target speech into acoustic classes is carried out. In order to perform the segmentation, pitch synchronous frames of speech are DFT analyzed and all the available power spectrums are resampled to the same number of points and energy normalized. Then, a clustering (e.g. a k-means clustering) is performed to find k acoustic classes. Once source and target classes are determined, the most central class vector is found for each class. To map source and target classes, source central class vectors are transformed by means of Dynamic Frequency Warping (DFW). Then, these transformed vectors are compared to the target central class vectors. The i^{th} source class is mapped to the j^{th} target class if the j^{th} target central class vector is the most similar to the i^{th} transformed vector.

Acoustic features	Mapping codebooks Techniques based on a pre-clustering with non-overlapping classes Continuous probabilistic transform functions Speaker interpolation methods Artificial neural networks systems Hidden Markov Model related conversions
LP residual signal	Artificial neural networks systems Speaker Transformation Algorithm using Segmental Codebooks Residual Prediction
Prosodic features	Stochastic mapping for pitch contours

Table 2.3: Summary of mapping functions.

2.4.5 Mapping Functions for Acoustic Features

The objective of the mapping functions is to represent the relationship between acoustic features of two different speakers uttering the same text. Most of the techniques used in VC come from the fields of speaker recognition and speaker adaptation for automatic speech recognition systems.

Mapping functions are trained by estimating the correspondence between spectral features of the source speaker with aligned features of the target speaker. Usually, a previous acoustic classification is carried out to allow a phone-dependent spectral relationship between speakers.

Mapping functions for acoustic parameters may be grouped according to the acoustic classification performed and the mathematical form of the conversion function (see table 2.3). When working with acoustic parameters derived from a source-filter model of the speech, some residual mapping may be carried out too. And, in any case, prosodic parameters need to be mapped to have a complete VC system.

Mapping Codebooks

One of the first published VC systems was based on vector quantization and spectral mapping [Abe88]. According to this approach, source and target acoustic spaces are hard partitioned with non-overlapping classes. The clustering of the acoustic spaces is performed by estimating a vectorial codebook for each of them. Once the source codebook and the target codebook are estimated, a mapping codebook is built in order to perform the conversion.

A mapping codebook is a codebook that describes a mapping function between the vector spaces of two speakers. To build the codebook, two speakers utter the same set of sentences. Then, all the words are vector-quantized frame by frame using speaker-dependent codebooks and a source-target alignment is carried out. The codevector correspondence between two speakers is accumulated as histograms. Using each histogram as a weighting function, the mapping code-

book is defined as a linear combination of target speaker codevectors. In particular, a codevector $\mathbf{C}_i^{(map)}$ of the mapping codebook, corresponding to the codevector $\mathbf{C}_i^{(source)}$ of the source codebook, is generated with the following linear combination of target codevectors:

$$\mathbf{C}_i^{(map)} = \sum_{j=1}^{M} w_{ij} \mathbf{C}_j^{(target)}, \tag{2.1}$$

$$w_{ij} = \frac{h_{i,j}}{\sum_{k=1}^{M} h_{ik}}, \tag{2.2}$$

where $\mathbf{C}_i^{(target)}$ denotes the i^{th} codevector of the target codebook with dimension M. w_{ij} is the relative frequency of the correspondence between the codevector $\mathbf{C}_i^{(source)}$ and the codevector $\mathbf{C}_j^{(target)}$ in the training data, computed using h_{ij} which is defined as:

$$h_{ij} = \text{counts}(\mathbf{C}_i^{(source)}, \mathbf{C}_j^{(target)}). \tag{2.3}$$

In the transformation step, the source speech features are vector coded with the source speaker's codebook and decoded by means of the mapping codebook. Figure 2.6 is a block diagram of the mapping codebook approach during the transformation mode.

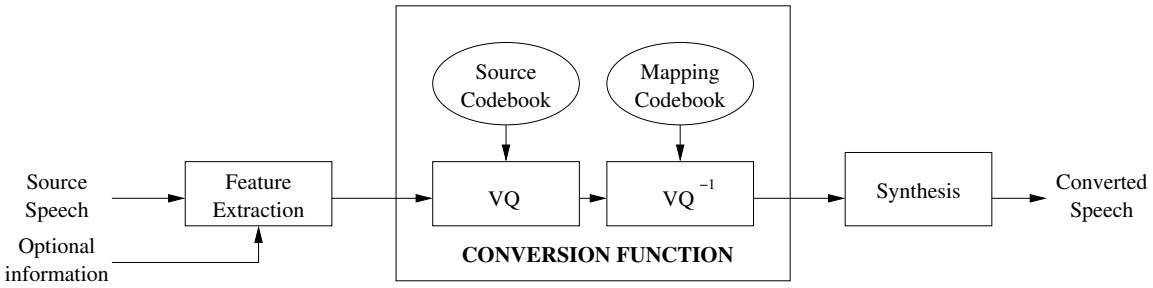


Figure 2.6: Block diagram of a mapping codebook VC system during the transformation mode.

Mapping codebooks were applied to spectrum parameters, power values and pitch frequencies. Some years latter, this approach was also applied to a parametric system [Miz95], where mapping codebooks were estimated for formant frequencies and spectral tilt.

Although mapping codebook systems produce converted voices resembled to the target voice, it was reported that the quality of the output was not as high as required in final applications. The main problem of mapping codebooks is the representation of all spectral realizations with a finite set of codevectors. This introduces spectral discontinuities in the converted speech.

In order to overcome the problem of quantization errors due to the limitation of the codebook size, fuzzy vector quantization was introduced by Shikano et al. [Shi91]. In the fuzzy VQ, vectors are represented not as one codevector but as a linear combination of several codevectors:

$$\hat{\mathbf{x}} = \sum_{n=1}^{M} \alpha_n \mathbf{C}_n^{source}, \tag{2.4}$$

$$\alpha_n = \frac{1}{d_n^{\left(\frac{1}{f-1}\right)}}, \tag{2.5}$$

$$d_n = \| \mathbf{x} - \mathbf{C}_n \|, \tag{2.6}$$

where $\hat{\mathbf{x}}$ denotes a decoded vector of an input vector $\mathbf{x}$ and f denotes fuzziness.

As in the previous case, the conversion is performed by replacing source speaker's codevectors by the mapping codevectors. The final expression for the converted vector $\hat{\mathbf{y}}$ is:

$$\hat{\mathbf{y}} = \sum_{n=0}^{M} \alpha_n \left(\sum_{j=1}^{M} w_n j \mathbf{C}_n^{target} \right). \tag{2.7}$$

Other techniques based on a pre-clustering with non-overlapping classes

After the mapping codebook approach, several techniques were published based on a partition of the acoustic space in non-overlapping classes, estimating a transformation function for each class.

Valbret et al. [Val92] applied two different conversion methods, taken from the speech recognition domain, to each acoustic class: Dynamic Frequency Warping (DFW) and Linear Multivariate Regression (LMR). DFW represents the correspondence between the source frequency axis and the target frequency axis by a warping function $\alpha(w)$, such as:

$$\mathbf{Y}(w) = \mathbf{X}(\alpha(w)), \tag{2.8}$$

were $\mathbf{Y}(w)$ and $\mathbf{X}(w)$ denote the target and source power spectrum, respectively. This function is calculated as the path that minimizes some selected distance between the source spectrum and the target spectrum. To determine the warping function for each acoustic class, all the optimal warping for the source-training spectrum pairs are estimated. The final warping function is defined as the median warping in each class. The main drawback of DFW is that only formant positions are moved, but their amplitudes can not be modified.

On the contrary, LMR modifies all the source spectrum shape to match the target spectrum. When using the LMR technique the mapping in each class is a linear transformation. If $\mathbf{C}^{s,q} = \{\mathbf{C}_j^{s,q}\}_{j=1}^{M_q}$ is the set of source spectrum vectors of the class q normalized by the mean vector of the class (M_q is the q^{th} class population), and $\mathbf{C}^{t,q} = \{\mathbf{C}_j^{t,q}\}_{j=1}^{M_q}$ is the corresponding aligned

target vector set, the linear regression transformation matrix $\mathbf{P}_q$ that minimizes the mean-square error between aligned source and target vectors is obtained by:

$$\mathbf{P}_q = \underset{\mathbf{P}}{\operatorname{argmin}} \sum_{k=1}^{M_q} \parallel \mathbf{C}_k^{t,q} - \mathbf{P}\mathbf{C}_k^{s,q} \parallel^2 \tag{2.9}$$

The solution of this minimization problem is straightforward:

$$\mathbf{P}_q = \mathbf{C}^{t,q}(\mathbf{C}^{s,q})^{\#} \tag{2.10}$$

where # indicates pseudo-inverse.

LMR may be interpreted as the search for the conditional expectation of the target spectral vector, knowing the aligned source vector, when the source-target joint distribution is Gaussian.

Recently, other researches [Sün03a] have extended the DFW systems working with Vocal Tract Length Normalization (VTLN) as a technique to warp the source spectrum by a determined parametric warping function. Several warping functions have been proposed, but the number of parameters is determinant when choosing the warping function due to estimation problems with few data. In particular, a linear warping function has been proposed in [Sün05d], since it has been reported that it performs similar to other more complex functions. As it has been already mentioned in DFW, VTLN moves only the formant positions, but their amplitudes can not be modified. Voice converted by VTLN resembled to a third speakers, nor the source nor the target speaker, while maintaining a high level of speech quality.

Another technique that comes from the speech recognition field, in particular from speaker adaptation, is VC using speaker selection and vector field smoothing [Has95]. To apply this approach multiple speakers have to be pre-recorded, and the most similar to the target speaker is selected. Then, a transfer vector is estimated to map the source acoustical feature space to the target space. The main drawback of this technique is the need of having multiple speakers recorded, making speaker selection and vector field smoothing not suitable for some situations.

A novel VC system was published by Arslan et al. [Ars99], called Speaker Transformation Algorithm using Segmental Codebooks (STASC). The first stage of STASC is the construction, during the training phase, of a parallel codebook for LSF parameters and a parallel codebook for residual frames. A parallel codebook refers to a pair of codebooks, one for the source speaker and another for the target speaker, with the same number of codevector and with the i^{th} source codevector related with the i^{th} target codevector. For instance, a parallel codebook can be estimated with one codevector per phoneme.

In the transformation phase, STASC system relays on the source-filter theory of the speech production to represent the spectrum of the source speaker as $\mathbf{X}(w) = \mathbf{G}_s(w)\mathbf{V}_s(w)$, where

$\mathbf{G}_s(w)$ and $\mathbf{V}_s(w)$ represent the source speaker excitation and the vocal tract transfer function, respectively for the incoming speech frame $\mathbf{x}(n)$. Based on this representation, the target speech spectrum $\mathbf{Y}(w)$ can be formulated as:

$$\mathbf{Y}(w) = \left[\frac{\mathbf{G}_t(w)}{\mathbf{G}_s(w)}\right]\left[\frac{\mathbf{V}_t(w)}{\mathbf{V}_s(w)}\right]\mathbf{X}(w), \tag{2.11}$$

where $\mathbf{G}_t(w)$ and $\mathbf{V}_t(w)$ represent codebook estimated target vocal tract and excitation spectra, respectively. This representation of the target spectrum can be thought of as an excitation filter $\mathbf{H}_g(w)$, followed by a vocal tract filter $\mathbf{H}_v(w)$:

$$\mathbf{Y}(w) = \mathbf{H}_g(w)\mathbf{H}_g(w)\mathbf{X}(w). \tag{2.12}$$

To perform the transformation, both $\mathbf{H}_g(w)$ and $\mathbf{H}_v(w)$ must be found. This filters are determined by a weighted combination of the codebook filters:

$$\mathbf{H}_v(w) = \sum_{i=1}^{L} v_i \frac{\mathbf{V}_i^t(w)}{\mathbf{V}_i^s(w)}, \tag{2.13}$$

$$\mathbf{H}_g(w) = \sum_{i=1}^{L} v_i \frac{\mathbf{U}_i^t(w)}{\mathbf{U}_i^s(w)}, \tag{2.14}$$

where $\mathbf{V}_i^t(w)$ and $\mathbf{V}_i^s(w)$ denote the vocal tract filters derived from the target and source LSF vectors for the i^{th} codeword, and $\mathbf{U}_i^t(w)$ and $\mathbf{U}_i^s(w)$ denote the average target and source excitation magnitude spectra for the i^{th} codeword, respectively.

To estimate the weights v_i, a perceptual weighted distance $\mathbf{D} = (d_1, d_2, ..., d_L)$ (L is the codebook size) between source LSF parameters and all the codevectors of the source LSF codebook is calculated. Based on the distance from each codebook entry, an approximated source LSF vector $\hat{\mathbf{x}}$ can be expressed as:

$$\hat{\mathbf{x}} = \sum_{i=1}^{L} v_i \mathbf{S}_i, \tag{2.15}$$

where $\mathbf{S}_i$ denotes the i^{th} source LSF codevector. The parameters v_i correspond to:

$$v_i = \frac{\mathbf{e}^{-\gamma d_i}}{\sum_{l=1}^{L} \mathbf{e}^{-\gamma d_l}} \qquad i = 1, ..., L. \tag{2.16}$$

The value of γ for each frame is found by an incremental search with the criterion of minimizing the perceptual weighted distance between the approximated LSF vector $\hat{\mathbf{x}}$ and the original vector $\mathbf{x}$. To further improve the estimation of γ a gradient descendant algorithm is also run. The

parameter γ can be regarded as information about the phonetic content of the current speech frame.

The weighted codebook representation of the target spectrum results in an expansion of the formant bandwidth due to the interpolation of the LSF. Therefore, a post-processing algorithm is applied to modify the bandwidth to be similar to the most likely target codeword.

Multiple extensions of this technique have been published. For example, STASC was also applied to sub-band based VC [Tur02, Tur03] for high sampling rate files.

It is difficult to compare the conversion results that have been published for the techniques exposed, since there is not a reference corpus and a common evaluation framework for all of them. Only DFW and LMR have been equitably compared, concluding that LMR performs better than DFW with respect to transforming voice quality, but producing some audible distortions. The LMR technique is the base of the following family of mapping functions, which intend to solve the problem of spectral discontinuities in the conversion.

Continuous Probabilistic Transform Functions

The step forward to resolve the problem of spectral jumps in VC was the introduction of continuous probabilistic transform functions [Sty98]. This method is based on the use of Gaussian Mixture Models (GMM) to model the probability distribution function of the source speaker spectral envelops. A GMM is a soft (probabilistic) partition of the space, opposite to early hard partitions. The source speaker is represented using Q Gaussian mixtures with $\mathbf{\Sigma}_q$ and μ_q as the means and covariance matrices. Using this representation, the transformation function $F(\mathbf{x})$ is a continuous parametric function that takes into account the probabilistic classification provided by the mixture model:

$$F(\mathbf{x}) = \sum_{q=0}^{Q-1} h_q(\mathbf{x})[\mathbf{v}_q + \mathbf{\Gamma}_q {\mathbf{\Sigma_q}}^{-1}(\mathbf{x} - \mu_\mathbf{q})], \tag{2.17}$$

where $h_q(\mathbf{x})$ is the posterior probability that the qth Gaussian component had generated the source frame $\mathbf{x}$. The parameters $\mathbf{\Gamma}_q$ and $\mathbf{v}_q$ are estimated by least squares, minimizing:

$$\varepsilon_{mse} = E[\| \mathbf{y} - F(\mathbf{x}) \|^2] \tag{2.18}$$

for $\mathbf{x}$ and $\mathbf{y}$ aligned training vectors from the source and target speakers.

Some years latter, it was proposed [Ye03] to include perceptual weights in the least squares minimization, to minimize a perceptual error instead of the squared error.

A review of different VC techniques was presented in [Bau96], and it was reported that the performance of a GMM system is as good as or better than mapping codebooks, linear regression and some approaches with artificial neural networks.

The GMM model was extended estimating the GMM on joint source-target data [Kai98a]. Taking into account the joint information, the components of the mixture are allocated depending on both source and target. The conversion function is estimated by regressing the joint GMM model:

$$F(\mathbf{x}) = E[\mathbf{y} \mid \mathbf{x}] = \sum_{q=0}^{Q-1} h_q(\mathbf{x})[\mu_q^{\mathbf{y}} + \mathbf{\Sigma}_q^{yx}\mathbf{\Sigma}_q^{xx^{-1}}(\mathbf{x} - \mu_q^x)]. \tag{2.19}$$

It is published [Kai98b], that the performance of least squares GMM and joint regression GMM are similar, but the last one requires less operations and less memory.

The reported quality of the converted speech for GMM based systems is higher than for mapping codebooks systems, because the spectral discontinuity is solved. However, it has been also reported that one of the problems of GMM based VC systems is that the converted spectrum is exceedingly smoothed, degrading the final speech quality. In order to improve the over-smoothing, several strategies have been proposed. The first technique proposed was a spectral enhancement by post-filtering, which consists in modifying the LPC filter coefficient values in order to reduce the formant bandwidth. This is a very used technique in speech coding [Boi04]. Toda et al. [Tod01a, Tod01b] proposed to mix the converted spectrum obtained from a GMM based algorithm with DFW, by a mixing weight frequency dependent. The performance of a DFW system is worse than a GMM system, because the spectral power cannot be converted. But the converted spectrum from a DFW system does not have the over-smoothing problem. According to the GMM/DFW solution, the converted spectrum $S_c(f)$ is estimated as:

$$\mid S_c(f) \mid = exp\left(w(f) \ln \mid S_G(f) \mid + (1 - w(f)) \ln \mid S_D(f) \mid\right) \qquad 0 \leq w(f) \leq 1, \tag{2.20}$$

where $S_G(f)$ and $S_D(f)$ denote the GMM-based converted spectrum and the DFW-based spectrum, respectively. The expression of the proposed mixing weight $w(f)$ is as follows:

$$w(f) = \left| \frac{2\pi f}{f_s} + 2\tan^{-1} \frac{a \sin(2\pi f / f_s)}{1 - a\cos(2\pi f / f_s)} \right| / \pi \qquad -1 \leq a \leq 1, -f_s/2 \leq f \leq f_s/2, \tag{2.21}$$

where f_s denoted the sampling frequency and a is the parameter which change the mixing weight and has to be tuned manually.

Another proposed solution to the over-smoothing was proposed in [Che03], which consist in ignoring the variance information in the regression function. At this moments, there is no concluding solution to the over-smoothing problem of GMM voice conversion systems.

Continuous Probabilistic Transform Functions are one of the most promising family of mapping functions for segmental acoustic parameters. Chapter 4 contains a deeper study of GMM properties and performance.

Artificial Neural Networks

Artificial Neural Networks (ANN) have been applied to VC [Nar95] in systems that use formants to represent the vocal tract features. The ANN technique is based on the property that a multilayered feed-forward neural network using non-linear processing elements can capture an arbitrary input-output mapping. This generalization property of ANN helps in the faithful transformation of formants across speakers, avoiding the use of large codebooks.

The training scheme of the referred conversion system based on ANN consists in a formant analysis phase, followed by a learning phase in which the implicit formant conversion for the first three formants is captured by a neural network. In the transformation phase, the three formants extracted from each frame of the source speaker's speech are given as input to the input layer of the trained ANN. The output of the network gives the converted formants. The converted formants together with the source pitch contour modified to suit the average pitch of the target speaker are used in a formant Vocoder to synthesize speech with the desired vocal tract system characteristics.

Both formants and pitch modifications introduce characteristics of the target speaker in the synthesized speech. However, several important characteristics of the target speaker are not incorporated in the transformation, specially the glottal pulse shape and prosodic features.

ANN have been also applied to transform LPC parameters. In particular, Watanabe et al. [Wat02] used Radial Basis Function (RBF) networks with Gaussian basis. RBF networks are usually three layer networks, containing Q radial functions $h_i(\mathbf{x})$ on the hidden layer. Each response of the radial functions is weighted by a connection vector $\mathbf{w_i} = [w_{i1} w_{i2} \dots w_{1Q}]^T$. The network outputs $\tilde{\mathbf{y}} = [\mathbf{y}_1 \mathbf{y}_2 \dots \mathbf{y}_p]^T$, where p is the vector dimension, are given as the linearly weighted sum of $h_i(\mathbf{x})$:

$$\mathbf{y}_k = \sum_{q=1}^{Q} h_q(\mathbf{x}) w_{kq} \qquad k = 1, 2, \dots, p. \tag{2.22}$$

Radial functions $h(\mathbf{x})$ are a special class of function. Their characteristic feature is that their response decreases (or increases) monotonically with distance from a central point. In particular, a Gaussian RBF monotonically decreases with distance from the center.

Although the mathematically basis of RBF with Gaussian basis and GMM are different, they have common properties and it is expected to have similar performances.

Hidden Markov Models related to Voice Conversion

The main motivation for the application of Hidden Markov Models (HMM) to the VC task is the introduction of dynamic information into the transformation. The former works of HMM

based VC [Kim97] used source and target HMM in order to segment the speech signal into states. Then, for each state a conversion function based on mapping codebooks was estimated. The publication [Mor03] presented a similar segmentation system based on HMM, but the transformation function associated to each state was based on Maximum Likelihood Linear Regression (MLRR). MLLR is a model adaptation technique that estimates a set of linear transformations for the mean and variance parameters of a Gaussian mixture HMM system. The effect of these transformations is to shift the component means and alter the variances in the initial system so that each state in the HMM system is more likely to generate the target data.

VC technology has been also used to modify the speech generated by a TTS system based on HMMs. Although actual HMM synthesis techniques [Mas96] do not produce the quality of unit selection systems, they provide flexibility for prosodic modifications that motivates their study. Since HMM synthesis uses phoneme HMM as speech units, the VC task consists in adapting each source HMM to target data. Masuko et al. [Mas97] proposed a combination of MAP (maximum a posteriori)/VFS(vector field smoothing) techniques, carried out sequentially, to adapt source phoneme HMM. MAP modifies the mean and covariances of the output distribution of HMM states, using adaptation data. It is effective to the problem of limited training data. Then, VFS is applied in order to interpolate new parameters of untrained distributions by MAP and to smooth estimated parameters of MAP trained distributions. Latter, Tamura et al. [Tam01] presented another strategy, maximum likelihood linear regression (MLRR), for the adaptation of a HMM based TTS. This strategy has been also applied to the problem of polyglot synthesis using a mixture of monolingual corpora [Mas05], where VC techniques are used to create an average polyglot speaker from different monolingual speakers.

HMM based VC systems will be detailed studied in Chapter 4, as an extension of GMM systems to include dynamic information.

Speaker Interpolation

Segmental acoustic features have been also transformed by interpolating multiple-speakers's acoustic patterns. The main motivation of speaker interpolation is the conversion to a target speaker with few training data (as few as one word), without loosing the dynamic characteristics of the speech. In [Iwa94] a speech spectrum transformation method was proposed by interpolating spectral parameters between a set of pre-stored speakers by a lineal function. The parameters of the interpolation function are estimated from a training set of sentences. Latter, in [Iwa95] the interpolation was carried out by the aid of Radial Basis Functions.

The main drawback of this approach is the necessity of having a set of speakers (four in this two references) pre-recorded.

2.4.6 LP Residual Signal Mapping

Many of the VC approaches use Linear Predictive Coding (LPC) parameters to perform vocal tract transformations. LPC is based on the source-filter speech signal model. Although ideally the residual LP signal is just a train of impulses for voiced frames or white noise for unvoiced frames, the truth is that it contains the glottal tract characteristics that have not been modeled by the linear prediction filter, and also all the factors that a minimum phase all pole filter can not model (such as nasalizations and the difference with the real phase). Synthesizing speech with real residual signal improve the quality of a simple Vocoder using ideal residuals. Even though the residual signal is not as influencing as LPC in speaker individuality, it contains information that can help to achieve the required conversion performance and quality. Therefore, some works have been published focusing in the residual signal modification task.

Based on the idea that speech signal contains non-linearities, mainly present in the residual signal, residuals have been modeled by a long-delay nonlinear predictor using a time-delay neural network [Lee96]. Once the predictor is estimated, a mapping codebook for neural net nonlinear predictors is built to transform the residual signal. It is reported that the naturalness of the converted speech increases when introducing the residual mapping, but some buzzy quality or click noises appear in regions with mixed voicing.

STASC [Ars99] deals with residual signals too. An excitation transformation filter is formulated for each codeword entry, using the excitation spectra of the source speaker and the target speaker, in the same way that the vocal tract conversion filter was built (see section 2.5.2).

A different approach from residual mapping was proposed by Kain et al. [Kai01a], where the residual signal is predicted from the vocal tract parameters, instead of transforming the source residual. The main assumption is that for a class of voiced speech the residuals signals of one pitch period are similar, and thus predictable from the LP parameters. Recently, The same prediction strategy has been adopted by other authors [Ye04, Sün05b].

2.4.7 Prosodic Conversion

Prosodic conversion refers to the transformation of the prosodic characteristic of a source speaker (mean and time evolution of the fundamental frequency, phoneme duration, loudness) to the prosodic characteristics of a target speaker. Prosodic conversion is out of the scope of this thesis dissertation. However, some relevant works will be referred in order to complete the state of the art of VC systems.

Prosodic conversion is the aspect less studied of VC systems. Few prosodic conversion systems have been published. On one hand, the most part of the works already presented for spectrum conversion only scales the pitch of the source speaker to resemble the target one, without dealing

with pitch evolutions nor phoneme durations. On the other hand, there are some that construct a prosodic conversion system similar to the spectrum mapping, such as STASC [Ars99, Tur03] or MLLR [Tam01].

A novel method to deal with pitch conversion was published by Ceyssens et al. [Cey02]. It consists in a stochastic system that transforms pitch contours taking into account multiple pitch parameters: pitch offset, pitch declination and variances according to the length of the utterances. The basic idea of this system is to model pitch evolutions of a phrase by a declination line plus a normal distribution to take into account the variation of the pitch around that line. During VC both the declination line and the derivations around it must be transformed independently. Pitch declination is converted in a deterministic way, by a linear function. Derivations are converted scaling the variance of the source normal distribution by the variance of the target normal distribution.

Recently, prosodic conversion has been studied in the framework of Speech-to-Speech translation, in order to improve the quality of the output prosody. In [Agü05], it is proposed the use of the intonation of the speaker of the source language to improve the quality of the intonation of the target language. To take into account the converted prosody, the following speech generation process is proposed. First, the prosodic features of the source speaker are estimated. Second, a prosodic mapping module performs the transformation of the estimated features in order to enrich the output of the translation module. Finally, the speech synthesis module produces the output waveform signal using prosody generated by the prosody generation module, which takes advantage of the enriched text.

2.4.8 Speech Production and Prosodic Modification Techniques

Methods of speech analysis and synthesis have evolved with the development of signal processing techniques. Despite the changes, the overall approach of analysis and synthesis has remained the same. First, the analysis step provides an alternative presentation of the speech signal. Acoustic and prosodic parameters derived from the alternative representation are the input of VC systems. Once the parameters have been converted, the new parameters can be used to synthesize a transformed voice.

The most simple analysis/synthesis system used in VC was a LPC Vocoder, with either impulse train or white noise as excitation. The prosodic modifications in this synthesis system are straightforward. Pitch information is generated artificially when producing the excitation signal, phoneme duration and speech rate is related to the time instants that the filter coefficients are changed. The Vocoder system was extended to a source-filter synthesis system, employing Overlap-and-Add (OLA) techniques to carry out prosodic modifications, such as TD-PSOLA or LP-PSOLA.

A speech signal model more elaborated also used in VC, that allows high quality modifications of the speech, is HNM (Harmonic+Noise Model) [Sty98]. HNM performs a pitch-synchronous harmonic+noise decomposition of the speech. In voiced sounds, the spectrum is divided into a harmonic and a noise part by a parameter called the maximum voiced frequency. Lower this frequency, speech spectrum is modeled as a sum of harmonic sine waves. Upper it, a noise component modulated by a time-domain amplitude envelope is estimated. The harmonic part is synthesized directly in the time domain as a sum of harmonics. The fundamental frequency of this harmonic signal is constant over the duration of the synthesis frame, whereas the amplitudes and duration of the harmonics are linearly interpolated between two successive frames. The noise part is obtained by filtering a unit-variance white Gaussian noise through an all-pole filter. If the frame is voiced, the noise part is filtered by a high pass filter with cut off frequency equal to the maximum voiced frequency.

Kain et al. [Kai01b] utilized a Harmonic model, a HNM simplified where all the frequency band is considered harmonic, in a VC system, using OLA for pitch and time-scale modifications.

Another alternative for speech analysis-synthesis is the STRAIGHT Vocoder (Speech Transformation and Representation using Adaptative Interpolation of weiGTHed spectrum) [Tod00]. STRAIGHT is a high quality analysis-synthesis method, which extracts fundamental frequency by using TEMPO (Time-domain excitation extractor using Minimum Perturbation Operator). This Vocoder uses pitch-adaptative spectral analysis combined with a surface reconstruction method in the time-frequency region in order to remove signal periodicity, and designs excitation source based on phase manipulation. STRAIGHT can manipulate such speech parameters as pitch, vocal tract length, and speaking rate, while maintaining high reproductive quality.

2.5 Summary

Speech sounds are produced by a human physiological process which involves the lungs, the larynx and the vocal tract. The inventory of human sounds can be described in terms of phonemes.

There are several parameters in the speech signal which contribute to the speaker variability. These parameters may be classified in acoustic cues (segmental and suprasegmental) and linguistic cues. Influencing segmental cues are the pitch, the glottal source spectrum and the vocal tract frequency response, while influencing suprasegmental cues are the fundamental frequency time evolution, the speech rate and the speech intensity. One of the main problems of VC is to find a way of representing the speaker individuality with a reduced number of parameters which are suitable to the conversion task.

This chapter have overviewed the most relevant works of the VC task, from the most former studies (mapping codebooks) to the state of the art systems (joint GMM regression plus residual

prediction). The next chapter will provide the framework for the systems developed in this dissertation.

Chapter 3

Voice Conversion System Framework

This chapter provides an overview of the common framework of all the studied VC systems in the dissertation. The outline of the chapter is as follows. First, training and evaluation corpora are described in section 3.1. Two different corpora have been used. The first corpus consisted in an already recorded database for TTS purposes. A second corpus was recordered during this thesis realization, which was specially designed for the VC task. Then, section 3.2 is dedicated to the speech analysis technique. In particular, LPC analysis-synthesis technique, with pitch-synchronous frames, was employed. In section 3.3 a lineal intra-phoneme alignment procedure is described and in section 3.4 the speech production and prosodic modification system selected are determined. Finally, several VC evaluations, both objective and perceptual, are reviewed in section 3.5.

3.1 Training and Evaluation Corpora

A speech corpus is a database that contains speech data files and complementary files with information related to the speakers, the speech, the recording conditions and the contents of the recordings. All the information needed to train and evaluate a VC system must be present in the corpus.

Two different corpora are used in the current work. In a first step, an already existing corpus, generated for a Spanish unit selection TTS system is used. This corpus consists in two professional speakers, one male and one female, uttering sentences of about 9 words. Speech and laryngograph signals were recorded in an acoustically isolated room, with a sampling frequency of 32kHz and 16 bits per sample. For this study, signals were decimated to 16kHz by software. Additionally, information about the segmentation into phonemes and pitch marks, both manually revised, is available. The total corpus size is more than one hour for each speaker.

Due to the particularities of VC systems and the procedure to evaluate them, a new corpus

was designed and recorded. The first requirement of the corpus was to select the text material assuring maximum phoneme and diphone coverage, in order to have well represented the characteristics of each sound. Therefore, the text material was extracted from the already existing corpus for the TTS, grouped in three sets: the training set, the validation set and the evaluation set. The sentences of each set were selected trying to include at least one apparition for each phoneme and one for each diphone. The software tool used to select the sentences was CorpusCrt v1.03, the corpus balancing tool of the Universitat Politècnica de Catalunya (UPC).

The amount of data of the corpus must be large enough to build a VC system and to evaluate it. Therefore, it was decided to record 20 sentences for the training set, 10 sentences for the validation set and 20 sentences for the evaluation set. The mean length of each sentence is nine words. Four non-professional Spanish native speakers, 2 males and 2 females, were recorded.

Another requirement for the VC corpus was a natural time-alignment and prosody evolution between identical sentences uttered by different speakers. This was achieve by the mimicking approach [Kai01a]. According to this approach, speakers tried to follow the timing and the accentuation pattern as well as the pitch contour of given template utterances while they were uttering the recordings. In the current VC corpus the selected template speaker was the female speaker of the TTS corpus.

The procedure to record the mimic corpus consisted in playing twice the template sentence to the new speaker. Each production of the sentence was preceded by three "beeps", in a way that the template sentence started at the same time than a four silent beep. During the first playback the new speaker was asked to listen to the template, while during the second playback the speaker uttered the sentence at the same time that the template. Then, another three "beeps" were played and the speaker uttered the sentence trying to mimic the template. Only this last utterance was recorded. All the sentences were showed in a screen to the new speaker as a prompt. The recordings were controlled by a technical person, i.e. the author, and they were repeated when an error occurs.

The main reason to choose a mimicking recording technique is for the evaluation of the proposed systems. Prosodic conversion between speakers is out of the scope of this thesis. Therefore, it will be easier to evaluate the VC performance if both source and target speaker have a similar timing and fundamental frequency evolution. With this corpus, listeners will not be influenced by the prosody when discriminating speakers.

The corpus was recorded following the recording specifications produced in the TC-STAR corpus [Bon06]. The recording place was an isolated room, whose reverberation time[1] was $RT60 < 0.3s$. The Signal to Noise ratio of the recorded speech was greater than 40dB.

[1]The reverberation time of a room is the time it takes for sound at 1kHz to decay by 60dB once the source of sound has stopped.

Speakers	
speaker profile	non-professional native Spanish
number of speakers	2 males and 2 females
Contents	
language	Spanish
domain	general sentences
text structure	sentences about 9 words
phonetic coverage	trying to include at least one apparition for each phoneme and one for each diphone
number of sentences	50
speaking style	neutral with mimic templates
Recording setup	
acoustical environment	isolated room
input devices	2 microphones (a large membrane and a close-talk) and 1 laryngograph
amplifier	RME OctaMic D
recording device	RME HDSP 9652 Audio Card
recording software	NannyRecord v2.0 (UPC)
Technical specifications	
sampling frequency	96 kHz
bit number	24
number of channels	3
speech file format	raw
Post-processing	
re-sampling	16 kHz
bit number	16
additional files	phonetic segmentation pitch marks

Table 3.1: Voice Conversion corpus specifications.

The recordings were carried out with three input devices: one large membrane microphone, one close-talk microphone and one laryngograph. The large membrane microphone recorded the signal for the final voices used in the VC systems. The distance from the microphone to the speaker should have been 60 cm or 30 cm with wind screen. The laryngograph signal served for the detection of pitch pulses (detect the start of the glottal closure). Also, a close-talk microphone, head mounted with a fixed distance of about 7 cm to the right of the mid-sagittal plane at the height of the upper lip, was used.

All signals were sampled synchronously with a sampling rate of 96 kHz, 24 bit per sample with the least significant byte first as (signed) integers. The bandwidth of the speech signal was at least 40 Hz - 20.000 Hz. Speech was stored in raw speech files.

For each utterance (speech file), the VC corpus provides: the prompt text used to record the utterance, the orthographic annotation, the phonetic transcription, the segmentation into phonemes and the pitch marks, associated with the glottal closure. The segmentation was carried out automatically. For each phoneme, the starting and ending time is provided. All the events (start and end positions) are indicated in seconds.

A two steps technique was employed to estimate the pitch marks. First, the laryngograph signal was differentiated to find out the negative peaks corresponding to the glottal closure instant. Second, the large membrane microphone signal was filtered by a low-pass filter. In order to synchronize the pitch marks with the speech signal, the estimated glottal closure instants were delayed to match with the instants of minimum energy prior to the speech signal peaks.

The experiments of the following chapters use speech sampled at 16kHz. Therefore, a post-resamplig was applied by software, and the resolution was changed to 16 bits per sample. Table 3.1 in page 37 resumes the characteristics of the mimicked corpus.

3.2 Speech Analysis

The speech analysis method selected in the current work is based on linear predictive coding, also known as LPC analysis or auto-regressive modeling. LPC analysis is fast and simple. It is also effective to estimate the main parameters of the speech signal. LPC model considers the speech signal $s(n)$ as the output of a system $H(z)$ whose input is an excitation signal $e(n)$, related to the physical glottal flow (see figure 3.1).

The filter $H(z)$ is an all-pole filter, with enough number of poles to be a good approximation of speech signals. Thus, the expression for $H(z)$ is:

$$H(z) = \frac{S(z)}{E(z)} = \frac{1}{A(z)} = \frac{1}{1 - \sum_{k=1}^{p} a_k z^{-k}}, \tag{3.1}$$

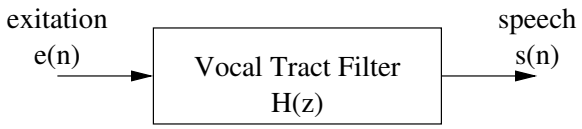


Figure 3.1: Speech model diagram.

where p is the order of the LPC analysis. In the current work an order 20 was selected for all the experiments. The inverse filter $A(z)$ is defined as:

$$A(z) = 1 - \sum_{k=1}^{p} a_k z^{-k}. \tag{3.2}$$

The filter coefficients a_k are called the LP (linear prediction) coefficients. According to this model, the current sample can be predicted as a linear combination of its past p samples:

$$\hat{s}(n) = \sum_{k=1}^{p} a_k s(n-k) \tag{3.3}$$

The results of the LP analysis are: the polynomial coefficients a_k and the error signal $e(n)$, also called LP residual signal. The error signal is the difference between the input speech and the estimated speech:

$$e(n) = s(n) - \hat{s}(n) = x(n) - \sum_{k=1}^{p} a_k s(n-k). \tag{3.4}$$

The autocorrelation method was used to estimate the LP coefficients, which relays on the Levinson-Durbin algorithm. The autocorrelation method always leads to a stable vocal tract filter $H(z)$, due to the minimum-phase of $A(z)$ assumption. Two operations should be carried out to the speech signal before the estimation of LP coefficients: a windowing operation and a pre-emphasis filtering.

Speech signal is a random variable signal. Hence, speech characteristics varied over the time. A short term analysis of the signal assumes that the signal remains unchanged over short periods of time. To carry out a short term analysis the signal is divided into segments, also called frames, and each segment is analyzed by itself. The operation of multiplying a speech signal $s(n)$ by a analysis window $w(n)$ in order to extract a particular segment is called windowing. The analysis window used in this work is an asymmetrical Hamming window (see figure 3.2), consisting of two halves of a raised cosines function of a pitch period length:

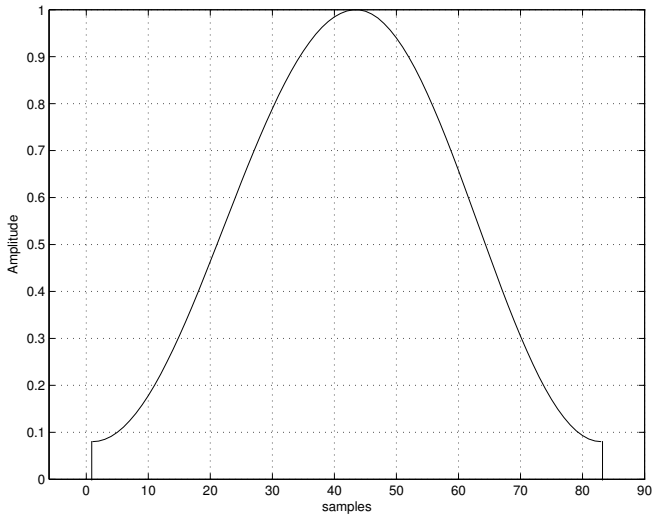


Figure 3.2: Asymmetrical Hamming window for $N_r = 40$ and d $N_l = 43$.

$$w(n) = \begin{cases} 0.54 - 0.46\cos\left(\frac{2\pi n}{N_r - 1}\right) & 0 \leq n \leq \text{floor}\left(\frac{N_r-1}{2}\right) \\ 0.54 - 0.46\cos\left(\frac{2\pi(n+\text{floor}(\frac{N_l-1}{2}) - \text{floor}(\frac{N_r-1}{2}))}{N_l - 1}\right) & \text{floor}\left(\frac{N_r-1}{2}\right) + 1 \leq n \leq \text{floor}(\frac{N_l-1}{2}\Big) \\ 0 & \text{otherwise} \end{cases} \tag{3.5}$$

where N_r and d N_l are the right and left half lengths.

Pre-emphasis filtering is an habitual practice in LPC analysis. This filter emphasizes the high frequencies, improving the numerical stability of further estimations. In this work, a fix pre-emphasis filter has been used $H(z) = 1 - 0.9375z^{-1}$.

An alternative representation of the LP coefficients more adequated to the VC task are Line Spectral Frequencies (LSF). LSFs are derived from computing the roots of the $(p+1)^{th}$ order symmetric and antisymmetric polynomials $P(z)$ and $Q(z)$ defined as:

$$P(z) = A(z) + z^{-(p+1)}A(z^{-1}), \tag{3.6}$$

$$Q(z) = A(z) - z^{-(p+1)}A(z^{-1}). \tag{3.7}$$

Three main properties of $P(z)$ and $Q(z)$ are [Soo84]:

- The roots of $P(z)$ and $Q(z)$ lie on the unit circle.
- Once the roots are sorted, the roots of $P(z)$ and $Q(z)$ are distinct and alternate.
- There are exactly two roots at +1 and -1.

The angles of the complex conjugate roots of $P(z)$ and $Q(z)$ are the Line Spectral Frequencies. A general procedure for computing the LSF values is as follows:

1. Compute $P(z)$ and $Q(z)$ and eliminate the roots at +1 and -1.
2. Project the roots onto the real axis, resulting in real polynomials whose roots are real, distinct, and lie in the range -1 to +1.
3. Find the roots of these polynomials derived from $P(z)$ and $Q(z)$.
4. Take the inverse cosine of each root, to get the angle of the corresponding LSF.

Published techniques for computing line spectral frequencies generally avoid root finding methods in step 3 because of concerns about convergence and complexity. Root finding methods are usually iterative procedures which are subject to convergence problems, are sensitive to roundoff errors, and have unpredictable processing delays. Exhaustive searches are a typical alternative to root finding algorithms.

However, the method used in the current work avoid convergence problems taking advantage of the special structure of the polynomials that must be solved. In particular, a version of the Decimation-in-Degree (DID) transformation [Rot99b] has been used to express P(z) and Q(z) as a function of Chebyshev polynomials and the root finding algorithm is an accelerated version of the Newton method [Rot99a].

LSFs are related closely to formant frequencies, but in contrast LSFs can be estimated more reliable. Usually, a formant is surrounded by LSFs and its bandwidth is dependent on the closeness of the corresponding LSFs. Figure 3.4 in page 43 illustrates the relationship between formant frequencies and LSFs.

The most important characteristic of LSFs for the VC task is their sensitivity: a perturbation of one coefficient generally results in a spectral change only around that coefficient. This property is illustrated in figure 3.5 in page 43. The solid line and dash-dot line envelope representations only differs in the third LSF value.

Moreover, LSFs have a fix range $[0, \pi]$, which makes them attractive for real-time DSP implementations and the filter stability condition is simply checked. The minimum phase property of A(z), i.e. the filter stability, is preserved if all the frequencies are different, ordered and between the range $[0, \pi]$ after any modification of LSF parameters.

3.3 Speech Alignment

The speech alignment technique adopted in the current system framework is a lineal alignment using phoneme boundaries as anchor points. Both source and target frame repetitions were

allowed. Figure 3.3 illustrates the alignment procedure.

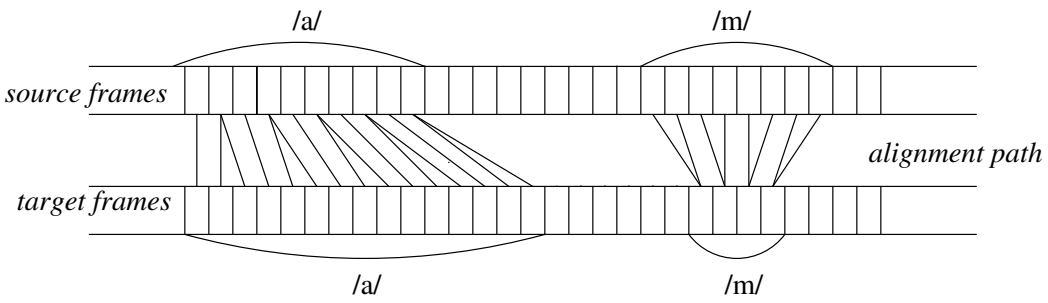


Figure 3.3: Frame alignment representation.

In an initial development step of the thesis, DTW was also used as an alignment technique, but lineal alignment using phoneme boundaries as anchor points has provided better training data for VC systems.

3.4 Speech Production and Prosodic Modifications

The speech production system of the current work relays on the source-filter model of the speech production (see figure 3.1 in page 39). The analysis of the speech by means of LPC produces an error signal or LP residual signal. The speech production module takes the residual signal as an input. The input is filtered by the synthesis filter 1/A(z) generating the output speech signal $\tilde{s}(n)$:

$$\tilde{s}(n) = \sum_{k=1}^{p} a_k \tilde{s}(n-k) + e(n). \tag{3.8}$$

When a pre-emphasis filtering technique has been used in the analysis, the equivalent de-emphasis filtering must be carried out to generate the final $\tilde{s}(n)$.

If $e(n)$ is different from the residual signal of the analysis or the synthesis filter is not the inverse of the analysis filter, the synthesized speech signal $\tilde{s}(n)$ will not be the same than the original signal $s(n)$. In VC systems the synthesis filter is derived from the mapped acoustic features and the residual signal of the analysis is modified to better match the target characteristics.

The objective of prosodic modifications is to change the amplitude, duration and pitch of a speech segment. Amplitude modification is a straightforward change by direct multiplications of the speech by a desired factor, but duration and pitch changes are more elaborated procedures. The prosodic modification method adopted in the current work is Time Domain Pitch Synchronous Overlap and Add (TD-PSOLA). PSOLA [Mou90] allows prosodic modifications

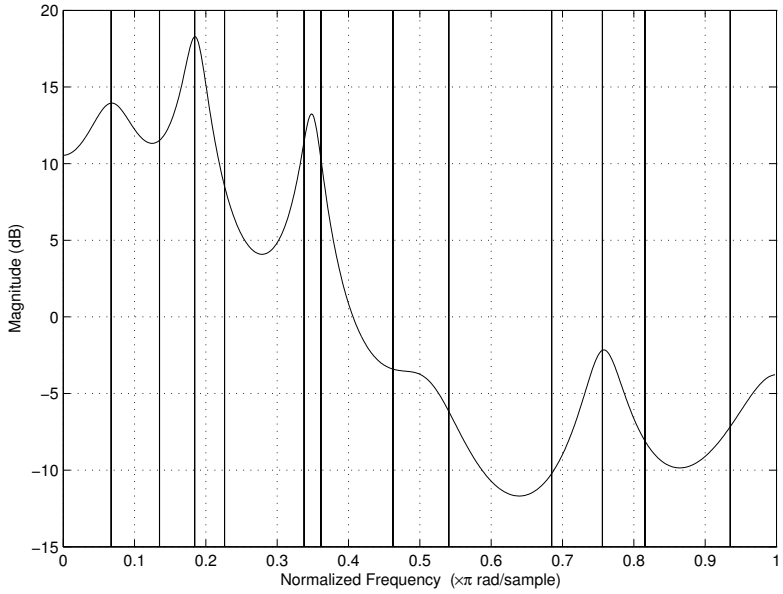


Figure 3.4: Representation of the spectral envelope and the LSF parameters (vertical lines) for a realization of the vowel /a/, for a filter of order 12.

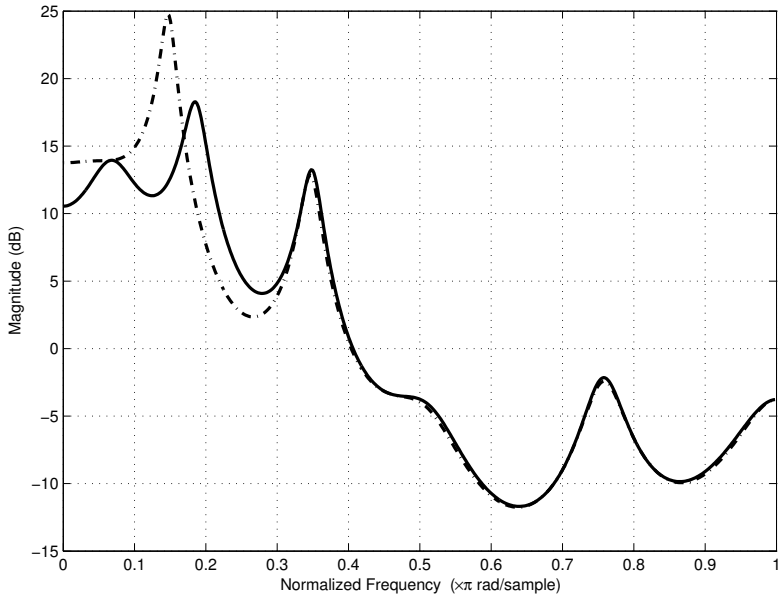


Figure 3.5: Two spectral envelopes differing in one LSF value.

without assuming any speech model. It is based on the periodicity of the speech signal. In the analysis step, the signal is divided into overlapping frames, usually of two pitch periods length, multiplying each frame by a window to attenuate the border samples. To reconstruct the signal, frames are added. Repetitions or deletions of frames are allowed if it is desired to change the duration of the signal. To changed the F0 value, frames must be move closer or separate before the addition. Figure 3.6 illustrates a duration an a pitch modification.

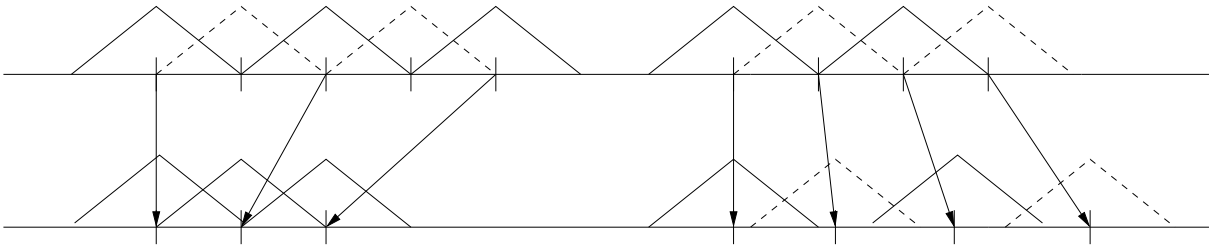


Figure 3.6: Two prosodic modifications by TD-PSOLA: a time reduction on the left and a F0 lowering on the right.

TD-PSOLA allows prosodic modifications with high quality. However, some concatenation problems may arise because the assumption of perfect periodicity is not completely true in natural speech.

3.5 Voice Conversion Evaluation

VC systems are evaluated in two different aspects: the degree of the personality change of the converted speech and the final sound quality. Moreover, every aspect can be evaluated by objective tests and perceptual tests. The first group measures the ability of the system to convert acoustic features in a determined mapping. Usually, objective tests are distance measures between target and transformed features defined in the feature domain or performance indices that take also into account the original distance between source and target features.

Although objective tests are useful in the developing step, the correlation between objective tests and performance is not clearly defined. Therefore, as it happens in TTS and speech and audio coding, the final evaluation of conversion systems must be perceptual. In order to prevent the evaluators from interpreting their decisions, all the listeners of the perceptual tests should not be familiar to the background of the test nor be familiar with the speaker voices. In particular, they must not know the contents of the evaluation plan.

This section describes several tests and strategies for the VC evaluation carried out in the current work. They are similar to the proposed evaluation in the integrated European project TC-STAR [Sün05a].

Vocal Tract Distances

In the transformation of vocal tract parameters three different distances are of particular interest: the source-target distance, the converted-target distance and the converted-source distance. Vocal tract distances between LSF vectors are defined as:

$$D(\mathscr{X}, \mathscr{Y}) = \frac{1}{N} \sum_{n=0}^{N-1} IHMD(\mathbf{x}_n, \mathbf{y}_n), \tag{3.9}$$

where N is the number of vocal tract frames $\mathbf{x}$ and $\mathbf{y}$. The function $IHMD$ indicates the Inverse Harmonic Mean Distance [Lar91], whose expression is:

$$IHMD(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{p=0}^{P-1} c(p)(\mathbf{x}(p) - \mathbf{y}(p))^2}, \tag{3.10}$$

where the weight $c(p)$ is defined as:

$$c(p) = \text{argmax}\{c_x(p), c_y(p)\}, \tag{3.11}$$

$$c_x(p) = \frac{1}{x(p) - x(p-1)} + \frac{1}{x(p+1) - x(p)} \tag{3.12}$$

and

$$c_y(p) = \frac{1}{y(p) - y(p-1)} + \frac{1}{y(p+1) - y(p)}, \tag{3.13}$$

with $w(0) = 0$ and $w(P+1) = \pi$ (p is the vector dimension).

The $IHMD$ introduces perceptual information in the distance, since the mismatch in spectral picks is more weighted than the mismatch in spectral valleys.

Vocal Tract Mapping Performance Index

When transforming a collection of source LSF vectors $\mathscr{X}$ into a collection of target LSF vectors $\mathscr{Y}$ the performance index of the conversion can be expressed as:

$$P = 1 - \frac{IHMD(\tilde{\mathscr{Y}}, \mathscr{Y})}{IHMD(\mathscr{X}, \mathscr{Y})} \tag{3.14}$$

where $\tilde{\mathscr{Y}}$ denotes the set of converted vectors. It can be observed that a conversion system that doesn't change the source speech ($\tilde{\mathscr{Y}} = \mathscr{X}$) will led to $P = 0$. A value of $P = 0.4$ means that the distance between the source and target speakers has been reduced about 40% when the conversion is performed.

The maximum value of P is one, but it is not expected in the experiment results despite the correct performance of the system. A speaker may utter the same phonetic content in many

different ways. Therefore, the performance index will not be one although it had been measured on real speech from two different realizations of the same sentence for the same speaker.

This performance index is useful for the comparison between different speaker combination conversions, because of its normalization to the initial source-target distance.

Extended ABX Test

The extended ABX test is a subjective test where listeners are presented a list of ABX questions. Each question consists in three speech files, where A is source or target speaker, B is the other one and X is one of the evaluation files. The listener must rate the voice identity of the X file as follows:

1	2	3	4	5
X is speaker A	X is similar to speaker A	X is neither A nor B	X is similar to speaker B	X is speaker B

This extended version of the ABX test does not force the listener to make a decision between A nor B. The listener may report that the transformed speech does not resemble neither the source nor the target speaker.

In the experiments reported in this dissertation, each speech file consists in one sentence randomly chosen from each one of the speakers. Listeners are not forced to listen to the complete file; they can stop the play whenever they want. Source, target and converted sentences are different, because the prosody do not have any influence in the personality decision. The same test is evaluated by 20 listeners. All the tests have been balanced designed, as for the number of different conversion systems tested, as for the different source-target speakers pairs as for the order of the speech files of each question.

ABX test is a direct way of evaluating conversion systems and very useful to compare different works because its use is very spread. However, the ABX test contains information that can help the listener to decide to rate the system in a favorable way. From the listener's point of view, it is easy to rate two speech files as being similar if there is a third speech file very different from both files.

Similarity Test

A set of pairs of speech files are presented and listeners are asked to decided if they correspond to the same speaker or to two different speakers. One of the files of every pair is either the source or target speaker, and the other one corresponds to the converted voice. Some pairs of source

and target files, in any order, are also added. In any case, two different sentences are selected for every file of the pair.

The converted speech files of the similarity test are a pitch modified version of the extended ABX test speech files. The converted files are modified in order to have the same mean pitch that the speaker with which is being compared. The pitch modification is carried out by the TD-PSOLA technique.

All the evaluated similarity tests have been balanced designed, in the same way that the extended ABX tests.

MOS for speech quality

VC systems need to be evaluated of two different perceptual aspects: the voice identity and the speech quality. Usually, there is a trade off between both aspects when designing a transformation function. Extended ABX test and Similarity test are used to evaluate the voice identity. A MOS test is used to evaluate the speech quality.

The Mean Opinion Score (MOS) test is a widely used test in telecommunications for measuring the speech quality. During this test, listeners are asked to rate a speech file according to its quality: (1) bad; (2) poor; (3) fair; (4) good; (5) excellent. The mean opinion score is the arithmetic mean of all individual scores of the same speech class. In particular, MOS tests will be carried out for natural and converted speech.

3.6 Summary

This chapter has provided an overview of the framework that will be shared for all systems developed in the following chapters. In particular, the corpora, the speech analysis and synthesis technique, the frame alignment between source and target speakers and the prosodic modification system have been explained. Additionally, several objective and perceptual test have been described.

Two different corpora are used to train and evaluate the VC systems developed in this dissertation. The first corpus consists in data of a male and a female speakers, recordered for a TTS. A second corpus was designed specially for the VC task, and two males and two females were mimicked recordered. The mimic strategy allows a natural time-alignment and prosody evolution between identical sentences uttered by different speakers. This is a desired recording method when studying VC systems without dealing with prosodic conversion.

The corpora speech is parametrized by LP coefficients plus LP residual signal, estimated from a pitch-synchronous LPC analysis. The polynomial coefficients are transformed to LSF

parameters, due to LSF parameters are more suitable to the conversion task. Therefore, VC systems developed in this dissertation will be organized in two processes. On one hand, LSF parameters will be converted by a vocal tract mapping system. On the other hand, the LP residual signal will be modified in order to better match the target residual.

Before the training of the mappings, an alignment between source and target frames must be carried out. The employed alignment is a lineal interpolation using phoneme boundaries as anchor points. Prosodic modifications of the re-synthesized signal are attained by means of TD-PSOLA.

Both objective and perceptual test are of interest when evaluating a VC conversion systems. The objective test are focused on the system performance to change the speaker voice personality, whereas perceptual test are also used to evaluate the quality of the converted speech.

Next two chapters will explore several techniques to perform the mapping of the vocal tract parameters and the modification of the LP residual signal.

Chapter 4

Vocal Tract Conversion

This chapter is dedicated to vocal tract conversion. First, the basic ideas about what is desired in systems dealing with vocal tract conversion are detailed, and two selected methods to convert the characteristics of the spectral envelope are presented. In particular, in section 4.2 Gaussian Mixture Models are presented as a mathematically base of state of the art conversion systems, which are detailed in section 4.3.

Afterwards, extensions and new approaches studied in this thesis dissertation will be explained. In section 4.4, the estimation algorithm for GMMs is modified to include non-parallel source data. In section 4.5 Hidden Markov Models are presented as an extension of GMMs to include dynamic information to Voice Conversion systems. In section 4.6 decision tree systems, where the strategy of soft classification is not applied, are studied.

Finally, results of objective and perceptual test are discussed in section 4.7.

4.1 Introduction to Vocal Tract Conversion

All the methods that deal with vocal tract conversion are based on the idea that each speaker has his/her own way of uttering a specific phone, so the relationship between different speakers should depend on what they are uttering in every moment. Therefore, the spectral mapping function has to take into account some phonetic/acoustic information in order to choose the most appropriate relationship for each speech frame to be converted. Generally, the conversion process can be divided in three stages: a model of the acoustic space with a structure by classes, an acoustic classification machine and a mapping function (see figure 4.1).

The main characteristic of all the approaches studied in this chapter is that they perform a continuous transformation of acoustic features from a soft classification, mathematically based on a Gaussian Mixture Model. The advantages that these methods present above other state of the art systems are:

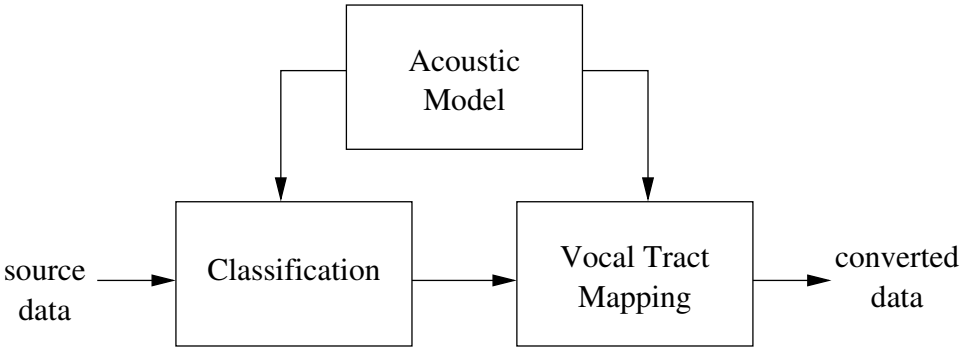


Figure 4.1: Vocal tract conversion block diagram.

Continuous transformations opposite to discrete transformations [Abe88] where each acoustic class is represented with only one spectral vector, reducing all the acoustic space to some discrete points. Therefore, with continuous transformations the output of the transformation can be positioned in any region of the acoustic space, obtaining a converted voice more rich and more natural.

Soft acoustic classification. The use of classification indices, instead of hard classification, improves the final synthesis quality. In VC systems the classification is carried out on the acoustic space. Due to the limited set of acoustic training data, it is not possible to model each class perfectly. Also, in continuous speech there are transitions between classes because of co-articulation. Therefore, if hard classification is used it will result in synthesis artifacts when a class change occurs. It is expected that working with soft classification the artifacts generated by unnatural discontinuities in the transformation, which typically occurs in the VQ model when a vectors jumps from one class to the other, will be avoided.

Gaussian Mixture Models (GMM) are probabilistic models that may represent the acoustic space of one or more speakers with a set of overlapping classes, by non-supervised training. These acoustic classes are somehow related to phonetic events. The description of each acoustic class is performed by a normal distribution: a mean vector representing the mean value of the class, and a covariance matrix representing the shape of the dispersion of the vectors around the mean. Thanks to their structure, once a GMM is estimated any acoustic realization can be classified applying Bayes' rule. The classification is probabilistic and a continuous function of the spectra parameters.

The mapping function will be based on the GMM that model the acoustic space, taking advantage not only of the soft classification, but also of the complete description of the acoustic classes. Therefore, the model as well as the transformation function are continuous and complete.

4.2 Definition of Gaussian Mixture Models

4.2.1 GMMs as Probability Density Functions

A GMM [Dud01] is a probability density function built as a weighted sum of Q Gaussian components given by the equation:

$$p(\mathbf{x};\theta) = \sum_{q=0}^{Q-1} \alpha_q \mathcal{N}(\mathbf{x};\theta_q), \tag{4.1}$$

where $\mathbf{x} = [x_0 x_1 ... x_{p-1}]^T$ is a p-dimensional random vector, $\mathcal{N}_q(\mathbf{x};\theta_q)$ are the component densities and α_q the mixture weights. Each component is a p-dimensional Gaussian function:

$$\mathcal{N}_q(\mathbf{x} \mid \theta_q) = \frac{1}{(2\pi)^{p/2}} \mid \mathbf{\Sigma}_q \mid^{-1/2} \mathbf{e}^{\left[-\frac{1}{2}(\mathbf{x}-\mu_q)^T \mathbf{\Sigma}_q (\mathbf{x}-\mu_q)\right]}, \tag{4.2}$$

with μ_q the p-dimensional mean vector and $\mathbf{\Sigma}_q$ the p x p covariance matrix of the Gaussian distribution. The scalar mixture weights α_q of the GMM are non-negative, $\alpha_q \geq 0, \forall q = 0, ..., Q-1$, and normalized to 1, $\sum_{q=0}^{Q-1} \alpha_q = 1$. Due to this two restriction, the Gaussian mixture is a probability density function (pdf). The parametric representation of a GMM is completely defined by $\theta = \{\alpha_q, \mu_q, \mathbf{\Sigma}_q\}$ for $q = 0, ..., Q-1$.

A GMM is a mathematical model that can approximate a probability density of a collection of independent and identically distributed (iid) vectors $\mathscr{X} = \{\mathbf{x}_n\}$, $n = 0 \ldots N-1$. GMMs are only suitable to model vector spaces in problems where the sequence of the observations parametrized by vectors is assumed non-relevant.

In vocal tract conversion problems, GMMs are used to model speech spectrum parameters, such as LPC derived coefficients or mel frequency cepstrum vectors. In this context, the components of the GMM model acoustic classes which may be associated to hidden phonetic events. Each class is represented by a Gaussian function, defined in all the acoustic space in a continuous way. Mixture weights α_q represent the normalized frequency of each class in the training data set.

4.2.2 Estimation of GMM Parameters

The estimation of the GMM parameters is usually stated as a maximum likelihood parameter estimation problem [Bil98]. The goal of this method is to find the model parameters that maximize the probability of observation of a training set.

The probability density function of a GMM (Eq. 4.1) is totally defined by the set of parameters $\theta = \{\alpha_q, \mu_q, \mathbf{\Sigma}_q\}$. Having a a set of N p-dimensional iid vectors $\mathscr{X} = \{\mathbf{x}_n\}$, $n = 0 \ldots N-1$, supposedly generated by this distribution, the likelihood function for θ is defined as:

$$P(\mathscr{X};\theta) = \prod_{n=0}^{N-1} P(\mathbf{x}_n;\theta), \tag{4.3}$$

and the log-likelihood is derived as:

$$L(\theta \mid \mathscr{X}) = log\left(P(\mathscr{X};\theta)\right) = \sum_{n=0}^{N-1} log(P(\mathbf{x}_n;\theta)). \tag{4.4}$$

The formulation of the maximum likelihood method can be stated as:

$$\theta^* = \underset{\theta}{\operatorname{argmax}}\, L(\theta \mid \mathscr{X}). \tag{4.5}$$

The maximization of this function according to θ is not analytically straight, but the parameters can be estimated using the Expectation-Maximization (EM) algorithm [Dem77]. The EM is an iterative algorithm that consist in increasing the likelihood of the model parameters in each iteration by successive maximizations of auxiliary functions.

EM is usually applied to maximum likelihood problems where the density probability functions models a given data set when the data is incomplete or has missing values. In GMM parameter estimation for vocal tract conversion, the incomplete data set $\mathscr{X}$, i.e. speech spectral vectors, can be completed by the acoustic class $\mathscr{Y}$ corresponding to each observation, which is considered the hidden variable. Then, with the complete data set $\mathscr{Z} = (\mathscr{X}\mathscr{Y})$ the complete likelihood function can be formulated as:

$$L(\theta \mid \mathscr{Z}) = L(\theta \mid \mathscr{X}, \mathscr{Y}) = log\left(P(\mathscr{X};\theta)\right). \tag{4.6}$$

The basic idea of the EM algorithm is to estimate a new model $\hat{\theta}$ from an initial model θ such that $L(\hat{\theta}) \geq L(\theta)$. Then, the new model becomes the initial model and the process is repeated until some determined threshold is reached. A general description of the EM routine is explained in the next paragraphs.

The algorithm consists in the following steps [Sty96]:

Initialization: Initialization of the model parameters θ^0.

E-step: To estimate the posterior probabilities of each hidden acoustic class using current model parameters θ^m:

$$P(w_q \mid \mathbf{x};\theta_q^m) = \frac{\alpha_q \mathcal{N}(\mathbf{x};\mu_q,\mathbf{\Sigma}_q)}{\sum_{j=0}^{Q-1} \alpha_j \mathcal{N}(\mathbf{x};\mu_j,\mathbf{\Sigma}_j)}, \tag{4.7}$$

for all the acoustic classes w_q, $q = 0, ..., Q-1$.

M-step: Using the posterior probabilities estimated above, estimate the new model θ^{m+1} using the following equations:

$$\alpha_q^{m+1} = \frac{1}{N}\sum_{n=0}^{N-1} P(w_q \mid \mathbf{x}_n; \theta_q^m), \tag{4.8}$$

$$\mu_q^{m+1} = \frac{\sum_{n=0}^{N-1} P(w_q \mid \mathbf{x}_n; \theta_q^m)\mathbf{x}_n}{\sum_{n=0}^{N-1} P(w_q \mid \mathbf{x}_n; \theta_q^m)}, \tag{4.9}$$

$$\mathbf{\Sigma}_q^{m+1} = \frac{\sum_{n=0}^{N-1} P(w_q \mid \mathbf{x}_n; \theta_q^m)(\mathbf{x}_n - \mu_q^{m+1})(\mathbf{x}_n - \mu_q^{m+1})^T}{\sum_{n=0}^{N-1} P(w_q \mid \mathbf{x}_n; \theta_q^m)}. \tag{4.10}$$

As stated previously, these expressions leads to $L(\theta^{(m+1)}) > L(\theta^m)$. E-step and M-step are repeated until some threshold of the likelihood increment is reached.

The EM algorithm converge to a local maximum [Xu96]. In practice, the initialization values of the parameters has small influence in the value of the final likelihood reached. However, the convergence rate depends on the initialization point. A regularization technique, which consist in adding a small value to the covariance diagonal at every step, may prevent from non-convergence due to numerical problems.

4.2.3 GMMs for Soft Classifying

A GMM provides a soft classification between the several Gaussian components through classification indices. The classification indices of a new vector correspond to the posterior probabilities of the acoustic classes given the observed vector. Applying Bayes' rule to the GMM expression the classification indices $c_q(\mathbf{x})$ are:

$$c_q(\mathbf{x}) = P(w_q \mid \mathbf{x}) = \frac{\alpha_q \mathcal{N}(\mathbf{x}; \mu_q, \mathbf{\Sigma}_q)}{\sum_{j=0}^{Q-1} \alpha_j \mathcal{N}(\mathbf{x}; \mu_j, \mathbf{\Sigma}_j)} \qquad q = 0, ..., Q-1, \tag{4.11}$$

where w_q indicates the q^{th} acoustical class.

Thanks to the smooth and continuous evolution of the classification indices, discontinuities in the synthesis phase are prevented due to jumps between acoustical classes.

4.3 GMMs as a Base of Mapping Functions: Previous Studies

In speech technologies, GMMs had been used efficiently for text-independent speaker recognition [Rey95], because it is is highly suitable to model a speaker's acoustic space.

In the vocal tract conversion task, GMMs are used not only to model the speaker acoustic space and to provide a soft classification of new speech vectors, but they are also the base

of the conversion function. The mapping function associates the acoustic space of the source speaker with the target speaker using the complete description of each component of the GMM, considering these components as clusters rather than simple vectors as in VQ approaches. The mixture of Gaussians splits the acoustic space according acoustic information, and learns a mixture of linear regression functions.

Several authors have published works based on GMMs [Sty96, Kai01a, Tod01b]. In the next two sections the theoretical basis of the most relevant GMM based VC systems are revised.

4.3.1 Least Squares GMM Mapping Function

GMMs were first introduced in the VC field by Stylianou et al. [Sty95]. The system was inspired by the mapping codebook approach [Abe88] and attempted to convert the whole spectral envelope without extracting specific phonetic features. The GMM is used as a model of the acoustic space of the source speaker. To carry out the conversion, the spectral parameter vectors of the source speaker are soft classified according to the acoustic classes of the GMM. The conversion function applied is formulated by a continuous parametric function which takes into account the probabilistic classification provided by the mixture model. The parameters of the mapping are estimated by least square optimization on the training data.

The form of the mapping function was chosen by analogy with the results obtained in the case where the GMM is reduced to a single component. If the source vectors follow a normal distribution, and source and target are jointly Gaussian, the minimum mean square error for the converted vector is given by [Kay98]:

$$\mathbb{E}[\mathbf{y} \mid \mathbf{x} = \mathbf{x}_n] = \mathbf{v} + \mathbf{\Gamma}\mathbf{\Sigma}^{-1}(\mathbf{x} - \mu), \tag{4.12}$$

where $\mathbf{\Gamma}$ is the cross covariance matrix of the source and target vectors and $\mathbf{v}$ is the mean target vector:

$$\mathbf{v} = \mathbb{E}[\mathbf{y}], \tag{4.13}$$

$$\mathbf{\Gamma} = \mathbb{E}[(\mathbf{y} - \mathbf{v})(\mathbf{x} - \mu)^T]. \tag{4.14}$$

This result was extended to the case of the GMM by weighting each expression 4.12, corresponding to each Gaussian of the mixture, by the conditional probabilities that the vector $\mathbf{x}_n$ belongs to the class w_q. The parametric form for the conversion function can be formulated as:

$$F(\mathbf{x}) = \sum_{q=0}^{Q-1} c_q(\mathbf{x})[\mathbf{v}_q + \mathbf{\Gamma}_q\mathbf{\Sigma}_q^{-1}(\mathbf{x} - \mu_q)], \tag{4.15}$$

where $c_q(\mathbf{x})$ is the posterior probability that the q^{th} Gaussian component had generated $\mathbf{x}$.

The parameters $\{\mathbf{v}_q, \mathbf{\Gamma}_q\}$ are computed by least-square optimization to minimize the total squared conversion error between the converted and the target data in the training set:

$$\varepsilon_{mse} = \mathbb{E}[\| \mathbf{y} - F(\mathbf{x}) \|^2]. \tag{4.16}$$

Stylianou et al. [Sty98] applied this conversion strategy to cepstral coefficients, studying two types of mappings according to the structure of $\mathbf{\Gamma}$: full conversion (full covariance matrix $\mathbf{\Gamma}$) and diagonal conversion (diagonal covariance matrix $\mathbf{\Gamma}$). It was reported that numerical errors in matrix inversions, computation load and storage requirements were a problem for using the full conversion mapping. Therefore, diagonal conversion was preferred. It was also reported that least square GMM method outperformed other already published methods in robustness as well as in efficiency.

4.3.2 Joint GMM Regression

Kain et al. [Kai98a] extended the least squares GMM conversion combining source and target spectral feature vectors $\mathbf{z} = [\mathbf{x}^T \mathbf{y}^T]^T$ in order to model the joint density $p(\mathbf{x}, \mathbf{y})$. In such situation, the conversion function that minimizes the mean square error between converted and target vectors is the regression of $\mathbf{y}$ given $\mathbf{x}$:

$$F(\mathbf{x}) = \mathbb{E}[\mathbf{y} \mid \mathbf{x}] = \int \mathbf{y} p(\mathbf{y} \mid \mathbf{x}) d\mathbf{y} = \sum_{q=0}^{Q-1} c_q(\mathbf{x}) \hat{y}_q, \tag{4.17}$$

where

$$c_q(\mathbf{x}) = \frac{\alpha_q \mathcal{N}(\mathbf{x}; \mu_q^x, \mathbf{\Sigma}_q^{xx})}{\sum_{j=0}^{Q-1} \alpha_j \mathcal{N}(\mathbf{x}; \mu_j^x, \mathbf{\Sigma}_j^{xx})} \tag{4.18}$$

and

$$\hat{y}_q = \mu_q^{\mathbf{y}} + \mathbf{\Sigma}_q^{yx} \mathbf{\Sigma}_q^{xx^{-1}} (\mathbf{x} - \mu_q^x), \tag{4.19}$$

with $\mathbf{\Sigma}_q = \begin{bmatrix} \mathbf{\Sigma}_q^{xx} & \mathbf{\Sigma}_q^{xy} \\ \mathbf{\Sigma}_q^{yx} & \mathbf{\Sigma}_q^{yy} \end{bmatrix}$ and $\mu_q = \begin{bmatrix} \mu_q^x \\ \mu_q^y \end{bmatrix}$.

No inversions of large and sometimes poorly conditioned matrices are required when modeling the joint acoustic space. The joint density estimation method makes no assumption about the target distributions: clustering takes place on observations of both source and target vectors. In theory, modeling the joint density should lead to more judicious allocation of mixtures for the regression problem.

Kain et al. [Kai98b] reported that least squares and joint GMM methods generated similar results. It seems to indicate that the target distributions are similar to the source distributions in

respect to their variance. But depending on the amount of training data, joint GMM regression was better than the least square method due to numerical errors. Therefore, it was concluded that joint GMM regressions are more robust to few training data.

A common problem of all the GMM based vocal tract systems is the broadening of formant bandwidths, due to (at least in part) an over-averaging introduced by the transformation function. Figure 4.2 illustrates this effect, clearly observable in the formant of the highest frequency. Perceptually, the broadening of formant bandwidths results in a muffling effect of the converted speech.

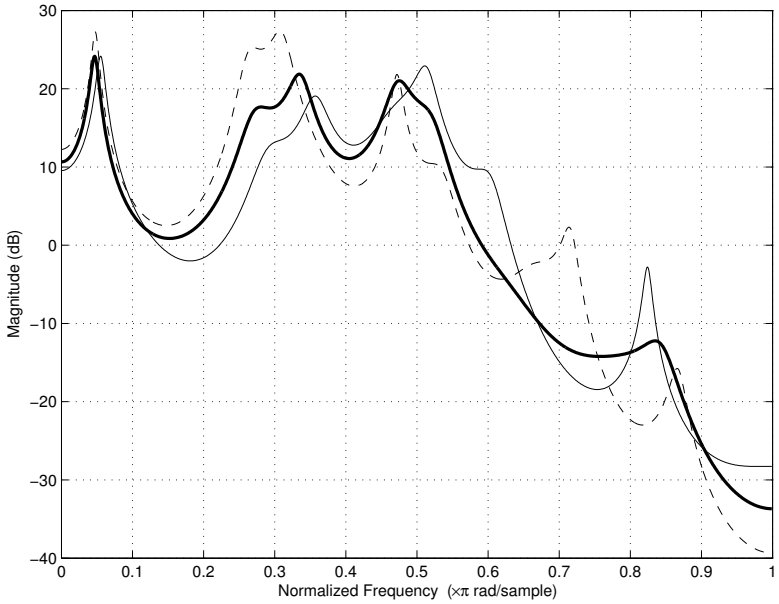


Figure 4.2: Spectral envelope of the source speaker (dashed line), target speaker (solid line) and converted speaker (thick solid line).

4.4 Non-parallel Source Data in GMM Systems

There are many applications of VC systems that, in the training step, it is available more data from the source speaker than from the target speaker. For example, in the personalization of a TTS as many source sentences as desired can be generated with an acceptable speech quality, due to the source speaker database is large (usually several hours). However, it is required to use as few sentences as possible from the target speaker.

Systems based on GMMs are well suited to vocal tract conversion, but they can not deal with source data without its corresponding parallel target data. In this section some modifications to the estimation algorithm for the GMM parameters are presented, in order to study if a set of source parameters, without the corresponding target ones, can improve the performance of the joint GMM system. Previous studies [Mil97] have shown that, for specific applications, including

unlabeled data in classification problems increases the performance of the classification. The current purpose is to apply this idea to the regression field. Some preliminary results of this study were published in [Dux03] by the author.

The parameters of the mixture model are calculated with the criterion of maximizing the likelihood function of the available source-target vector pairs (see Eq. 4.6). This likelihood function must be modified to include non-parallel data from the source speaker. The new likelihood function can be formulated as:

$$L(\theta \mid \mathscr{X}, \mathscr{Y}) = \prod_{i=0}^{N-1} P(\mathbf{x}_i, \mathbf{y}_i) \prod_{j=N}^{N+M-1} P(\mathbf{x}_j), \tag{4.20}$$

where

$$p(\mathbf{x}, \mathbf{y}) = \sum_{q=0}^{Q-1} \boldsymbol{\alpha}_q \mathcal{N}\left((\mathbf{x}, \mathbf{y}), \begin{pmatrix} \mu_q^x \\ \mu_q^y \end{pmatrix}, \begin{pmatrix} \boldsymbol{\Sigma}_q^{xx} & \boldsymbol{\Sigma}_q^{xy} \\ \boldsymbol{\Sigma}_q^{yx} & \boldsymbol{\Sigma}_q^{yy} \end{pmatrix}\right) \tag{4.21}$$

and

$$p(\mathbf{x}) = \sum_{q=0}^{Q-1} \boldsymbol{\alpha}_q \mathcal{N}(x, \mu_q^x, \boldsymbol{\Sigma}_q^{xx}). \tag{4.22}$$

N is the number of source-target vector pairs and M the number of source vectors without parallel target ones.

In the following two sections some assumptions are taken to simplify the estimation of the new GMM parameters. Comparative results of presented techniques are summarized in section 4.4.3.

4.4.1 Fixed Covariance Matrices

For the sake of simplicity, let's assume that the non-parallel source vectors will only significantly affect the means and weights of the joint source-target mixtures, but not the covariance matrices. It means that the shape of each acoustical class will be kept, while their position inside the acoustical space will be modified to take into account the new information.

To solve the maximum likelihood problem stated in Eq. 4.20 with the fixed covariance matrix restriction, an EM algorithm must be derived to recalculate the means and weights, without modifying the covariance matrices, from the parallel and non-parallel training data. The expressions for the GMM parameters of this modified EM at each iteration are:

$$\alpha_q^{m+1} = \frac{1}{N+M}\left(\sum_{n=0}^{N-1} P(w_q \mid \mathbf{x}_n, \mathbf{y}_n; \theta_q^m) + \sum_{n=N}^{N+M-1} P(w_q \mid \mathbf{x}_n; \theta_q^m)\right), \tag{4.23}$$

$$\mu_q^{x,m+1} = \frac{\sum_{n=0}^{N-1} \mathbf{x}_n P(w_q \mid \mathbf{x}_n, \mathbf{y}_n; \theta_q^m) + \sum_{n=N}^{N+M-1} \mathbf{x}_n P(w_q \mid \mathbf{x}_n; \theta_q^m)}{\sum_{n=0}^{N-1} P(w_q \mid \mathbf{x}_n, \mathbf{y}_n; \theta_q^m) + \sum_{n=N}^{N+M-1} P(w_q \mid \mathbf{x}_n; \theta_q^m)}, \tag{4.24}$$

$$\mu_q^{y,m+1} = \frac{-\Sigma_q^{yx,m}\Sigma_q^{xx,m}\left(\sum_{n=0}^{N-1}(\mathbf{x}_n - \mu_q^{x,m+1})P(w_q \mid \mathbf{x}_n, \mathbf{y}_n; \theta_q^m)\right)}{\sum_{n=0}^{N-1} P(w_q \mid \mathbf{x}_n, \mathbf{y}_n; \theta_q^m)} + \frac{\sum_{n=0}^{N-1} \mathbf{x}_n P(w_q \mid \mathbf{x}_n, \mathbf{y}_n; \theta_q^m)}{\sum_{n=0}^{N-1} P(w_q \mid \mathbf{x}_n, \mathbf{y}_n; \theta_q^m)}, \tag{4.25}$$

$$\mathbf{\Sigma}_q^{m+1} = \mathbf{\Sigma}_q^0, \tag{4.26}$$

where θ_q^m are the model parameters of the component q at the previous iteration. Details about the derivation of the expression 4.23, 4.24 and 4.25 can be found in Appendix A.

The procedure to estimate a GMM from parallel and non-parallel data is as follows. First, an initial GMM is estimated from the parallel data as in section 4.3.2. Afterwards, this GMM is used as an initial model of the modified EM algorithm. It is expected that this re-estimation of the GMM will be fast and converge in few iterations, as the highest dimensional parameters are not re-calculated and the initial GMM should be near a maximum of the parallel/non-parallel likelihood function.

4.4.2 Completion of Non-Parallel Data

An alternative to reduce the computational complexity of the estimation of the parallel/non-parallel likelihood function given by the expression 4.20 is to complete the missing data. It means to make a guess about the target parallel parameters of non-parallel training source vectors. One possible way of completing data is to include transformed vectors as parallel vectors. In such situation, the new likelihood function will be stated as:

$$L(\theta \mid \mathscr{X}, \mathscr{Y}) = \prod_{i=0}^{N-1} p(\mathbf{x}_i, \mathbf{y}_i) \prod_{j=N}^{N+M-1} p(\mathbf{x}_j, \mathbf{x}_j^{trans}), \tag{4.27}$$

where $\mathbf{x}_j^{trans}$ are the converted vectors corresponding to the non-parallel source data, transformed by an initial vocal tract conversion system.

In order to apply the completion strategy, it should be assumed that a starting conversion function, estimated from only the parallel data, is good enough to estimate an initial set of transformed vectors.

The procedure to estimate a vocal tract conversion system with the completion strategy is as follows. First, an initial GMM is estimated from the parallel data as in section 4.3.2 and the non-parallel source vectors are converted. Once source-transformed vector pairs are built, a

second EM algorithm will be applied over the joint parameter set of source-target and source-transformed vectors in order to estimate the final GMM.

Note that during the completion strategy the basic EM algorithm is applied twice: first using the parallel source-target vectors as training data, and then using the parallel source-target and source-transformed vectors as training data.

4.4.3 Experimental Results

The corpus used for the non-parallel source data experiments consisted in a corpus of two speakers, one male and one female, recorded for a TTS development (see chapter 3 for the corpus and speech analysis details). In this study, speech signals were decimated to a sampling frequency of 8kHz and 12 order LSF vectors were estimated. The resulting source and target LSF vectors were lineally aligned using phoneme boundaries as anchor points as explained in section 3.3.

A reference joint GMM regression system was trained using 20 pairs of male-female aligned sentences (training Set_20; about 1 minute of speech) and also with a reduced set of 5 pairs of male-female aligned sentences (training Set_05; about 15 seconds of speech). For each training set, different numbers of mixtures have been considered. The performance of the conversion from a male speaker to a female speaker was computed with independent test data, consisting in 20 sentences. The performance is calculated as the index P:

$$P = 1 - \frac{D(\tilde{\mathscr{Y}}, \mathscr{Y})}{D(\mathscr{X}, \mathscr{Y})}, \tag{4.28}$$

where the function $D(\cdot)$ indicates the distance between the source vocal tract vectors $\mathscr{X}$, the target vocal tract vectors $\mathscr{Y}$ and the converted vocal tract vectors $\tilde{\mathscr{Y}}$. The results are shown in table 4.1.

# sentences/components	2	4	8	16	32	64
Set_20	-	-	0.259	0.255	0.239	0.209
Set_05	0.202	0.210	0.179	0.144	-	-

Table 4.1: Performance index for the reference vocal tract conversion system.

According to table 4.1, a regression function trained with 20 sentences performs the best for 8 Gaussian mixtures, and a regression function trained with 5 sentences performs the best for 4 mixtures. When more mixtures are trained an over-fitting problem appears.

The evaluation of the effect of adding non-parallel data using both methods explained in the previous sections have been carried out for the two training sets (Set_20 and Set_05) and

for each number of mixtures (Q). The results of the reference joint GMM regression systems trained with enlarged sets of male-female aligned data will be also presented as a comparative value. This top-line system will be called enlarged-GMM.

Figures 4.3 and 4.4 compares the results for Set_20 and Set_5 training sets between the reference system, the enlarged-GMM system and the fixed covariance matrices system. The first figure presents, from left to right, groups of experiments where 10, 20, 40 and 80 source sentences have being added to the parallel training data. In the second figure 5, 10 and 15 non-parallel source sentences were added. For each group, the evaluation index P is represented, as a function of the number of mixtures.

As it was expected, adding more parallel data resolves the over-fitting problem of the reference system and increases the performance of the conversion. Therefore, enlarged-GMMs perform better than other methods.

The slightly increasing in the performance showed in 4.3 when using non-parallel source data is not significant, independently from the number of non-parallel sentences used.

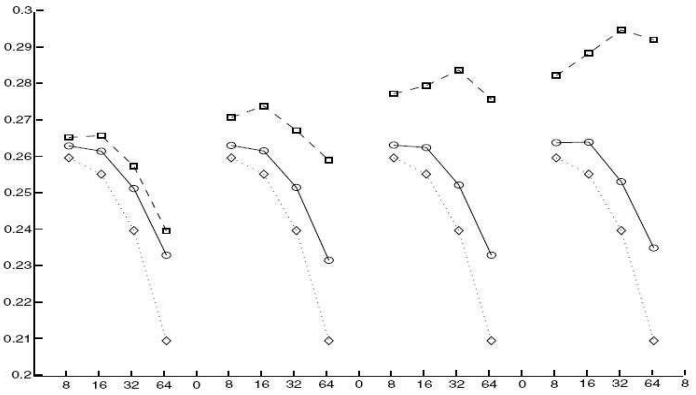
On the other hand, the results presented in figure 4.4 for the Set_05 training set show significant improvements in the performance. Actually, adding 10 unaligned sentences and estimating a GMM with 4 mixtures is computationally fast, and the performance increases 18% (from 0.190 to 0.224).

It can be concluded that when dealing with models estimated over few data, modifications only in the mixture weights and means increase the performance. But, when the GMM has been estimated with enough data to be a good model, the restriction over the covariance matrices is a hard constraint and the results do not improve significantly.

Figures 4.5 and 4.6 show the performance indices for experiments about completion of non-parallel data. To build the source-transformed pairs, the reference GMM with 8 Gaussian mixtures was used in Set_20, and the reference GMM with 4 mixtures in Set_5. The same sets of non-parallel source data as in the previous experiments have been added.

It can be observed that a better performance is obtained with the completion of non-parallel data than with fixed covariance matrices for both reference GMMs. In fact, for the 20 initial sentences model, it is possible to achieve an equivalent performance adding 10 source-target parallel sentences (50% more of the initial set), than adding 20 non-parallel source sentences. The inclusion of more unaligned sentences does not increase the performance; this means that it exists a relationship between the two different sets of data.

In the case of a reference GMM trained with 5 sentences, the performance increment adding non-parallel data is more remarkable, since in the reference system a limited amount of information about the source and the target is available.



;

Figure 4.3: Performance index for the fixed covariance method (circles), compared to the reference GMM (diamonds) and enlarged-GMM (squares), for training Set_20. Number of sentences added, from left to right: 10, 20, 40 and 80.

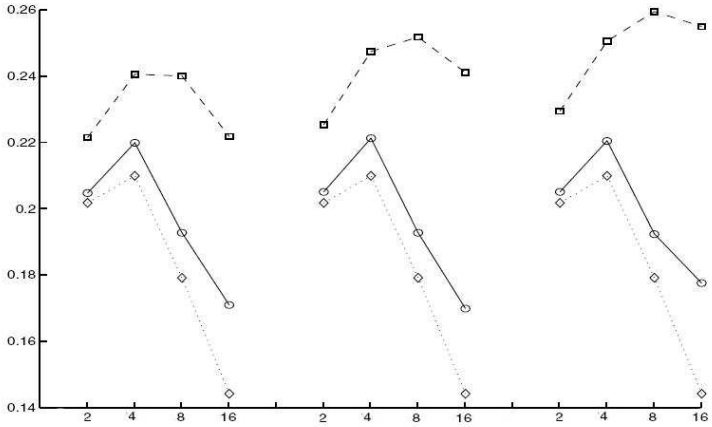
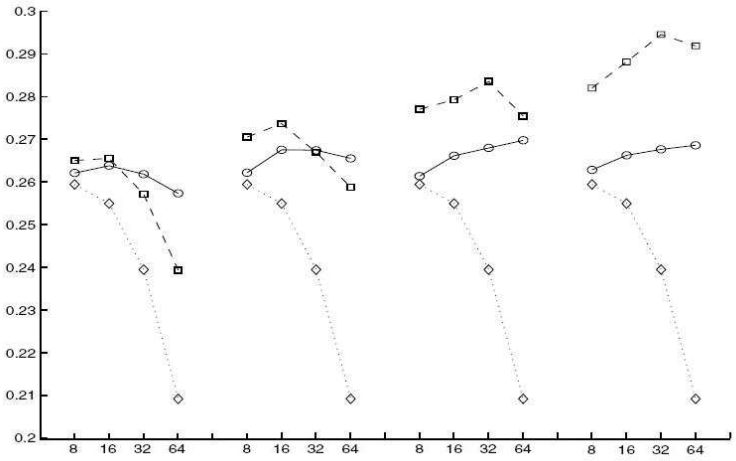


Figure 4.4: Performance index for the fixed covariance method (circles), compared to the reference GMM (diamonds) and enlarged-GMM (squares), for training Set_05. Number of sentences added, from left to right: 5, 10 and 15.



;

Figure 4.5: Performance index for the completion of non-parallel data method (circles), compared to the reference GMM (diamonds) and enlarged-GMM (squares), for training Set_20. Number of sentences added, from left to right: 10, 20, 40 and 80.

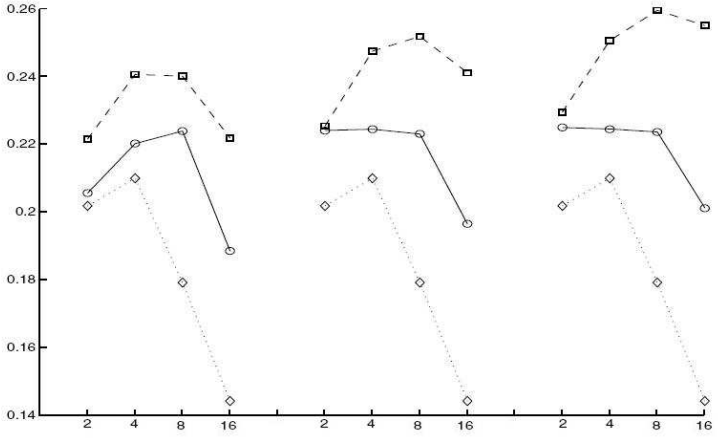


Figure 4.6: Performance index for the completion of non-parallel data method (circles), compared to the reference GMM (diamonds) and enlarged-GMM (squares), for training Set_20. Number of sentences added, from left to right: 5, 10 and 15.

According to the results, the strategy of completion of non-parallel data is appropriated to increase the conversion performance when a limited set of source-target parallel training vectors are available.

4.4.4 Conclusions

This section focused on increasing the performance of a vocal tract conversion system based on joint source-target GMM regression, using non-parallel source data. It has been shown that a combined learning with parallel source-target data and source-transformed data increases the conversion performance, mainly when few training data is available. In this latest situation, to re-estimate only means and mixture weights also increases the performance, with a very reduced computational time.

4.5 HMM based Vocal Tract Conversion

Although GMM based systems are good candidates for the vocal tract conversion task, they have drawbacks. Some of the drawbacks are caused by the GMM own structure. GMM based systems work on a frame-by-frame basis. That means that the information about past and future frames is not relevant for the conversion. This is a simplification of the real speech production mechanism, where co-articulation affects phoneme pronunciation. The propose of this section is to include dynamic information in the vocal tract conversion task, in order to transform one frame according to previous and next frames. The objective of including dynamic information is to better convert phoneme boundary frames.

Hidden Markov Models (HMM) are well-known models which can capture the dynamics of the training data by using states. A HMM models the probability distribution of a feature vector according to its actual state, and also models the dynamics of vector sequences with transition probabilities between states. Thus, HMMs were chosen to extent GMMs in the vocal tract conversion task.

Conversion systems based on a HMMs were almost inexistent in the literature. HMMs were first introduced in [Kim97], with the purpose of segmenting the speech signal according to states and applying a discrete state-dependent transformation function. An introductory study about HMM based vocal tract conversion systems was published in [Dux04b] by the author, where mapping functions were formulated as Gaussian functions, extending previous discrete mappings, but HMMs were still used to hard segment the source speech into states. In the following sections, a HMM conversion system based on soft classification and continuous transformation is presented.

4.5.1 HMMs as Probability Density Functions

HMMs can model problems with temporality, situations that evolve in time. In such problems, a state $s(t)$ at time t depends directly on a state $s(t-1)$ at time $t-1$. The training data to model is a set of N vector sequences $\mathscr{X} = \{\mathbf{X}_n^{T_n}\}$, $n = 0 \dots N-1$, where each sequence $\mathbf{X}_k^{T_k} = (\mathbf{x}_0^k \mathbf{x}_1^k \dots \mathbf{x}_{T_k-1}^k)$ consist of T_k vectors $\mathbf{x} = [x_0 x_1 \dots x_p - 1]^T$ of dimension p.

A HMM may be represented by nodes, acting as hidden states, interconnected by links describing the conditional probabilities of a transition between the states $a_{ij} = P(s(t) = j | s(t-1) = i)$. Each hidden state has an associated probability function $b_i(\mathbf{x}) = P(X_t = x | s(t) = i)$ of emitting a visible state $\mathbf{x}$.

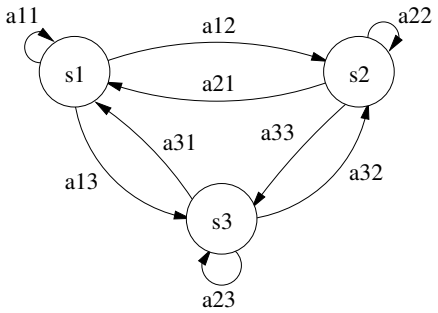


Figure 4.7: HMM diagram

At every time step, from t to the $t+1$, some transition between hidden states must occur and a visible symbol must be emitted. Thus, the model has the following normalization conditions:

$$\sum_j a_{ij} = 1 \qquad \forall i, \tag{4.29}$$

$$\int_x b_j(\mathbf{x}) dx = 1 \qquad \forall j. \tag{4.30}$$

All the studied HMMs are ergodic, i.e. all the states are connected with each other and also with themselves. The emission probability function for each state is defined as a Gaussian Mixture Model, sometimes limited to only one component. The studied HMMs are first order discrete time Markov Models, which means that the probability at $t+1$ depends only on the state at t.

The probability that the model produces a sequence $\mathbf{X}^T$ of observations is:

$$P(\mathbf{X}^T) = \sum_{r=1}^{r_{max}} P(\mathbf{X}^T \mid \mathbf{s}_r^T) P(\mathbf{s}_r^T), \tag{4.31}$$

where each s_r is a sequence $\mathbf{s}_r^T = (s(1), s(2), ..., s(T))$ of T hidden states. If the HMM has S hidden states there will be S^T possible terms in the sum of Eq. 4.31, corresponding to all possible sequences of length T.

According to Eq. 4.31, to find the probability of the HMM to produce the sequence $\mathbf{X}^T$ all the possible sequences of hidden states must be found, then for each sequence the probability of generating $\mathbf{X}^T$ must be calculated and finally all of the contributions must be sum up. Here, the parallelism with the previous model can be observed: each state sequence in a HMM is acting as each component in a GMM (see Eq. 4.1).

HMMs will be used to model spectrum parameters in the vocal tract conversion task. In particular, joint source-target vectors were used. In this context, the hidden states of the HMM correspond to hidden acoustic classes which can be associated to hidden phonetic events. Note that each hidden acoustic class is modeled by a GMM. The main difference with GMMs systems is that HMMs capture information about the sequence of acoustic events.

4.5.2 Estimation of HMMs Parameters

The HMM parameters $(a_{ij}, b_i(\mathbf{x}), \pi_i)$, where a_{ij} indicates the transition probability matrix (transition probabilities among hidden states), $b_i(\mathbf{x})$ the emission probability function of the i^{th} state and π_i the initial probability of the i^{th} state, can be estimated iteratively from training samples using the Baum-Welch (or forward-backward) algorithm. Only the basic concepts and notation of this algorithm will be introduced here, as they will be used in the conversion function description. For a detailed explanation of this algorithm refer to [Rab93].

The forward-backward algorithm is an instance of the expectation-maximization algorithm. In the forward part we define $\alpha_i(t)$ as the probability that the model is in the state $s(t) = i$, from now $s_i(t)$, and has generated the target sequence up to the time t.

$$\alpha_i(t) = \begin{cases} 0 & t = 0 \text{ and } i \neq \text{initial state} \\ 1 & t = 0 \text{ and } i = \text{initial state} \\ \left[\sum_j \alpha_j(t-1)a_{ji}\right] b_i(\mathbf{x}(t)) & \text{otherwise} \end{cases}$$

The backward algorithm is the time-reversed version of the forward algorithm. We define $\beta_i(t)$ to be the probability assuming that $s(t) = i$ and will generate the remainder of the given target sequence, that is, from $t+1$ to T.

$$\beta_i(t) = \begin{cases} 0 & w_i(t) \neq w_0 \text{ and } t = T \\ 1 & w_i(t) = w_0 \text{ and } t = T \\ \sum_j \beta_j(t+1)a_{ij}b_j(\mathbf{x}(t)) & \text{otherwise} \end{cases}$$

The $\alpha_i(t)$ and $\beta_i(t)$ parameters are determined at each iteration of the algorithm using the

a_{ij} and $b_j(\mathbf{x}(t))$ of the previous iteration. In order to write an updating expression for a_{ij} and $b_j(\mathbf{x}(t))$, first an auxiliary variable is introduced: $\xi_{ij}(t)$, the probability of transition between $s_i(t-1)$ and $s_j(t)$, given the model generated the entire training sequence $\mathbf{X}^T$ by any path:

$$\xi_{ij}(t) = \frac{\alpha_i(t-1)a_{ij}b_j(\mathbf{x}(t))\beta_j(t)}{P(\mathbf{X}^T|\Theta)}, \tag{4.32}$$

where $P(\mathbf{X}^T|\Theta)$ is the probability that the model has generated the sequence $\mathbf{X}^T$ by any path.

Therefore, the new model parameters a_{ij} can be estimated by:

$$a_{ij} = \frac{\sum_{t=1}^{T}\xi_{ij}(t)}{\sum_{t=1}^{T}\sum_{k}\xi_{ik}(t)}, \tag{4.33}$$

This process is repeated until a convergence criterion is reached.

4.5.3 HMMs for Soft Classifying

A HMM provides two types of classification: according to the hidden variables and according to the observations. Thanks to the continuous structure of the presented HMMs, both kinds of classification can be soft, i.e. with indices between $[0, 1]$ for each class. Usually, when working with HMMs in speech processing, e.g. for speech or speaker recognition, a hard classification is used for the hidden variables, using a dynamic programming algorithm to find the optimal path through the Markov model. The problem of finding the most probable sequence of hidden states given a sequence of visible states X^T is a decoding problem.

However, in vocal tract conversion task we are interested in a soft classification between the hidden states, through classification indices. Similar to GMM, the classification indices correspond to the posterior probability of a hidden state s_i at time t given the observed vector sequence:

$$c_i(t, \mathbf{X}) = P(s_i(t) \mid \mathbf{X}) = \gamma_i(t) = \frac{\alpha_i(t)\beta_i(t)}{\sum_{j=1}^{S}\alpha_j(t)\beta_j(t)}, \tag{4.34}$$

where $s_i(t)$ indicates the i^{th} hidden state at time t and S the number of hidden states.

4.5.4 Vocal Tract Conversion Based on HMMs

The spectral conversion based on HMMs presented is an approximation similar to joint GMMs, but with the advantages of using sequence information. The concept of soft classification is extended not only to the probability density function of the emissions, but also to the state being.

Extended source-target vector sequences $\mathbf{Z}_n^{T_n}$ for $n = 0 \dots N-1$ are used to estimate a HMM as it is explained in the previous sections. To carry out the conversion, first spectral parameter vectors of the source speaker $\mathbf{x}$ are soft classified according to the hidden states of the Markov model. The conversion function applied is formulated by a continuous parametric function which takes into account the probabilistic classification provided by the model.

To estimate the conversion function, the expectation of a sequence $\mathbf{Y}$ of the target speaker given the sequence $\mathbf{X}$ of the source speaker is calculated by means of:

$$\mathbb{E}[\mathbf{Y} \mid \mathbf{X}] = \Big[\mathbb{E}[\mathbf{y}_0 \mid \mathbf{X}] \qquad \mathbb{E}[\mathbf{y}_1 \mid \mathbf{X}] \qquad \dots \qquad \mathbb{E}[\mathbf{y}_{T-1} \mid \mathbf{X}]\Big], \tag{4.35}$$

where

$$\mathbb{E}[\mathbf{y}_t \mid \mathbf{X}] = \int \mathbf{y}_t p(\mathbf{y}_t \mid \mathbf{X}) d\mathbf{y}_t \qquad \text{for } t = 0 \dots T-1. \tag{4.36}$$

The conditional probability of a vector $\mathbf{y}_t$ given the sequence $\mathbf{X}$ can be decomposed according to the hidden states:

$$p(\mathbf{y}_t \mid \mathbf{X}) = \sum_{i=1}^{S} p(\mathbf{y}_t, s_i(t) \mid \mathbf{X}) = \sum_{i=1}^{S} p(\mathbf{y}_t \mid s_i(t), \mathbf{X}) p(s_i(t) \mid \mathbf{X}). \tag{4.37}$$

Thus, the expectation expression can be rewritten as:

$$\mathbb{E}[\mathbf{y}_t \mid \mathbf{X}] = \sum_{i=1}^{S} p(s_i(t) \mid \mathbf{X}) \int \mathbf{y}_t p(\mathbf{y}_t \mid s_i(t), \mathbf{X}). \tag{4.38}$$

Eq. 4.38 consist of two different terms: one corresponding to the indices of the soft classification according to the hidden states and another corresponding to the converted vector for each state. Finally, the conversion function for a HMM system can be expressed as:

$$\mathbb{E}[\mathbf{y}_t \mid \mathbf{X}] = \sum_{i=1}^{S} c_i(t, \mathbf{X}) \hat{\mathbf{y}_t^i}, \tag{4.39}$$

where $\hat{\mathbf{y}_t^s}$ is the transformation by the GMM corresponding to the i^{th} state explained in the last section.

The structure of the conversion function is similar to one of the GMM based conversion system. Both functions are a weighted sum of the individual Gaussian component regressions, but the value of the weights differ in each case. When working with GMMs, weights are estimated according to the probabilities of belonging to each acoustic class. However, when working with HMMs, weights are estimated according to the probability of being in the hidden states, introducing dynamic information into the system.

4.6 Decision Tree based Vocal Tract Conversion

At the output of a TTS there is not only available the synthetic speech signal, but it is also available phonetic information of the generated utterances. All the vocal tract conversion approaches presented until now use spectral features derived from the speech signal to estimate the acoustic models by maximum likelihood, ignoring the phonetic data. Phonetic data is highly reliable, as it does not come from any estimation. Moreover, phonetic data may be usefull in the classification task, because the acoustics are somehow related to the phonetics.

Probabilistic GMM and HMM based vocal tract conversion approaches have two main characteristics: the conversion functions are continuous transformations and there is a prior soft classification according to acoustic variables. In order to include phonetic information to the vocal tract conversion task, the classification machine has to be able to handle with categorical data (variables that take a finite number of values without any natural notion of similarity nor order).

Decision trees allow working with numerical data as well as categorical data. Thanks to this property, non metric data, such as phonetic information, was incorporated into the classification task prior to the regression estimation by the author in [Dux04a]. In particular, the variables that have been used in the current work are: voicing of each frame, the vowel/glide/consonant category, point and manner of articulation for consonants, height and backness for vowels and glides. Phonemes are not used as a categorical variable due to the scarceness of data.

Unlike GMM and HMM based vocal tract conversion systems, the classification provided by decision trees is hard. Although soft classification is more flexible and less sensible to errors due to jumps between classes, it is belived that phonetic data carries information that allows to better split the acoustic space according to the transformation error.

Moreover, it is expected that two continuous source vectors will produce two similar transformed vectors, if conversion functions are reliable estimated. In such situation, to assure the converted spectrum continuity is not as important the soft classification as transformation functions continous and defined in all the acoustic space. Therefore, decision trees with a Gaussian regression related to each leave are studied as a classification machine prior to the spectral transformation.

4.6.1 CART Decision Trees

CART (classification and regression trees) [Bre98] is a general framework to organize data, in order to build a classification or a regression machine. As CART handles with numerical and categorical data, phonetic information can be incoporated into the acoustic regression.

To start the construction of a decision tree all the available training data $\mathscr{X}$ is asigned to the root node. Then, this data is splited into subsets assigned to descendant nodes. This process is repeated for each descendant node, until some stopping criteria is fullfiled. Figure 4.8 represents a binary decision tree.

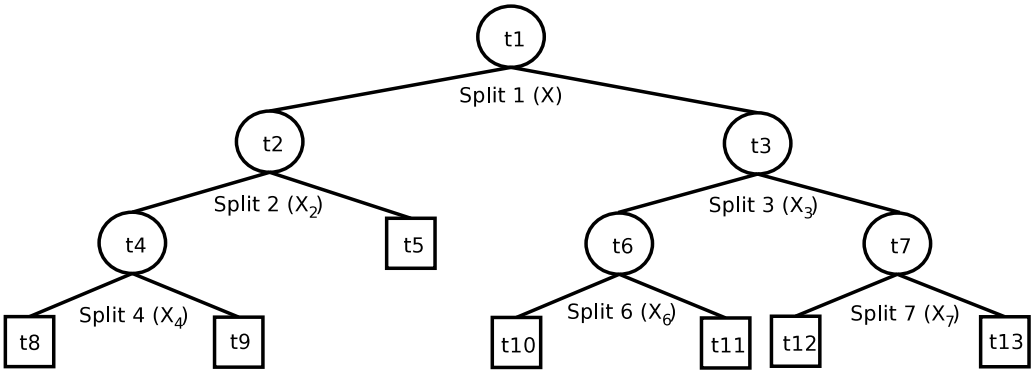


Figure 4.8: Decision tree diagram.

Nodes which are not splited are called terminal nodes or leafs. For each node t_i, the data subsets of its descendant nodes are disjoint. Data subsets of the tree leafs form a partition of $\mathscr{X}$. Trees applied to VC define a conversion function for each leaf. The required elements to determine and to build a tree are:

1. A maximum number of splits for each node.

2. A set Q of questions to test in each node.

3. A rule to select a split at every intermediate node, also called the splitting rule.

4. A rule for determining when a node is terminal.

5. An estimation of a conversion function for every leaf.

All the trees studied in this disertation are binary, i.e. each node is splited into two child nodes. The set Q is formed by binary questions of the form *is* $\mathbf{x} \in A$, $A \subset \mathscr{X}$, where A represents a phonetic characteristic of the frame $\mathbf{x}$, in particular: a vowel/glide/consonant flag, the point of articulation, the manner of articulation, the height, the backness and voicing (see table 4.2).

Items three and four are explained in the next section, and the estimation of a conversion function for every leaf is presented in section 4.6.4.

vowel/glide/consonant flag	vowel, glide, consonant
point of articulation	alveolar, bilabial, dental, interdental, labiodental, palatal, velar
manner of articulation	fricative, affricate, aproximant, lateral, nasal, plosive, tap, trill
height	open, mid-open, close, mid-close, schwa
backness	back, center, front
voicing	voiced, unvoiced

Table 4.2: Possible values for phonetical variables.

4.6.2 CART Growing for Voice Conversion

There are multiple methods to grow a decision tree. Although all the methods share the basic idea of building new nodes while spliting the training data up to the leafs, their main difference is how this data is handled. First, a straightforward method to grow a CART with all the available data for training is presented. Afterwards, two methods that deal with validation data sets are presented: one using pre-pruning and another based on post-pruning.

The straightforward procedure to grow the tree uses all the available data in each node to estimate a GMM based vocal tract conversion system. The spliting decision is taken according to the increment of the accuracy of the conversion of the same training data. In particular, a joint GMM based vocal tract conversion system is estimated from a training data set for the parent node t (the root node in the first iteration), and an error index $E(t)$ for all the elements of the training data set belonging to that node is calculated. The error index used is the mean of the Inverse Harmonic Mean Distance [Lar91] between target and converted frames, calculated as:

$$E(t) = \frac{1}{|t|} \sum_{n=0}^{|t|-1} IHMD(\tilde{\mathbf{y}}_n, \mathbf{y}_n), \tag{4.40}$$

where $|t|$ is the number of frames in the node t, $\mathbf{y}$ is a target frame and $\tilde{\mathbf{y}}$ its corresponding converted frame. The distance $IHMD(\tilde{\mathbf{y}}, \mathbf{y})$ (see Eq. 3.10 in page 45) weights more the mismatch in spectral picks than the mismatch in spectral valleys when working with LSF vectors.

All the possible question of the set Q are evaluated at node t and two child nodes t_L and t_R are populated for each question q. The left descendant node is formed by all the frames which fulfill the question and the right node by the rest.

For each child node, a joint GMM based vocal tract conversion system is estimated, and the error indices $E(t_L, q)$ and $E(t_R, q)$ for the training vectors corresponding to the child nodes t_L and t_R obtained from the question q are calculated. The increment of the accuracy for the

question q at the node t can be calculated as:

$$\Delta(t,q) = E(t) - \frac{\big(E(t_L,q)|t_L|\big) + \big(E(t_R,q)|t_R|\big)}{(|t_L| + |t_R|)} \qquad |t| = |t_L| + |t_R| \tag{4.41}$$

where $|t|$ indicates the number of spectral vectors of the training set belonging to the node t.

The increment of the accuracy is evaluated for each question and the question q^* corresponding to maximum increment is selected. If the increment of the accuracy is positive and the number of training frames and the number of validation frames are higher than minimum threshold the node t is split by the question q^*. Otherwise, that node is declared a leaf.

The tree is grown until there is no node candidate to be split. At this point, the root, non-terminal nodes and leaves are determined, as well as the selected question of each splitting. The flow chart of page 72 resumes the CART growing procedure.

Decision trees grown with the previous procedure may suffer over-fitting to the training data. Due to the same data is used to construct the tree and to decide either to stop or to continue with the splitting, small particularities or estimation errors of the training data may influence the final structure of the system. This effect is more relevant in the deepest levels of the tree, once the main dependencies have been captured by higher nodes. To solve the problem of over-fitting some pruning algorithms have been proposed. Pruning algorithms are divided in two categories: pre-pruning methods and post-pruning methods.

Pre-pruning methods avoid large trees by introducing in the stop splitting rule an arbitrary threshold or a dependency with a validation data set. The method adopted in the current work consists in dividing the training set into two subsets: one called training and the other one validation. To decide if a node will be split or not, the increment of accuracy for the training subset is evaluated for each question and the question q^* corresponding to the maximum increment is selected. Then, the increment of accuracy for the validation subset for the question q^* is calculated, and only if it is greater than zero the node will be split. Note that to construct a tree with this pre-pruning method not only the training subset must be dragged through the tree but also the validation subset.

Pre-pruning methods have a drawback called "the horizon effect". The decision about stopping the splitting is taken locally in each node, without considering future splits. Therefore, a node declared a leaf could become the root of a subtree with better accuracy if the splitting had not been stopped.

To avoid the "the horizon effect", post-pruning algorithms start with a complete tree and recombine leafs and nodes to achieve a smaller tree with good accuracy. In this dissertation the Reduced Error Pruning algorithm (REP) [Qui87] is studied. This algorithm finds the smallest optimally pruned tree with respect to a validation subset.

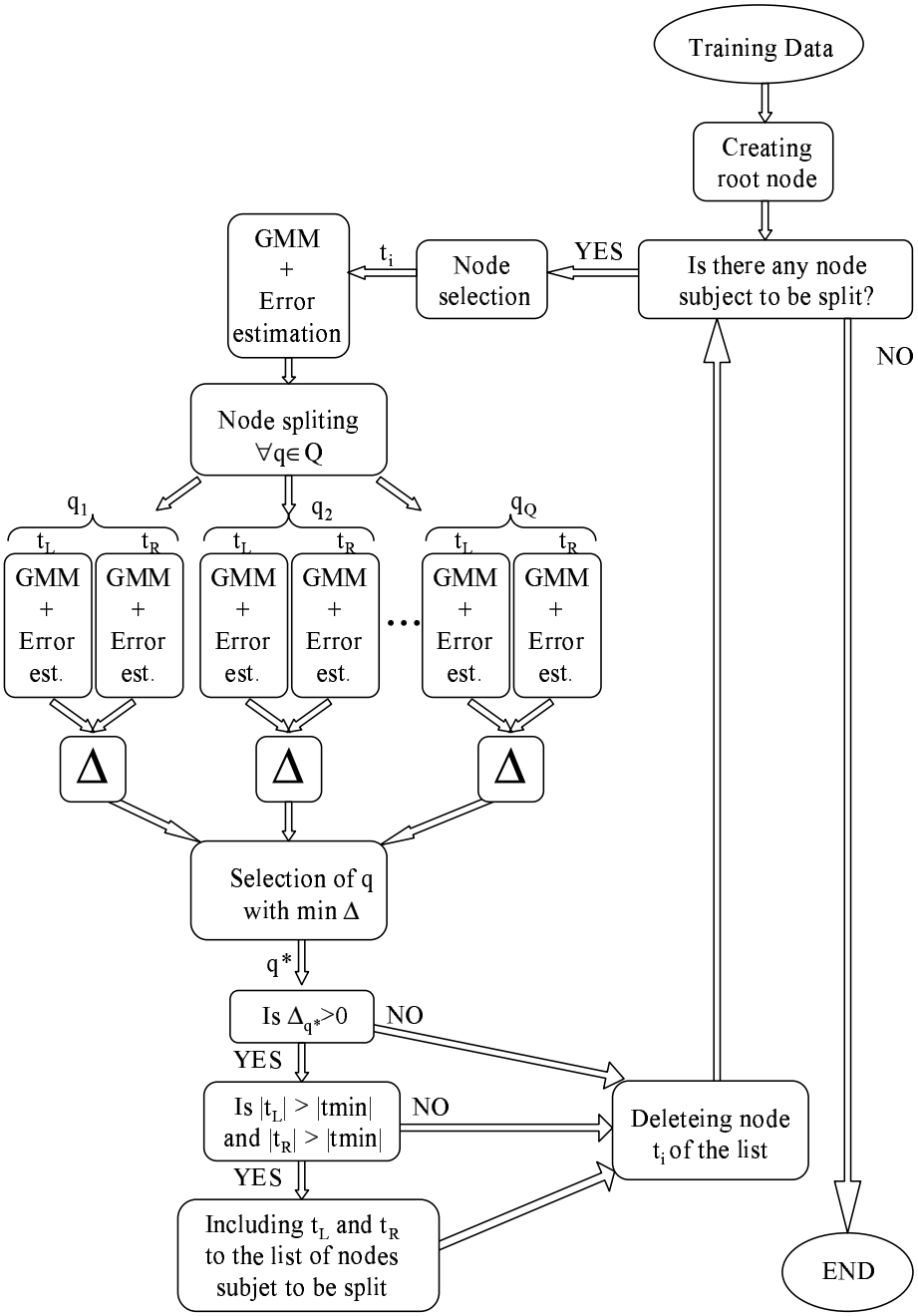


Figure 4.9: Flow chart of CART growing for VC systems.

REP consist in building a decision tree with the data of the training subset, for example with the straightforward method already presented. Then, for each non-terminal node t it compares the accuracy of the validation subset when the subtree rooted at t is kept with the accuracy of the validation subset when the node t becomes a leaf. If the accuracy of the simplified tree is higher, and the tree rooted at t contains no other subtree with a better accuracy, the node t is pruned. This process is repeated until any further pruning may decrease the accuracy with respect to a validation subset. In practice, REP algorithm is applied in a single-scan bottom-up manner to find the optimum tree.

4.6.3 Decision Trees for Hard Classifying

The decision tree constructed as explained in the previous section can be used to divide the acoustic space in overlapping classes determined by phonetic properties. The difference between the CART approach and the probabilistic approaches, as GMM and HMM, is that the classification into these acoustic classes is hard.

The classification of a new vector begins at the root node, which evaluate a particular phonetic property of the frame. Based on the answer the appropriated link to a child node is followed. This process is repeated until a leaf is reached. Each leaf represents a hidden acoustical class and has defined a conversion function.

4.6.4 Vocal Tract Conversion based on Decision Trees

Up to now, it has been seen how a decision tree is grown based on phonetic information and how it is used for classification into hidden phonetic classes. To carry out the conversion a regression function for each class is needed. Their estimation is straightforward. First, all the available data (training set or training plus validation subsets if some pruning is applied) is hard classified by the tree. Then, the data of each class is used to estimate a joint GMM and the transformation function related is derived.

It must be remarked that, although the transformation function of each leave is estimated with data of a single phonetic class, the transformation is continuous and defined in all the acoustic space. Both properties are a requirement to assure a high quality of the converted speech.

To transform new source vectors, they are classified into leafs according to their phonetic features by the decision tree. Then, each vector is converted according to the joint GMM regression belonging to its leaf.

4.7 Experiments and Results

This section summarizes the most relevant experiments that have been carried out in order to evaluate the effects of the vocal tract conversion and to compare different conversion systems. In particular, the four mappings presented in the previous sections have been evaluated: joint GMM regression, as a baseline system, and HMM based vocal tract conversion, CART with pre-pruning systems and CART with post-pruning systems, as new proposed systems. The experiments have been designed to study three important aspects of the vocal tract conversion:

- The architecture determination of GMM and HMM based systems.
- The objective performance of the different mapping functions according to the source-target speaker pair and the number of training sentences.
- An evaluation of the speaker personality change perceived by listeners when transforming the vocal tract.

The outline of this section is as follows. First, a common experiment framework is established for all the system evaluations. Then, the number of components for GMM systems and the number of states for HMM systems are selected by means of v-fold cross-validation. Next, the results for the objective test are presented. The objective test consists in a normalized performance index, useful to make comparisons between systems trained with different number of sentences and with different source-target speaker pairs. Finally, the results for the perceptual tests are presented. Two different perceptual test have been carried out: the extended ABX test and the similarity test. Additionally, a MOS test to evaluate the quality of the converted speech was performed.

4.7.1 Experiment Framework

The corpora used for the experiments consisted in two different corpora, one containing a male and a female speaker (called Male_1 and Female_1 speakers) recorded for a TTS and another corpus containing four speakers, two males (Male_2 and Male_3) and two females (Female_2 and Female_3), "mimicked" recorded. Speech signals were resampled to 16kHz, and a pitch-synchronous LPC analysis of order 20 was carried out in order to estimate the LSF parameters. See chapter 3 for a detailed description of the corpora and speech parameterization.

All the experiments have been carried out varying the source-target speaker pair, the vocal tract mapping function and the number of training sentences. Every available speaker combination is evaluated, what results in 12 intra-gender and 18 cross-gender conversions. Four different vocal tract mappings have been considered: joint GMM regression, HMM-based mapping, CART

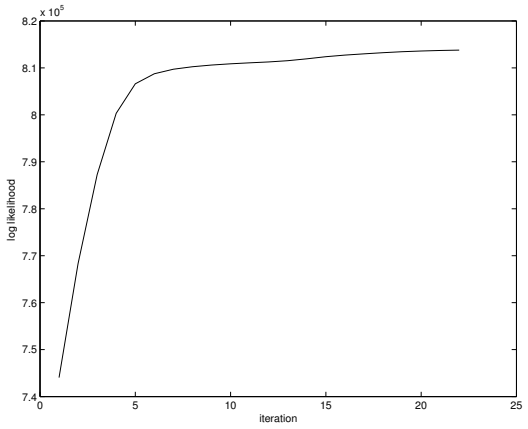


Figure 4.10: Likelihood evolution during EM iterations for a GMM model.

with pre-pruning and CART with post-pruning systems. Four different training sets have been used. Set_30, Set_20, Set_10 and Set_05 refer to systems trained with 30 sentences (about 1.5 minutes), 20 sentences (about 1 minute), 10 sentences (about 0.5 minutes) and 5 sentences (about 15 seconds) respectively. Each system have been evaluated with a set of 20 sentences, about 1 minute of speech, not included in any of the training sets. The contents of each set were selected with a maximum phoneme and diphone appearance criteria.

The initial values for means and covariances of the GMM models were set equal to the centroids and class dispersions of a previous k-means clustering. The EM algorithm for GMM estimation has been iterated until the increment of the likelihood was less than 0.01% of the likelihood of the previous iteration. Figure 4.10 illustrates the likelihood evolution during the EM iterations for an estimation of a joint GMM instance.

The HMMs used in this dissertation have been estimated with the Hidden Markov Model Toolkit HTK[1].

4.7.2 Component and State Number Selection

The architecture of GMM and HMM systems depends on a parameter to be chosen before the model estimation: the number of GMM components and the number of HMM states respectively. These parameters will have a direct influence in the final system performance when using the models for vocal tract conversion. In a general trend, a model with more components/states will be able to represent the data more accurately. However, when increasing the component/state number without increasing the amount of training data an over-fitting problem may appear. Over-fitting refers to the effect of fitting the parameters of a model to a training data until the model looses its generalization property.

[1]HTK is a portable toolkit for building and manipulating hidden Markov models (http://htk.eng.cam.ac.uk/)

A strategy to determine the component/state number is validation. Validation consists in dividing the available training data in two subsets: the train subset and the validation subset. Several models are estimated with the train subset, using different parameter values. Each model performance is evaluated on the validation subset. The model with the best performance on the validation subset is chosen. The main drawback of the validation strategy is that not all the available data is used to estimate the model. In problems with few available data, as in the VC task, this lost of information may be inadmissible.

The strategy used in the current experiments to determine the GMM component number and HMM state number is v-fold cross-validation, with $v = 10$. The number of normal distributions in the HMM states has been fixed to one, as it is desired that each state model only one acoustic class.

V-fold cross-validation is a method for estimating a model performance based on resampling. In v-fold cross-validation, the available data is divided into v subsets of approximately equal size. The model is estimated v times, each time leaving out one of the subsets from training, but using only the omitted subset to compute the performance. The final estimation for the model performance is the average of the v performances.

V-fold cross-validation can be used for parameter selection by estimating v times several models with different parameter values. The parameter that results in a model with the best estimated final performance is chosen. Once the parameter value has been determined, a new model can be estimated using all the available data. Therefore, there is no lost of training information in the model estimation.

Tables of pages 77 and 78 contain the GMM component number and HMM state number for all the combinations of source-target speaker pairs and sets of training sentences under study, determined by 10-fold cross-validation. Systems with $\{2, 4, 8, 16\}$ GMM components and $\{2, 4, 8, 16\}$ HMM states have been considered. A previous informal study showed that it was no necessary considering more than 16 GMM components or HMM states.

According to the results, the number of training sentences has more influence in the component/state number selection than the source-target speaker pair. A GMM with 2 components and a HMM with 2 states are always selected for systems trained with 5 sentences, whereas up to 16 GMM components and 8 HMM states were selected for systems trained with 30 sentences. For the sake of comparisons, figure 4.11 displays the component/state number selection according to the number of training sentences, averaged over all the source-target speaker pairs. It must be remarked that the number of HMM states is never greater than the number of GMM components for the same number of training sentences. It is expected that HMMs require more training data for their estimation in order to capture dynamic information.

There are few situations where a greater number of components/states were selected for a

Speaker	Male_1	Female_1	Male_2	Male_3	Female_2	Female_3
Male_1	-	8	4	4	4	4
Female_1	8	-	16	8	4	16
Male_2	8	4	-	8	8	16
Male_3	8	8	8	-	4	8
Female_2	8	4	4	8	-	8
Female_3	8	16	8	8	8	-

Table 4.3: Number of GMM components for the conversion from the speaker of the first column to the speaker of the first line, using 30 sentences for training.

Speaker	Male_1	Female_1	Male_2	Male_3	Female_2	Female_3
Male_1	-	4	4	4	4	4
Female_1	4	-	8	8	4	4
Male_2	8	4	-	8	4	4
Male_3	4	4	8	-	8	4
Female_2	4	4	4	8	-	8
Female_3	4	4	16	8	8	-

Table 4.4: Number of GMM components for the conversion from the speaker of the first column to the speaker of the first line, using 20 sentences for training.

Speaker	Male_1	Female_1	Male_2	Male_3	Female_2	Female_3
Male_1	-	4	4	4	4	4
Female_1	4	-	4	8	4	4
Male_2	4	4	-	4	4	4
Male_3	4	4	4	-	4	4
Female_2	4	8	4	4	-	4
Female_3	4	4	8	8	4	-

Table 4.5: Number of GMM components for the conversion from the speaker of the first column to the speaker of the first line, using 10 sentences for training.

Speaker	Male_1	Female_1	Male_2	Male_3	Female_2	Female_3
Male_1	-	2	2	2	2	2
Female_1	2	-	2	2	2	2
Male_2	2	2	-	2	2	2
Male_3	2	2	2	-	2	2
Female_2	2	2	2	2	-	2
Female_3	2	2	2	2	2	-

Table 4.6: Number of GMM components for the conversion from the speaker of the first column to the speaker of the first line, using 5 sentences for training.

Speaker	Male_1	Female_1	Male_2	Male_3	Female_2	Female_3
Male_1	-	8	4	4	4	4
Female_1	8	-	8	8	4	4
Male_2	8	8	-	8	4	4
Male_3	8	8	8	-	4	4
Female_2	8	8	4	4	-	4
Female_3	8	8	8	8	8	-

Table 4.7: Number of HMM states for the conversion from the speaker of the first column to the speaker of the first line, using 30 sentences for training.

Speaker	Male_1	Female_1	Male_2	Male_3	Female_2	Female_3
Male_1	-	4	4	4	4	4
Female_1	4	-	8	8	4	4
Male_2	8	4	-	4	4	4
Male_3	4	4	4	-	4	4
Female_2	8	4	4	4	-	4
Female_3	4	4	4	8	4	-

Table 4.8: Number of HMM states for the conversion from the speaker of the first column to the speaker of the first line, using 20 sentences for training.

Speaker	Male_1	Female_1	Male_2	Male_3	Female_2	Female_3
Male_1	-	4	4	4	4	4
Female_1	4	-	4	4	4	4
Male_2	4	4	-	4	4	4
Male_3	4	4	4	-	4	4
Female_2	4	8	4	4	-	4
Female_3	4	4	4	8	4	-

Table 4.9: Number of HMM states for the conversion from the speaker of the first column to the speaker of the first line, using 10 sentences for training.

Speaker	Male_1	Female_1	Male_2	Male_3	Female_2	Female_3
Male_1	-	2	2	2	2	2
Female_1	2	-	2	2	2	2
Male_2	2	2	-	2	2	2
Male_3	2	2	2	-	2	2
Female_2	2	2	2	2	-	2
Female_3	2	2	2	2	2	-

Table 4.10: Number of HMM states for the conversion from the speaker of the first column to the speaker of the first line, using 5 sentences for training.

Speaker	Male_1	Female_1	Male_2	Male_3	Female_2	Female_3
Male_1	-	12	9	12	5	11
Female_1	7	-	8	8	7	4
Male_2	11	10	-	10	6	9
Male_3	14	11	10	-	6	8
Female_2	12	12	7	10	-	7
Female_3	15	8	9	9	11	-

Table 4.11: Number of CART with pre-pruning leaves for the conversion from the speaker of the first column to the speaker of the first line, using 30 sentences for training.

Speaker	Male_1	Female_1	Male_2	Male_3	Female_2	Female_3
Male_1	-	4	6	5	2	6
Female_1	7	-	6	5	6	7
Male_2	11	11	-	9	6	5
Male_3	8	11	4	-	4	4
Female_2	13	13	4	3	-	7
Female_3	11	9	7	9	4	-

Table 4.12: Number of CART with pre-pruning leaves for the conversion from the speaker of the first column to the speaker of the first line, using 20 sentences for training.

Speaker	Male_1	Female_1	Male_2	Male_3	Female_2	Female_3
Male_1	-	2	6	6	3	4
Female_1	3	-	4	6	3	1
Male_2	7	9	-	4	1	3
Male_3	6	5	1	-	2	4
Female_2	6	9	3	3	-	3
Female_3	8	9	1	7	5	-

Table 4.13: Number of CART with pre-pruning leaves for the conversion from the speaker of the first column to the speaker of the first line, using 10 sentences for training.

Speaker	Male_1	Female_1	Male_2	Male_3	Female_2	Female_3
Male_1	-	1	3	2	1	1
Female_1	2	-	10	5	2	1
Male_2	2	2	-	1	1	1
Male_3	3	6	1	-	1	1
Female_2	2	12	6	6	-	6
Female_3	8	14	12	7	1	-

Table 4.14: Number of CART with pre-pruning leaves for the conversion from the speaker of the first column to the speaker of the first line, using 5 sentences for training.

Speaker	Male_1	Female_1	Male_2	Male_3	Female_2	Female_3
Male_1	-	12	9	12	5	11
Female_1	7	-	11	8	7	6
Male_2	14	17	-	12	9	12
Male_3	11	13	10	-	12	14
Female_2	15	12	7	10	-	10
Female_3	15	16	9	12	11	-

Table 4.15: Number of CART with post-pruning leaves for the conversion from the speaker of the first column to the speaker of the first line, using 30 sentences for training.

Speaker	Male_1	Female_1	Male_2	Male_3	Female_2	Female_3
Male_1	-	8	8	9	10	6
Female_1	11	-	6	7	15	11
Male_2	19	16	-	13	14	11
Male_3	12	11	9	-	12	11
Female_2	15	19	16	15	-	7
Female_3	17	14	7	9	9	-

Table 4.16: Number of CART with post-pruning leaves for the conversion from the speaker of the first column to the speaker of the first line, using 20 sentences for training.

Speaker	Male_1	Female_1	Male_2	Male_3	Female_2	Female_3
Male_1	-	9	12	6	13	13
Female_1	11	-	12	10	13	14
Male_2	15	17	-	16	16	15
Male_3	10	10	12	-	12	15
Female_2	12	17	20	16	-	14
Female_3	18	17	14	14	16	-

Table 4.17: Number of CART with post-pruning leaves for the conversion from the speaker of the first column to the speaker of the first line, using 10 sentences for training.

Speaker	Male_1	Female_1	Male_2	Male_3	Female_2	Female_3
Male_1	-	17	16	17	21	16
Female_1	14	-	18	16	19	15
Male_2	19	19	-	19	19	15
Male_3	16	18	17	-	19	16
Female_2	21	22	20	22	-	19
Female_3	15	18	15	19	21	-

Table 4.18: Number of CART with post-pruning leaves for the conversion from the speaker of the first column to the speaker of the first line, using 5 sentences for training.

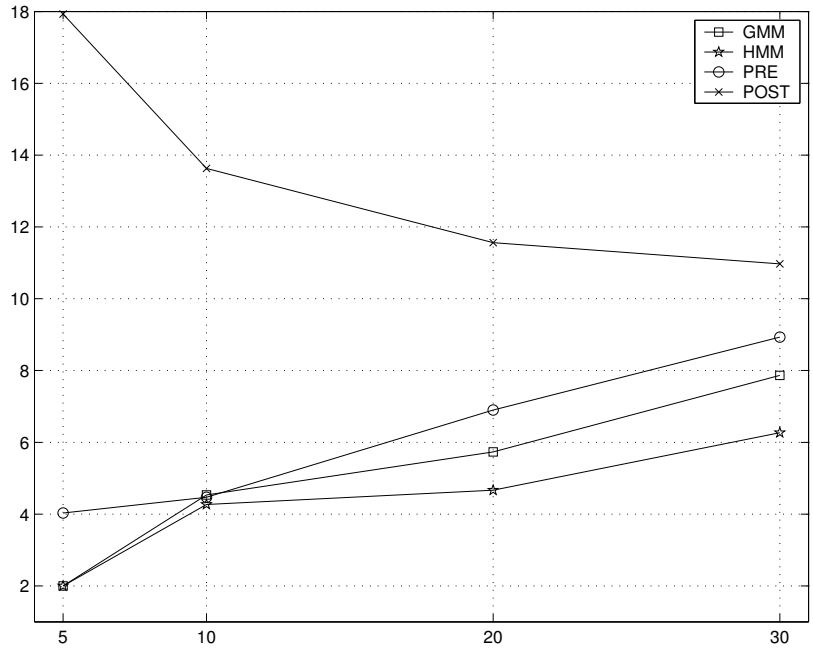


Figure 4.11: Number of components/states/leaves for GMM (squares), HMM (stars), CART with pre-pruning (circles) and CART with post-pruning (crosses) systems.

system trained with less training data. For example, the number of GMM components selected for the speaker pair Female_3-Male_2 was 8 for 30 training sentences and 16 for 20 training sentences. This is due to the component/state number selection process. For every number of training sentences, the component/state number was selected from the set $\{2, 4, 8, 16\}$, without taking into account the other already trained systems.

In order to compare the number of phonetic classes detected by both CART systems with the number of acoustic classes modeled by GMM and HMM systems, tables of pages 79 and 80 contain the number of leaves for the CART with pre-pruning and CART with post-pruning systems trained with all the available source-target speaker pairs and sets of training sentences. Those tables illustrate how the pre-pruning method prevents the tree from growing in a major degree than the post-pruning method. As expected, trees grown with post-pruning have more leaves than trees grown with pre-pruning.

As in the GMM and HMM cases, the number of leaves for both CART systems depends on the number of training sentences. However, the source-target speaker pair has slightly more influence in CART systems than in the previous studied systems.

When comparing the number of acoustic classes of GMM systems and CART systems, it is seen that CART with pre-pruning detects a number of phonetic classes similar to the number of GMM components. This is not the case of CART with post-pruning. It must be remarked the suspicious performance of the CART post-pruning systems trained with 10 or 5 sentences (see

figure 4.11). The high number of phonetic classes detected may be a symptom of over-fitting to the validation data when doing the pruning. This effect will be explored in the next section.

4.7.3 Objective Test: Results and Discussion

This section presents the results and discussion of the objective test for the vocal tract conversion task. The objective test consisted in calculating the mean performance of the four systems under study: joint GMM regression (labeled as GMM), HMM based vocal tract conversion (labelled as HMM), CART with pre-pruning (labelled as PRE) and CART with post-pruning (labelled as POST). The four systems were trained with sets Set_30, Set_20, Set_10 and Set_05, for all the possible source-target speaker pairs.

The performance is calculated as the index P:

$$P = 1 - \frac{D(\tilde{\mathscr{Y}}, \mathscr{Y})}{D(\mathscr{X}, \mathscr{Y})} \tag{4.42}$$

where the function $D(\cdot)$ indicates a perceptually weighted distance between the source vocal tract vectors $\mathscr{X}$, the target vocal tract vectors $\mathscr{Y}$ and the converted vocal tract vectors $\tilde{\mathscr{Y}}$. The detailed presentation of the performance index is given in chapter 3, page 45. A VC system that does not modify the source voice will lead to $P = 0$, whereas the optimal conversion will result in $P = 1$.

Figure 4.12 shows the performance of the four systems under study averaged over all the source-target speaker pairs, when the systems have been trained with the four different sets of sentences. We first observe that as the number of training sentences increases the performance of the four systems also increases. It is mainly remarkable for the CART with post-pruning system, which presents the larger differences in performance values for consecutive training sets. Only the HMM system trained with Set_05 has a better mean performance than the HMM system trained with Set_10, but this is a marginal result.

The performance index of the HMM based vocal tract conversion system is always lower than the performance index for joint GMM regression, and the difference become greater as the training set contains less sentences. Although HMMs include dynamic information and theoretically they are expected to perform better than GMMs in the vocal tract conversion task, it seems that the amount of training data is a major constriction for HMM estimation. Therefore, HMMs estimated with the available data result in models with worse performance. This lack of data makes the model unreliable for training sets containing few sentences.

CART systems presents better performance than joint GMM regression for training sets Set_30, Set_20 and Set_10. Moreover, the performance of both CART systems trained with Set_20

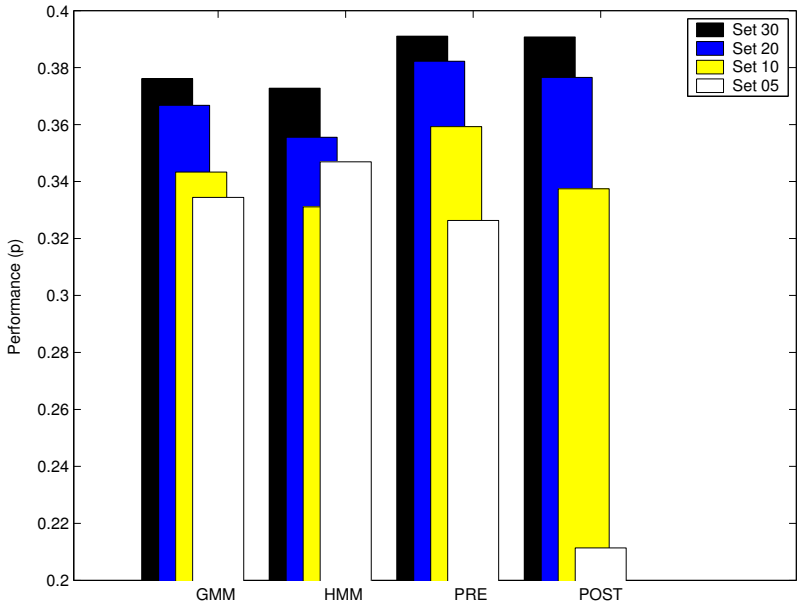


Figure 4.12: Mean performance for GMM, HMM, CART with pre-pruning and CART with post-pruning systems trained with Set_30, Set_20, Set_10 and Set_05.

are greater than the performance of the joint GMM regression system trained with Set_30:

$$P_{CART}^{Set\_20} > P_{GMM}^{Set\_30}.$$

CART systems are more suitable to the vocal tract conversion task, due to the inclusion of phonetic information in the clustering of the acoustic space previous to the application of the transformation function. Using CART systems it is possible to have the same performance than with joint GMM regression, but with less training sentences. However, there is a lower limit of training information, and using less data will result in CART systems with worse performance than GMM systems. This effect is noticeable in the CART systems trained with 5 sentences. When there is few amount of training data, acoustic information is more relevant than phonetic information to carry out the clustering for the conversion task.

If we compare CART with pre-pruning and CART with post-pruning, the former system presents better performance index values. The main difference between both systems is the process by a node is declared a leaf. CART with pre-pruning is a *conservative* system, i.e. the stop splitting rule is taken locally, the first time that there is an accuracy reduction estimated on the validation data. On the other hand, the post-pruning method relays on the validation data to keep branches that in deeper splits increment the tree accuracy. When dealing with problems with a medium amount of validation data, post-pruning methods will prevent the tree to stop the splitting in an early step. However, in the vocal tract conversion task, few validation data is available. For this reason, the CART with post-pruning system has fitted the tree architecture too much to the validation data and the tree has lost its generalization property.

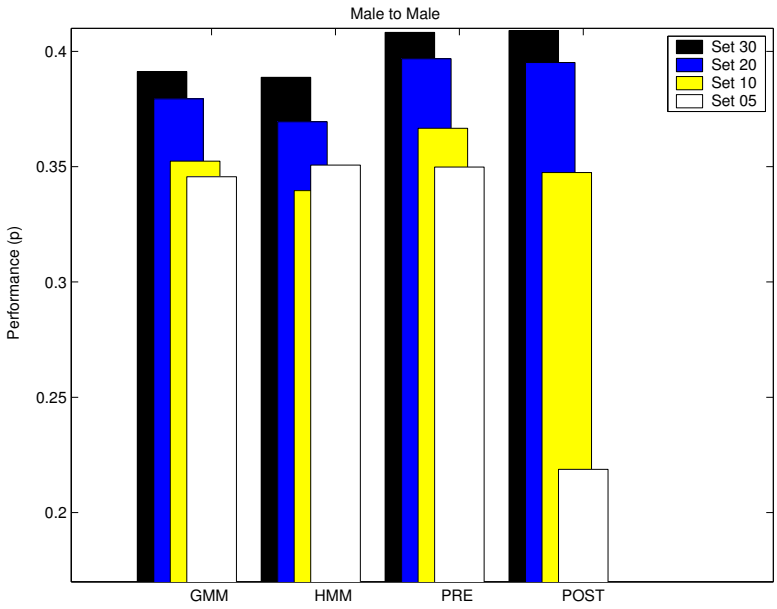


Figure 4.13: Mean performance for GMM, HMM, CART with pre-pruning and CART with post-pruning systems trained with Set_30, Set_20, Set_10 and Set_03 for male source speakers and male target speakers.

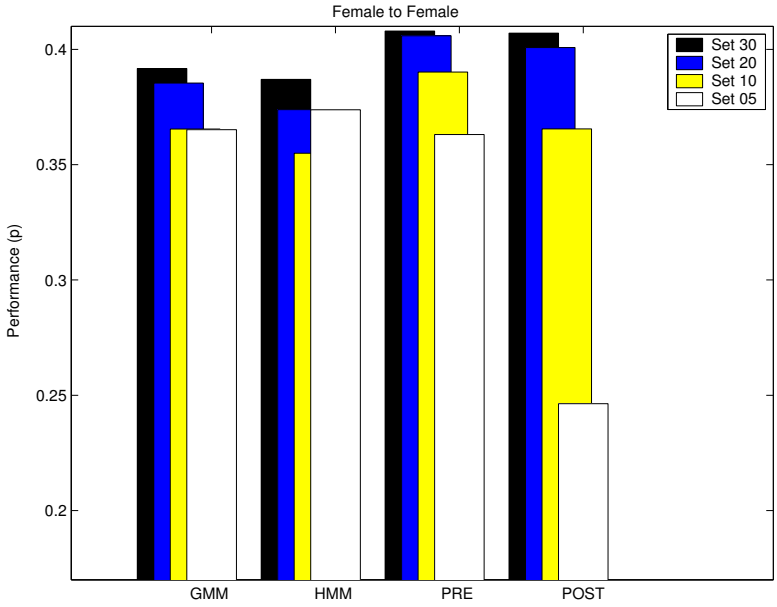


Figure 4.14: Mean performance for GMM, HMM, CART with pre-pruning and CART with post-pruning systems trained with Set_30, Set_20, Set_10 and Set_03 for female source speakers and female target speakers.

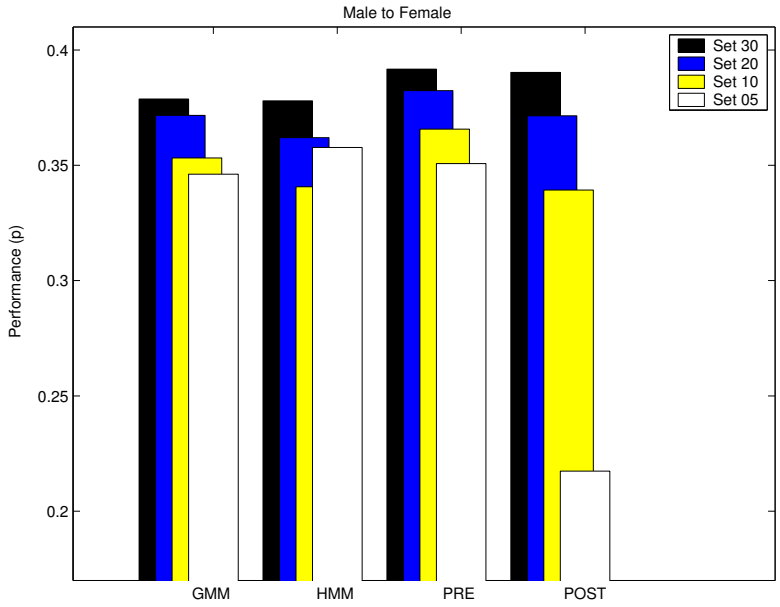


Figure 4.15: Mean performance for GMM, HMM, CART with pre-pruning and CART with post-pruning systems trained with Set_30, Set_20, Set_10 and Set_03 for male source speakers and female target speakers.

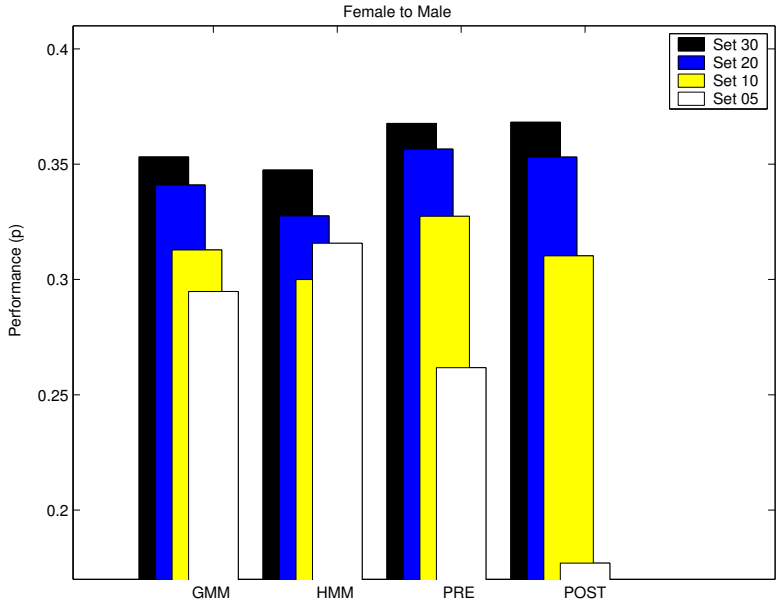


Figure 4.16: Mean performance for GMM, HMM, CART with pre-pruning and CART with post-pruning systems trained with Set_30, Set_20, Set_10 and Set_03 for female source speakers and male target speakers.

Figures from 4.13 to 4.16 display the performance of the four systems under study according to the gender of the source and target speakers. Intra-gender conversion performance is better for all the studied systems than cross-gender conversion performance. In particular, the conversion from a female speaker to a male speaker is the worst case. Although all the speakers have different vocal tract frequency response, there are some aspects particular to the gender of the speaker. For instance, in average the formant locations for female speakers are higher than for male speakers, due to the vocal tract length of females speakers is usually shorter than the vocal tract length for males speakers. Therefore, in average, the vocal tract frequency response of two speakers of the same gender are closer than the vocal tract frequency response of two speakers of different gender.

The detailed results for each one of the source-target speaker pair trained systems have been included in the Appendix B. The most relevant conclusion of these results is that systems involving the professional and manually supervised corpus speakers (MALE_1 and FEMALE_1) have better performance values than other similar systems. And what's more, cross-gender conversions between a manually supervised speaker and another speaker have better performance than intra-gender conversions involving two non-supervised speakers. This statement is true for all the studied systems: joint GMM regression, HMM based vocal tract conversion, CART with pre-pruning and CART with post-pruning systems.

The voice quality of the recordings of the two professional speakers (an actor and an actress) is with no doubt higher than the voice quality of the four non-professional speakers, in terms of vocalization, noises due to breathings, etc. Therefore, it is expected than experiments involving professional speakers will lead to better results.

The supervision of the professional speaker corpus consisted in a manually validation or correction of the phonetic transcription, the phonetic segmentation and the position of the pitch marks. Phonetic transcription and segmentation have been used in the alignment procedure, shared by the four systems. The alignment determines the training data to estimate the mapping functions. Therefore, a better alignment will result in a mapping with better performance.

Phonetic information is also critical in the CART system estimation. In order to study the effect of the two phonetic annotation systems used (the supervised one and the non-supervised one), the system performances have been averaged according to the involved speakers. Figure 4.17 displays the average performance of the trained systems involving at least one manually supervised speaker and figure 4.18 displays the average performance of the trained systems involving two non-supervised speakers. CART systems estimated with at least one supervised speaker are clearly better than joint GMM regression and HMM based vocal tract systems, but the performance of CART systems estimated with two non-supervised speakers decays to the level of GMM and HMM systems. Therefore, CART mappings will be only useful for systems trained from data with good phonetic transcription and segmentation. This is not a problem for

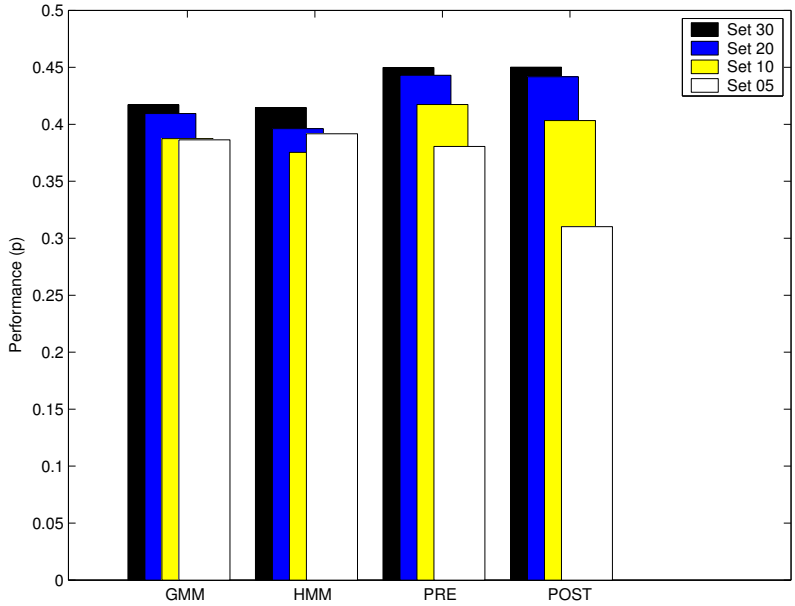


Figure 4.17: Mean performance for GMM, HMM, CART with pre-pruning and CART with post-pruning systems trained with at least one supervised speaker.

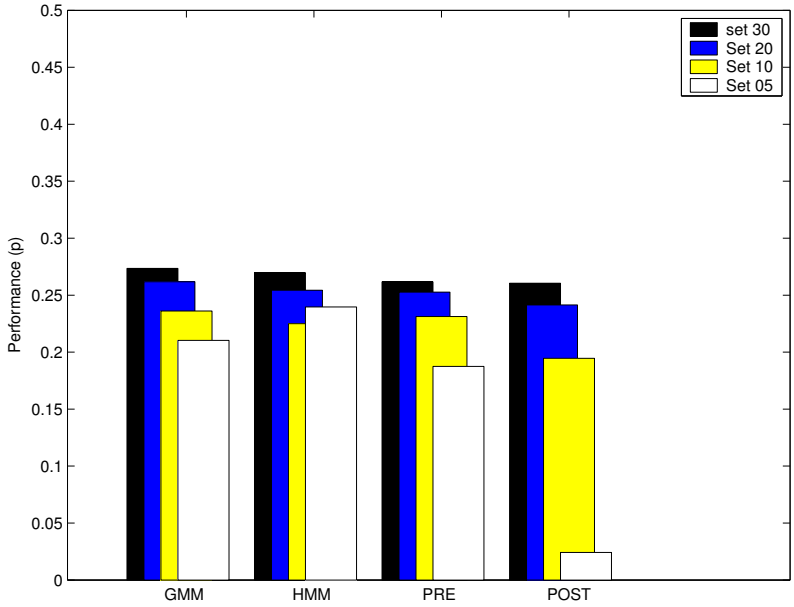


Figure 4.18: Mean performance for GMM, HMM, CART with pre-pruning and CART with post-pruning systems trained with two non-supervised speakers.

the application of CART systems as a post-processing block in a TTS, because the corpus of the TTS is almost always manually supervised and uttered by a speech professional (actors/actresses, announcers, etc.).

Figure 4.20 and 4.21 display a representative example of vocal tract conversion between one supervised speaker and one non-supervised speaker according to joint GMM regression and according to CART with pre-pruning. The converted vocal tract by the CART system fits better in low frequencies and has a greater dynamic range than the converted vocal tract by the GMM system. This is due, at least in part, that there is less averaging operations in the CART system than in the GMM system. Therefore, the converted by CART vocal tract response does not tend to flatter as much as the converted by GMM vocal tract response.

Finally, figure 4.19 illustrates the evolution of the performance index from frame to frame for the GMM and CART with pre-pruning systems. Although in a general trend both GMM and CART systems have irregular evolution of the performance index, the CART with pre-pruning system presents less deep valleys.

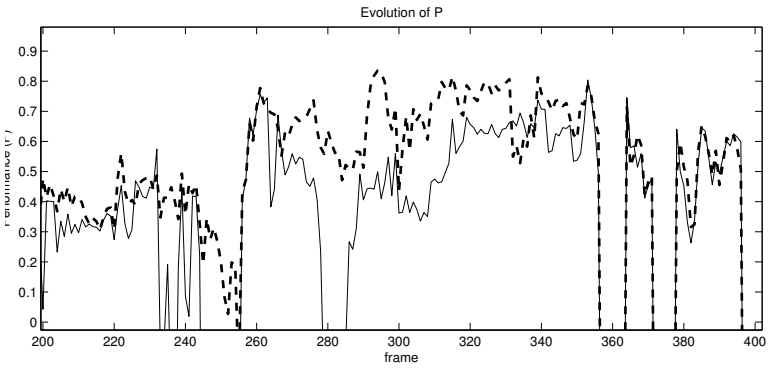


Figure 4.19: Performance index evolution according to the frame number. Solid line: GMM conversion. Dashed line: CART conversion.

4.7.4 Perceptual Tests: Results and Discussion

Objective results have shown that the performance index depends on the source-target speaker pair, the number of training sentences and the vocal tract conversion system which carries out the transformation. To validate the objective results and to study the correlation of the performance index P with the human perception, a perceptual test set have been carried out. Two vocal tract conversion systems have been selected to be evaluated: a joint GMM regression trained with Set_30 and a CART with pre-pruning system trained with the same set of sentences. Transformations have been done with two source-target speaker pairs: one cross-gender conversion, from FEMALE_1 to MALE_1, and one intra-gender conversion, from MALE_1 to

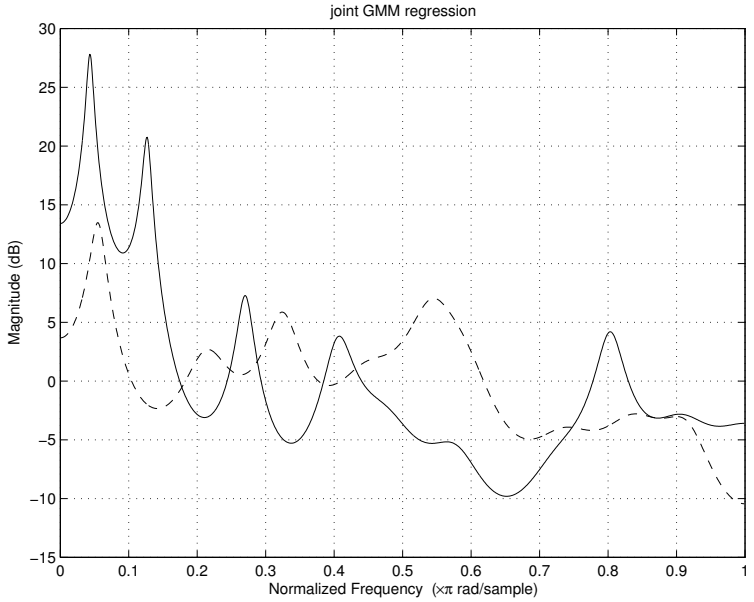


Figure 4.20: Vocal tract frequency response. Target speaker: solid line. Converted speaker by GMM: dashed line.

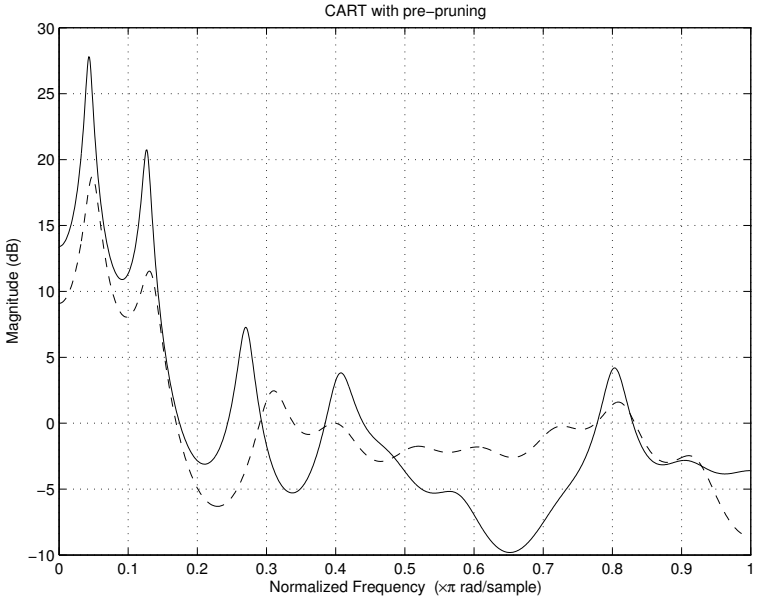


Figure 4.21: Vocal tract frequency response. Target speaker: solid line. Converted speaker by CART: dashed line.

MALE_2 [2].

The goal of the perceptual tests is the evaluation of the vocal tract conversion, independently of other speech characteristics. Therefore, the converted voice was generated with the converted vocal tract parameters, the target LP residual signal of an aligned target sentence and the target prosody. It is assumed to have an ideal residual and prosodic transformation system.

Two different perceptual tests have been carried out to evaluate the system ability to change the voice speaker identity: an extended ABX test and a similarity test. All the tests have been completed by twenty listeners. In the extended ABX test, listeners have to rate the similarity between X and A, B speech files. A and B files are from the source and target speakers, in a random order. X file is either the GMM converted voice, the CART converted voice, the source original voice or the target original voice. The three files correspond to different sentences.

The mean results for the extended ABX test are:

X	**Mean Score**
original	4.53
GMM	4.40
CART	4.20

Table 4.19: Mean Score of the extended ABX test.

where the scale has been normalized to:

1	**2**	**3**	**4**	**5**
X is the source speaker	X is similar to source speaker	X is neither the source nor the target	X is similar to target speaker	X is the target speaker

Both GMM and CART systems achieve the goal of changing the speaker identity of the source voice to the speaker identity of the target voice, as GMM and CART mean scores are between "similar to" and "the same than" the target speaker. It must be remarked that original recordings, when evaluated by means of an extended ABX test, do not achieve a score of 5. In some questions, two original recording of the same speaker have been rated as "similar", but not as "the same" speakers.

In the similarity test, listeners have to decided if two speech files come from the same speaker or from two different speakers. One of the speech files is either the source or the target speaker, while the other speech file can be the source, the target, the GMM converted or the CART

[2]Representative examples of the different evaluated conversions are in http://gps-tsc.upc.ed/veu/research/vc.php3.

converted voice. When comparing any voice with the source speaker, the converted voice is modify to match with the source prosodic characteristics. The two files of each pair correspond to different sentences.

The mean scores for the similarity test are displayed in table 4.20. The results correspond to the percentage of times that one speaker of the first row have been rated as the same/different speaker when compared with one speaker of the first column. The *original* speaker refers to both the source and target speakers.

	original		**source**		**target**	
	same	**different**	**same**	**different**	**same**	**different**
original	92.59	7.41				
GMM			2.78	97.22	86.11	13.89
CART			2.86	97.14	86.11	13.89

Table 4.20: Mean Score of the similarity test.The results correspond to the percentage of times that one speaker of the first row have been rated as the same/different speaker when compared with one speaker of the first column.

As GMM systems as CART systems place the converted speaker and the source speaker at a distance that about the 97% of the times converted and source speakers are rated to be different speakers. When the comparison is between converted and target speech files, both systems succeed another time. However, the percentage is 86.11%, whereas for original speech files the percentage is 92.59%.

According to the results, we conclude that both GMM systems and CART systems are able to transform vocal tract parameters in order to convert the source voice into the target voice. Moreover, the performance of both systems is very similar.

To finish the perceptual study of GMM and CART systems for the vocal tract conversion, a MOS test has been carried out. Listeners were asked to rate the speech quality of a set of sentences, read by the source, the target, the GMM converted and the CART converted speakers. Results are displayed in table 4.21.

	MOS
original	4.42
GMM	2.42
CART	2.50

Table 4.21: MOS results.

Converted speech were rated lower in quality than original speech. Further studies in vocal tract conversion should be focus on maintaining the original quality through the conversion. Quality of CART based systems was perceived slightly better than quality of GMM systems.

In a previous study published by the author [Dux04b], listeners of a preference test were presented couples of sentences, and they were asked to select the most natural sentence of each couple. When listeners had to choose between GMM converted speech files and CART converted speech files, 71% of the time listeners preferred CART converted speech in front of GMM converted speech. The listeners preference for CART systems may be due, at least in part, to the less number of averaging operations included in CART systems than in GMM systems.

4.7.5 Conclusions

Four different vocal tract conversion systems have been compared: a baseline system, based on joint GMM regression, and three novel systems, which include new information to the acoustic model. In particular, the first proposed system uses HMMs to include dynamic information of the speech signal. The other two proposed systems were based on CARTs to include phonetic information to the task.

The algorithm to estimate CART with pre-pruning systems is simpler than the estimation algorithm for GMMs and HMMs. CARTs do not need any parameter tuning, unlike the number of components of GMMs and the number of states of HMMs that have to be determined prior to the model estimation. Moreover, in practical situations the number of operations to estimate a CART with pre-pruning system is lower than the number of operations to estimate a GMM model, once the component number has been selected. We can not provide a theoretically prove of this statement, as the number of final leaves for CART systems and the number of iterations of the EM algorithm are unknown a priori, but for 30 training sentences GMM estimation lasts about 1 hour and CART estimation about 10 minutes.

Objective results have shown that the performance of HMM vocal tract conversion systems for a limited number of training data is not higher that the performance of joint GMM regression systems. As a consequence, HMM based conversion has been left out of further perceptual evaluations. However, it is still remaining the possibility that for enough training data HMM systems will over-perform GMM systems.

CART systems, specially CART with pre-pruning systems, resulted in the best performance when the training set contains at least 10 sentences. However, CART systems are more sensible to errors in the speech phonetic segmentation and transcription than the other systems. It seems to be a requirement that at least one of the involved speakers in the conversion is a professional and has a high quality phonetic information.

Perceptual test have corroborate than both GMM and CART with pre-pruning systems achieve the goal of changing the voice identity of the source speaker. Listeners reported a slightly better quality for CART systems than for GMM systems. The listeners preference for CART

systems may be due, at least in part, to the less number of averaging operations included in CART systems than in GMM systems.

4.8 Summary

Vocal tract conversion can be generally divided in three stages: a model of the acoustic space with a structure by classes, an acoustic classification machine and a mapping function. In the state of the art vocal tract conversion systems, GMMs are used to model the speech spectrum parameters. GMMs provide a soft classification into their components, which can be seen as acoustical classes. Moreover, mapping functions to convert vocal tract parameters are based on the Gaussian mixtures, by performing a regression of joint source-target probability density functions.

Systems based on GMMs are well suited to the vocal tract conversion task, but they can not deal with source data without its corresponding parallel target data. Two approaches for the used of source non-parallel data in GMM systems have been proposed: a modified EM algorithm with fixed covariance matrices, and a strategy to complete non-parallel data by including transformed vectors as parallel vectors. According to objective results, a combined learning with parallel source-target data and source-transformed data increases the conversion performance, mainly when few training data is available. In this latest situation, to re-estimate only the means and the mixture weights also increases the performance, with a very reduced computational time.

As an alternative to GMMs, two novel vocal tract conversion systems have been proposed: a HMM based vocal tract conversion systems and a CART based vocal tract conversion systems.

HMM based vocal tract conversion systems include dynamic information into the conversion task, in order to transform one frame according to previous and posterior frames. The objective of including dynamic information is to better convert phoneme boundary frames. However, objective results have reveled that, for a limited amount of training data, GMM systems perform better than HMM systems. Although HMMs include dynamic information and theoretically they are expected to perform better than GMMs in the vocal tract conversion task, it seems that the amount of training data is a major constriction for HMM estimation. The lack of training data makes the model unreliable for training sets containing few sentences.

CART based vocal tract conversion systems have been proposed to include phonetic information in the conversion process. In particular, a decision tree is used to classified the acoustic parameters prior to the application of the mapping function. Unlike GMM and HMM based vocal tract conversion systems, the classification provided by decision trees is hard. Although soft classification is more flexible and less sensible to errors due to spectral jumps, it is believed that phonetic data carries information that allows to better split the acoustic space according to

the transformation error. Moreover, it is expected that two continuous source vectors produce two similar transformed vectors, if conversion functions are reliable estimated.

According to objective and perceptual results, CART systems, specially CART with pre-pruning systems, succeed in the vocal tract conversion task. The tree estimation, compared with the GMM estimation, is simpler and does not need any parameter tuning. Moreover, listener reported a slightly preference for CART converted voices in front of GMM converted voices. The listeners preference for CART systems may be due, at least in part, to the less number of averaging operations included in CART systems than in GMM systems.

Next chapter is focused on the LP residual signal modification. Vocal tract mapping plus LP residual modification will constitute a complete VC system able to change the timbre of a source speaker's voice into the timbre of a target speaker's voice.

Chapter 5

LP Residual Signal Modification

This chapter discuss the motivation of modifying the source LP residual signal in order to achieve an effective change of the speaker personality.

After an introduction, the most relevant studies on the current topic are analyzed in section 5.2, showing that two different strategies have been used to face the residual signal modification problem. On one hand, the converted residual can be obtained by a modification of the source residual signal. On the other hand, the converted residual can be predicted from the converted vocal tract parameters. Both strategies are compared in section 5.3. Finally, a Phonetic Residual Prediction and Smoothing method is proposed in section 5.4, as a new system which alleviates the computational load of the residual selection state of the art systems.

5.1 Introduction to Residual Modification

Many of the research efforts in VC have been carried out in the field of vocal tract mapping. Vocal tract parameters, such as LSF vectors, condense the most relevant part of the information about speaker identity. However, speech generated with converted vocal tract parameters using the source LP residual signal as an excitation contains characteristics of the source speaker that prevents an effective impersonalization. Informal listening tests have reveled that when transforming only vocal tract parameters and adjusting the mean pitch value, the converted signal sounds as a third speaker, not the source neither the target speaker.

The most part of the VC systems published do not work with a vocal tract model that takes into account a really glottal flow model [Dux02]. For instance, in this thesis dissertation LPC is used as a model of vocal tract. Therefore, the LP residual signal, as the error of the LPC analysis, contains all the effects of the speech signal which are not taken into account by the assumptions of the LPC model.

Some of these effects are:

- Resonances not modeled by the LPC filter due to order restrictions.
- Spectral zeros.
- Glottal pulse shape characteristics, such as secondary glottal pulses or noise in the closed phase.
- Phase incoherence due to the minimum phase assumption of the LPC filter.

These effects are speaker dependent. Due to that, LP residual signals of two different speakers are not interchangeable.

An alternative to use source LP residual signal as excitations is the use of a LPC Vocoder. A LPC Vocoder generates the output speech by feeding LPC filters with artificial residual signal. Residuals are produced by a train of impulses when the speech is voiced and by white Gaussian noise when the speech is unvoiced. The main drawback of LPC Vocoders is the lack of naturalness of the generated speech. A Vocoder voice is not rated as natural as a real voice. Spectral details of real speech are lost by the simplification of the artificial residual signal.

Due to the previous discussion, we conclude that to achieve a high speech quality, naturalness and an effective change in the speaker individuality it is also needed to adapt the glottal flow characteristics of the source speaker to those of the target speaker. To sum up, LP residual modification may be seen as a way of boosting the personality change and reintroducing spectral details to the converted speech.

5.2 Previous Studies on Residual Modification

The residual modification problem is a very new subject of study. There is few bibliography on this topic and a comparative study about different published strategies is almost inexistent. The goal of this section is to provide a complete overview of previous works on LP residual signal modification for the VC task, in order to understand and compare the different strategies used to solve the problem. In particular, the five most relevant techniques will be described.

5.2.1 Transformation based on Nonlinear Prediction Analysis

Lee et al. presented a conversion system [Lee96] where voiced residuals signals are modeled by a long-delay neural net predictor, whose parameters are converted by a mapping codebook.

Neural net predictors can be used for modeling non-linear data without any prior assumption about the form of the non-linearity. When performing a LPC estimation, the linearly predictable

component of the speech signal is removed. Therefore, neural nets are highly suitable to model the residual signal.

In particular, the neural net predictor used in the referred work consists in a three layer network, which contain three units in the input layer, two units in the hidden layer and one unit in the output layer (see figure 5.1).

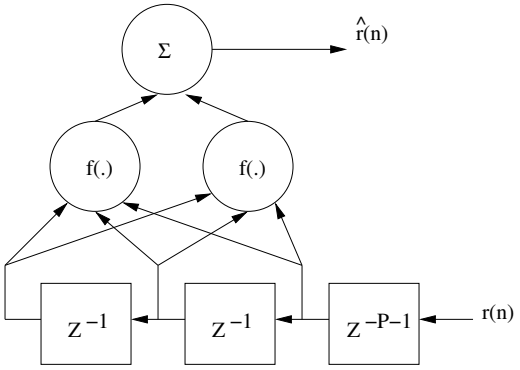


Figure 5.1: Long delay neural net predictor.

The mathematical expression of the predictor can be stated as:

$$\hat{\mathbf{r}}(n) = \sum_{j=1}^{2} w_j f\Big(\sum_{i=-1}^{1} v_{ij} \mathbf{r}(n - P - i)\Big), \tag{5.1}$$

where $\mathbf{r}(n)$ is the residual frame to be modeled and P its pitch period. Therefore, the number of parameters of the residual model is eight, two w_j of the hidden layer, and six v_{ij} of the input layer.

The training of the residual transformation system consists in modeling the voiced residual frames of the source and target training data by the neural net predictor. Afterwards, a mapping codebook, whose codevectors represents the eight parameters of the predictors, is estimated.

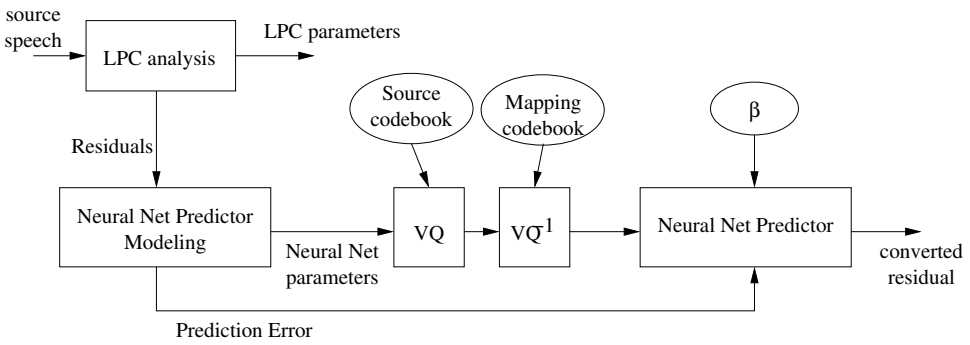


Figure 5.2: Block diagram of the operation mode of the Transformation based on Nonlinear Prediction Analysis for voiced speech.

Figure 5.2 illustrates the residual transformation process. The first step to transform a new source residual signal is the modification of the pitch period by the average pitch modificator factor β, defined as:

$$\beta = \frac{\bar{P}_{target}}{\bar{P}_{source}}, \tag{5.2}$$

where $\bar{P}_{target}$ and $\bar{P}_{source}$ are the average pitch periods of the target and source speaker. Finally, the converted residual signal $\tilde{r}(n)$ can be expressed by:

$$\mathbf{r}_c(n) = \sum_{j=1}^{2} \hat{w}_j f\Big(\sum_{i=-1}^{1} \hat{v}_{ij} \hat{\mathbf{r}}_s(n - \beta P - i)\Big) + \mathbf{e}_s(n), \tag{5.3}$$

where $\mathbf{e}_s(n)$ is the predictive error of the source residual signal given by $\mathbf{e}_s(n) = \hat{\mathbf{r}}_s(n) - \mathbf{r}_s(n)$, and $\hat{w}$, $\hat{v}$ are the mapped parameters.

Transformation based on Nonlinear prediction analysis only deals with voiced frames, while keeping source unvoiced residual frames in the converted speech.

5.2.2 Speaker Transformation Algorithm using Segmental Codebooks

Arslan [Ars99] published the Speaker Transformation Algorithm using Segmental Codebooks (STASC) system, where the conversion applied to each source residual frame depends on a weighted sum of residual transformation filters.

This system is based on the construction, during the training phase, of two mapping codebooks: one for LSF parameters and another for LP residual frames. Each codebook contains one codevector per phoneme. To built the codebooks, speech is automatically phonetically segmented. Then, LSF parameters (or residual frames) are collected for each phoneme, and the centroids for each codebook class are calculated as an average of the LSF vectors (or magnitude spectrum of the residual signals) corresponding to each phoneme.

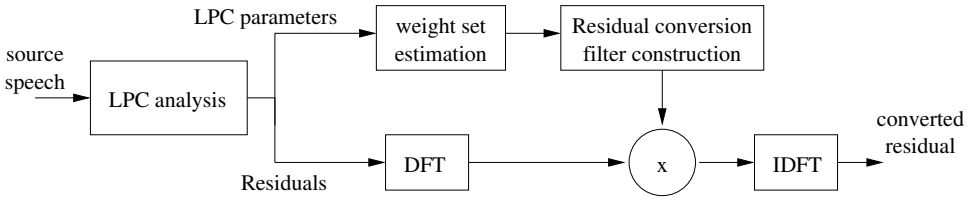


Figure 5.3: Block diagram of the operation mode of the STASC.

Figure 5.3 shows the block diagram of the residual conversion process. To carried out the conversion of a new source residual frame, a perceptual weighted distance $\mathbf{D} = (d_1, d_2, ..., d_L)$

(L is the codebook size) between its LSF parameters and all the codevectors of the source LSF codebook is calculated. Based on the distance from each codebook entry, an approximated source LSF vector $\hat{\mathbf{x}}$ can be expressed as:

$$\hat{\mathbf{x}} = \sum_{i=1}^{L} v_i \mathbf{S}_i, \tag{5.4}$$

where $\mathbf{S}_i$ denotes the i^{th} source LSF codevector. The parameters v_i correspond to:

$$v_i = \frac{\mathbf{e}^{-\gamma d_i}}{\sum_{l=1}^{L} \mathbf{e}^{-\gamma d_l}} \qquad i = 1, ..., L. \tag{5.5}$$

The value of γ for each frame is found by an incremental search with the criterion of minimizing the perceptual weighted distance between the approximated LSF vector $\hat{\mathbf{x}}$ and the original vector $\mathbf{x}$. To further improve the estimation of γ a gradient descendant algorithm is also run. More details about the estimation of γ can be found in [Ars99].

The parameter γ can be regarded as information about the phonetic content of the current speech frame. Once γ is obtained, the same set of weights v_i is used to construct the residual conversion filter:

$$\mathbf{H}(w) = \sum_{i=1}^{L} v_i \frac{\mathbf{U}_i^t(w)}{\mathbf{U}_i^s(w)}, \tag{5.6}$$

where $\mathbf{U}_i^s(w)$ and $\mathbf{U}_i^t(w)$ are average source and target excitation spectra for the i^{th} codeword respectively. The converted excitation signal $\mathbf{r}_c(n)$ can be obtained by applying this filter to the source speaker excitation signal $\mathbf{r}_s(n)$:

$$\mathbf{r}_c(n) = \mathrm{IDFT}(\mathbf{H}(w)\,\mathrm{DFT}(\mathbf{r}_s(n))). \tag{5.7}$$

It must be remarked that STASC uses information about LSF parameters in the residual modification process only by computational reasons, but they use can be avoided. Source LSF parameters are employed to find the set of weights v_i, which provide phonetic related information into the conversion. In the publication [Ars99], the authors explained that although one may benefit from estimating a different set of codebook weights for the residual domain, they chose to apply the same set of weights derived from LSF parameters, mainly for computational reasons.

5.2.3 Residual Codebooks by LPC Classification

The residual modification system proposed in [Kai01b] consists in a LPC parameter classifier and a LP residual codebook, where each class of the classifier is associated with an entry in

the codebook. In fact, this system can not be called a conversion system, due to the source speaker's residual signal is not modified to produce the converted residual. In contrast, converted residuals are generated as a weighted sum of predicted target residuals. The residual prediction is performed based on the converted vocal tract parameters.

To build the LPC classifier, a GMM with Q components modeling the LPC cepstrum parameters of voiced frames of the target speaker is estimated, according to the method exposed in chapter 4. The LP residual codebook is formed by a set of Q pairs of codevectors, representing the magnitude and phase spectrum of residuals frames resampled to 100 points. The magnitude codevector for the word q is generated as follows:

$$\mathbf{m}_q = \sum_{i=0}^{N-1} \mathbf{M}_i \frac{c_q(\mathbf{x}_i)}{\sum_{j=0}^{N-1} c_q(\mathbf{x}_j)} \qquad q = 0 \dots Q-1, \tag{5.8}$$

where $\mathbf{M}_i$ is the magnitude spectrum of the i^{th} target residual vector $\mathbf{x}_i$ of a training set of size N. Each entry $\mathbf{m}_q$ is the normalized, weighted sum of all training residual magnitude spectra, where the weights $c_q(\mathbf{x}_i)$ correspond to the degree of membership to the particular class q. In particular, the weights $c_q(\mathbf{x}_i)$ are the posterior probabilities of the previously estimated target GMM:

$$c_q(\mathbf{x}_i) = P(w_q \mid \mathbf{x}_i) = \frac{\alpha_q N(\mathbf{x}_i; \mu_q, \mathbf{\Sigma}_q)}{\sum_{j=1}^{Q} \alpha_j N(\mathbf{x}_i; \mu_j, \mathbf{\Sigma}_j)} \qquad q = 0 \dots Q-1. \tag{5.9}$$

This probabilistic approach is an averaging and a smoothing operation.

A similar probabilistic approach can not be used for determining the phase of the codevectors, because their values are given in modulo 2π. The residual phase vector of the centroid of each codebook class is determined by the phase vector corresponding to the residual with the highest posterior probability of that class:

$$\mathbf{p}_q = \mathbf{p}(\mathbf{x}_k) \qquad | \qquad x_k = \underset{\mathbf{x}_i}{\operatorname{argmax}}(c_q(\mathbf{x}_i)) \qquad q = 0 \dots Q-1, \tag{5.10}$$

where $\mathbf{p}(\mathbf{x}_i)$ denotes the i^th training residual phase vector.

The residual prediction process is illustrate in figure 5.4. To carry out the residual generation, the posterior probabilities for all Q classes are calculated from the converted cepstral vector $\hat{\mathbf{x}}_i$. The frequency-normalized residual magnitude spectrum for the i^{th} frame is given by the sum of the magnitude codebook entries weighted by the posterior probabilities:

$$\hat{\mathbf{M}}_i = \sum_{q=0}^{Q-1} \mathbf{m}_q c_q(\hat{\mathbf{x}}_i), \tag{5.11}$$

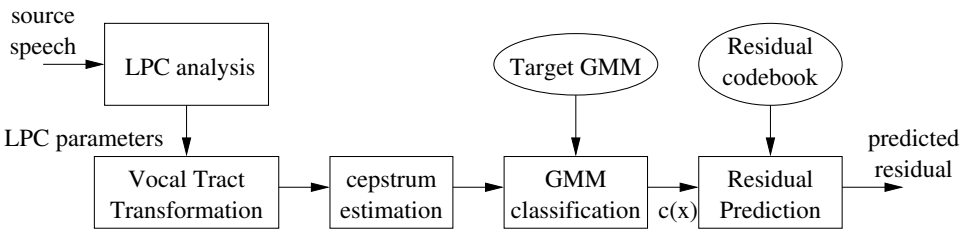


Figure 5.4: Block diagram of the operation mode of the residual prediction based on the Residual Codebooks by LPC Classification method for voiced speech.

and the phase spectrum is given by the most likely phase codebook entry:

$$\hat{\mathbf{p}}_i = \mathbf{p}_k \qquad | \qquad k = \underset{q}{\operatorname{argmax}}(c_q(\hat{\mathbf{x}}_i)). \tag{5.12}$$

It was reported that the direct speech generation from the converted magnitude and phase produces an audible degradation of the final speech, most often perceived as roughness [Kai01b]. To alleviate this problem, the trajectories of each harmonic phase over all the frames are unwrapped and the voiced regions smoothed by zero-phase filtering with an eight-point Hanning window.

To finish the conversion, the mean and the variance of the fundamental frequency are modified to match the target by resampling the residual predicted frames. For unvoiced speech, the target residual spectra are a resampled versions of the source speaker's residual spectra.

It must be remarked that the source residual signal is not used in any step of this system. Converted residuals are generated with information of the target speaker and the converted vocal tract parameters.

5.2.4 Residual Selection and Phase Prediction

Ye et al. published a novel method of residual prediction for synthesizing natural phase dispersion [Ye04], where residuals are selected from a database extracted from the target training data. Each entry of the database is formed by $[\mathbf{r}_n; \mathbf{v}_n]$, $n = 0 \ldots N-1$, where $\mathbf{r}_n$ is the log magnitude spectrum of the n^{th} target training residual and $\mathbf{v}_n$ is defined as $\mathbf{v}_n = [f_1, f_2, \ldots, f_d, \Delta f_1, \Delta f_2, \ldots, \Delta f_d]^T$. Line spectral frequencies are indicated as f_i and Δf_i are their increments between the current and the previous frame.

To produce the converted speech, once the vocal tract has been transformed, a residual signal is selected from the database. The criteria used to select the residual $\mathbf{r}_k$ for the converted envelope $\tilde{\mathbf{v}}$ is to choose that residual whose associated LSF vector $\mathbf{v}_k$ minimizes the following square error:

$$E = (\mathbf{v}_k - \tilde{\mathbf{v}})'(\mathbf{v}_k - \tilde{\mathbf{v}}). \tag{5.13}$$

In addition to the residual selection, the referred system includes a phase prediction function in order to improve the quality of the converted speech. To estimate the predictor, the input speech signal $\mathbf{s}(n)$ is expressed as a weighted sum of Q speech prototypes $\mathbf{t}_q(n)$:

$$\tilde{\mathbf{s}}(n) = \sum_{q=0}^{Q-1} \mathbf{t}_q(n) c_q(\mathbf{v}), \tag{5.14}$$

where $c_q(\mathbf{v})$ is the posterior probability of the q^{th} component of a GMM modeling the LSF distribution of the training LSF vectors.

The speech prototypes are calculated by minimizing the following coding error over all the target training data:

$$E = \sum_{n=0}^{N-1} \big(\mathbf{s}(n) - \mathbf{T}\mathbf{p}(v_t)\big)'\big(\mathbf{s}(n) - \mathbf{T}\mathbf{p}(v_t)\big), \tag{5.15}$$

where $\mathbf{T} = [\mathbf{t}_0, \ldots, \mathbf{t}_{Q-1}]$ and $\mathbf{p}(v) = [c_0(\mathbf{v}), \ldots, c_{Q-1}(\mathbf{v})]$.

During the conversion, the waveform shape of the converted signal is predicted for each frame as $\tilde{\mathbf{s}}(n) = \sum_{q=0}^{Q-1} \mathbf{T}_q(n) c_q(\mathbf{v})$. The required phases are obtained from the predicted waveform using the analysis routine and pitch-scale modification algorithm of sinusoidal modeling. Then, this phases are imposed to the selected residual $\mathbf{r}_k$ to generated the converted residual $\mathbf{r}_c$. Figure 5.5 is a representation of the residual generation process by Residual Selection and Phase Prediction.

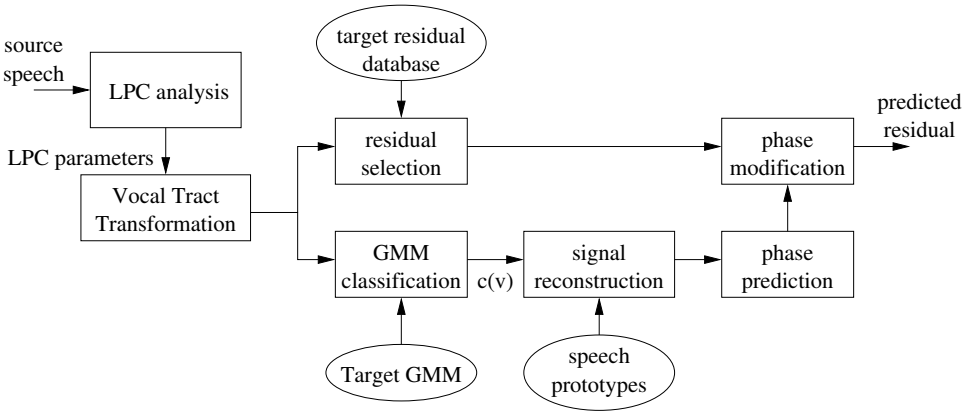


Figure 5.5: Block diagram of the operation mode of the Residual Selection and Phase Prediction system for voiced speech.

The work of Ye et al. deals with unvoiced sounds by unit selection and concatenation of source speech portions, without converting their vocal tract parameters.

The main difference between this work and residual prediction based on Residual Codebooks by LPC Classification is the residual material. Whereas in the last study residuals are generated from a weighted average of a limited set of codevectors, Residual Selection and Phase Prediction method uses real target residuals. One of the consequence is the system has to storage all the residuals seen in the training step.

5.2.5 Residual Selection and Smoothing

The technique of Residual Selection and Smoothing, published by Sündermann et al. [Sün05b], is a refinement of the residual selection technique that deals with the inaccuracies of phase dispersion and the treatment of voiced and unvoiced frames by means of a time-variant residual smoothing. The motivation of this improvement of the residual selection technique is that when listening to converted speech, in particular voiced regions, some artifacts can be perceived, due to improper residuals selections. In voiced regions, the signal should be almost periodic. Therefore, voiced residuals are not expected to change abruptly. However, in unvoiced regions the residuals should resemble white noise that changes from frame to frame. These considerations led to the idea of a voicing-dependent residual smoothing.

Once the sequence of predicted residual frames $\{\hat{\mathbf{r}}_n\}$ is obtained according to the residual selection technique, they are smoothed according to a voiceness degree. See figure 5.6 for a block diagram of the system on the operation mode.

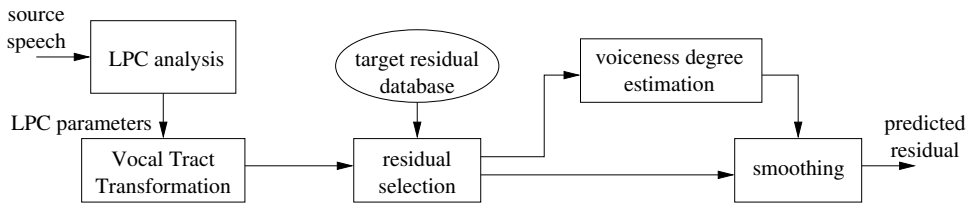


Figure 5.6: Block diagram of the operation mode of the Residual Selection and Smoothing system.

The voiceness degree σ is determined by the correlation coefficient of the source residual to be converted and the next one. The final residual $\tilde{\mathbf{r}}_c$ is obtained by applying a normal distribution function to computed a weighted average over all the residual predicted vectors:

$$\tilde{\mathbf{r}}_c = \sum_{n=-K+k}^{K+k} \mathcal{N}((n-k); 0, \boldsymbol{\alpha}\sigma_k)\hat{\mathbf{r}}_n \qquad k = 0, ..., K, \tag{5.16}$$

where $\boldsymbol{\alpha}$ is the voiceness gain that has to be tuned manually. For voiced regions ($\sigma \approx 1$) the

average is over neighbor elements by a wide bell, as for unvoiced regions ($\sigma \approx 0$) there is no local smoothing.

In order to carry out the smoothing all the residual frames must be of the same length. Therefore, a prior length normalization to the mean pitch period is carried out.

5.2.6 Analysis of Previous Studies

After an analysis of the previous descriptions, two main strategies to face the LP residual modification problem have been identified. Both *Transformation based on Nonlinear Prediction Analysis* and *Speaker Transformation Algorithm using Segmental Codebooks* converts the source residual signal in order to better match the target one. We will call this strategy LP Residual Conversion. Residual conversion systems are the equivalent to vocal tract conversion systems, where residual parameter (or signal) vectors are modified according to a source-target relationship. The fundamental idea of conversion systems is that given aligned speech signals of two speakers it can be found a mapping between their residuals according to acoustic information.

On the other hand, the three remaining systems (*Residual Codebooks by LPC Classification*, *Residual Selection and Phase Prediction* and *Residual Selection and Smoothing*) predict the residual signal from the converted vocal tract parameter vector, without taking into account any source residual information. We will call this strategy LP Residual Prediction. Residual Prediction assumes that the residual is not completely uncorrelated with the spectral envelope, making the prediction possible. The residual magnitude spectrum contains the errors caused by the spectral envelope fit and the phase spectrum contains information about the natural phase dispersion of the signal, opposed to the minimum phase assumption of the LPC model. Therefore, it can be expected that the residuals are similar and predictable for a particular speaker in a phonetic class.

Differences between conversion and prediction of LP residual signal can be summarized in the following two statements:

- Conversion and Prediction strategies use different material to obtain the transformed residual: modified source residuals for conversion systems in front of target residuals for prediction systems.
- Conversion and Prediction strategies use different information to determine the modification procedure: source speech information for conversion systems in front of converted vocal tract information for prediction systems.

Although a partial comparative between some residual selection techniques has been published [Sün05c], a formal comparative between both strategies should be carried out, in order to

determine the basis for a novel residual modification technique.

5.3 A Comparative between Conversion and Prediction of Residual Signals

As it has been explained in the previous section, residual signal modification can be faced from two different points of view: as a source residual signal alteration or as a signal prediction from the converted vocal tract parameters. The key point to measure the fitting of both strategies to the residual modification problem is the strength of relationship between source-target residuals or vocal tract-residuals of the same speaker. In this section, a comparative study of these two strategies published in [Dux06a] by the author is detailed, before the proposed residual modification method is presented in section 5.4. Also, a new strategy to reconstruct the transformed speech avoiding large noises due to phase discontinuities is presented, in order to carry out perceptual evaluations.

The outline of this section is as follows. First, both Residual Conversion and Residual Prediction systems used for the comparative will be described, including a presentation of the new converted speech generation process. Next, comparative results will be discussed in order to choose the most appropriate strategy.

5.3.1 Residual Modification Systems

Residual Conversion

The objective of the Residual Conversion system evaluation is to find out if the relationship between source residuals and aligned target residuals is the best option to built a residual modification system for a VC task.

The system that have been used to carry out the evaluation is based on mapping codebooks. The proposed system is similar to the first published method in the vocal tact conversion field [Abe88], also applied to the residual problem in [Lee96]. However, we are not concern about errors due to the modelization or quantization of the residual signals, since the point of interest is the comparison between conversion and prediction.

Therefore, the source residual codebook contains all the source residual frames of the aligned training data as codevectors. In the same way, the target residual codebook contains all the aligned target residual frames. The mapping codebook associates the i^{th} source codevector with its aligned i^{th} target codevector. With this codebook construction strategy, there is no reduction of the dimensionality of the training data. All the training information is used in the conversion step.

To convert a source residual signal the most similar source codevector r_k^s must be found. The similarity measure used is spectral distortion:

$$SD(r_1, r_2) = 20 \log \Big(\frac{1}{N} \sum_{n=0}^{N-1} \sqrt{(S_{r_1}(n) - S_{r_2}(n))^2} \Big), \tag{5.17}$$

where $S_{r_i}(n)$ denotes the normalized power spectrum of the residual r_i. The converted residual signal is determined by the target residual codevector r_k^t associated with the k^{th} source codevector by the mapping codebook.

Residual conversion has been applied only to voiced frames. Source residual signal has been kept for unvoiced frames, because they do not contain as many glottal characteristics of the speaker as voiced frames.

Residual Prediction

The objective of the Residual Prediction system evaluation is to find out if the relationship between vocal tract parameters and residual signals of the same speaker is the best option to built a residual modification system for a VC task.

The selected system to carry out the evaluation is similar to the residual conversion system. As in residual conversion, we are not concerned about errors due to the quantization of the residual signal nor errors due to the modelization of the search space into regions less populated. Therefore, the prediction approach uses all the available information and the acoustic space is not hard or soft parted into classes. In particular, the strategy followed was to build parallel codebooks for LSF and their LP residual signals associated, with as many codevectors as available frames in the training step.

The prediction procedure is as follows. Converted LSF are compared with all the LSF codevectors and the residual signal associated to the most similar codevector is chosen. The similarity measure for LSF is the Inverse Harmonic Mean Distance, which weight more the mismatch in spectral picks than the mismatch in spectral valleys.

As in residual conversion, residual prediction has been applied only to voiced frames.

Speech Generation

The speech reconstruction strategy applied to the converted LSF and modified residual signals is inverse filtering and TD-PSOLA for frame concatenation. A straightforward application of TD-PSOLA over the speech frames obtained from a complete VC system results in concatenation noises, mainly due to no similarity criteria over neighbor pitch period residual signals is imposed.

In order to avoid synthesis artifacts due to residual phase discontinuities from frame to frame a Harmonic Model for each of the modified residuals is estimated [Dep97]:

$$s(n) = \sum_{k=1}^{K} a_k cos(2\pi f_k n + \phi_k). \tag{5.18}$$

The number of harmonic frequencies is the integer $K \leq f_{sampling}/(2f0)$. The harmonic phases ϕ_k are modified so that continuity is assured in every voiced region. It means that the initial phase of the harmonic k of the frame i is calculated as:

$$\phi_k^i = 2\pi \frac{f_k^{i-1} + f_k^i}{2} N_k, \tag{5.19}$$

where N_k is the number of points between the center of the $(i-1)^{th}$ frame and the i^{th} frame. When a birth of a harmonic frequency occurs it is assigned a random initial phase.

Duration, loudness and speech rate of the source speaker are kept. The mean value of pitch (μ) and its variance (σ) are estimated for source and target speakers from the training data. When the speech generation is carried out, the pitch of the utterance is modified to adjust its mean and variance according to Eq. 5.20.

$$f0_{converted} = \mu_{target} + \frac{\sigma_{target}}{\sigma_{source}}(f0_{source} - \mu_{source}) \tag{5.20}$$

5.3.2 Comparative Results

Two sets of experiments have been carried out in order to evaluate the performance of conversion and prediction residual systems:

1. Converted LPC residual signal filtered by mapped LSF, with $F0$ modification.
2. Predicted LPC residual signal filtered by mapped LSF, with $F0$ modification.

The method used to map the vocal tract parameters is based on a decision tree classification, the CART with pre-pruning system already presented in chapter 4. The amount of training data has been chosen equal to a practical number of sentences in a real VC application, in particular 30 sentences for training and 20 for test. Four different speakers, two males (MALE_1 and MALE_3) and two females (FEMALE_1 and FEMALE_2), were used as source and target speakers. Twelve listeners, who work in speech technologies, completed the evaluation.

Three different tests have been evaluated: an extended ABX test, a similarity test and a MOS test. In the extended ABX test, listeners rated every question according to the following

rule: 1-Very close to A; 2-Close to A; 3-Neither A nor B; 4-Close to B; 5-Very close to B. In order to analyze the results, ratings have been normalized in order to identify A as the source speaker. In the similarity test, listeners were asked to rate the similarity of two speech files from 1 (different speakers) to 5 (the same speaker). Twelve listeners completed the evaluation.

Table 5.1 resumes the results of the extended ABX test. Each column corresponds to the proportion of times that listeners have rated the transformed speech signal as source (very close or close), target speaker (very close or close) or neither of them. Slightly better results were obtained with Predicted Residuals than with Converted Residuals. To be more precise, table 5.3 contains the numerical results of the extended ABX test once ratings have been normalized in order to identify A as the source speaker.

Experiment / %	**Source**	**Neither**	**Target**
Conversion	2	26	72
Prediction	4	20	76

Table 5.1: ABX test results.

Results of the similarity test are displayed in Table 5.2. The desired results are few similarity between Source-Converted speakers (rate of 0) and high similarity between Target-Converted speakers (rate of 5). Although the similarity test was the most difficult task for the listeners, the similarity between source and converted speech files was less rated than the similarity between target and converted files for both experiments. This result confirms that both proposed VC systems achieve the goal of moving away the voice individuality from a source speaker to a target speaker. The only remark is about the experiment 1, Converted Residuals, where rates are lower in both questions (1.38 in front of 1.48 and 2.93 in front of 3.48). This result can be explained with the results of the MOS test.

Experiment	**Source-Converted**	**Target-Converted**
Conversion	1.38	2.93
Prediction	1.48	3.48

Table 5.2: Similarity test results.

The results of the MOS test are showed in table 5.3. When Predicted Residual signals fed the mapped LPC filter, MOS results moves to 2.22 (natural speech was rated 4.42 in the perceptual tests of chapter 4), but the reduction in quality is more severe when converted residual signal are used. The low quality of Converted Residuals may cause the listeners perceive the transformed speech signal away as from the source speaker as from the target speaker.

	Conversion	Prediction
ABX	3.90	3.92
MOS	1.74	2.22

Table 5.3: ABX and MOS test results.

The lack of quality in transformed speech signals comes from three main reasons: errors in the vocal tract mapping, mismatches between vocal tract and residuals, and frame concatenation errors. Errors due to vocal tract mapping are shared by the two systems tested and they are not the object of the current comparison. Although a Harmonic Model and phase modification have been used for speech signal reconstruction in order to alleviate the frame concatenation errors, not all the artifacts has been avoided. Further studies will be able to increase both prediction and conversion quality approaches. In particular, a new method based on phonetic residual prediction and fixed smoothing is presented in section 5.4.

The key difference between Converted and Predicted residual systems is the mismatch between vocal tract and residuals. According to the results, the relationship between LSF parameters and their residuals results in a better morphing quality than the relationship between residuals of the source and target speakers. Therefore, next sections will be focus on Predicted Residual systems.

5.4 Phonetic Residual Selection and Fixed Smoothing

This section presents a new residual prediction system that utilizes phonetic information for the reduction of the selection space and fixes the length of a posterior smoothing.

Previous residual prediction works perform the residual selection by weighting a limited number of codevectors [Kai01b], or by a search in a large database [Ye04,Sün05b]. The drawback of the former system is that the effect of the average of the codevectors is added to the effect of the average of the vocal tract. This accumulated average may result in an over-smoothing, a characteristic of the vocal tract conversion that residual modification tries to improve. On the other hand, a search in a large database, as the one performed in the second group of approaches, could slow the execution time of the VC routine, critical in some applications.

The solution proposed in this section is the construction of a collection of databases in the training step according to phonetic characteristics of the speech that will allow to keep all the training information in the operation phase of the prediction system, but the amount of searching operations will be reduced at any desired degree. Moreover, the inclusion of phonetic information

will be studied as a factor to achieve a better residual prediction. Our hypothesis is that given a LPC filter corresponding to some phonetic class, the application of a residual from a similar phonetic class will result in a better speech quality then the application of an unknown origin residual.

Once the sequence of residual frames has been predicted, a constant length smoothing for voiced frames will be applied. Unvoiced residual frames will be generated as samples of a white Gaussian noise.

5.4.1 Collection of Databases

The process to obtain the collection of databases consists in three steps. First, an initial database is built with all the appropriate LSF vector and residual signal pairs of the target speaker. Second, a classification tree based on phonetic decisions is estimated to cluster the LSF-residual pairs. Finally, the collection of databases is populated.

The initial database is formed by all the LSF vector and residual signal voiced pairs of the target training data. Then, the mean μ and standard deviation σ of the speaker's pitch period length is estimated and all the residual signals not included in the $[\mu - a\sigma, \mu + a\sigma]$ segment are excluded from the database ($a > 0$, $a \in \Re$). In the experiments, a value of $a = 1.5$ has been used. The remaining residual signals are length normalized to the mean.

The objective of the clustering is to divide the initial database in N databases of any desired minimum and maximum number of elements according to the conversion most efficient way. The clustering is performed by a CART system, trained with all the LSF vectors and residual signal pairs of the initial database grouped in a training subset and a validation subset. The same set Q of phonetic questions used in the vocal tract conversion (see table 4.2 in page 70) is used to split the nodes.

The procedure to grow the tree is as follows. The parent node (the root node for the first iteration) is split by every phonetic question. For each splitting option, a right and a left databases are built with all the training LSF-residual pairs of the right and left child nodes. Once these databases are determined, a residual is selected for each one of the validation LSF vectors of the child nodes. In order to select the residuals, for each LSF validation vector $\mathbf{x}_i$ the most similar training LSF vector $\mathbf{x}_j$ is found according to the Inverse Harmonic Mean Distance (IHMD). The selected residual is the corresponding training residual signal $\mathbf{r}_j$ of the training LSF vector $\mathbf{x}_j$.

When all the residuals for the validation LSF vectors have been selected, an increment error index is calculated for each spiting option as:

$$\Delta(t,q) = D_t - \frac{(D_{t_L}(q)|t_L|) + (D_{t_R}(q)|t_R|)}{(|t_L| + |t_R|)} \qquad |t| = |t_L| + |t_R| \tag{5.21}$$

where $|t|$ denotes the number of validation elements belonging to the node t and $D(\cdot)$:

$$D(q) = \sum_{i=0}^{|t|-1} \frac{(\mathbf{r}_i - \mathbf{r}_j)^2}{\mathbf{r}_i^2} \tag{5.22}$$

where $\mathbf{r}_i$ is the i^{th} validation residual signal, $\mathbf{r}_j$ the selected residual for the i^{th} LSF validation vector and q denotes the question by which the node has been split.

At every growing step of the tree, all the $\Delta(t, q)$ are compared. The node t^* and the question q^* that satisfies $\{q^*, t^*\} = \text{argmax}_{q,t}\, \Delta(t, q)$ are selected, and the node t^* is split by q^* if $|t_L|$ and $|t_R|$ are greater than the minimum number of elements allowed.

The decision to stop the tree growing depends on the number of elements in each node. When all nodes have a number of training plus validation elements less than the maximum number allowed, or if having more than the maximum number all the possible splittings results in nodes less populated than the minimum population, the growing process is stop and all the nodes without children are declared leafs.

A different behavior between CART growing for vocal tract conversion and CART growing for residual modification must be remarked. The increment error index for residual modification $\Delta(t, q)$ can be negative, in contrast to vocal tract conversion, because the goal of the clustering is to divided the data in smaller databases despite that the selection residual error increases.

Finally, to populate the collection of databases all the available data (training plus validation subsets) is classified according to the decision tree. Then, LSF vector and residual signal pairs are ordered in databases according to their leaf label.

With this procedure, databases containing at least a minimum number of elements and at most a maximum number of elements determined by the designer are obtained. Only databases with more than the maximum number of elements are allowed when their splitting will result in databases with less elements than the minimum. The collection of databases will improve the search of the best residual in the operation phase, but the total memory load of the system has not changed. All the available information in the training step is kept.

5.4.2 Phonetic Residual Selection

Phonetic residual prediction is an easy operation in the working phase. For every source frame, its converted LSF vector is classified according to phonetic characteristics, determining which one of the databases in the collection is the most adequate to carry out the selection. Then, a search in the selected database is performed in order to look for the most similar target LSF vector to the converted one. Once an entry of the database is selected the source residual signal is replaced by the residual signal of the database.

This operation is defined only for voiced frames. Unvoiced source residual signals are generated as white Gaussian noise.

5.4.3 Residual Smoothing

The reconstruction strategy applied to the converted LSF and transformed residual signals is the inverse filtering and TD-PSOLA method for prosodic modifications. As it has been mentioned in previous sections, a straightforward application of TD-PSOLA over the speech frames obtained from a complete VC system results in concatenation noises, mainly due to no similarity criteria over neighbor pitch period residual signals is imposed. This is a reason to apply some smoothing to the residual signals, once they are selected from the collection of databases.

A previous published method [Sün05b] applies a variable length Gaussian averaging over all residual frames. The length of the averaging depends on a voiceness degree estimated on the source speech to be converted. Two similar smoothings are proposed in this dissertation.

The first proposed smoothing is a variable length Gaussian averaging, where the length depends on a voiceness degree estimated on the target speaker in the training phase of the system. Therefore, the collection of databases are enriched by adding the correlation coefficient of each target residual and the next target residual in the original sentences. Two main reasons motivated this approach. On one hand, there are speakers that exhibit different voiceness degree in some special phonemes. As the residual selection is performed based on phonetic information, the voiceness degree may be also selected from the database, in order to take into account the speaker dependent characteristics. On the other hand, estimating the voiceness degree on the target speaker reduces the computational load in the operating phase, as the correlation coefficient has been estimated in the training phase.

The second proposed smoothing is a fixed length Gaussian averaging, with different lengths for voiced and unvoiced speech frames. This is the most simple smoothing. However, it will be useful to compare if more sophisticate smoothings are really needed in the VC task.

5.5 Experiments and Results

This section discuss the evaluation of the Phonetic Residual Selection and Fixed Smoothing and compares its performance when the collection of databases consists in only one database, shared by all the phonetic classes. This latest system is very similar to the one published in [Sün05b].

The Phonetic Residual Selection was performed by setting the minimum number of elements to 16 and the maximum to 512. Transformations have been carried out with the training set Set_30 and with the same two source-target speaker pairs of chapter 4: one cross-gender conver-

sion, from FEMALE_1 to MALE_1, and one intra-gender conversion, from MALE_1 to MALE_2. The initial database of 5606 entries for MALE_1 was divided in 27 sub-databases (a mean of 208 entries for sub-database), and the initial database of 6392 entries for MALE_2 resulted in 21 sub-databases (a mean of 304 entries for sub-database).

After several informal listening tests, the final designed systems employ a fixed smoothing length for voiced speech and does not apply any smoothing for unvoiced speech. Therefore, the final voiced residual $\mathbf{r}_c$ are obtained by applying a normal distribution function to computed a weighted average over all the residual predicted vectors:

$$\mathbf{r}_c = \sum_{n=-K+k}^{K+k} \mathcal{N}((n-k); 0, \boldsymbol{\alpha})\hat{\mathbf{r}}_n \qquad k = 0, ..., K, \tag{5.23}$$

where a voiceness gain of 2 was chosen. Listeners of the informal test reported that a variable length smoothing was almost indistinguishable form a fixed length smoothing. Due to the reports, the simplest system was selected.

The evaluation consisted in two different perceptual tests: an extended ABX test and a similarity test, in the same format than the evaluation tests of vocal tract conversion explained in chapter 4. In addition, a MOS test to evaluate the converted speech quality was carried out. Twenty listeners completed the tests. The converted speech played to the listeners came from two different complete VC systems. One system was based on Phonetic Residual Selection, whereas the other one used only one database. This latter system will be called Residual Selection. Both systems performed a fixed smoothing length for voiced speech, and the vocal tract conversion was carried out by means of a CART with pre-pruning mapping.

The mean results for the extended ABX test are summarized in table 5.4, where ratings have been normalized in order to identify A as the source speaker.

	Mean Score
original	4.58
Residual Selection	3.97
Phonetic Residual Selection	3.94

Table 5.4: Mean score of the extended ABX test.

The mean scores for the similarity test are displayed in table 5.5. The results denotes the percentage of times that one speaker of the first row have been rated as the same/different speaker when compared with one speaker of the first column. The *original* speaker refers to both the source and target speakers.

Extended ABX scores decrease when a residual modification technique is applied, in front of

	original		source		target	
	same	**different**	**same**	**different**	**same**	**different**
original	97.37	2.63				
Res. Selec.			2.63	97.37	76.32	23.68
Phonetic Res. Selec.			0	100	79.49	20.51

Table 5.5: Mean score of the similarity test.The results denotes to the percentage of times that one speaker of the first row have been rated as the same/different speaker when compared with one speaker of the first column.

using the aligned target residual (rated 4.20 in experiments of chapter 4). Residual modifications techniques employ real target residual frames to generate the converted speech. Therefore, these systems are not expected to result in converted voices closer to the source speaker than using aligned target residual frames to generated the converted speech. This statement can be proved by the similarity test. The time percentage that source and converted speakers have been confused (between 0% and 2.63%) are similar for the current experiments than for the experiments of chapter 4. However, the distance between target and converted speakers has increased (about 80% in the current experiments in front of 86% in the aligned target residual experiments). A probable explanation of this worse performance of systems dealing with residuals, in front of ideal residual modification systems, is the poor quality of the generated speech.

Quality results are displayed in table 5.6. The reported quality for both Residual Selection and Phonetic Residual Selection need to be improved to be acceptable in an application. Phonetic Residual Selection has rated with lower quality than systems using only one database to perform the residual selection. Therefore, it can be conclude that phonetic information is not a help in the residual selection. Only when computational requirements forces it, the Phonetic Residual Selection should be selected to carry out residual modifications in a VC task.

	MOS
original	4.6
Residual Selection	1.56
Phonetic Residual Selection	1.24

Table 5.6: MOS results.

The quality of speech generated by residual modification systems presented in this section is, with no doubt, higher than the quality of speech generated by the residual prediction system presented in section 5.3.1. However, MOS results were better for the latter system. There are two probable explanations for this phenomenon, due to differences in the evaluation set.

The main reason for these differences may be that natural sentences have been introduced to be evaluated in the MOS test presented in the current section, whereas in the MOS test carried out previously only converted voices were evaluated. Natural sentences quality may have moved down the rates for converted sentences quality.

Another reason for the differences may be the personal preferences of listeners who completed the test. The most part of the twelve listeners of the test discussed in section 5.3.1 were people who work in speech technologies, used to listen to slightly differences in similar synthesized voices. In contrast, only a few proportion of listeners of the current evaluation work in speech technologies.

In order to validate the statement that the quality of speech generated by residual modification systems presented in this section is higher than the quality of speech generated by the residual prediction system presented in section 5.3.1, an informal listening test have been carried out with some listeners who had contributed to both evaluations. All the asked listeners support the statement.

5.6 Conclusions

Two residual modification systems for a VC task have been compared. Phonetic Residual Selection uses a collection of databases in order to select the most appropriated target residual to the converted vocal tract parameters. Phonetic information determines in which database the selection will be carried out. In contrast, Residual Selection uses only one database, shared by all the phonetic classes. Both systems perform a fixed smoothing length for voiced speech, and the vocal tract conversion was carried out by means of a CART with pre-pruning mapping.

The main advantage of Phonetic Residual Selection, in front of the other system, is that the computational load in the operation phase is reduced. However, Phonetic Residual Selection has been rated with lower quality than the Residual Selection system in perceptual results. Therefore, only when computational requirements forces it, the Phonetic Residual Selection should be selected to carry out residual modifications in a VC task.

Perceptual tests have revealed that the quality of both systems under evaluation is not good enough to be used in practical applications. At the actual level of development of VC technologies, it is recommended the use of conversion systems without residual modification for practical applications. Mapping vocal tract systems using source residuals to generate the converted speech, or Vocoders taking into account converted LSF, may produce converted speech not so close to the target speaker or less natural, but with an acceptable quality. In any case, further work is required to improve the quality of residual modification systems.

5.7 Summary

VC research has been mainly focused on the mapping of vocal tract parameters. However, in order to achieve an effective change of the speaker individuality some modification of the LP residual signal should be performed, since residuals contain many speaker dependent characteristics.

After an analysis of previous published works, two strategies to face the residual modification problem have been identified. On one hand, Residual Conversion systems modify source residuals by some mapping function. On the other hand, Residual Prediction systems predict the converted residuals from the converted vocal tract parameters. A comparative of both strategies have been carried out, concluding that the relationship between vocal tract parameters of a given speaker and their corresponding residuals works better than the relationship between aligned source-target residuals, in terms of conversion performance.

Based on the previous conclusion, a Phonetic Residual Selection and Fixed Smoothing system has been proposed for generating voiced converted residuals. This systems selects the most appropriated residual for the converted vocal tract parameters from a collection of databases which entries are target LSF-residual pairs. Once the sequence of voiced residual frames has been selected, a fixed length smoothing is applied. Unvoiced residual frames are generated as white Gaussian noise.

The collection of databases is built in the training step, including all the target training information organized by phonetic characteristics. The use of a collection of databases, in front of using only one database, will allow to keep all the training information in the system operation phase, while reducing the amount of searching operations at any desired degree. Moreover, the inclusion of phonetic characteristics has been studied as a factor to achieve a better residual selection.

The performance of the proposed system has been studied by means of a perceptual evaluation. Test results have revealed that the quality of the proposed system is not good enough to be used in practical applications. In addition, a perceptual comparative between converted residuals generated with and without phonetic information has been carried out. Based on the results, we conclude that phonetic information does not improve the quality of the converted speech. Further work is required to improve the quality of residual modification systems.

Next chapter describes the final complete VC system proposed in this dissertation and discuss alternatives to new speaker generation for a TTS when few data is available.

Chapter 6

A Complete Voice Conversion System

This chapter sums up the conclusions of the vocal tract conversion study, presented in chapter 4, and the LP residual modification study, presented in chapter 5, to design a complete VC system. Although the proposed VC system has been mainly designed to work as a post-processing block for a TTS, since manual supervised phonetic information is assumed to be available, its extension to other applications are also described.

The outline of the chapter is as follows. First, section 6.1 reviews why it is important to have a large corpus when working with a corpus based TTS and presents alternatives to new speaker generation when few data is available. Section 6.2 describes the complete VC system proposed in this dissertation as an alternative to produce new speakers for a TTS, when at least one corpus-based speaker is already generated. In the same section, a novel text-independent training is also described to built the proposed VC system when the training corpus has not parallel source-target sentences. Finally, in section 6.3 the performance of the proposed system has been evaluated in the periodic evaluation campaign organized in the framework of the integrated European project Technology and Corpora for Speech-to-Speech Translation (TC-STAR).

6.1 Alternatives to New TTS Speaker Generation with Few Data

Actual high quality TTS systems are based on acoustic unit concatenation for speech generation. The speech generation process of theses systems, also called corpus-based TTS, consists in two steps. First, the most appropriated acoustic units are selected from a speaker-dependent database. Then, the prosody of the selected units is modified, if it is necessary, and a strategy of concatenation is applied to joint the selected units together.

Recent researches have provided efficient algorithms for acoustic unit selection from large databases using dynamic programming. Due to techniques such as beam search, unit selection can be performed in real time. However, the concatenation and modification of acoustic units that come from disjoint speech segments is critical for the performance of a TTS. In order to avoid concatenation noises (i.e. "clicks") and speech distortions due to prosodic modifications, it is required to have a large collection of acoustic units. Mainly, there are two reasons for this requirement. On one hand, the number of concatenation points of an utterance can be reduced and the spectral continuity on the concatenation points is smoother if large acoustic strings that match with the phonetic content are present in the corpus. On the other hand, the output speech will be less distortioned if there are acoustic units in the corpus with similar prosody to the output prosody.

However, there are many TTS applications where it is not possible to collect or to store a large corpus. For example, in applications that require a large amount of voices (voice chats, games with characters with the players' voices, etc.) it is too time consuming and expensive to build a corpus-based TTS speaker for each voice. In other applications, the target speaker is not available to record a large corpus, for example when creating virtual characters from famous actors/actresses. Three different alternatives to produce new corpus-based TTS speakers from few data are presented in the following paragraphs.

The first alternative, and the most classical, is the construction of a corpus-based speaker with the few data available. This is the alternative that introduces less modifications to the TTS. Mainly, only one new component should be added to a corpus-based TTS: a unit replacement for phonetic units not present in the new speaker corpus. This unit replacement may be as simple as replacing a missing unit for another unit of the corpus with similar phonetic properties, or as complex as generating an artificial unit. The artificial unit can be generated with information of the current speaker or with information of previous corpus-based speakers, built in advance as a tool of the TTS.

A corpus-based TTS speaker generated directly with few data will produce synthetic speech with a lot of artifacts, because there will be a lot of concatenation points and large prosodic modifications of the selected acoustic units. Nevertheless, there are some situations where a corpus-based TTS speaker with few data may be useful. For example, when building a TTS speaker for a very limited domain such as a talking clock [Bla00] a corpus-based TTS speaker with few data may result in a good synthetic speech quality. Due to linguistic variations in a very limited domain are reduced, recording only a short list of sentences will assure to have good coverage of the domain. The main drawback of this strategy is that the resulting TTS is not flexible to introduce large variations on the generated speech if we will to maintain the output quality. As a consequence, corpus-based TTS speakers with few data are not a suitable alternative for application domains that evolve in time including lots of new words or phonetic

contents.

VC is the future technology to built new speakers for a TTS, with at least one corpus-based speaker, when few data is a requirement. The second alternative to produce new TTS speakers is the conversion of all the former speaker database in order to produce a complete converted speaker with a large converted corpus. This alternative is useful for applications where it is required a real time speech generation, because no computational load is added to the TTS operating time. The main drawback of this strategy is the memory load, because a large corpus must be stored for each one of the speakers. Moreover, an extended time of speaker creation will be needed, since all the original corpus must be converted. Therefore, this strategy will be only suitable for central applications serving the generated speech on-line, when new voices are desired. It is the case of non-final user TTS enterprises, who want to use/sell "speakers" but not the capacity of the TTS customization.

The third alternative to produce new speakers for a TTS with at least one corpus-based speaker is the use of a VC system as a post-processing block. With this strategy there are no problems when generating speech for a general domain, and also there is no problem with the memory load. Moreover, the inclusion or deletion of as many speakers as desired will be fast and simple. However, a real time operation is needed, what limits the system operation mode and thus the actual system output quality.

Nevertheless, this third alternative is thought to be the future line for creating new TTS speakers for a general domain with few data. Future research should integrate the VC system with the acoustic unit selection, in order to increase the output speech quality by better selecting the acoustic units of the source speaker to be converted.

The complete VC system to work as a post-processing block in a corpus-based TTS proposed in this dissertation is presented in the next section.

6.2 A Complete Voice Conversion System for a TTS

This section, after determining the completed VC system proposed in this dissertation, describes a new procedure to train the conversion system where the target speaker is not constrained to utter specific sentences.

6.2.1 System Architecture

The complete VC system proposed in this dissertation deals with vocal tract parameters and LP residual signals, without dealing with prosodic nor linguistic information. Only a re-scale of the pitch is performed to adjust the mean and standard variation of the speaker fundamental

frequency.

The vocal tract conversion system selected is the CART with pre-pruning system presented in chapter 4. Both objective and perceptual results have shown that CART with pre-pruning systems are the vocal tract conversion systems with the best performance when more than 5 sentences are used for training. The results have also shown that a speech professional speaker with reliable phonetic information is recommended for at least one of the speakers involved in the conversion. This is not a problem to apply CART systems as a post-processing block in a TTS, since the original speaker of the TTS (the source speaker of the conversion) is usually an actor/actress or an announcer with a corpus manually supervised.

The residual modification system used for the proposed approach is Residual Selection with Fixed Smoothing. Although Phonetic Residual Selection would be more adequate for systems working in real time, the converted speech quality has been a key factor to choose the use of selection systems without phonetic restrictions for the proposed approach.

The only prosodic modification, i.e. the adjustment of the mean (μ) and variance (σ) of the pitch, is carried out by means of the following expression, whose parameters are estimated in the training step:

$$f0_{converted} = \mu_{target} + \frac{\sigma_{target}}{\sigma_{source}}(f0_{source} - \mu_{source}). \tag{6.1}$$

The block diagram of the operating mode of the complete VC system is displayed in figure 6.1.

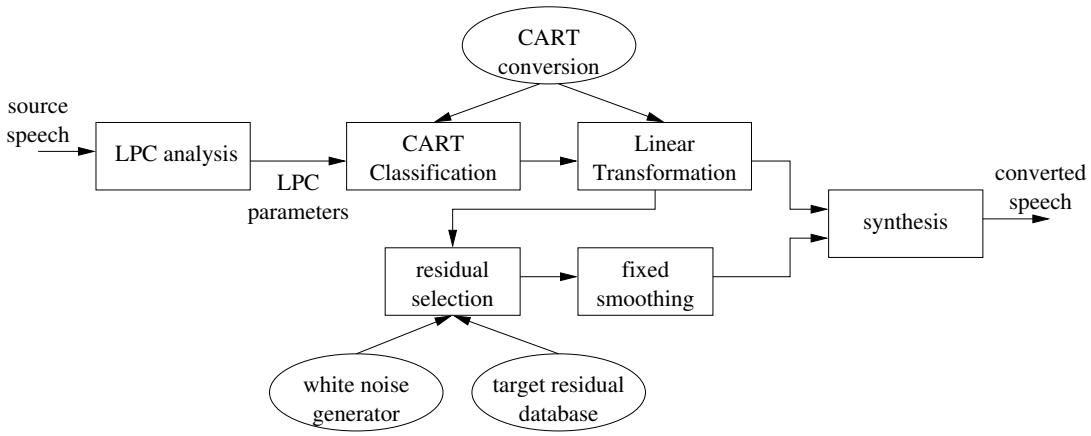


Figure 6.1: Block diagram of the operation mode of the proposed complete VC system.

6.2.2 Text-independent Training

Many VC systems require that, in the training step, source and target speakers utter the same set of sentences. Based on these parallel sentences, a transformation function is estimated. However, there are situations when it is not possible to ask the target speaker to utter a specific set of sentences present in the source corpus, for example when a set of sentences with high phonetic coverage are generated after the source corpus collection, or when the VC system is applied to a speech-to-speech translation task. Moreover, the acceptance of non-intrusive systems by most part of people is higher than the acceptance of intrusive systems. Therefore, VC systems have to evolve towards text-independent trained systems.

Intra-lingual text-independent training for VC systems applied to a TTS is not a major issue, as any source utterance may be generated by the TTS. Therefore, parallel sentences will be always available, because given N_t training sentences of the target speaker the same N_t source sentences can be synthesized by the TTS with an acceptable quality.

In order to extend this text-independent aspect to any VC system, a method for speech alignment, which can be applied to intra-lingual as well as cross-lingual VC tasks, is proposed [Dux06b]. The basis of this alignment strategy are found in VC systems applied to TTS, although it can be used for any VC task without assuming that many source speaker data is available.

The proposed approach obtains the phonetic alignment information using the unit selection module of a speech synthesizer. Having N_s training sentences of the source speaker and N_t training sentences of the target speaker, with N_t similar to N_s and N_s not large enough to build a general TTS with an acceptable output quality, the alignment procedure consists in the following two steps. First, a corpus-based speaker is built with all the N_t target training sentences. Then, the N_s training source utterances are resynthetized by a TTS, using the target corpus. This resynthesis copies the prosody (fundamental frequency contour, duration and energy) from the source utterance, but the selection module is forced to use the database corresponding to the target speaker.

At the output of the selection module, the acoustic units of the target speaker corresponding to the phonetic segmentation of the training data of the source speaker are obtained. With this information, a source-target lineal alignment using phoneme boundaries as anchor points can be carried out, as in the parallel training data situation.

Some of the constraints of the unit selection algorithm of a TTS needed to be relaxed when using it in the text-independent training task. By default, the selection works with either diphones or triphones as acoustic units, but in the text-independent training task the reduced size of the database implied that some units are missing.

The proposed text-independent training can easily be applied to cross-lingual voice conver-

sion systems if the source speaker is bilingual. In such situation, the transformation needs to be trained in the language of the target speaker and applied in the other language.

6.3 Experiments and Results

The complete proposed VC approach, including the text-independent training method, have been submitted to an evaluation of the integrated European project Corpora for Speech-to-Speech Translation (TC-STAR), where severals VC systems from universities and research companies are compared. In the following sections, there is a brief description of the integrated European project and the results that the proposed system achieved.

6.3.1 Integrated European TC-STAR Project

TC-STAR, Technology and Corpora for Speech-to-Speech Translation (www.tc-star.org), is a project funded by the European Commission with in the Sixth Program to advance in the three core technologies for Speech-to-Speech Translation: speech recognition, translation and speech synthesis. In speech synthesis, the main goals are to produce high quality speech (even for ill-formed sentences) and with the speaker identity of the source speaker. The particular framework is the translation of the speeches at the European Parliament.

The participants of TC-STAR consortium are balanced between research and technology partners, and include centers for languages resources distribution and validation.

TC-STAR organizes periodical evaluations open to external partners in all the speech-to-speech translation technologies, including speech synthesis and voice conversion. In the second campaign (March 2006), voice conversion has been evaluated in English, Mandarin and Spanish. For Spanish-English, one specific track was cross-lingual voice conversion.

Three research companies and one university have participated in the VC tasks: IBM, Nokia, Siemens, each one submitting one systems, and UPC (Universitat Politècnica de Catalunya), which submitted three different systems [Dux06b]. One of the submissions consisted in the proposed VC system based on CART for vocal tract conversion and Residual Selection with Fixed Smoothing for LP residual signal modification. Another one consisted in a TTS-back-end that uses the phonetic and prosodic representation of the source speech. The synthetic speech is produced using a concatenative synthesizer built using the target training data. This last system will serve as a comparison between VC technology and the TTS classical technology with limited data.

6.3.2 Voice Conversion Task Evaluation

VC task has been evaluated by means of two perceptual tests. The first tests consisted in a comparison of speaker identities. The second test was an evaluation of overall speech quality. Therefore, two metrics are needed: one for rating the success of the transformation in achieving the desired speaker identification, and one for rating the quality. This is needed since strong changes usually achieve the desired identity at the penalty of degrading the quality of the signal.

To evaluate the performance of the identity change, the human judges were presented with examples from the transformed speech and the target one. They have to decide using a 5-point scale if the voices comes from different speakers (1) or from the same speakers (5). Some natural examples source-target were also presented as a reference. The judges rate the transformed voice quality using a 5-points MOS scale, from bad (1) to excellent (5).

For each language (Chinese Mandarin, English, Spanish), 20 subjects were recruited to complete the tests. The subjects were between 18 and 40 years old native speakers with no known hearing problem. They were not experts in speech synthesis; they were paid for the task. Perceptual tests were carried out via the web. Subjects were required to have access to high-speed/ADSL Internet connection and good listening material.

The language resources for English and Spanish evaluations include 4 bilingual speakers English/Spanish, who recorded around 200 sentences in each language. Recordings were asked to be in a mimic style, for those systems that require it. The 200 sentences were divided in two sets: 10 sentences for testing purposes and the rest for training. The sentences were specially selected to be phonetically rich. The recordings are of high quality (96kHz, 24 bits, three synchronized channels, including a laryngograph signal). Each evaluation participant received the CGI based on the laryngograph output (text files with the time of epoch closure), the phoneme segmentation and the additional information files (text, prosodic information, etc.) additionally to the audio files. Details about the language resources can be found in [Bon06].

Spanish intra-lingual VC were developed for four different conversion pairs: two intra-gender conversions (male-male and female-female) and two cross-gender conversions (male-female and female-male). Four different systems were submitted for the Spanish intra-lingual VC task. Table 6.1 display the results of the identity test and the speech quality test for the complete VC system proposed and the TTS classical technology. Also, the results of the other two submitted systems are displayed (the company/university names are hidden for confidentiality reasons).

The last row shows how the original voices from the source and target speakers are judged to be different (rate 1.96) and to have good quality (> 4.5).

Perceptual results show that the proposed VC system and a TTS built with few training data

System	Identity	Quality
Proposed system	3.47	2.25
TTS	3.35	3.2
SRC-TGT	1.96	4.8/4.62
xxx	2.29	3.025
yyy	3.175	2.38

Table 6.1: TC-STAR evaluation results.

perform similar when changing the voice identity. In particular, the proposed system was the highest rated in identity. This is a promising result for the VC technology, because it means that, when using enough training data, a similar identity is obtained using only target speech than using only converted speech. However, with respect to speech quality, the proposed VC system needs to be improved to be acceptable in a real application. According to the results, the best VC systems in the identity evaluation were rated the worst in the quality evaluation. A tradeoff between personality change and speech quality should be the focus of further studies. At the actual level of the VC technology development, a corpus-based TTS speaker with few training data achieves better quality than any converted voice by the TC-STAR submitted systems.

6.3.3 Conclusions

Formal evaluation for Spanish language in the TC-STAR project have shown that the use of a TTS based on the almost 200 target sentences achieve significantly better speech quality than the VC systems submitted. Therefore, the state of the art of VC technology needs to be improved in terms of converted speech quality.

Of course, it is expected that the quality of the TTS will be very sensible to the amount of training data and the quality of the recordings. Therefore, a similar evaluation should be carried out with only 20 training sentences. This amount of data is more real for practical VC systems, and TTS based on 20 sentences will results in worse quality than TTS based on 200 sentences.

The proposed VC system is quite successful in achieving the identity of the target speaker. However, the quality of the speech is not good enough to be used in practical applications.

6.4 Summary

This chapter has review the three main alternatives to build a new TTS speaker, when few data is available: building a corpus-based speaker, transforming a complete database and using a VC system as a post-processing block.

A complete VC system, based on a CART with pre-pruning for vocal tract conversion and Residual Selection and Fixed Smoothing for LP residual signal modification, has been proposed. The system performance, in terms of identity conversion and speech quality, has been evaluated in the TC-STAR project framework.

TC-STAR evaluations concluded that although the proposed VC system achieves a speaker personality conversion, VC technology needs to improve the converted speech quality to be acceptable in a real application.

Chapter 7

Conclusions and Future Work

In this dissertation, the study and design of Voice Conversion (VC) systems have been addressed. In particular, the goal of this thesis was to develop a VC system to work as a post-processing block for a Text-To-Speech system (TTS). VC systems applied to a TTS have mainly two particular characteristics: source data is unlimited, as any utterance can be generated by the TTS, and phonetic information is available forehand. Both characteristics have been explored in order to improve the performance of the state of the art VC systems.

Next section summarizes the most relevant conclusions from the studies carried out in the previous chapters. Finally, in section 7.2 lines for future research that can be considered as extensions of the work developed in this dissertation are described.

7.1 Conclusions

The speech parameterization used in this dissertation is based on the source-filter theory of the speech production. In particular, LSF vectors have been used as vocal tract parameters and LP residuals have been estimated. As a consequence, the developed VC systems consisted in two different transformation sub-systems: one focus on the vocal tract parameters and another focus on the LP residual signal modification.

Chapter 4 was centered on the vocal tract transformation topic. The first study of the chapter considers the use of non-parallel source data in a vocal tract transformation system based on Gaussian Mixture Models (GMM). Previous studies had shown that including unlabelled data in classification problems increased the performance of the classification for specific applications. A similar hypothesis has been made for regression problems in this dissertation.

Two approaches have been proposed for the used of source non-parallel data in GMM systems: a modified EM algorithm with fixed covariance matrices, and a strategy to complete non-parallel data by including transformed vectors as parallel vectors. The conclusion of the

study, based on objective evaluations, was that a combined learning with parallel source-target data and source-transformed data increases the conversion performance, mainly when few training data is available. In this latest situation, to re-estimate only means and mixture weights also increases the performance, with a very reduced computational time.

The dissertation contribution to the vocal tract transformation topic also includes two novel conversion systems: Hidden Markov Models (HMM) based systems and Classification and Regression Trees (CART) systems.

HMM systems have been studied to include dynamic information in the acoustic model in order to better convert phoneme boundary frames. However, objective results have shown that the performance of HMM vocal tract conversion systems for a limited number of training data is not higher that the performance of joint GMM regression systems. Further studies should explore the possibility that for enough training data HMM systems will over-perform GMM systems.

CART systems have been studied to include dynamic information in the classification of the acoustic parameters prior to the application of the mapping function. Our hypothesis was that phonetic data carries information that allows to better split the acoustic space according to the transformation error. Both objective and perceptual results corroborated this hypothesis. CART systems, specially CART with pre-pruning systems, resulted in the best performance when the training set contains at least 10 sentences.

Moreover, the algorithm to estimate CART with pre-pruning systems is simpler than the EM algorithm for GMMs, what makes the use of CART systems very attractive. However, CART systems are more sensible to errors in the speech phonetic segmentation and transcription than the other studied systems. It seems to be a requirement that at least one of the involved speakers in the conversion have a high quality phonetic information.

In chapter 5, the LP residual modification problem was studied. After an analysis of previous published works, two strategies to face the residual modification problem have been identified. On one hand, Residual Conversion systems modify source residuals by some mapping function. On the other hand, Residual Prediction systems predict the converted residuals from the converted vocal tract parameters. A comparative of both strategies have been carried out, concluding that the relationship between vocal tract parameters of a given speaker and their corresponding residuals works better than the relationship between aligned source-target residuals, in terms of conversion performance.

An extension of a state of the art residual selection system has been proposed, in order to improve the residual selection by using a collection of databases organized by phonetic characteristics. However, perceptual results have reveled that the use of the collection of databases decreases the system performance. Consequently, Phonetic Residual Selection systems should

be only employed when computational requirements forces it.

In chapter 6, a complete VC system, based on a CART with pre-pruning for vocal tract conversion and Residual Selection with Fixed Smoothing, has been proposed. The complete system has been submitted to a formal evaluation of the integrated European project TC-STAR. Perceptual test results of the open evaluation have revealed that the proposed system succeed in achieving the identity of the target speaker. However, the converted speech quality of state of the art VC systems is not good enough to be used in practical applications, non of the submitted systems was rated with better quality than a corpus-based TTS speaker built with the target training data. Further work is required to improve the quality of complete VC systems.

7.2 Future work

There are several lines for future research that can be considered as extensions of the work developed in this dissertation. Four major areas, which will benefit greatly from further research, are briefly discussed in the following paragraphs.

Speech Model Due to the vocal tract and the glottal tract are located physically in different places, it can be assumed that two different relationships will connect the vocal tract of two speakers and the glottal tract of the same two speakers. The LPC analysis technique that has been used in the current dissertation can not separate the contribution of the vocal tract from the contribution of the glottal tract. Thus, the LP residual signal obtained is an error signal, somehow related to the speaker's glottal flow, but not the proper glottal flow. Moreover, the obtained vocal tract parameters also model some glottal characteristics. Therefore, a more sophisticated vocal tract model, which takes into account the speaker's glottal flow characteristics, may provide better speech features to the conversion task.

Speech Alignment Speech alignment generates the training data used to estimate mapping functions. Therefore, its performance is critical. Although several alignment methods are used in the state of the art VC systems, we haven't found any comparative study about the topic. A new alignment, which operates based on conversion results, may improve the conversion performance of the proposed systems.

As another future research, the text-independent alignment procedure may also be extended to perform cross-lingual VC.

Vocal Tract Mapping Functions Two main lines of future research can extend the work carried out in the vocal tract mapping. As a first line, HMM systems may be further study in situations where more training data is available. Objective results have shown

that HMM systems are able to convert vocal tract parameters from a source speaker to a target speaker, but their performance was lower than GMM performance. This lower performance can be due to estimation problems of HMMs. Therefore, it will be interesting to study HMM and GMM performance when more training data is available.

As a second line of future research, we propose to replace the linear transformation functions associated to each CART leaves for non-linear mappings. Non-linear functions may capture the source-target relationship more accurately. These non-linear mappings should be continuous functions and defined in all the acoustic space, in order not to degrade the converted speech quality.

Residual modification techniques Residual modification techniques are the less developed aspect of the Voice Conversion systems, and thus future works must be carried out to increase their performance, in terms of speech quality. The improvement of these systems may begin with a parameterization of the target residual signal, in order to reduce the memory load of residual selection techniques. Moreover, residual parameterization may allow to concatenate sequences of residual signals without artifacts.

Finally, a detailed conversion for prosodic characteristics, such as fundamental frequency time evolution, phoneme duration or intensity, is required to have a complete Voice Conversion system. And in a higher level, linguistic characteristics of the target speaker may be incorporated to the converted speech.

Appendix A

Derivation of the EM Algorithm Estimated over Parallel+Non-Parallel Data with Fixed Covariance Matrices

Expression of the incomplete-data log-likelihood of a GMM model, estimated with parallel and non-parallel data:

$$L(\theta^t \mid \mathbf{X}, \mathbf{Y}) = \prod_{i=0}^{N-1} p(\mathbf{x}_i, \mathbf{y}_i) \prod_{j=N}^{N+M-1} p(\mathbf{x}_j), \tag{A.1}$$

where

$$p(\mathbf{x}, \mathbf{y}) = \sum_{q=0}^{Q-1} \boldsymbol{\alpha}_q N\left((\mathbf{x}, \mathbf{y}), \begin{pmatrix} \mu_q^x \\ \mu_q^y \end{pmatrix}, \begin{pmatrix} \boldsymbol{\Sigma}_q^{xx} & \boldsymbol{\Sigma}_q^{xy} \\ \boldsymbol{\Sigma}_q^{yx} & \boldsymbol{\Sigma}_q^{yy} \end{pmatrix}\right) \tag{A.2}$$

and

$$p(\mathbf{x}) = \sum_{q=0}^{Q-1} \boldsymbol{\alpha}_q N(x, \mu_q^x, \boldsymbol{\Sigma}_q^{xx}). \tag{A.3}$$

Expression of the complete-data log-likelihood of a GMM model, estimated with parallel and non-parallel data:

$$L(\theta^t \mid \mathbf{X}, \mathbf{Y}, \mathbf{K}) = \sum_{i=0}^{N-1} \log\left(\alpha_{ki} N(\mathbf{x}_i, \mathbf{y}_i \mid \mu_{ki}, \boldsymbol{\Sigma}_{ki})\right) + \sum_{i=N}^{N+M-1} \log\left(\alpha_{ki} N(\mathbf{x}_i \mid \mu_{ki}^x, \boldsymbol{\Sigma}_{ki}^{xx})\right). \tag{A.4}$$

E step. The E-step consists in finding the expected value of the complete-data log-likelihood with respect to the unknown data $\mathbf{K}$, given the observed data $\mathbf{X}$ and $\mathbf{Y}$ and the current parameter estimates. We define $Q(\theta^t, \theta^{t-1}) = E\left[\log p(\mathbf{X}, \mathbf{Y} \mid \theta^t) \mid \mathbf{X}, \theta^{t-1}\right]$, where t denotes the current iteration.

$$Q(\theta^t,\theta^{t-1}) = \sum_{k\in K}\left(\sum_{i=0}^{N-1}\log(\alpha_{ki}N(\mathbf{x}_i,\mathbf{y}_i \mid \mu_{ki},\mathbf{\Sigma}_{ki}))\right)\prod_{i=0}^{N-1}\frac{\alpha_{ki}^{t-1}N(\mathbf{x}_i,\mathbf{y}_i \mid \mu_{ki}^{t-1},\mathbf{\Sigma}_{ki}^{t-1})}{\sum_{q=0}^{Q-1}\alpha_q^{t-1}N(\mathbf{x}_i,\mathbf{y}_i \mid \mu_q^{t-1},\mathbf{\Sigma}_q^{t-1})}+$$

$$\sum_{k\in K}\left(\sum_{i=N}^{N+M-1}\log(\alpha_{ki}N(\mathbf{x}_i,\mathbf{y}_i \mid \mu_{ki}^{x},\mathbf{\Sigma}_{ki}^{xx}))\right)\prod_{i=0}^{N-1}\frac{\alpha_{ki}^{t-1}N(\mathbf{x}_i,\mathbf{y}_i \mid \mu_{ki}^{x,t-1},\mathbf{\Sigma}_{ki}^{xx,t-1})}{\sum_{q=0}^{Q-1}\alpha_q^{t-1}N(\mathbf{x}_i,\mathbf{y}_i \mid \mu_q^{x,t-1},\mathbf{\Sigma}_q^{xx,t-1})} \quad \text{(A.5)}$$

This expression can be simplified:

$$Q(\theta^t,\theta^{t-1}) = \sum_{q=0}^{Q-1}\sum_{i=0}^{N-1} log(\alpha_q)p(q \mid \mathbf{x}_i\mathbf{y}_i,\theta^{t-1}) + \sum_{q=0}^{Q-1}\sum_{i=0}^{N-1} log\left(N(\mathbf{x}_i\mathbf{y}_i \mid \theta^{t-1})\right)p(q \mid \mathbf{x}_i\mathbf{y}_i,\theta^{t-1})+$$

$$\sum_{q=0}^{Q-1}\sum_{i=N}^{N+M-1} log(\alpha_q)p(q \mid \mathbf{x}_i,\theta^{t-1}) + \sum_{q=0}^{Q-1}\sum_{i=N}^{N+M-1} log\left(N(\mathbf{x}_i \mid \theta^{t-1})\right)p(q \mid \mathbf{x}_i,\theta^{t-1}). \quad \text{(A.6)}$$

M step. The M step consists in maximizing the expression of the E step, that is: $\theta^t = \text{argmax}_\theta\, Q(\theta^t,\theta^{t-1})$. To maximize this expression, we can maximize the term containing α_q and the term containing θ_q independently, since they are not related.

To find the expression for α_q, the Lagrange multiplier λ with the constraint that $\sum_{q=0}^{Q-1}\alpha_q = 1$ is introduced:

$$\frac{\delta}{\delta\alpha_q}\left[\sum_{q=0}^{Q-1}\sum_{i=0}^{N-1} log(\alpha_q)p(q \mid \mathbf{x}_i\mathbf{y}_i,\theta^{t-1}) + \sum_{q=0}^{Q-1}\sum_{i=N}^{N+M-1} log(\alpha_q)p(q \mid \mathbf{x}_i,\theta^{t-1}) + \lambda\left(\sum_{q=0}^{Q-1}\alpha_q - 1\right)\right] = 0. \quad \text{(A.7)}$$

The resulting expression for α_q is:

$$\alpha_q^t = \frac{1}{N+M}\left(\sum_{n=0}^{N-1}P(w_q \mid \mathbf{x}_n,\mathbf{y}_n;\theta_q^{t-1}) + \sum_{n=N}^{N+M-1}P(w_q \mid \mathbf{x}_n;\theta_q^{t-1})\right) \qquad q = 0\ldots Q-1. \quad \text{(A.8)}$$

When maximizing the term containing θ_q, we should take into account that only μ_q^x and μ_q^y are variables. Therefore, the resulting expressions for μ_q^x and μ_q^y are:

$$\mu_q^{x,t} = \frac{\sum_{n=0}^{N-1}\mathbf{x}_nP(w_q \mid \mathbf{x}_n,\mathbf{y}_n;\theta_q^{t-1}) + \sum_{n=N}^{N+M-1}\mathbf{x}_nP(w_q \mid \mathbf{x}_n;\theta_q^{t-1})}{\sum_{n=0}^{N-1}P(w_q \mid \mathbf{x}_n,\mathbf{y}_n;\theta_q^{t-1}) + \sum_{n=N}^{N+M-1}P(w_q \mid \mathbf{x}_n;\theta_q^{t-1})}. \quad \text{(A.9)}$$

$$\mu_q^{y,t} = \frac{-\Sigma_q^{yx}\Sigma_q^{xx}\left(\sum_{n=0}^{N-1}(\mathbf{x}_n-\mu_q^{x,t})P(w_q \mid \mathbf{x}_n,\mathbf{y}_n;\theta_q^{t-1})\right)}{\sum_{n=0}^{N-1}P(w_q \mid \mathbf{x}_n,\mathbf{y}_n;\theta_q^{t-1})}+$$

$$\frac{\sum_{n=0}^{N-1}\mathbf{x}_nP(w_q \mid \mathbf{x}_n,\mathbf{y}_n;\theta_q^{t-1})}{\sum_{n=0}^{N-1}P(w_q \mid \mathbf{x}_n,\mathbf{y}_n;\theta_q^{t-1})}. \quad \text{(A.10)}$$

Appendix B

Objective Test Results by Source-Target Speaker Pair

This appendix contains the results of the objective evaluation of the vocal tract conversion systems under study. The following figures display the performance index P values for joint GMM regression, HMM based conversion, CART with pre-pruning and CART with post-pruning systems, broken down by the source speaker and the number of training sentences. The figures are presented in groups of four, two figure on the left page and two figures on the right page, to facilitate comparisons.

Performance conversions with MALE_1 as source speaker:

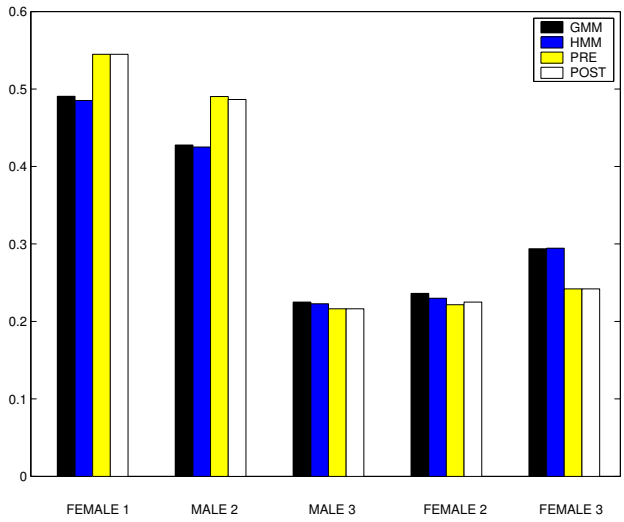


Figure B.1: Performance for conversions with MALE_1 as source speaker trained with Set_30.

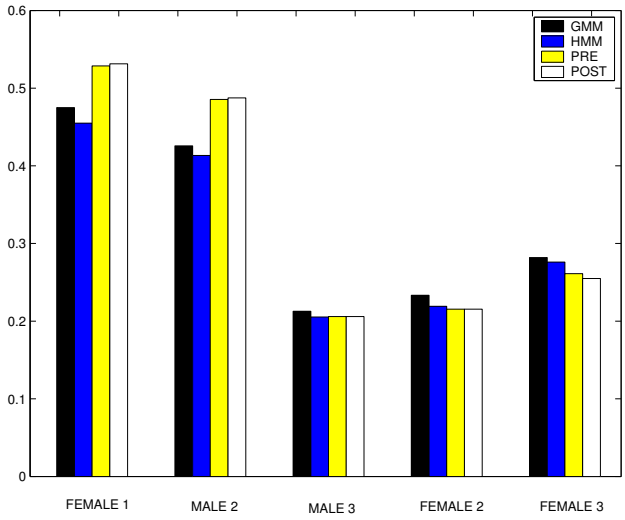


Figure B.2: Performance for conversions with MALE_1 as source speaker trained with Set_20.

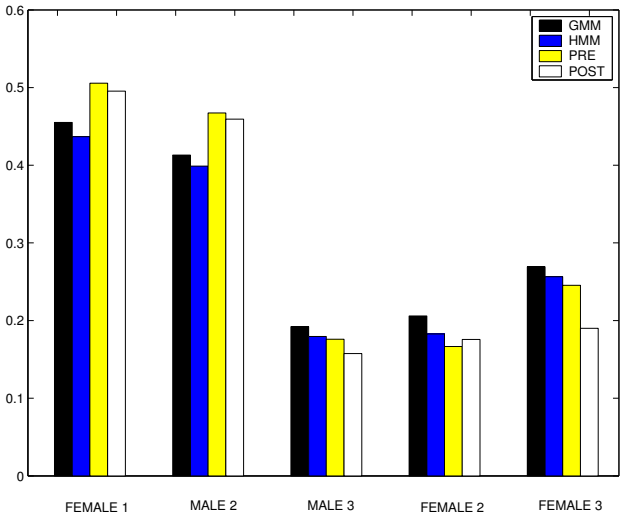


Figure B.3: Performance for conversions with MALE_1 as source speaker trained with Set_10.

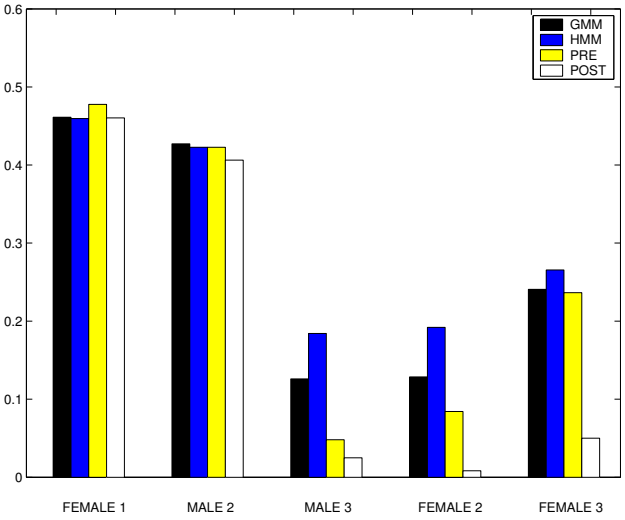


Figure B.4: Performance for conversions with MALE_1 as source speaker trained with Set_05.

Performance conversions with FEMALE_1 as source speaker:

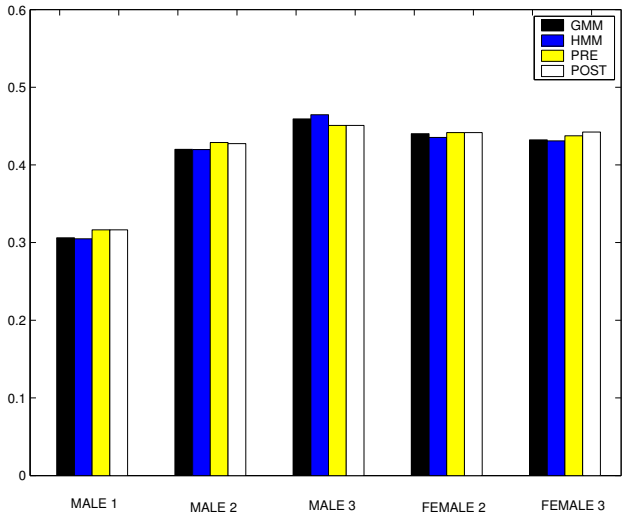


Figure B.5: Performance for conversions with FEMALE_1 as source speaker trained with Set_30.

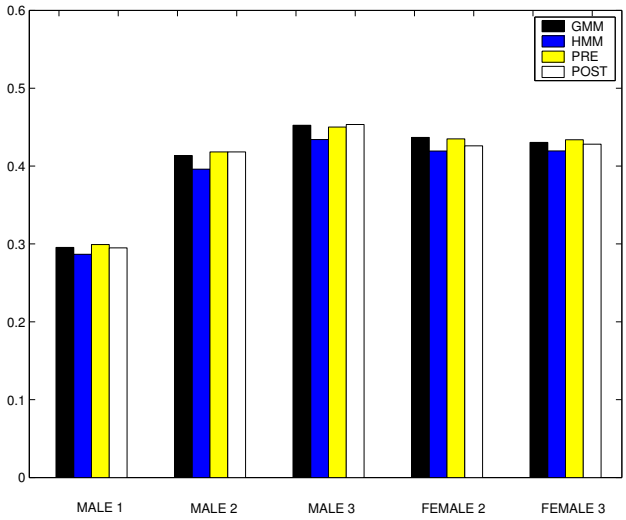


Figure B.6: Performance for conversions with FEMALE_1 as source speaker trained with Set_20.

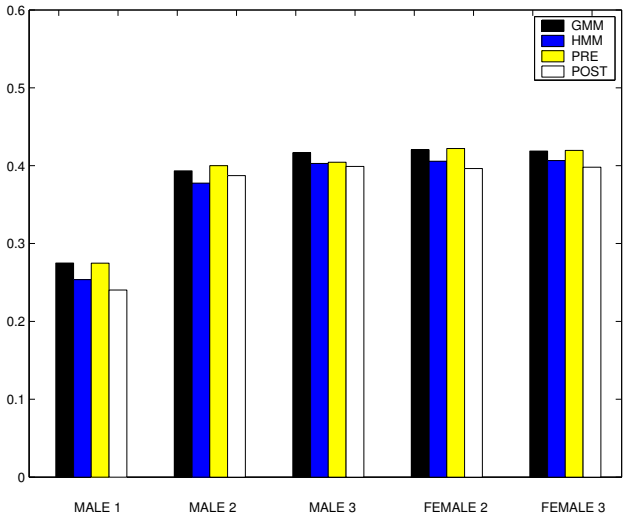


Figure B.7: Performance for conversions with FEMALE_1 as source speaker trained with Set_10.

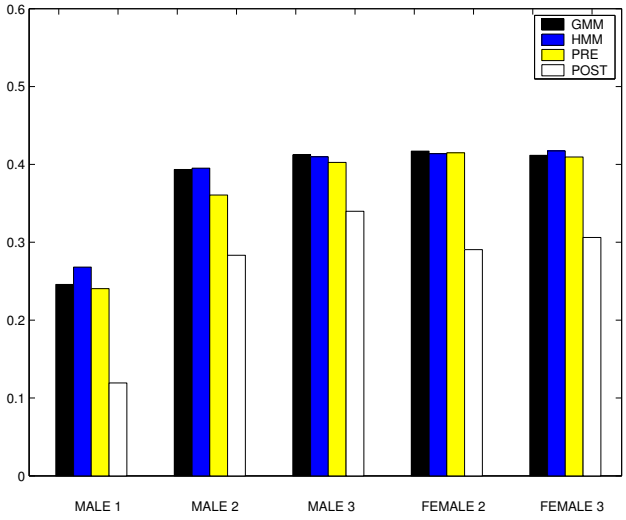


Figure B.8: Performance for conversions with FEMALE_1 as source speaker trained with Set_05.

Performance conversions with MALE_2 as source speaker:

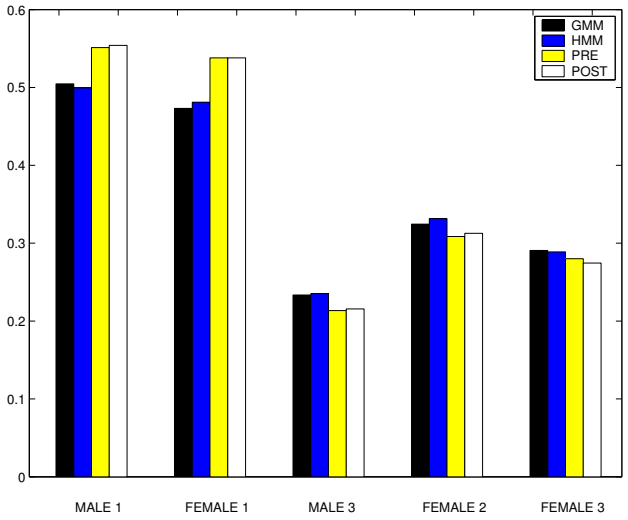


Figure B.9: Performance for conversions with MALE_2 as source speaker trained with Set_30.

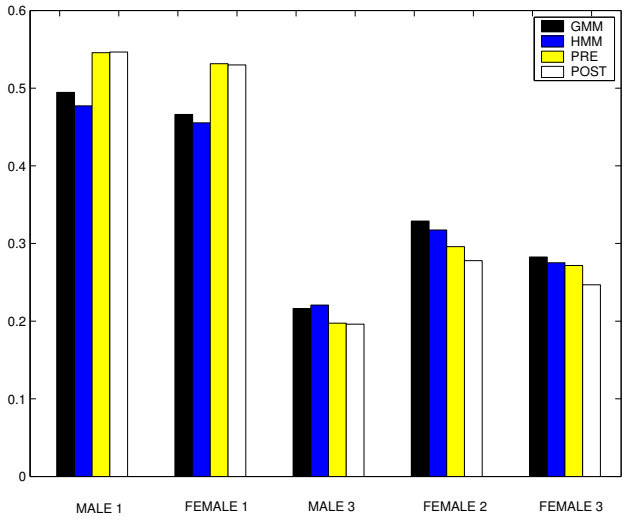


Figure B.10: Performance for conversions with MALE_2 as source speaker trained with Set_20.

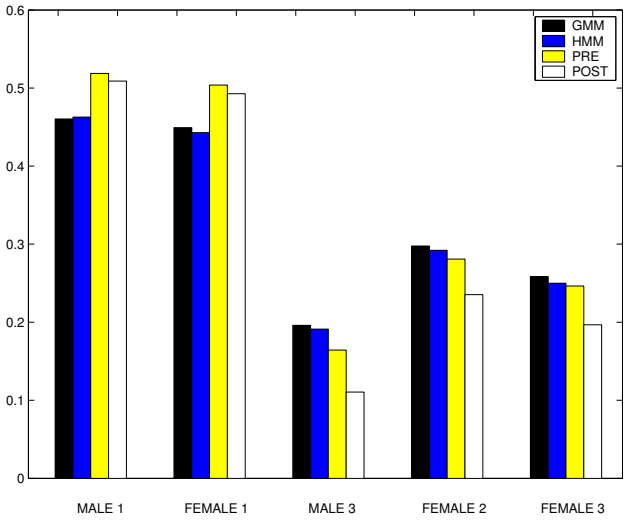


Figure B.11: Performance for conversions with MALE_2 as source speaker trained with Set_10.

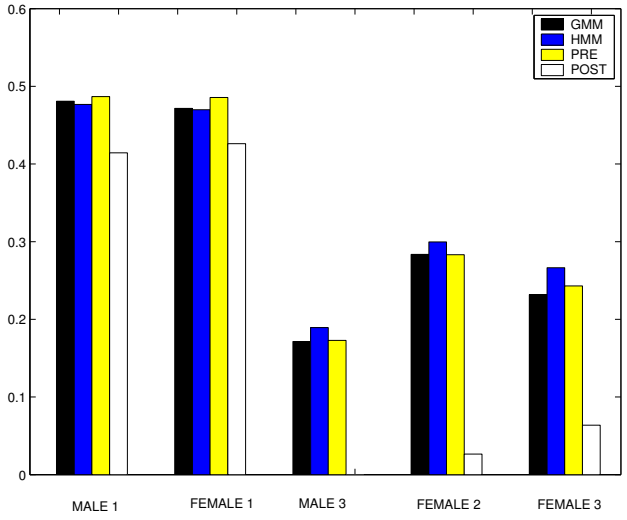


Figure B.12: Performance for conversions with MALE_2 as source speaker trained with Set_05.

Performance conversions with MALE_3 as source speaker:

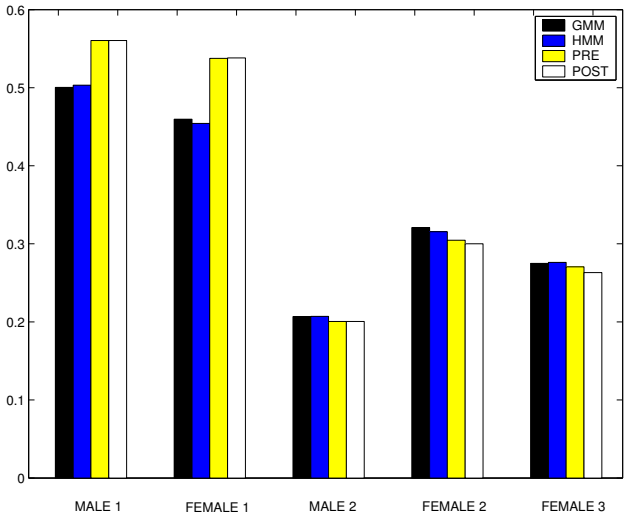


Figure B.13: Performance for conversions with MALE_3 as source speaker trained with Set_30.

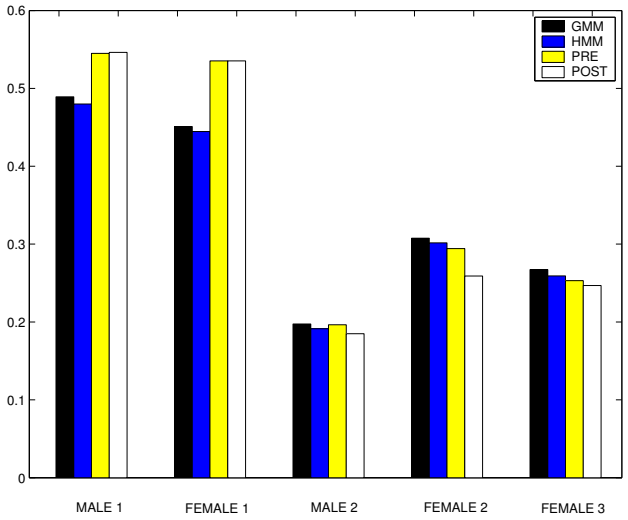


Figure B.14: Performance for conversions with MALE_3 as source speaker trained with Set_20.

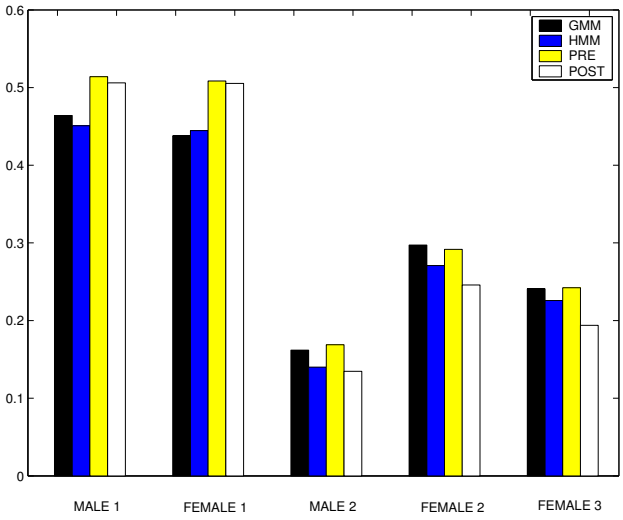


Figure B.15: Performance for conversions with MALE_3 as source speaker trained with Set_10.

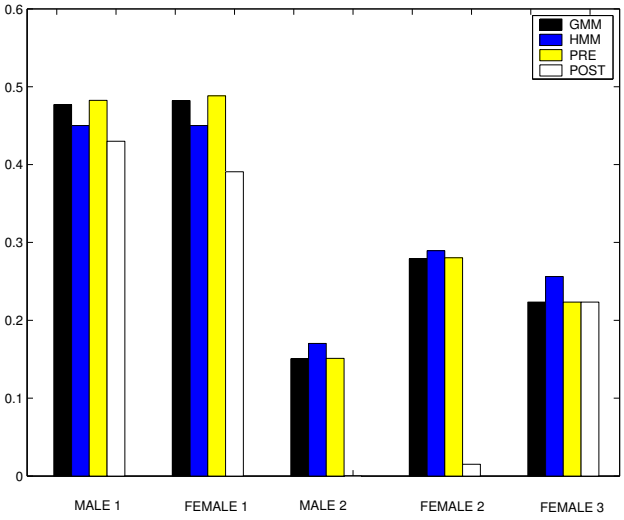


Figure B.16: Performance for conversions with MALE_3 as source speaker trained with Set_05.

Performance conversions with FEMALE_2 as source speaker:

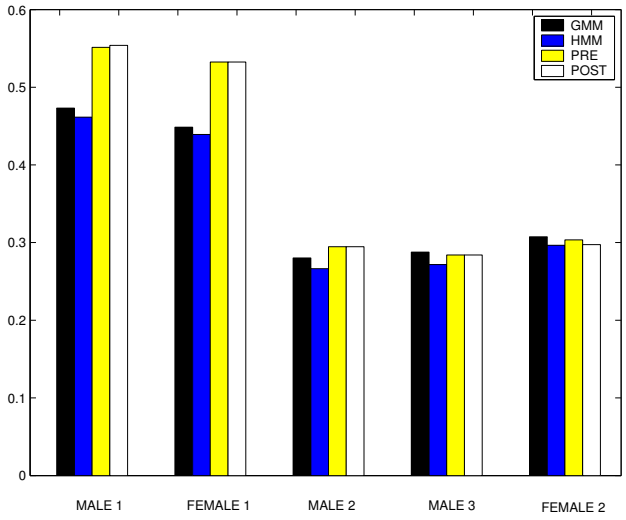


Figure B.17: Performance for conversions with FEMALE_2 as source speaker trained with Set_30.

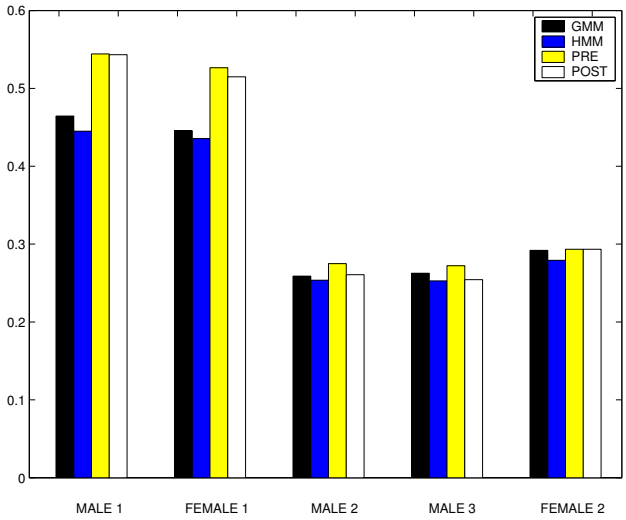


Figure B.18: Performance for conversions with FEMALE_2 as source speaker trained with Set_20.

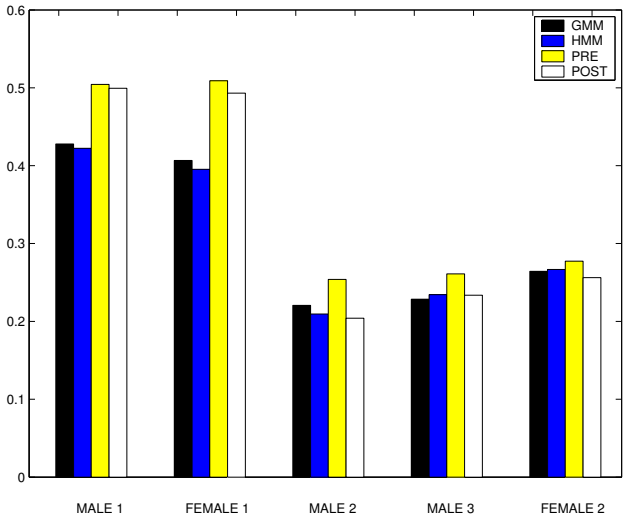


Figure B.19: Performance for conversions with FEMALE_2 as source speaker trained with Set_10.

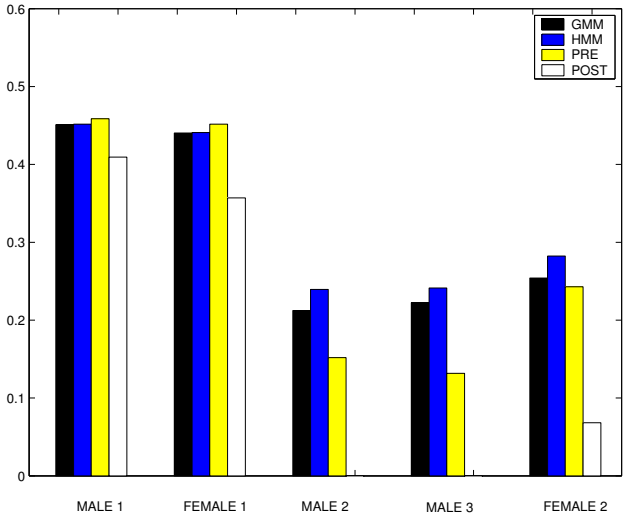


Figure B.20: Performance for conversions with FEMALE_2 as source speaker trained with Set_05.

Performance conversions with FEMALE_3 as source speaker:

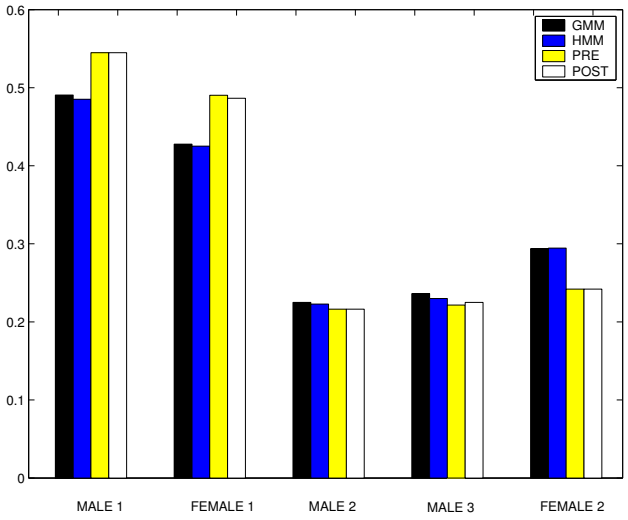


Figure B.21: Performance for conversions with FEMALE_3 as source speaker trained with Set_30.

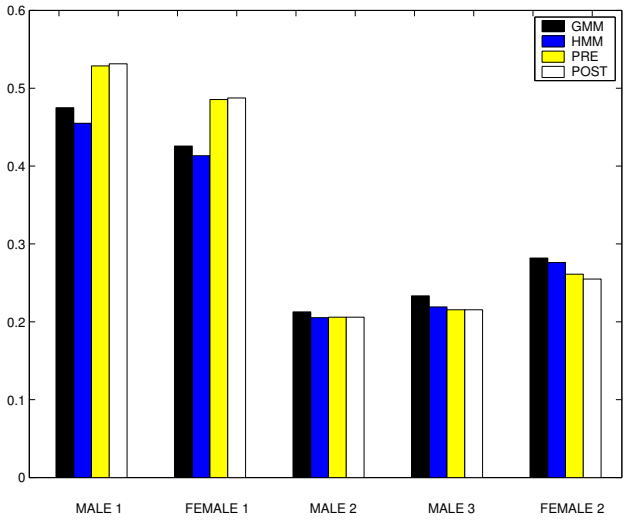


Figure B.22: Performance for conversions with FEMALE_3 as source speaker trained with Set_20.

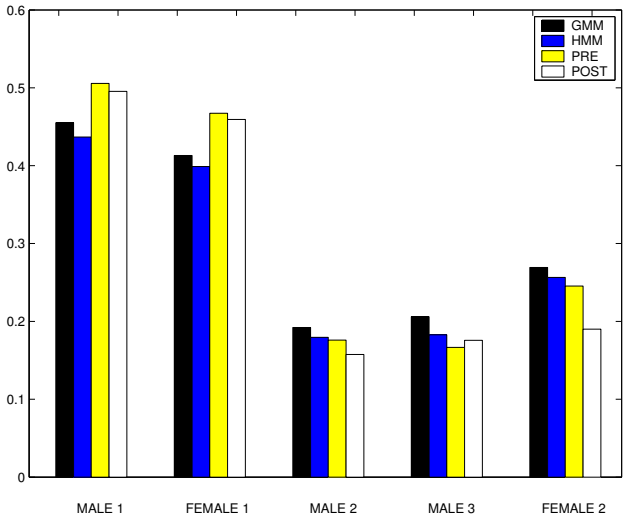


Figure B.23: Performance for conversions with FEMALE_3 as source speaker trained with Set_10.

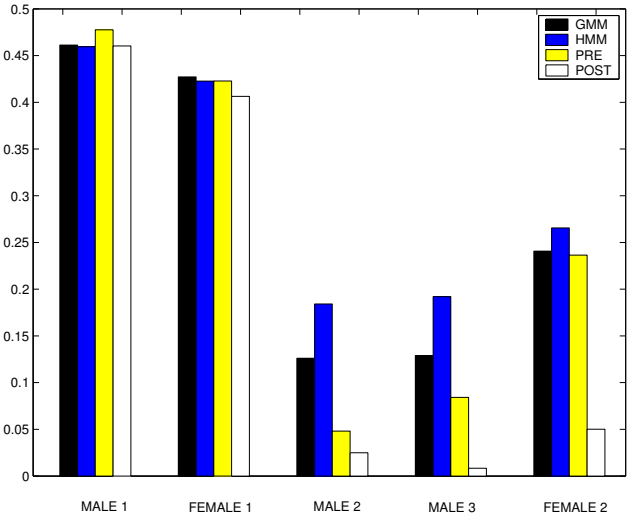


Figure B.24: Performance for conversions with FEMALE_3 as source speaker trained with Set_05.

Bibliography

[Abe88] M. Abe, S. Nakamura, K. Shikano, and H. Kuwabara, "Voice conversion vector quantization", *International Conference on Acoustics, Speech, and Signal Processing*, 1988.

[Agü05] P.D. Agüero, J. Adell, and A. Bonafonte, "Improving TTS quality using pitch contour information of source speaker in S2ST framework", *International Workshop Advances in Speech Technology 2005*, 2005.

[Ars99] L.M. Arslan, "Speaker Transformation Algorithm using Segmental Codebooks (STASC)", *Speech Communication*, vol. 28, pp. 211–226, 1999.

[Bau96] G. Baudoin, and Y. Stylianou, "On the transformation of the speech spectrum for voice conversion", *International Conference on Spoken Language Processing*, pp. 1405–1408, 1996.

[Bil98] Jeff A. Bilmes, "A Gentle Tutorial of the EM Algorithm and its application to parameter estimation for Gaussian Mixture and Hidden Markov Models", *International Computer Science Insitute Technical Reports*, 1998.

[Bla00] A. Black, and K. Lenzo, "Limited domain synthesis", *International Conference on Spoken Language Processing*, 2000.

[Boi04] M.A. Boillot, and J.G. Harris, "A loudness enhancement technique for speech", *International Symposium on Circuits and Systems*, pp. V–616 – V–618, 2004.

[Bon06] A. Bonafonte, H. Höge, I. Kiss, A. Moreno, U. Ziegenhain, H. van der Heuvel, H.U. Hain, X.S. Wang, and M.N. Garcia, "TC-STAR: especifications of language resources and evaluation for speech synthesis", *LREC*, 2006.

[Bre98] L. Breiman, *Classification and regression trees*, Chapman & Hall, 1998.

[Cey02] T. Ceyssens, W. Verhelst, and P. Wambacq, "On the construction of a pitch conversion system", *Europen Signal Processing Conference*, 2002.

[Che03] Y. Chen, M. Chu, E. Chang, J. Liu, and R. Liu, "Voice conversion with smoothed GMM and MAP adaptation", *European Conference on Speech Communication and Technology*, pp. 2413–2416, 2003.

[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*, 1977.

[Dep97] H. Depalle, and T. Hélie, "Extraction of Spectral Peak Parameters using a Short-Time Fourier Transform Modeling and No Sidelobe Windows", *ASSP Workshop on Applications of Signal Processing to Audio and Acoustics*, 1997.

[Dud01] R. O. Duda, P.E. Hart, and D.G. Stork, *Pattern Classification*, John Wiley and Sons Inc., 2001.

[Dux02] H. Duxans, and A. Bonafonte, "Revisión de técnicas de estimación del pulso glotal", *II Jornadas de Tecnologías del Habla*, 2002.

[Dux03] H. Duxans, and A. Bonafonte, "Estimation of GMM in voice conversion including unaligned data", *European Conference on Speech Communication and Technology*, 2003.

[Dux04a] H. Duxans, A. Bonafonte, A. Kain, and J. van Santen, "Including dynamic and phonetic information in voice conversion systems", *International Conference on Spoken Language Processing*, 2004.

[Dux04b] H. Duxans, A. Bonafonte, A. Kain, and J. van Santen, "Including dynamic information in voice conversion systems", *XX Congreso de la Sociedad Española para el Procesamiento del Lenguaje Natural*, 2004.

[Dux06a] H. Duxans, and A. Bonafonte, "Residual Conversion versus Prediction on Voice Morphing Systems", *International Conference on Acoustics, Speech, and Signal Processing*, 2006.

[Dux06b] H. Duxans, D. Erro, J. Perez, F. Diego, A. Bonafonte, and A. Moreno, "Voice Conversion of Non-aligned Data using Unit Selection", *TC-Star Speech to Speech Translation Workshop*, 2006.

[GA98] J.M. Gutierrez-Arriola, Y.S. Hsiao, J.M. Montero, J.M. Pardo, and D.G. Childers, "Voice Conversion based of parameter transformation", *International Conference on Spoken Language Processing*, 1998.

[Has95] M. Hashimoto, and N. Higuchi, "Spectral mapping for voice conversion using speaker selection and vector field smoothing", *European Conference on Speech Communication and Technology*, pp. 431–434, 1995.

[Hos03] J.P. Hosom, A.B. Kain, T. Mishra, J.P.H. van Santen, M. Fried-Oken, and J. Staehely, "Inteligibility of modifications to dysarthric speech", *International Conference on Acoustics, Speech, and Signal Processing*, pp. 924–927, 2003.

[Hua01] X. Huang, A. Acero, and H.W. Hon, *Spoken Language Processing*, Prentice Hall PTR, 2001.

[Iwa94] N. Iwahashi, and Y. Sagisaka, "Speech Spectrum transformation by speaker interpolation", *International Conference on Acoustics, Speech, and Signal Processing*, 1994.

[Iwa95] N. Iwahashi, and Y. Sagisaka, "Speech spectrum conversion based on speaker interpolation and multi-functional representation with weighting by radial basis function networks", *Speech Communication*, vol. 16, pp. 139–151, 1995.

[Kai98a] A. Kain, and M. W. Macon, "Spectral voice conversion for text-to-speech synthesis", *International Conference on Acoustics, Speech, and Signal Processing*, 1998.

[Kai98b] A. Kain, and M. W. Macon, "Text-to-Speech voice adaptation from sparse training data", *International Conference on Spoken Language Processing*, 1998.

[Kai01a] A. Kain, *High resolution voice transformation*, PhD Thesis, OGI school of science and engineering, 2001.

[Kai01b] A. Kain, and M. W. Macon, "Design and evaluation of a voice conversion algorithm based on spectral envelope mapping and residual prediction", *International Conference on Acoustics, Speech, and Signal Processing*, 2001.

[Kaw03] H. Kawanami, Y. Iwami, T. Toda, H. aruwatari, and K. Shikano, "GMM-based voice conversion applied to emotional speech synthesis", *European Conference on Speech Communication and Technology*, pp. 2401–2404, 2003.

[Kay98] S.M. Kay, *Fundamentals of statistical signal processing*, Englewood Cliffs Prentice-Hall, 1998.

[Kim97] E.K. Kim, S. Lee, and Y.H. Oh, "Hidden Markov model based voice conversion using dynamic characteristics of speaker", *European Conference On Speech Communication And Technology*, pp. 1311–1314, 1997.

[Lar91] R. Laroia, N. Phamdo, and N. Farvardin, "Robust and efficient quantization of speech LSP parameters using structured vector quantizers", *International Conference on Acoustics, Speech, and Signal Processing*, pp. 641–644, 1991.

[Lee96] K.S. Lee, D.H. Youn, and I.W. Cha, "A new voice transformation method based on both linear and nonlinear prediction analysis", *International Conference on Spoken Language Processing*, pp. 1401–1404, 1996.

[Mas96] T. Masuko, K. Tokuda, T. Kobayashi, and S. Imai, "Speech synthesis using HMMs with dynamic features", *International Conference on Acoustics, Speech, and Signal Processing*, pp. 389–392, 1996.

[Mas97] T. Masuko, K. Tokuda, T. Kobayashi, and S. Imai, "Voice Characteristics Conversion for HMM-based speech synthesis system", *International Conference on Acoustics, Speech, and Signal Processing*, pp. 1611–1614, 1997.

[Mas01] M. Mashimo, T. Toda, K. Shikano, and N. Campbell, "Evaluation of cross-language voice conversion based on GMM and STRAIGHT", *European Conference on Speech Communication and Technology*, 2001.

[Mas02] M. Mashimo, T. Toda, H. Kawanami, H. Kashioka, K. Shikano, and N. Campbell, "Evaluation of cross-language voice conversion using bilingual and non-bilingual databases", *International Conference on Spoken Language Processing*, 2002.

[Mas05] T. Masuko, K. Tokuda, T. Kobayashi, and S. Imai, "Polyglot Synthesis using a Mixture of Monolingual Corpora", *International Conference on Acoustics, Speech, and Signal Processing*, 2005.

[Mil97] D.J. Miller, and H.S. Uyar, "A mixture of experts classifier with learning based on both labelled and unlabelled data", *Advances in Neural Information Processing Systems*, pp. 571–577, 1997.

[Miz95] H. Mizuno, and M. Abe, "Voice conversion algorithm based on piecewise linear conversion rules of formant frequency and spectrum tilt", *Speech Communication*, 1995.

[Mor03] H. Mori, and H. Kasuya, "Speaker conversion in ARX-based source-formant type speech synthesis", *European Conference on Speech Communication and Technology*, pp. 2421–2424, 2003.

[Mou90] E. Moulines, and F. Chanpentier, "Pitch Synchronous Waveform Procesing Techniques for Text-tp-Speech Synthesis Using Diphones", *Speech Communication*, 1990.

[Nar95] M. Narendranath, H.A. Murthy, S. Rajendran, and B. Yegnanarayana, "Tranformation of formants for voice conversion using artificial neural networks", *Speech Communication*, 1995.

[Qui87] J.R. Quinlan, "Simplifying decision trees", *International Journal of Man-Machine sudies*, 1987.

[Rab93] L. Rabiner, and B. H. Juang, *Fundamentals of Speech Recognition*, Prentice Hall PTR, 1993.

[Ren03] D. Rentzos, S. Vaseghi, Q. Yan, C. Ho, and E. Turajlic, "Probability models of formants parameters for voice conversion", *European Conference on Speech Communication and Technology*, 2003.

[Rey95] D.A. Reynolds, and R.C. Rose, "Robust test-independent speaker identifications using Gaussian Mixture speaker models", *SAP*, 1995.

[Rot99a] J. Rothweiler, "A Rootfinding Algorithm for Line Spectral Frequencies", *International Conference on Acoustics, Speech, and Signal Processing*, 1999.

[Rot99b] J. Rothweiler, "On Polynomial Reduction in the Computation of LSP Frequencies", *IEEE Transactions on Speech and Audio Processing*, September 1999.

[Shi91] K. Shikano, S. Nakamura, and M. Abe, "Speaker Adaptation and Voice Conversion by Codebook Mapping", *IEEE International Symposium on Circuits and Systems*, pp. 594–597, 1991.

[Soo84] F.K. Soong, and B.H. Juang, "Line Spectrum Pair (LSP) and Speech data compresion", *International Conference on Acoustics, Speech, and Signal Processing*, 1984.

[Sty95] Yannis Stylianou, Olivier Cappé, and Eric Moulines, "Statistical methods for voice quality transformation", *European Conference on Speech Communication and Technology*, 1995.

[Sty96] Yannis Stylianou, *Harmonic plus Noise Models for Speech, combined with statistical methods, for Speech and Speaker Modification*, PhD Thesis, Ecole Nationale Supérieure des Télécommunications, 1996.

[Sty98] Yannis Stylianou, Olivier Cappé, and Eric Moulines, "Continous Probabilistic Transform for Voice Conversion", *IEEE Transactions on Speech and Audio Processing*, 1998.

[Sün03a] D. Sündermann, and H. Höge, "VTLN-Based Cross-Language Voice Conversion", *IEEE Automatic Speech Recognition and Understanding Workshop*, pp. 676–681, 2003.

[Sün03b] D. Sündermann, and H. Ney, "An Automatic Segmentation and Mapping Approach for Voice Conversion Parameter Training", *International Workshop on Advances in Speech Technology*, 2003.

[Sün05a] D. Sündermann, A. Bonafonte, H. Duxans, and H. Höge, "TC-STAR: Evaluation Plan for Voice Conversion Technology", *German Annual Conference on Acoustics, DAGA*, 2005.

[Sün05b] D. Sündermann, A. Bonafonte, H. Ney, and H. Höge, "A Study on Residual Prediction Techniques for Voice Conversion", *International Conference on Acoustics, Speech, and Signal Processing*, 2005.

[Sün05c] D. Sündermann, H. Höge, A. Bonafonte, and H. Duxans, "Residual Prediction", *International Symposium on Signal Processing and Information Technology*, 2005.

[Sün05d] D. Sündermann, G. Strecha, A. Bonafonte, H. Hoege, and H. Ney, " Evaluation of VTLN-Based Voice Conversion for Embedded Speech Synthesis", *European Conference on Speech Communication and Technology*, 2005.

[Tam01] M. Tamura, T. Masuko, K. Tokuda, and T. Kobayashi, "Text-to-Speech synthesis with arbitrary speaker's voice from average voice", *European Conference on Speech Communication and Technology*, 2001.

[Tod00] T. Toda, J. Lu, H. Saruwatari, and K. Shikano, "STRAIGHT-based voice conversion algorithm based on Gaussian Mixture Model", *International Conference on Spoken Language Processing*, pp. 279–282, 2000.

[Tod01a] T. Toda, H. Saruwatari, and K. Shikano, "High quality voice conversion based on Gaussian Mixture Model with Dynamic Frequency Warping", *European Conference on Speech Communication and Technology*, 2001.

[Tod01b] T. Toda, H. Saruwatari, and K. Shikano, "Voice conversion algorithm based on Gaussian Mixture Model with Dynamic Frequency Warping of STRAIGHT spectrum", *International Conference on Acoustics, Speech, and Signal Processing*, pp. 841–844, 2001.

[Tur02] O. Turk, and L.M. Arslan, "Subband based voice conversion", *International Conference on Spoken Language Processing*, Bogazici University, Istanbul, 2002.

[Tur03] O. Turk, and L.M. Arslan, "Voice conversion methods for vocal tract and pitch contour modification", *European Conference on Speech Communication and Technology*, pp. 2845–2848, 2003.

[Val92] H. Valbret, E. Moulines, and J.P. Tubach, "Voice transformation unsing PSOLA technique", *Speech Communication*, vol. 11, pp. 175–187, 1992.

[Ver96] W. Verhelst, and J. Mertens, "Voice conversion using partitions of spectral feature space", *International Conference on Acoustics, Speech, and Signal Processing*, pp. 365–368, 1996.

[Wat02] T. Watanabe, T. Murakami, M. Namba, T. Hoya, and Y. Ishida, "Transformation of Spectral Envelope for Voice Conversion Based on Radial Basis Function Networks", *International Conference on Spoken Language Processing*, 2002.

[Xu96] Lei Xu, and Michael I. Jordan, "On convergence Properties of the EM Algorithm for Gaussian Mixtures", *Neural Computation*, pp. 129–151, 1996.

[Ye03] H. Ye, and S. Young, "Perceptually weighted linear transformations for voice conversion", *European Conference on Speech Communication and Technology*, pp. 2409–2412, 2003.

[Ye04] Hui Ye, and Steve Young, "High quality voice morphing", *International Conference on Acoustics, Speech, and Signal Processing*, 2004.