TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY
Julkaisu 671 • Publication 671

Marko Moberg

Contributions to Multilingual Low-Footprint TTS System for Hand-Held Devices

Tampere 2007

Tampereen teknillinen yliopisto. Julkaisu 671
Tampere University of Technology. Publication 671

Marko Moberg

Contributions to Multilingual Low-Footprint TTS System for Hand-Held Devices

Thesis for the degree of Doctor of Technology to be presented with due permission for public examination and criticism in Tietotalo Building, Auditorium TB109, at Tampere University of Technology, on the 17th of August 2007, at 12 noon.

Tampereen teknillinen yliopisto - Tampere University of Technology
Tampere 2007

ISBN 978-952-15-1798-3 (printed)
ISBN 978-952-15-1813-3 (PDF)
ISSN 1459-2045

Abstract

Speech technology in the form of automatic speech recognition (ASR) and speech synthesis (text-to-speech or TTS) has become common in everyday use. Applications such as in-car navigation, hands-free control of devices, aids for visually impaired people, telephone-based schedule and reservation services, military applications and even some dictation applications can be found on the market today. The advancement of technology has made it possible to provide voice-based applications also on smaller, hand-held devices.

Text-to-speech as an area of technology is quite interdisciplinary. It requires text processing, language specific knowledge and mimicking of human speech production. Despite the differences in implementation details, all TTS systems share the same basic structure. The "front end" of the system carries out natural language processing creating intonation and pronunciation. The "back end" provides synthesized waveform based on the information provided by the front end. The linguistic knowledge is necessary in providing text normalization, syntactic analysis, and modeling of pronunciation and prosody. On the other hand the characteristics of human speech production are utilized in the waveform generation stage. The practical implementation of a TTS system also requires software engineering skills.

This thesis focuses on describing the challenges and solutions in optimizing a multilingual text-to-speech (TTS) system for hand-held devices. The challenges in development are introduced by the mismatch between the application requirements and available resources. The requirements include application features, speech quality, language support and portability. The main resources are memory size, performance, development time and cost. For example in most cases, more features, more languages and higher speech quality require more memory. The trade-off between requirements and resources are especially challenging in cost-optimized embedded devices targeted for consumer market. For this reason none of the available TTS solutions fully meet the requirements of the applications on hand-held devices.

In this thesis, a multilingual TTS framework is designed and optimized to meet the application requirements according to the availability of various resources. The TTS system uses a Klatt88 formant synthesizer or unit selection synthesis depending on the configuration. Novel approaches and improvements are applied to text normalization, text-to-phoneme conversion, control of synthesis parameters, system framework, and development tools.

A memory efficient, multilingual rule framework is developed for number and abbreviation expansion. This text-normalization task is aimed to provide voice information during physical exercise (speed, duration, energy consumption etc.). The framework supports gender-based inflections in numerals as well as number agreement in units. The same system is also expandable for other purposes. The functionality of the system is verified by implementing the number, unit and time rules for 42 languages.

Enhancements are also made at the phonetic level. After applying the text-to-phoneme conversion, the phonetic representation of the utterance is modified. The described methods can be used to reduce memory consumption and rapidly provide larger language coverage. Memory consumption of US English formant synthesis parameters was reduced by 7% through decreasing the number of phones needed to present the sounds of a given language. The rapid expansion of the language coverage is done through cross-lingual phoneme mapping where several sounds are approximated with phones that are already available in the system. The support for new languages is quickly obtained with the cost of approximated phonetics and foreign intonation. However, for similar languages and for certain applications the described approach is adequate.

Some ways to improve the handling of the synthesis parameters are also investigated. It is shown that a fairly simple intonation model can be successfully used when short utterances such as names or other isolated words are synthesized. The synthetic intonation using either Classification And Regressions Trees (CART) or Fujisaki model is confused as comparable with natural intonation in listening evaluation tests. In addition to fundamental frequency F0, other formant parameters and their control mechanisms are studied. It is shown that it is adequate to model formant contours using only two dynamically placed control points for each phoneme and fill in the missing points with linear interpolation.

In addition to functional enhancements the system framework and development methods are improved. A script language is developed for expressing language dependent synthesis rules in a memory efficient way. The performance of the script was tested with 18 synthesis languages. The average memory saving was 69% compared with the dynamically linked library (DLL)-based implementation. The use of abstract script language to control synthesis parameters separates synthesis language development from actual programming. The software engineering and the linguistic tasks are further separated by having a special tool for creating and testing language rules. A Java-based platform independent integrated development environment allows editing and binary conversion of the rule files.

It is shown that commercially viable multilingual TTS-based applications can be created by the following four main methods. First, the limitations of the TTS technology can be hidden or alleviated by limiting the scope of the application. Second, optimization in memory consumption and performance makes the TTS technology more attractive for hand-held devices. Third, multilingualism and rapid development of new synthesis languages are enabled through system design and proper development methods and tools. Finally, the separated TTS engine software and language dependent data make it possible to hide the software engineering details by providing language developers an interface with a higher level of abstractness.

Preface

The research presented in this thesis has been carried out at Nokia Research Center and at Nokia Technology Platforms in Tampere, Finland in 2002-2007. In addition to the scientific aspects the research activities have been strongly driven by requirements of product and application development.

First, I would like to express my gratitude to my thesis advisor and Head of Institute of Signal Processing at Tampere University of Technology, Prof. Moncef Gabbouj for his continuing support, patience and guidance. I would also like to thank Dr. Sue Hertz and Dr. Martti Vainio for providing detailed and constructive advice in finalizing the work. Furthermore, I would like to thank my colleague Dr. Kimmo Pärssinen for his support and valuable contributions to our research and for making the work enjoyable.

Special thanks to my wife Katariina for making this possible. And thanks to my lovely daughters Laura and Noora for reminding me what really matters in life. Finally, I would also like to thank my parents for their unconditional support in all of my projects including post-graduate studies.

Tampere, June 2007

Marko Moberg

Contents

List of Publications

This thesis is written on the basis of the following publications. In the text, these publications are referred to as Publications [P1], [P2], …, [P9].

[P1] **M. Moberg** and O. Viikki, "Optimizing Speech Synthesizer Memory Footprint through Phoneme Set Reduction", *Proceedings of IEEE-SP Workshop on Speech Synthesis 2002*, Santa Monica, CA, 11-13 September, 2002, pp. 171-174.

[P2] **M. Moberg**, K. Pärssinen and J. Iso-Sipilä, "Cross-Lingual Phoneme Mapping for Multilingual Synthesis Systems", *Proceedings of ICSLP 2004*, Jeju, Korea, 4-8 October, 2004, pp. 1029-1032.

[P3] **M. Moberg** and K. Pärssinen, "Comparing CART and Fujisaki Intonation Models for Synthesis of US-English Names", *Proceedings of Speech Prosody 2004*, Nara, Japan, 23-26 March, 2004, pp. 439-442.

[P4] K. Pärssinen and **M. Moberg**, "Evaluation of Perceptual Quality of Control Point Reduction in Rule Based Synthesis", *Proceeding of ICSLP 2006*, Pittsburgh, PA, USA, 17-21 September, 2006, pp. 2070-2073.

[P5] K. Pärssinen and **M. Moberg**, "Multilingual Data Configurable Text-to-Speech System for Embedded Devices", *Proceedings of Multiling 2006*, Stellenbosch, South Africa, 9-11 April, 2006.

[P6] **M. Moberg** and K. Pärssinen, "Integrated Development Environment for a Multilingual Data Configurable Synthesis System", *Proceedings of Specom 2005*, Patras, Greece, 17-19 October, 2005, pp. 155-158.

[P7] **M. Moberg** and K. Pärssinen, "Multilingual Rule-Based Approach to Number Expansion: Framework, Extensions and Application", *International Journal of Speech Technology*, Published on-line 22 September 2006 (Online First^{TM}).

[P8] J. Iso-Sipilä, O. Viikki and **M. Moberg**, "Multi-Lingual Speaker-Independent Voice User Interface for Mobile Devices", *Proceedings of ICASSP 2006*, vol. 1, Toulouse, France, 15-19 May, 2006, pp. 1081-1084.

[P9] K. Pärssinen, **M. Moberg** and M. Gabbouj, "Reading text messages using a text-to-speech system in Nokia Series 60 mobile phones: Usability study and application," *Technical report, Tampere University of Technology, Report (3):2006*, Tampere, Finland.

List of Supplementary Publications

The following publications are related to the topic but are not included in this thesis. In the text, these publications are referred to as Publications [S1] and [S2].

[S1] K. Pärssinen, **M. Moberg**, M. Harju, and O. Viikki, "Development Challenges of a Text-to-Speech System for Multiple Languages", *Internationalizing W3C's Speech Synthesis Markup Language Workshop II*, 30-31 May 2006, Heraklion, Crete.

[S2] **M. Moberg** and K. Pärssinen, "Using Text-to-Speech in Mobile Phones", Fonetiikan päivät 2006 - The Phonetics Symposium 2006, Reijo Aulanko, Leena Wahlberg & Martti Vainio (Eds.), Publications of the Department of Speech Sciences, University of Helsinki, 53, Helsinki, Finland, 30-31 August, 2006, pp. 125-133.

List of Acronyms

API	Application Programming Interface
CART	Classification And Regression Tree
CPU	Central Processing Unit
DLL	Dynamically Linked Library
G2P	Grapheme-to-Phoneme
HMM	Hidden Markov Model
HW	Hardware
IPA	International Phonetic Alphabet
IDE	Integrated Development Environment
IS	Information Structure
LPC	Linear Predictive Coding
LTS	Letter-To-Sound
PC	Personal Computer
POS	Part Of Speech
SAMPA	Speech Assessment Methods Phonetic Alphabet
SIND	Speaker Independent Name Dialler
SMS	Short Messaging Service
SSML	Speech Synthesis Mark-up Language
SW	Software
TTP	Text-To-Phoneme
TTS	Text-To-Speech
UI	User Interface
VOT	Voice Onset Time
WFST	Weighted Finite State Transducer

List of Tables

List of Figures

Chapter 1

Introduction

THE technological advancement has gradually taken voice user interfaces from sci-fi television shows and books into practical everyday use. For example, speech recognition and speech synthesis are nowadays used in cars (navigation, hands-free control), mobile phones (call initiation, voice control), aids for visually impaired, military applications and even some dictation applications. User interfaces with synthetic speech are normally used in applications or situations where the user has no other convenient means of getting the feedback from the device. Synthetic speech may also augment the existing mode of interaction by offering a faster and an alternative or a parallel way to get information from the system.

1.1. BACKGROUND AND MOTIVATION

Synthetic speech has been studied since the 1770's when Russian professor Christian Kratzenstein managed to produce vowel sounds with some resonant tubes. The famous acoustic-mechanical speech machine by Wolfgang von Kempelen was created in 1791 [94]. It successfully modeled human speech production with bellows as lungs, vibrating reed as vocal cords and leather tube as vocal tract. The first elementary electrical synthesizers were made in the 1920's and 1930's. The first formant synthesizers emerged in the 1950's and the first text-to-speech (TTS) systems in the 1960's and 1970's [2]. The first computer implementations were made in the 1980's.

Apart from research uses, synthetic speech is nowadays commonly used for example in toys, train announcements, telephone services or call centers, car navigation systems and mobile phones. The synthesis technologies may vary from simple speech sample concatenation to parametric synthesizers or full text-to-speech systems with unit selection synthesis. The synthesis technology as well as the selection of the entire TTS system depends on the target application and its requirements. TTS systems running on a personal computer (PC) platform can consume several tens of megabytes of memory without being too large. However, the same memory footprint is hardly acceptable on mass-produced mobile devices whose memory size has a significant impact on the manufacturing cost.

Despite the vast offering of commercial and non-commercial TTS systems none of them were directly suitable for the specific use for mobile phones. Instead of a full TTS system capable of processing long, arbitrary sentences there was a need for a multilingual, very low-footprint synthesis system for short utterances such as names and isolated words. The main driver for this type of a TTS system was a speaker independent name dialing (SIND) application. In SIND, the user says the name of the phonebook entry and the TTS speaks out the recognition results to the user before initiating a phone call. Another example of an application requiring a short utterance TTS is a speaking dictionary.

1.2. OBJECTIVE OF THE THESIS

This thesis describes the main requirements of a multilingual low-footprint TTS system in hand-held devices and describes some novel techniques which can be applied to meet them. The requirements, which are addressed here, are rapid development of languages, support for multiple languages, easy maintenance and configurability, low memory consumption, adequate speech quality and portability to different implementation platforms. The main focus is on low memory consumption and multilingual aspects. The speech quality is considered adequate when it is acceptable for commercial products. The quality level is determined by application requirements and user perception.

1.3. OUTLINE OF THE THESIS

This thesis contains two main parts. The first part is an introductory part and the second is a collection of related publications presented in scientific journals and conferences. The first part explains the main concepts and terms and gives necessary background information so that the reader can understand the details and contributions described in the publications.

The introductory part has six chapters. The second chapter following this introduction describes the fundamentals of human speech production and perception. The information is needed to better understand the source-filter speech synthesis models. An example of such a model is a formant synthesizer, which is used and referred to in most of the presented publications. The third chapter briefly presents the most relevant areas of linguistics. It will be shown that engineering knowledge is not enough in development of voice user interfaces, especially speech synthesis systems. The fourth chapter gives an outline of a generic TTS system. Functionality and expertise needed in development of each module is described. The fifth chapter presents the special requirements of the multilingual TTS systems on mobile devices. Several alternative methods are listed to minimize the memory consumption and to allow the fast development of languages. The introductory part is complemented with the conclusions in chapter 6.

1.4. SUMMARY OF PUBLICATIONS

The publications presented in this thesis were written during a period of four years, from 2002 to 2006. They all concentrate on the issues of multilingual text-to-speech synthesis in mobile phones. However, most of the results are also applicable to any TTS system on hand-held devices or even to any TTS system in general. The main topics in the publications are (i) the techniques for improving formant synthesis in multilingual TTS systems and (ii) the description of the TTS framework and language development. It should be noted that the term *phoneme* has been used in the publications to indicate both phonemes and phones of a language.

Publications [P1][75] and [P2][74] describe some methods for manipulating phonetic representation of an utterance to offer some memory reduction and simplification in a TTS system. Both publications also measure the impact of manipulation on perceived speech quality. Publication [P1][75] describes a method for reducing the language-specific set of sounds by forming diphthongs and affricates by combining other sounds. Listening evaluation tests were carried out to measure differences in quality and intelligibility between the original and artificially generated compound-sounds. The results show that some of the sounds in US English could be successfully replaced with compound-sounds without a significant loss in perceived quality. Another simplification of the language-specific phoneme set is described in [P2][74]. It presents a method where the pronunciation in one language can be approximated with the sounds of another language. It is shown that the method can be successfully applied as long as the two languages resemble each other phonetically. It allows the fast addition of new languages into the synthesis system without the cost of significantly increased memory consumption. The drawback of the method is the inaccurate pronunciation, intonation (foreign accent), and possibly degraded intelligibility of synthetic speech.

Publications [P3][70] and [P4][83] concentrate on formant synthesis as a part of a TTS system. Publication [P3][70] compares two different intonation models in the context of formant synthesis. The models are CART-based statistical model and a parameter based mathematical model called the Fujisaki model. In the statistical method the decision trees are trained with annotated material. The pitch (F0) contours, which are created during the synthesis phase, are determined by features such as syllable stress, syllable location and so on. The Fujisaki model is controlled mainly with the location and amplitude of accent and phrase commands. The locations are determined for example using information about sentence boundaries and stressed words or syllables. The comparison of the two methods shows that there is no significant difference between CART and Fujisaki intonation models when they are applied to short US English utterances.

Different interpolation schemes for formant parameters are investigated in [P4][83]. The interpolation takes place between the so called control points. The publication measures the changes in perceptual quality as the number of control points per phoneme is reduced. It is shown that there is no perceptual difference in two or four control point models provided that the location of the points is sensibly selected. The reduced number of points simplifies the interpolation process. The result is also relevant for human speech perception and is aligned with the earlier research [43].

The other group of publications describes the TTS framework, application related topics and language development. The separation of program code and language specific data is one of the most important ways to allow the implementation of a multilingual TTS system and to enable rapid language development. The framework that utilizes the code and data separation is presented in [P5][82]. The rule-based system takes a phoneme sequence as input, creates control parameters for formant synthesis, and produces synthetic speech waveform as output. The rules that control the synthesis parameters and parameter transitions are stored in language specific packages. The synthesis engine itself stays the same for all the languages used. It is shown that the script language which is used in implementing the rules consume significantly less memory than the corresponding implementation as program code (dynamically linked library, DLL). The creation of the language packages can be made easier and faster using the integrated development environment described in [P6][71]. The environment allows manipulation of the rule files and testing of the resulting output waveform. Various debugging features including visualization of the output waveform support the iterative language development process.

The front end of a TTS system has also several language specific functionalities. The specific challenges in text normalization are presented in [P7][72]. The publication introduces a rule framework for creating language specific normalization rules for numbers, units and time expressions. It also presents the implementation of rules for over 40 languages.

The application oriented viewpoints of the multilingual TTS system are given in [P8][54], [P9][85], [S1][84] and [S2][73]. The first describes the speaker independent name dialing (SIND) system where a low-footprint TTS is used in providing feedback on recognition results to the user. The second application oriented publication [P9][85] describes the message reader application on Nokia Series 60 mobile phones and presents the results of the immediate usability tests. Supplementary publications provide a high-level overview of the challenges of multilingual TTS development and TTS applications on mobile phones.

1.5. AUTHOR'S CONTRIBUTION TO PUBLICATIONS

The work described in the publications involved mainly research, innovation, specification, simulation, implementation, evaluation and documentation. The author's contribution to each publication is described next.

In Publication [P1][75], the author was responsible for the modification of the synthesis framework and for reducing the total number of separate US English sounds by constructing them using other sounds of that language. The author also generated the test data, conducted the listening evaluation tests, analyzed the results, wrote and presented the publication.

In Publication [P2][74], several phonetic mappings between languages were carried out and evaluated. The author was responsible for finding a perceptually optimal mapping from one phonetic representation to the other. This procedure was repeated for several language pairs such as Hungarian and Finnish, Estonian and Finnish, Dutch and German, and Greek and Spanish. The author specified and conducted a listening test to evaluate the quality of the phonetic mapping, and co-wrote and presented the publication.

The author co-wrote Publication [P3][70] and was intensively involved with the design of the work presented. The author contributed in tuning synthesis parameters for the formant synthesizer and in training the classification and regressions trees (CART) for prediction of prosody. The author also produced the test data and conducted the listening tests to enable the objective comparison of different intonation models.

In publication [P4][83], the author was involved in specifying the comparison of different interpolation strategies and choosing the optimal control points for formant tracks. The author also co-wrote the paper and conducted the listening evaluation tests for assessing the perceptual difference between the different numbers of control points in formant track interpolation.

The author's contribution to publication [P5][82] was to develop synthesis rules for several languages including US English, Finnish, French, Swedish, Russian and Czech using the presented script format. The rules controlled the formant synthesis parameters based on the phonetic and prosodic information such as manner, place and syllable stress. The author also co-wrote the paper and carried out analysis and evaluations for developed synthesis languages.

In publication [P6][71], the author took part in the design and specification of the integrated development environment and evaluated the functionality of the implemented system. The author also co-wrote and presented the paper.

The author was the originator of the idea for text normalization utilized in publication [P7][72]. The author also implemented the number, unit and time expansion rules for several languages using the framework. The work also included supervising the rule development for other languages, implementation of required Microsoft Excel tools and evaluation of the results presented in the publication. The author co-wrote the paper.

In publication [P8][54], the author co-wrote the paper giving some insight to the speech synthesis part of the voice user interface system.

Publication [P9][85] was also co-written by the author. The author also took part in organizing the immediate usability tests and analyzing the test results.

Chapter 2

Fundamentals of human speech production and perception

UNDERSTANDING of human speech production and perception is necessary in the development of speech synthesis systems. Articulatory synthesis and synthesis using source-filter models are closely related to the actual human speech production mechanisms. Unit selection synthesis and other forms of concatenative synthesizers do not directly model the physics of the human speech production system. However, they benefit from the classification of the speech sounds based on human articulation and from the knowledge of the speech perception in the same way as the other synthesis techniques.

2.1. SPEECH ORGANS

The human speech organs can be divided roughly into three groups: 1) organs which reside above the larynx (supraglottal organs), 2) the larynx and the laryngeal cartilages, and 3) respiratory organs such as trachea, lungs and respiratory muscles [4]. A cross-section of a human head with the main organs involved in speech production is shown in Figure 1.

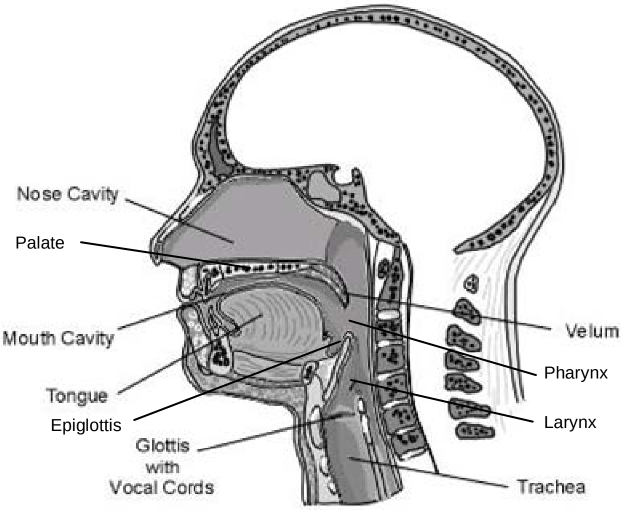

Figure 1. Overview of the human speech organs.

The supraglottal organs are mainly the articulatory organs including the lips, the teeth, the tongue, the alveolar ridge, the palate, and the velum. There are also three cavities used as resonating chambers is speech production. They are the oral cavity, the nasal cavity, and the pharynx. The larynx is a complex cartilaginous structure containing the vocal folds and glottis. It is located on the upper part of the trachea beneath the pharynx. The epiglottis prevents the intrusion of food or liquid into the lungs during swallowing. The trachea descends from the larynx into the chest cavity. It branches into two primary bronchi each leading to a separate lung. The airflow needed in speech production and breathing is generated with the muscles of respiration such as the diaphragm and the muscles controlling the rib cage (intercostals). The external intercostals elevate the sternum for inhalation whereas the internal intercostals lower the sternum during exhalation.

2.2. STAGES OF SPEECH PRODUCTION

The human speech production may be viewed as having three separate stages. The first stage provides air pressure for example from the lungs and is called *initiation*. The second stage is *phonation* where for example fundamental frequency is generated through vocal fold vibrations. The final stage is *articulation* where the generated signal is modulated to produce various speech sounds [4].

2.2.1. Initiation

The air flow needed in speech is generated during the initiation stage. There are three air stream mechanism locations used for speech production. They are the lungs (pulmonic), the larynx (glottalic), and the mouth (velaric). Most speech in natural languages involves a pulmonic egressive airstream where the air flows out of the lungs. At first the pressure is built up in the lungs and trachea below the glottis. This subglottal pressure is released when the glottis is opened and the air starts flowing from the lungs and out of the mouth or nose. Respiratory muscles control the airflow. The pulmonic egressive air stream is very similar to normal breathing and compatible with the respiratory cycle so it is well suited for speech.

It is also possible to create speech sounds with other mechanisms or with a different direction of air stream (e.g. pulmonic ingressive). Sounds classified as ejectives and implosives are initiated with glottalic egressive and ingressive streams, respectively. The glottalic air streams are generated by rising and lowering of the larynx. In some languages velaric airstreams are also used to generate click sounds. These velaric streams are generated in the mouth through actions of the tongue.

2.2.2. Phonation

Phonation takes place as the air passes through the glottis within the larynx. The air flow is modified in different ways depending on the type of phonation used. Different types are characterized by the mode of vocal fold vibrations. The basic types are voiceless phonation, voiced phonation (sometimes the term phonation is used to refer only to this mode) and whisper [4].

Voiced phonation may take place in different forms such as falsetto, creaky voice or breathy voice. Vocal fold vibrations are produced by a repeating glottal cycle. First, air pressure is built behind the closed vocal folds. As the pressure increases it forces the vocal folds open and the air starts flowing through the glottis. Due to Bernoulli effect the air pressure is reduced at the constriction and the vocal folds begin to close [68]. After the glottis is fully closed the air pressure starts building up and the cycle is repeating.

Whisper is produced without vocal fold vibrations. A small triangular shape passage is left in the glottis so that turbulent air flow (noise) is generated. In voiceless phonation there is no sound generated in the glottis, in other words neither periodic signal nor noise as in whisper is produced but instead the air is passed straight through the open vocal folds. Glottal stop is also usually mentioned as one of the modes although it is not really a phonation type. Glottal stop closes the vocal folds completely to stop the air flow and build up a pressure in subglottal space. Glottal stops are distinctive in Arabic language but also used in other languages, for example in a context where a word final and following word initial sound are the same.

2.2.3. Articulation

Articulation is the final stage in speech production. The periodical glottal pulses or noise generated during phonation are modified by the supraglottal organs. The posture and the positioning of the organs differentiate speech sounds. For example, the vowel sound in "sheep" is produced with the tongue positioned in the front part of the mouth whereas in "talk" the vowel is formed in the back of the mouth. In the case of voiceless sounds, the place of constriction and the channel shape determine the sounds. For example, [f] as in "fit" is produced by creating a constriction of the airflow with the lips and teeth. When the tongue is moved backwards and upwards to form a narrow air channel between the tongue and the alveolar ridge another fricative sound such as [ʃ] in "shout" is produced. Classification of the speech sounds is also based on the location of the articulation as will be shown in section 3.4.1.

2.3. MODELING SPEECH PRODUCTION

The configuration of the lips, the tongue, the velum and other articulatory organs contribute to the overall vocal tract shape during speech production. The natural resonant frequencies of the vocal tract are called formants and are often denoted with F1, F2, F3 and so on. The presence of formants can be clearly observed in voiced parts of speech as dark lines shown in the spectrogram in Figure 2. The vertical axis presents the frequencies from 0 to 4 kHz and the horizontal axis time from 0 to 3.7 seconds.

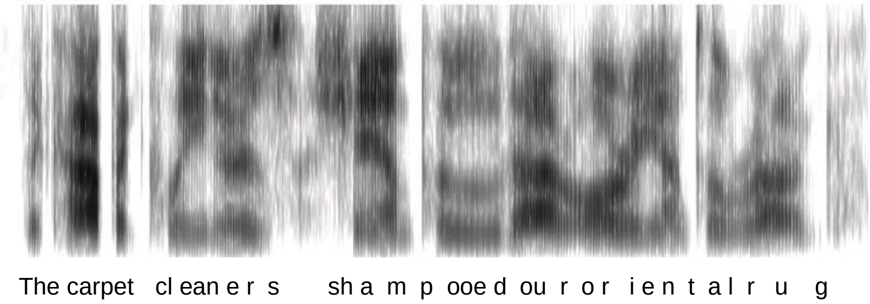

Figure 2. Spectrogram of the utterance "The carpet cleaners shampooed our oriental rug".

Some mathematical models of speech production have been developed for better understanding of human speech production [31]. Based on the analysis of resonant tubes and sound pressure propagation several formulas can be written for describing the speech production phenomenon. The models can be further used in speech synthesis.

The overall frequency response of the human speech signal or in other words the Fourier transform of the sound pressure $p_r(f)$ at distance r from the lips can be acquired by multiplying the transfer functions of the glottal source S(f), vocal tract model T(f) and the radiation characteristics R(f) as shown in (1) [97]. The radiation characteristics describe the behavior of the speech signal as it exits the mouth.

$$p_r(f) = S(f)T(f)R(f) \tag{1}$$

The glottal waveform of the excitation is related to the sound pressure reaching the ears of the listener. The pressure is closely approximated with the derivative of the volume velocity of the glottal waveform as presented by Stevens [97]. An example of the frequency response of the glottal pulse is shown in Figure 3. A detailed analysis of two common glottal models (LF and KLGLOTT88) is provided in [26]. The vocal tract transfer function T(s) can be modeled as

$$T(s) = K\frac{B(s)}{A(s)} = K\frac{(s-s_a)(s-s_b)(s-s_c)\ldots}{(s-s_1)(s-s_2)(s-s_3)\ldots} \tag{2}$$

where K is a constant, s_a, s_b, $s_{c,\ \ldots;}$ s_1, s_2, s_3, … are zeros and poles of T(s), respectively, and s=j2πf is a complex frequency variable. The poles represent the complex natural frequencies of the vocal tract, in other words formants.

The transfer function of the radiation characteristics resembles that of a high pass filter with a relatively wide transition band (slope about 6 dB per octave). Low frequencies are attenuated whereas higher frequencies are slightly amplified. The equations for radiation characteristics are also presented in [97]. All three components of the spectrum of the output sound pressure $p_r(f)$ are shown in Figure 3.

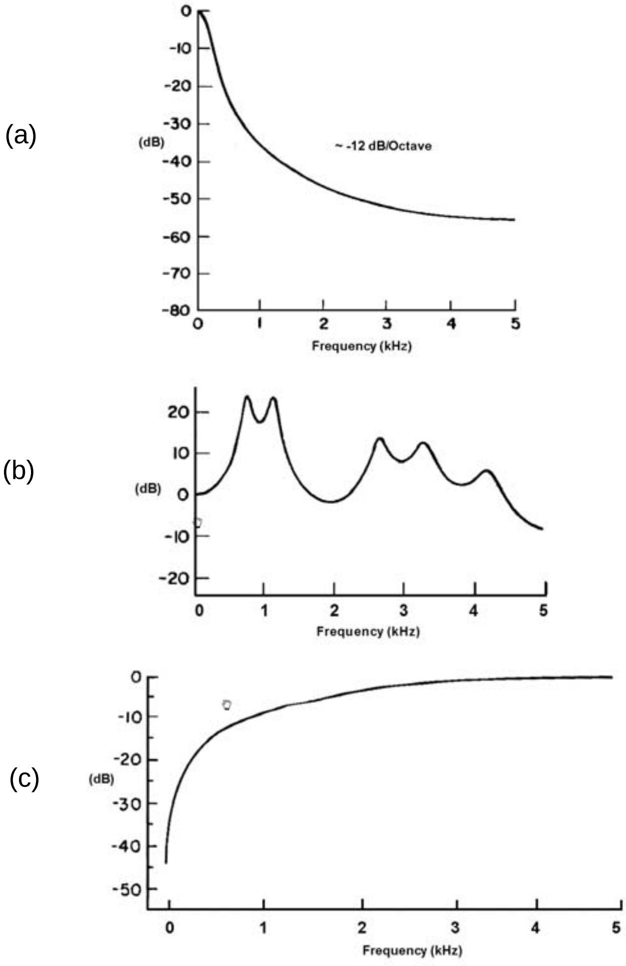

Figure 3. Three components of the overall transfer function of a speech signal: (a) glottal source, (b) vocal tract, and (c) radiation characteristics [58].

In some models such as Klatt88 [58], the effect of radiation characteristics has been incorporated into the glottal source. For this reason, the source creates a low-pass spectrum with -6 dB/octave fall-off instead of -12 dB/octave. The vocal tract spectrum shown in Figure 3 (b) is created when the model produces the US English vowel /A/ as denoted with Speech Assessment Methods Phonetic Alphabet (SAMPA) notation [89].

2.3.1. Source-filter synthesis

The theory of speech production can be directly applied to articulatory synthesis and to synthesizers which are based on a so called source-filter approach [30][31][32]. Articulatory synthesis uses models of human physiology and movement of organs to create the speech signal. Although the systems mimic the human speech organs they involve heavy calculation and are often too complex for practical applications.

The source-filter synthesizers model the excitation signal (source) and vocal tract (filter) to create synthetic speech. Examples of such synthesizers are Klatt formant synthesizer and linear predictive coding (LPC) synthesis. The Klatt formant synthesizer is described in more detail in section 4.8.1.

2.4. SPEECH PERCEPTION

The development of speech synthesis and TTS systems benefits a great deal not only from understanding of human speech production but also from the characteristics of speech perception. The changes in sound pressure emitted from a speaker's mouth are converted into mechanical movements of the tympanic membrane at the end of the ear canal. The displacement of the membrane causes a displacement of ear bones (malleus, incus, stapes) located in the middle ear. The mechanical excitation of the bones in the middle ear propagates into the fluids in the inner ear. The cochlea in the inner ear transforms the mechanical energy into electrical activity in the auditory nerve and further into the central nervous system [4][97].

The human auditory organs do not treat all frequencies equally. Also, the behavior of the brain affects the hearing experience. It is important to understand the capabilities of the human auditory system to be able to place the development effort into the right parts of the TTS system. Enhancements which do not have perceptual relevance can be left out whereas those which affect the acoustic cues of speech recognition are essential.

2.4.1. Perception of loudness and pitch

It is known that the human ear is not equally sensitive to all frequencies. The maximum sensitivity lies at 3000 Hz although the approximate frequency range of hearing is from 20 Hz to 20 kHz. The normal speech signal uses frequencies between 125 Hz and 8000 Hz but the intelligibility is preserved even if the frequencies above 4000 Hz are removed. The human hearing does not provide linear responses to sound level or pitch changes when the frequency is modified. Instead two different frequency sounds may sound equally loud although their amplitudes were different [4]. It has also been shown that human hearing is especially sensitive to changes in frequencies, levels and bandwidths associated with the spectral location of formants [97]. Another interesting effect is masking. The presence of one signal has an effect on perception of another signal that is in its temporal or spectral vicinity. It is also possible that a masking signal makes the other signal completely inaudible. The masking effect can occur simultaneously or it may have an impact on preceding or succeeding sounds [97].

The relative amplitudes of spectral prominences determine how the signal is heard. If for example, there are two spectral peaks close to each other in frequency, a human listener hears only a single frequency sound. The perceived frequency is somewhere between the two prominent peaks. The location depends on the relative sound levels so that it is closer to the stronger component. However, if the peaks are further apart, the listener perceives a single frequency which equals to one of the prominent peaks. In other words, no frequency blending occurs when the frequency components are far from each other [21]. These observations are relevant to the classification of vowels in human auditory system [97].

2.4.2. Acoustic cues

The most important acoustic cues in speech perception are the ones which enable listeners to separate speech sounds from one another. The artificial re-production of the cues is an important task in parametric speech synthesis. A better realization of the cues implies a better differentiation of speech sounds, thus providing higher intelligibility. Cues can be classified for example into phonation cues, manner cues, place cues, cues on segmental length and so on.

One of the differentiating factors in spoken consonants is the presence of voicing. For example, US English bilabial plosives [p] and [b] in word initial position have the same place of articulation but they differ in voicing. The plosive [b] is often referred as voiced sound whereas [p] is said to be voiceless. However, it can be shown that measuring of voicing itself is not adequate in differentiating abovementioned sounds. In some contexts like in words "bit" and "pit" both plosives can be measured to be voiceless [68]. The perceptual difference between the two sounds is not only due to voicing but also due to voice onset time (VOT). Voice onset time describes the time from the release after occlusion to the beginning of voicing. VOT is negative if voicing begins before the release and positive if it begins after the release. When voicing coincides with the release, VOT is said to be zero. In US English voiced plosives have small VOT whereas in voiceless plosives the VOT is larger. During the time from the release to the beginning of voicing aspiration is heard [4]. Thus voiceless plosives with large VOT are sometimes referred to as aspirated plosives.

The place of articulation is perceived through formant transitions [4]. The perceptually most important formant is F2 which means that special attention has to be paid in generating accurate F2 contours in speech synthesis. The manner of articulation is recognized for example through the presence of formants (laterals and approximants), length of the formant transitions, presence of noise (fricatives, affricates) or breaks in spectral content (plosives). The vowels are generally sufficiently defined with F1 and F2 values but for example rounding changes the distribution of formant energy compared with the unrounded vowel. The vowel length may also be a contrastive property in some languages such as in Danish.

In addition to the segmental cues there are some properties that extend over several speech segments. For example, in tonal languages (e.g. Mandarin Chinese, Cantonese, Thai) pitch or fundamental frequency is also an important cue. The pitch pattern or tone is contrastive and may change the meaning of the word. For example a one syllable word with a rising tone may have a different meaning than the same word with a falling tone. Even in non-tonal languages, speakers are able to provide additional information and better intelligibility with stress, intonation and timing. It is even possible to speak the message in such a way that the meaning is the opposite.

2.5. TERMINOLOGY

Many phenomena in speech production and perception can be studied from different points of view [64]. Table 1 provides a brief overview of related terms from physiological, acoustic, perceptual, and linguistic points of view.

Table 1. Essential terms in speech production and perception

Physiological	**Acoustic**	**Perceptual**	**Linguistic**
articulatory timing	signal in time domain	duration	quantity, tempo
phonation	fundamental frequency, F0	pitch	tone, intonation
generation of air pressure	intensity, amplitude	loudness	word and sentence level stress

For example, physiological activity called phonation produces acoustically measured fundamental frequency. The human ear perceives that as pitch. From the linguistic point of view the changes and patterns in fundamental frequency appear in tones and intonation. The same applies for linguistic functions of quantity and stress. The quantity is created in timing of articulatory sequences, which can be measured from the signal in the time domain. Humans perceive such a phenomenon as duration. The perception of loudness and stress relates to intensity and amplitude of the physical signal. The signal, on the other hand, is created with organs modifying the airflow and resulting speech spectrum. Some of the terms presented in the table are used interchangeably in this thesis.

Chapter 3

Linguistics

LINGUISTICS is a field of science that studies languages. Understanding similarities, differences and characteristics of different languages is important in providing natural language processing in multilingual text-to-speech systems. This chapter gives a brief introduction to those sub-fields of linguistics that play the most significant role in TTS system development. The purpose of the linguistic analysis in a TTS system is to gather necessary information for determining pronunciation and prosody.

3.1. MORPHOLOGY

Morphology is a field that studies of structure of words. For example, in languages such as English and French, words may be viewed as a combination of a word stem (or root) and an affix (suffix and prefix). The root part of the word may also be altered by inflexions. Morphological information may be used in determining the part of speech label for a word. For example in English, a word having a suffix "-ed" suggests that it might be a verb in the past tense. A minimal unit of meaning of grammatical function is called a *morpheme* [115]. The morphemes can be categorized into free morphemes (lexical or functional) or bound morphemes (derivational, inflectional). Depending on the focus, the field of study may be divided into inflectional morphology or derivational morphology [28]. Inflectional morphology covers words which are altered by, for example, a plural suffix such as in "cow" → "cows". The derivational morphology involves word formation through the use of more than one lexeme (words conventionally listed as separate entries in dictionary). For example, the relationship between the words "morpheme" and "morphology" is in the field of derivational morphology. An example of the morphological analysis of an English word "unintentionally" is shown in Figure 4. The abbreviations "Adj" and "Adv" denote "adjective" and "adverbial", respectively.

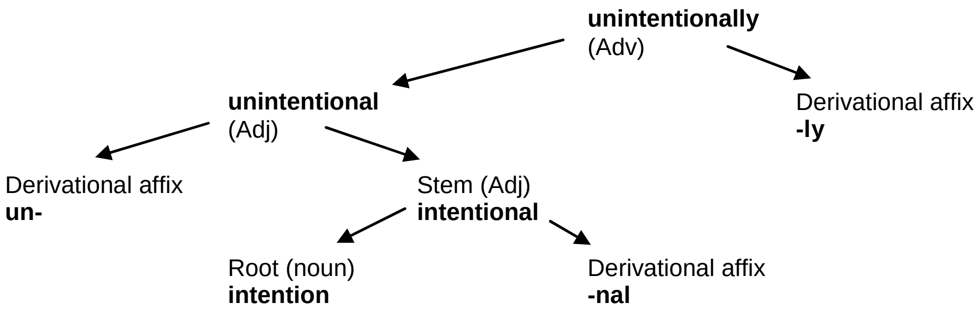

Figure 4. Morphological break-down of the word "unintentionally".

Such a tree-like structure can be formed for every word in a sentence in languages that follow the similar morphological partitioning. However, the context of the word should also be considered to accurately determine the morphemes.

Morphology can also be useful in representing all the words of a language in a dictionary. The size of the dictionary can be significantly reduced if morphological affixes and modifications are intelligently represented so that all possible forms of a single word do not have to be separately listed.

3.2. SYNTAX AND GRAMMAR

Syntax describes how phrases and sentences are constructed from words. The language specific grammar provides a template for correct structure of a sentence. The sentence structure is usually closely related to the way the sentence is spoken. It determines for example which words to emphasize, where to place pauses, and what kind of intonation is needed [68]. The grammatical description of a sentence may include part of speech information (noun, adjective, verb, adverb, preposition, pronoun, conjunction) and other traditional grammatical categories including number, person, tense, voice and gender [115]. The grammatical and syntactic analysis of a sentence results in a tree-like structure where each word and the corresponding categorization are presented. An example of a phrase-structure or syntactic tree is shown in Figure 5.

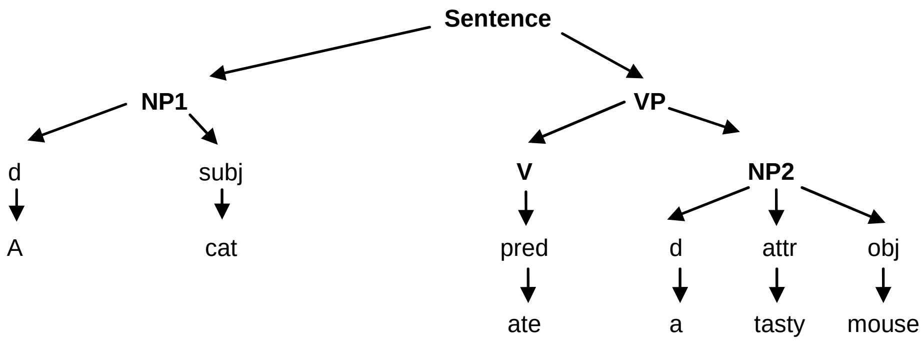

Figure 5. Syntactic tree structure of a sentence "A cat ate a tasty mouse".

The presented tree shows how the sentence is first divided into a noun phrase (NP1) and a verb phrase (VP). Then the verb phrase is further divided into a verb (V) and a noun phrase (NP2). Each phrase if further divided into words and their corresponding part of speech. The part of speech tags used in the figure are the following: d = determiner, subj = subject, pred = predicate, attr = attribute, obj =object.

3.3. SEMANTICS AND INFORMATION STRUCTURE

The term semantics is used to refer to the meaning of words, phrases and sentences. The study is focused on what the words conventionally mean rather than on what kind of message a speaker is trying to convey [115]. Semantics further restricts the possible structure and word selection in the sentences constructed according to syntax and grammars. For example, a sentence could be grammatically correct but quite implausible due to the selection of words with no apparent relation to each other. Semantics also has an impact on the actual realization of spoken utterances. The meaning conveyed via orthography may be contradicted by appropriate stressing of words and certain intonation patterns.

Information structure (IS) is an inherent aspect of meaning [61]. IS reflects the relation and interaction between the contextually dependent items in an utterance. According to Steedman [96] there are two dimensions in an IS; *theme* and *rheme*. Theme is a part of a sentence that a speaker believes to be a matter of mutual interest. Rheme is a part of a sentence that comments the theme or provides some information about the theme. There is a strong correspondence between the IS and intonation structure at least in English which means that a proper construction of IS can provide useful information for modeling intonation in a TTS system.

3.4. PHONETICS AND PHONOLOGY

Phonetics is the science of speech. It studies different speech sounds, speech production mechanisms and also speech perception and acoustics across all languages [4]. Functional, contrastive role of sounds is studied by phonology as stated by Kirchhoff [56]. The most essential areas of phonetics for speech synthesis are classification of speech sounds and suprasegmental properties of speech including coarticulation. For source-filter type synthesis, acoustic phonetics provides useful information.

There are two different notations for speech sounds used in this thesis. Phonemes which are differentiable units for a speaker of a given language are written between slash characters "/ /". Phones or phoneme realizations which cover all variants of each phoneme are marked with brackets "[ ]".

3.4.1. Classification of speech sounds

The standardized notation for speech sounds or segments is provided in International Phonetic Alphabet (IPA) [53]. The phonetic symbols in IPA are language-independent, thus they can be used in describing the pronunciation in every language. There are also several other notations for speech sounds. For example, SAMPA [89] notation offers a way to present IPA sounds in American Standard Code for Information Interchange (ASCII) characters. However, SAMPA notation does not provide correct mappings for all the languages. Due to lack of common notation to present IPA symbols on computer there are quite a few different methods to annotate databases especially in languages which are not covered by SAMPA.

The speech sounds are differentiated from each other using several parameters. The classification could be done using the air-stream mechanism, force of articulation, prolongability, state of the velum, manner and place of articulation, and so on. In practice, the so called three-term labels are often used. It is sufficient to use only information about phonation, place of articulation and manner of articulation to describe obstructed speech sounds as shown in Table 2. Most of the other parameters can be derived from these three attributes.

Table 2. Classification of speech sounds based on consonant-type articulation.

Parameter	**Example values**
1. Phonation	voiceless, voiced, whisper
2. Manner of articulation	stop, fricative, affricate, approximant, vowel
3. Place of articulation	bilabial, labio-dental, dental, alveolar, post-alveolar, alveo-palatal, retroflex, palatal, velar, uvular, pharyngeal, glottal

A similar classification is also available for vowel-type sounds. Instead of using a full list of parameters another set of three terms is defined. The vowel sounds are characterized by tongue position (height and front/back location) and the lip shape as shown in Table 3. The other parameters such as air-stream mechanism and state of the velum can be derived from the three-term labels.

Table 3. Classification of speech sounds based on vowel-type articulation.

Parameter	Example values
1. Tongue height	close (high), close-mid, open-mid, open (low)
2. Tongue position (anterior-posterior)	front, central, back
3. Lip shape	rounded, unrounded

The description of each speech sound can be mapped to IPA symbols. Some examples of the symbols and their three-term descriptions are presented here:

[ʃ]	voiceless post-alveolar fricative as in "shoot",
[b]	voiced bilabial plosive as in "bat",
[ð]	voiced dental fricative as in "this",
[ʌ]	unrounded open-mid back vowel as in "cut".

3.4.2. Suprasegmentals

In addition to the segmental characteristics, human speech has several features which extend over multiple speech segments. These suprasegmental features are often referred to as prosodic components of speech including stress, intonation, timing, and tone. The definition of suprasegmentals given by Lehiste [64] states that "suprasegmentals are features whose arrangement in contrastive patterns in the time dimension is not restricted to single segments defined by their phonetic quality".

Stress is an emphasis or a prominence of one or more syllables in a word. In some languages such as English stress has an important role in pronunciation. Incorrect location of stress may change the meaning of the word (e.g. "object", "subject", "conflict", "record" as verb or as noun) or make it unintelligible to a native speaker. There are usually three different levels of stress that can be identified. Those are primary (the strongest), secondary, and the weak stress. The number of stress levels depends on the language and the interpretation. The stressed syllables may be characterized by longer duration, increased loudness, raised fundamental frequency or by a clearer and more distinct pronunciation of a vowel in the syllable. The realization of stress is a language-dependent property, which means that the characteristics may vary. In addition to the syllable stress at word level (phonemic and morphological stress) there may be a word stress at the phrase level. The prominence of words is used to emphasize something in a phrase, to clarify the meaning of the phrase or just to separate content words from function words such as articles and prepositions. The part of speech and the semantics also influence the prominence of the words in a phrase or sentence.

Intonation can also be viewed as the melody of speech and it is realized as changes in the fundamental frequency (F0 or pitch). The factors affecting the fundamental frequency can be divided into several components such as intrinsic, segmental and suprasegmental. Intrinsic properties affect the pitch for some vowel sounds. For example, high vowels tend to have higher fundamental frequency than low vowels. Intrinsic pitch properties are not particularly salient although they are present in all the languages. Also segmental features such as phonetic context may introduce changes in pitch. The change may occur before or after a certain segment. For example, it is likely that after voiceless consonants the F0 is higher. Suprasegmental features affecting fundamental frequency are stress and tone sandhi (tone interaction over larger segments) in tonal languages. In tonal languages, the tone (pitch pattern) is a distinctive part of the word. Changes in tone can also mean the change in the meaning of a word. The tones of consecutive words interact with each other changing the original pitch patterns. Of course, the intonation may also be viewed as an outcome of linguistic features such as semantics, information structure, syntax and grammar. The suprasegmental changes in the fundamental frequency are grouped into so called breath groups (also referred to as tone groups or intonation phrases by some authors). The determination of the breath groups may be based on the grammatical and phonological aspects or on some additional criteria. Each breath group may have its own intonation pattern. Question phrases often have higher pitch and a rise at the end whereas statement phrases tend to have a fall at the end. The breath groups and intonation patterns are produced naturally and with ease by each native speaker of a given language.

Timing includes the speaking rate, the use of pauses and the speech rhythm. As the rate of speech increases the less precise the articulation becomes and the larger impact the neighboring speech sounds have on each other (co-articulation, see section 3.4.3). Changes in speech rate also modify the duration of pauses in an utterance. In addition, the presence of a pause may provide grammatical information for the listener. The rhythm of speech is language dependent and closely related to the grammar and prominent syllables [68]. Languages can be roughly grouped by different rhythm patterns. The traditional, yet controversial classification contains two rhythm groups: stress-timed rhythm and syllable-timed rhythm [81]. In languages with stress-timed rhythm (e.g. English) the prominent syllables occur after constant intervals in time. In languages with syllable-timed rhythm (e.g. French) each syllable is allocated an equal amount of time so that the overall length of the spoken sentence depends on the number of syllables in it. The traditional classification has been challenged by other views such as grouping based on phonological properties of languages [23].

3.4.3. Co-articulation

Individual speech sounds or speech segments are pronounced differently depending on the neighboring sounds. This phenomenon is called co-articulation. Co-articulation is caused either by physiological constraints in human speech production (language universal) or by rules in a given language (language specific) [68]. Co-articulation can be classified based on the direction of influence, parameters of articulation or functional aspect [4].

Speech segments may influence either the previous segments (anticipatory, regressive) or the following segments (perseverative, progressive). For example in English [k] gets fronted velaric articulation before a front vowel [i]. This is also an example of an allophonic variation where the changed sound is still perceived as a phoneme /k/. In some cases, phonemic assimilation occurs and the speech sound is completely changed due to co-articulation.

For example, "ten men" gets pronounced as [tem men] in English. Co-articulation may also have an influence beyond the neighboring segments (feature spread) [4].

It is necessary to take co-articulatory effects into account in speech synthesis. Both language universal and language specific properties should be adequately modeled to produce natural sounding speech.

3.4.4. Phonetic similarities and differences between languages

A group of phones (or sounds) that the speakers of a given language feel to be one sound and which are never used in a contrastive way, forms a phoneme of that language. The contrast in a language means that a word can change meaning if a certain sound is replaced with another one. For example, /p/ and /b/ are contrastive in English since "pin" and "bin" are considered separate words. There are many variations of a phoneme /p/ even within a language. These variations are called allophones and they are not contrastive. Whether /p/ is realized as aspirated $[p^h]$ in "pot" or unaspirated [p] in "spot" it is still understood as /p/ in English. However, the same variation may be contrastive in another language such as Thai.

Allophonic variations are partially due to articulatory physics (intrinsic allophones) and partially due to rules (e.g. grammar) of a language [68]. The phonetic variation is realized with properties such as duration, aspiration (aspirated vs. non-aspirated), breathiness (breathy vs. non-breathy), release status (released vs. unreleased), nasalization (nasalized vs. non-nasalized).

The phonetic similarity is sometimes challenging to define. The same phoneme may have various acoustic realizations even within a language depending on the phonetic context as shown above. It is also possible that the same acoustic realization results in a different phoneme in different contexts. The phones (phoneme realizations) for the same phoneme in two languages are very likely to be different. This is an important fact to recognize in multilingual systems. The realization of a phoneme in one language may also overlap with the phoneme realizations of a different phoneme in another language. An example of the last case can be found in comparing a phone $[t^s]$ in German, Canadian French and English. In German $/t^s/$ is a phoneme of its own (e.g. in "Zank"). In Canadian French $[t^s]$ is just an allophone of /t/ in certain contexts (e.g. "type") and in English the sound is a combination of two separate phonemes /t/ and /s/ as in "bits" [68].

Due to the described reasons, the phones in a multilingual system can be viewed as forming partially overlapping regions in a "phoneme plane". Realizations within each region are determined by likelihood given the phonetic context and language.

In addition to variations in phoneme realizations between different languages there are also language dependent restrictions on phoneme sequences. Certain phonemic combinations are not possible in some languages whereas in other ones they may cause some co-articulatory effects that need to be considered when the language is modeled.

Chapter 4

Introduction to text-to-speech

THE main function of a text-to-speech system is to provide an automated way of speaking out text stored in a digital format. The development of a TTS system is a very interdisciplinary effort requiring knowledge of human physiology, linguistics and software engineering. Mathematics in the form of digital signal processing and statistics are also frequently used in the abovementioned fields. Table 4 shows how the expertise of different fields is applied to each functional module of a TTS system. The citations in the table provide further information or examples of each methodology.

Table 4. Interdisciplinary field of text-to-speech.

Language dependency	**Field**	**TTS system modules**	**Methodologies**
Language-dependent	Linguistics	Natural language processing	Rules [39][44] Stochastic models (n-grams) [37] Non-stochastic models [87]
	Phonetics	Text-to-phoneme	Rules [7] Dictionaries [105] CARTs [98] Neural networks [86][90]
		Prosody generation	Rules [2] Templates [51] CARTs [14][102] Databases (prosody embedded in units) [52]
Language-independent	Human speech production & perception	Waveform generation	Formant synthesis [58][59][66] LPC synthesis [79] Hidden Markov Model (HMM) synthesis [106][107] Concatenative synthesis [19] Variable length unit selection synthesis [5][52] Articulatory synthesis [36]

As shown, the conversion of textual input into speech can be carried out in many different ways using various methods. This chapter gives an overview of different classes of TTS systems. In addition, the main functional modules of a general TTS system are presented and different implementation methodologies for each module are described.

4.1. CLASSIFICATION OF TTS SYSTEMS

TTS systems may be categorized in many different ways. The classification is usually based on various system parameters and it is possible that a single TTS system belongs to multiple classes. Some examples of TTS system classes are presented in Table 5.

Table 5. Classification of TTS systems.

Parameters	Examples
Generality of input	generic / limited domain
Number of languages	multilingual / monolingual
Synthesis method	concatenative / rule-based
Memory consumption	embedded / PC / server
License	shareware / commercial

The generic systems do not set any limitations to the input text. They can process any type of text with any given content. They know how to filter out parts of the text which are not pronouncable and convert everything else into spoken format. The limited domain system on the other hand sets requirements and limitations on the type of input the system can handle. Such a system may be optimized just to handle the text dealing with a certain subject matter, for example weather. It may also introduce limitations regarding the format of the input text such as sentence structure or sentence length.

Classification also indicates whether a TTS system may be multilingual or monolingual. A monolingual system can only support a single language whereas a multilingual system supports multiple languages. Such a system may, depending on the implementation, use one language at the time or even process text written in mixed languages.

Another way to classify TTS systems is based on the synthesis technology or synthesis strategy used. The systems may be referred to as rule-based systems or concatenation-based (corpus-based) systems. The rule-based systems usually model the phonation mechanism with a set a parameters. Rule models may vary by the chosen approach. For example, it is possible to use phonetic feature-structures instead of plain phonemes as an input to the rule system [22][67]. The rich phonological representation would make the handling of allophonic variations easier. For example, phoneme /p/ may have strong aspiration, weak aspiration or no aspiration depending on the context. With plain phoneme input, several context dependent rules need to be created to choose the right allophone. This information is inherent in phonological structure. The proper allophone is selected based on the phone's role and location in a syllable. Concatenation-based systems operate on speech segments stored in a large database. Different segments are available for different contexts to allow good reproduction of co-articulary effects and allophonic variations. The systems may vary based on the concatenation algorithm and the speech database format. Speech segments may be for example, diphones or larger units and they may be encoded to provide memory reduction and allow concatenation in parametric plane. Thus in more detailed level synthesis systems may, for example, be classified as formant synthesis, LPC synthesis, diphone concatenation synthesis, unit selection synthesis, hidden Markov model (HMM) synthesis system, and so on.

The amount of memory that a system consumes can also be used in classification. The size of the implementation is often closely related to the synthesis method. For example the largest systems targeted for server platforms are usually based on the unit selection method and the smallest ones are rule-based systems using formant synthesis or compressed concatenation units. The largest systems require tens of megabytes of memory whereas the most compact ones can function with tens of kilobytes.

4.2. TTS APPLICATIONS

TTS technology is nowadays used in many different applications. The main application areas are voice user interfaces, automated announcement systems, hands-free/eyes-free applications, aids for the visually impaired, and entertainment applications (e.g. games).

In voice user interfaces TTS technology is used to complement automatic speech recognition (ASR) technology. In some telephone services there might also be other input mechanisms such as dual-tone multi-frequency (DTMF) tones. The most common services using TTS or recorded prompts are reservation, ordering and query services, which may be available 24-hours a day.

Another domain that benefits from automatic reading of text is in the field of information services. Weather reports, share prices in stock market, and any other form of continuous information stream can be provided with TTS. Such applications are usually domain specific which makes it possible to optimize the system for the best speech quality.

TTS is also an essential part of eyes-free operation of different devices. Synthetic speech can be used in reading e-mail or short messaging service (SMS) messages [P9][85]. Furthermore, TTS system can decrease the memory needs and promote multilingualism in navigation system so that the entire vocabulary does not need to be pre-recorded.

Another application category using the eyes-free feature is aimed especially to aid visually impaired people. The proper design and planning of the voice prompts provides a way for blind persons to access various devices. TTS can be used in producing the voice prompt or just in providing spoken output for the dynamic user interface content. An example of such an application called VoiceAid is described in [S2][73].

Finally, TTS can be used in entertainment and educational applications to give users richer audio experiences, to provide information or just to amuse. Such applications are for example games, educational software (e.g. talking dictionaries) and talking heads, where a TTS system output is synchronized with an animated image.

4.3. BLOCK DIAGRAM OF A GENERIC TTS SYSTEM

A good overview of a TTS system is given by Dutoit [28] and Holmes & Holmes [49]. A generic TTS system may be viewed as having a front end and a back end. The front end contains all the text processing and analysis functions, whereas, the back end generates the actual synthetic waveform.

The main purpose of the front end is to analyze the input text and find out all the essential information which may have an impact on pronunciation and prosody. The front end may also be called a natural language processing (NLP) module as suggested by Dutoit [28]. The simplistic approach of text analysis processes the input text in a sequential manner. However, it is easily shown that the different blocks within the NLP module must interact. For example, the expansion of an English text string "$5" depends on the context. It may be either "five dollars" or "five dollar" if it is connected to a noun (e.g. "five dollar bill") [94].

The back end uses the information on pronunciation and prosody to create synthetic speech. The waveform generation is quite independent of the NLP module. The high-level information provided by the NLP module is the same in most TTS systems. However, different systems may parameterize the information in different ways depending on the synthesis technology used in the back end of the TTS system. A block diagram of a generic TTS system is presented in Figure 6.

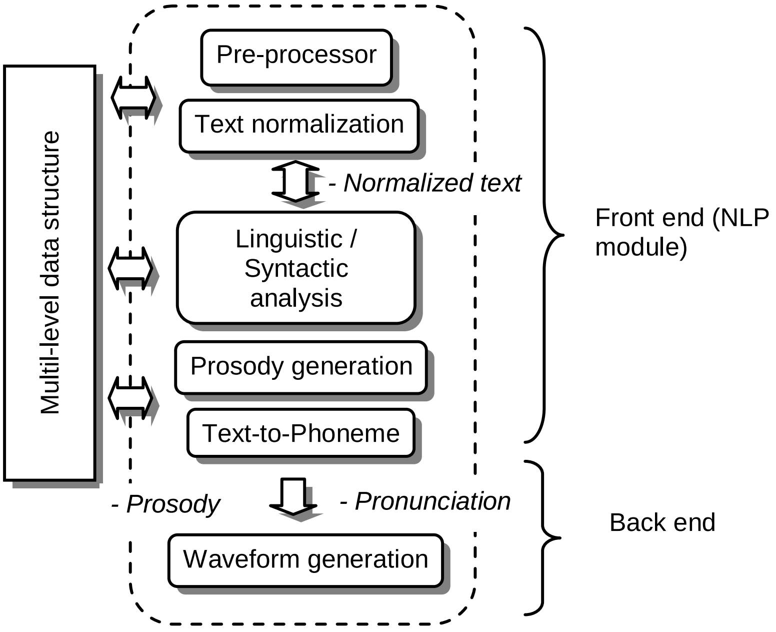

Figure 6. Block diagram of a generic TTS system.

The figure presents a rough breakdown of the functional modules of both the front and back end of a TTS system. The front end consists of pre-processor, text normalization, linguistic analysis, prosody generation and text-to-phoneme mapping modules. It should also be noted that all the text analysis modules are able to share the information in the multi-level data structure, which may contain a phrase structure, part of speech and other pieces of information utilized by several processing modules.

4.4. PRE-PROCESSING AND NORMALIZATION

The first processing module provides the high level manipulation and filtering of the input text. It might be required that prior to actual text normalization some information is filtered out. The filtering or pre-processing removes all unpronounceable characters from the input stream and also applies some application specific rules if needed. For example, a TTS system for e-mail reading could remove some fields from the e-mail header. If special annotations are used such as Speech Synthesis Markup Language (SSML) [112], the pre-processor extracts the tags (e.g. assigned paragraphs, emphasis, pauses) and passes the information to the other processing blocks of the TTS system.

The normalization or pre-processing stage includes also text segmentation and sentence detection. Text segmentation splits sequences of characters into words, numbers or punctuation marks. The segments are usually separated by one or more white space characters. The detected segments are further grouped into sentences. The sentence end detection may rely on punctuation such as period, colon or question mark. However, since the period character may also be used in numbers, dates or abbreviations, the context needs to be studied to determine the end of the sentence.

Another task of the normalization process is to expand numbers, abbreviations, web addresses (Uniform Resource Locators or URLs), dates, time expressions, currencies, and other character sequences which carry a meaning (e.g. smilies). The expansion is often context dependent so additional information about the surrounding sentence is needed. The output from the text normalization is pure text that is passed forward in the processing chain.

The text normalization may be constructed as a simple look-up table but such an approach lacks the flexibility to utilize full context information. A set of hand written rules could also be used to carry out the normalization task. Rules are accurate but it may take quite a bit of time and effort to create them. It has been suggested that weighted finite-state transducer (WFST) could be used in abbreviation and number expansion taking into account lexically dependent pronunciation. Probabilities are associated with state transitions so it is easy to calculate the most probable or the lowest cost paths across the transducer model [94]. Decision trees and n-gram language models have also been evaluated as a mean for providing text normalization for non-standard words [95].

4.5. LINGUISTIC ANALYSIS

One of the most challenging blocks in the TTS system is the module responsible for linguistic analysis. The processing is language-specific but several common steps can be found across languages. The term linguistic analysis is used here to cover morphological analysis, syntactic analysis and contextual analysis (e.g. grammars). The ultimate purpose of this analysis stage is to provide a description of a sentence with prosodic groups. This is done by breaking down the sentence into phrases and words with part of speech tags, word-level stress information, and syllabification. This information is often hierarchical and also essential for the generation of natural intonation, rhythm and phrasing in speech.

The sentences can be analyzed through classification of words. Detection of function words (determiners, pronouns, prepositions etc.) and content words can be used together with assigning part of speech (POS) tags to each word. The number of function words is often limited. The content words may appear in many different formats including singular, plural, feminine, masculine, neuter, past tense and other inflected forms. The morphological analysis recognizes the root word or word stems although the actual word may be modified for example by gender or number [28]. The TTS lexicon or dictionary can be significantly smaller when modified forms of the words do not have to be stored. The recognition of affixes gives additional information about the number, case, gender, conjugation and so on. This information can be further utilized in syntactic analysis. The study of morphology of words will also help in finding pronunciation for some compound words where some regular pronunciation rules do not apply.

After word-level analysis has been carried out, it is likely that there are still some ambiguities left in POS assignment. Some context dependent analysis is needed to finalize word classification. Contextual analysis may be performed with either probabilistic or deterministic methods. In probabilistic methods, word categories for ambiguous words are chosen using transition probabilities between successive word categories in a sentence (modeled e.g. with n-grams) [24][37]. The trigram model for POS assignment is described in [94]. The assignment of categories may also be carried out using neural networks [62]. The deterministic method assigns the category based on rules, which operate on the possible word categories. An example of rule-based POS assignment is so called Brill's tagger first introduced in 1992 [15] but enhanced in 1994 [16].

4.6. PROSODY GENERATION

Prosody of a synthetic utterance determines the perceived naturalness together with the accuracy of articulation. Even if speech segments were ideal, synthetic speech would not sound perfect without the natural intonation, proper phrasing and accurate location of stress. Prosodic components can be determined using information acquired during linguistic analysis. There are several methods to incorporate prosody into speech synthesis including classification and regression trees (CART) [14], mathematical models, knowledge-based rules [44], speech databases and pitch templates [60].

CARTs provide a data-driven statistical method for predicting prosodic features such as stress, intonation and segment duration [46][102]. Stress or pitch values for a given syllable or word can be determined using the same features that were used during the training of the trees. Word level stress or accentuation is determined by the position of the word within a sentence and context features such as discourse markers (e.g. "well", "now"). In some languages such as English content words will have different accent than function words. For intonation, the features include syllable stress, location of the syllable in a phrase and phonetic labels. Prosodic phrasing, which determines intonation patterns for longer speech segments, can be predicted with punctuation, length of utterance, distance in syllables, accentuation, syllable stress, part of speech and so on [94]. CARTs can also be used in predicting segment durations. The modeling of the segment duration must take into account co-articulatory effects and suprasegmental influences which means that each speech segment does not necessarily have an intrinsic duration [94]. Features which affect timing include segment identity, context, syllabic stress, word emphasis, location of the syllable in a word, and location of the syllable in a phrase.

The performance of the statistical prediction is heavily based on the regularity and consistent correlation between features in the training data and the feature being predicted. If, for example, a primary stress always causes a rise in intonation, accurate stress assignment would lead to correct intonation using statistical CART method. Details of using CARTs for intonation modeling is given in [P3][70], [100] and [102].

There are also other models for intonation modeling. For example, some popular models are the so called tone sequence model introduced by Pierrehumbert [80] and Tilt model [103]. There is also superposition model by Fujisaki [33][34][35]. In tone sequence model the intonational phrase is presented as a sequence of high and low tones for pitch accent, phrase accent and boundary tones. This way, intonation is described as relative changes in pitch rather than absolute values. The superposition model adds together influences of accent and phrase commands to form a complete pitch contour.

Prosody might also be embedded in the speech unit database as in some synthesis systems using waveform concatenation. Based on features produced by the linguistic analysis module, speech units providing the best phonetic and prosodic match are fetched from the database. Quality of the prosody depends on the richness of the database and the accuracy of the linguistic analysis.

Tonal languages such as Mandarin Chinese, Cantonese and Thai require specific support from pitch modeling. Tone in tonal languages is a contrastive feature which means that by changing pitch pattern, the meaning of the word is changed. Modeling of the tone can be carried out using any of the previously described methods but for more accurate pitch contours, templates might be more attractive [51]. Pitch templates provide normalized contours for each tone of a given syllable or word. The intonation pattern for the entire phrase can then be constructed as a combination of tones and sentence level intonation.

4.7. TEXT-TO-PHONEME CONVERSION

Text-to-phoneme (TTP, also sometimes called grapheme-to-phoneme, G2P or letter-to-sound, LTS) module creates pronunciation for words from their corresponding orthographic representation. Ideally the phonetic representation should take into account the context where the word occurs. For example, in some languages the pronunciation varies depending on the surrounding words. For example the English verb "read" is pronounced differently depending on the tense (present vs. past). The TTP module must therefore interact with the linguistic analysis module to obtain the necessary information. The pronunciation itself can be produced in several different ways. It can be generated using dictionaries where each word entry has a corresponding phonetic representation. In some languages (e.g. Finnish) where there is a straightforward relationship between orthography and pronunciation it is possible to write a complete set of rules to make the TTP conversion. In addition to these exact methods there are statistical approaches such as decision trees [P8][54][98][110] and neural networks [86]. They provide the most likely pronunciation for given words based on the features in the material used in training the decision trees and neural networks. Data-driven techniques are preferred when detailed language-specific knowledge is not available and automated development methods are needed. However, sophisticated but sometimes laborious knowledge-based approaches (e.g. in ETI-Eloquence) can offer better performance than data-driven systems especially in embedded, low-memory footprint systems [104].

4.8. WAVEFORM GENERATION

Prosodic parameters together with the pronunciation provide necessary information for generation of synthetic speech waveform. The set of parameters may be different depending on the synthesis technology but in all cases the core information is the same. The actual waveform generation may be based for example on rules and parameters such as in formant synthesis or it may use recorded speech databases of, for example, diphones or variable length units. This section gives an overview of various synthesis technologies.

4.8.1. Formant synthesis

The Klatt formant synthesizer developed by Dennis Klatt during the late 1970's is the most well known and widely used formant synthesizer [58]. The synthesizer was further enhanced in 1988 (Klatt88) [59]. The Klatt formant synthesizer has a model for glottal sound source that can generate either periodic glottal pulses or noise for aspiration or frication sounds. The vocal tract is modeled with a cascade or parallel bank of second-order resonators. In addition there are two pole-zero pairs for modeling nasal and tracheal effects in speech productions. The cascade arrangement is intended for generating sonorous sounds such as vowels, whereas the obstruent sounds should be produced using parallel construction. The simplified block diagram of the Klatt88 synthesis engine is shown in Figure 7.

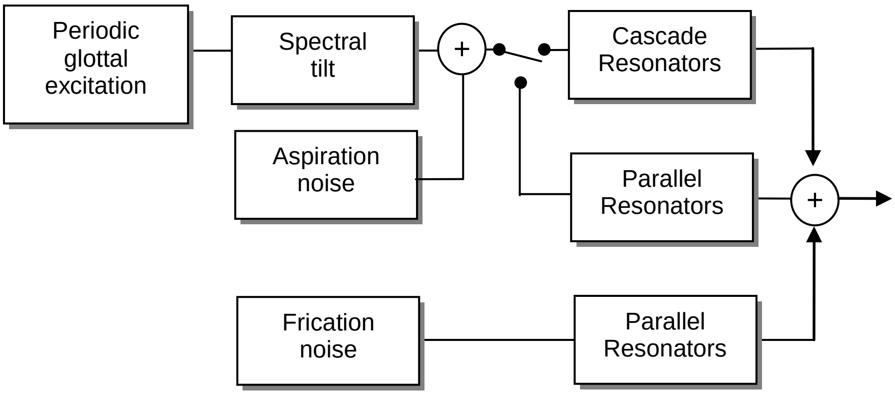

Figure 7. Simplified block diagram of a Klatt88 formant synthesizer.

Klatt88 has 12 parameters which are often constant for the synthesized utterance and do not have to be updated in every synthesis frame. They determine for example the sample rate, overall gains and number of formants in the system. In addition, there are 48 modifiable parameters which may be updated once in every synthesis frame. The most essential parameters are the formant frequencies and bandwidths for cascade resonators, formant amplitudes and bandwidths for parallel resonators, voicing gain, frication gain, aspiration gain, and fundamental frequency. The full list of parameters and their usage can be found in [58] and [59].

It has been shown that Klatt88 is capable of producing almost natural sounding speech via copy synthesis [47][48]. This means that human speech quality can be reached by manually tuning the synthesis parameters for each synthesis frame to match the spectral representation of natural speech. The automatic extraction of copy synthesis parameters is difficult due to interdependencies between parameters. The similar copy synthesis procedure is not possible in a TTS system because there is no reference speech signal available. Instead, there is just plain text as input. Hence the most challenging part of the formant synthesis development is to provide a way to extract and automatically control synthesis parameters. The available information for such a control module is the linguistic structure including phonetic representation of an utterance with syllable boundaries, and prosodic information including syllable stress, segmental durations and fundamental frequency.

4.8.2. Waveform concatenation

Waveform concatenation is a synthesis strategy which generates synthetic speech simply by putting recorded speech samples together. The quality of the concatenated speech depends for example on the smoothness of the spectral properties over the concatenation boundaries and the ability to choose units with perceptually-appropriate phoneme realizations. Also the amount of variation in voice quality of the speech samples should be minimized to guarantee good quality results. Several studies have been carried out to investigate various concatenation algorithms and their contributions to minimize spectral mismatches that are perceptually relevant [57][113]. Prosody matching techniques are required to modify pitch and/or duration of the segments. The typical scaling factor for pitch and duration is between 0.5 and 2 [28]. This indicates that there are limited means of modifying prosodic realizations in speech database units.

The basic speech segments in concatenation have traditionally been diphones. Diphones contain parts of two sounds thus embedding some of the effects of co-articulation. For example, the word "cat" would be a result of the following diphones: [sil-k] [k-œ] [œ-t] [t-sil], where "sil" denotes silence. Although there are more diphones than phones in each language, the number of units is still manageable. The total number of diphones varies from 500-1500 depending on the language [94]. The size of the unit database can be reduced using parameterizations and compression of speech samples. Furthermore, parameterized samples allow spectral and temporal manipulation to smooth the concatenation points. The modification of prosody is very likely needed during synthesis. Although the matching segments were found from the database, the durations and the pitch would need to be modified. One popular parameterization method in synthesis is linear predictive coding (LPC). Instead of storing the recorded speech samples the corresponding LPC parameters and excitation signal are stored. LPC-based waveform concatenation synthesis systems have been developed by Festival project, SVOX and Bell laboratories [8][79][94].

The most popular concatenation techniques used in diphone concatenation are Time Domain – Pitch-Synchronous OverLap-Add (TD-PSOLA) and Multi-Band Resynthesis OverLap-Add (MBROLA) [19][27]. TD-PSOLA algorithm uses pitch marks for periodic speech segments which are stored in the database. The manipulation of pitch and duration in the synthesis phase is carried out by shortening or lengthening the pitch periods through appropriate windowing of the signal. Because the signal modifications are done pitch synchronously there are no abrupt changes in pitch at concatenation points. MBROLA technique (developed from MBR PSOLA) is an enhanced version of TD-PSOLA. It incorporates some spectral smoothing methods and allows high data compression ratio but keeps the actual synthesis simple [29]. It should be noted that co-articulatory effects of some features of speech extend over four or five segments [101]. It is thus clear that a diphone-based system cannot fully capture all the variations (unless the database size is significantly increased to contain diphones in many different contexts).

Some of the shortcomings of diphone concatenation such as the need for signal processing and limited modeling of coarticulation can be alleviated using longer units or larger speech databases [5]. Longer units can better model the co-articulation, whereas larger speech databases provide more suitable segments for concatenation when appropriate context information is used in searching the database. The current state-of-the-art concatenation systems are using the unit selection technique [52]. Unit selection can be viewed as a generalization of diphone concatenation. Instead of storing only one copy of each diphone into the database, the unit selection database contains a large amount of speech data offering multiple copies of each diphone in various contexts. Unlike diphone synthesis, the unit selection does not apply major signal processing to units to minimize the degradation resulting from concatenation.

The concatenation units are chosen to minimize the total cost. The cost is a combination of a target cost and a concatenation cost. The target cost C^t describes how well the selected unit u_i corresponds to the desired unit t_i (phone, context, pitch, etc.). It is calculated as a weighted sum of differences between the elements of the target and candidate feature vectors as shown in (3).

$$C^t(t_i, u_i) = \sum_{j=1}^{p} w_j^t C_j^t(t_i, u_i) \tag{3}$$

The differences C^t_j are the sub-costs for each of the *p* features. The symbols *t* and *u* correspond to the target and candidate unit, respectively. Multiplier *w* is the weight assigned for each sub-cost. The concatenation cost, on the other hand, measures how well the selected unit matches with the neighboring units. The similar formula for concatenation cost is given in (4). [52]

$$C^c(u_{i-1}, u_i) = \sum_{j=1}^{q} w_j^t C_j^t(u_{i-1}, u_i) \tag{4}$$

The unit selection algorithm chooses the units in such a way that the total cost is minimized. The search for the optimal units may be carried out using for example The Viterbi algorithm [111].

Unit selection may operate on different unit sizes. Sub-phonetic units have been used for example in HMM-based synthesis and in AT&T NextGen system, which uses half phones [12][25]. Phone-sized units are being used in Festival framework (Festvox) [9]. Although most of the systems use fixed size units, the search algorithm may pick multiple consecutive units from a recorded utterance thus forming a "longer" unit.

4.8.3. Other synthesis methods

There are also other techniques to perform low level synthesis. Articulatory synthesis models the physiology of the human speech organs [36]. The model may include for example the movement of the articulatory muscles, vocal chords and vocal tract. Articulatory synthesis approaches can be divided into geometrical, physiological and statistical categories. The synthesis models are very complex and they are still in a research stage.

Another method which has received more attention during the past years is HMM-based synthesis [106][107]. HMM synthesis allows changes in synthetic voice quality such as speaking styles and emotions. As for speech recognition purposes, HMMs need to be trained before they can be used for synthesis. Spectrum and excitation parameters are extracted from a database and modeled by context dependent HMMs. In the synthesis phase, HMMs are concatenated. Advantages of the HMM-based synthesis are that it guarantees smooth transition between speech segments and predictable/stable output. Also, the memory footprint is quite small (e.g. one megabyte) compared with the unit selection systems (e.g. tens of megabytes).

There have also been some attempts and some on-going activities to combine different synthesis methods. For example the combination of waveform concatenation and formant synthesis has been experimented by Öhlin [116] and also by Hertz [45]. Öhlin's approach uses concatenation for voiceless segments whereas Hertz experimented with many other phonetic classes. It has been shown that perception of the voice quality is mainly determined by the stressed vowels i.e. stressed syllable nuclei. Therefore many consonants and unstressed vowels can be generated for example with formant synthesis without significant impact on perceived quality.

Chapter 5

Multilingual TTS on hand-held devices

ALTHOUGH the basic functionality of every TTS system is the same, there are numerous ways to implement it. A generic TTS system is not necessarily suitable for specific tasks with special vocabulary and characteristics. The implementation may also require a large amount of memory or a specific platform. Hand-held devices also have their own specific requirements which need to be addressed in TTS system development. This chapter describes the main challenges in the development of a multilingual TTS system for hand-held devices and provides some novel solutions. The new approaches have an impact on several functional blocks of a TTS system, from text normalization to waveform generation. The introduced methods have been applied to a multilingual low-footprint TTS system for mobile phones.

5.1. MAIN CHALLENGES

The selection of a suitable TTS system depends on several factors. The high-level requirements are set by the TTS application and constrained by the available resources such as implementation platform and development cost as shown in Figure 8.

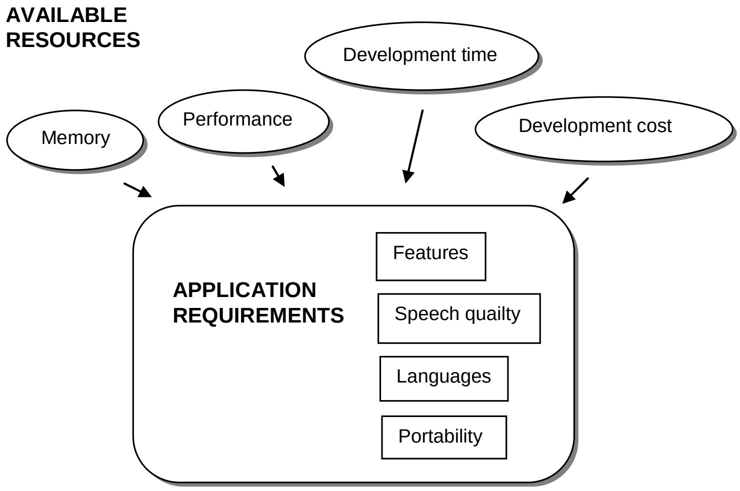

Figure 8. Main challenges for TTS system development.

Meeting the requirements set by the application can be quite challenging especially in embedded, hand-held devices such as mobile phones where resources are limited. The available memory and the central processing unit (CPU) time are usually lower than in desktop devices. Also the software cost (development or licensing) has a large impact on the total cost of the product. Therefore some compromises need to be made to balance the requirements and the available resources. The dependencies and trade-offs between the resources and requirements are complex. This section does not give a full, in-depth analysis of all possible issues but presents the main challenges encountered in acquiring or in developing a multilingual TTS system for hand-held devices.

5.1.1. Features of a text-to-speech system

The application requirements set by the features are, for example, generality of the input text, sentence length, specific pre-processing and normalization needs, number of voices, speech rate and needs for customization.

Some simplifications in the actual implementation may be utilized if the input to the TTS system is constrained. For example, a smaller lexicon can be adequate in limited vocabulary or limited domain applications. This also simplifies the pronunciation modeling, text normalization and makes the prosody modeling more accurate. In many cases, it is also possible to obtain higher speech quality and comprehensibility when the system is tuned to a specific vocabulary or domain [65]. Limited domain applications are, for example, automated systems for reading out weather reports or stock market information or aids for navigation. The system may also be limited to proper nouns, more specifically names. This constraint does not alleviate the challenges in pronunciation modeling but the characteristics of such utterances makes the prosody modeling simpler. Names can be used for example in mobile phones where calls can be initiated using speech recognition, and TTS used to provide feedback on the recognition results [P8][54]. In such application, the inputs to the TTS system are the phonebook entries, which are names of persons or companies.

Limiting the size of the input sentence can also significantly reduce the complexity of the TTS system. Especially the text normalization and syntactic analysis become simpler when only isolated words or short (e.g. 1-3 word) sentences are used as input. The prosody modeling is also rather straightforward for isolated words where there is no sentence context that needs to be taken into account. The TTS applications on mobile phones that take advantage of limited sentence length include talking dictionaries, name dialing, voice commands and spoken caller's identification [S2][73]. Also any other application which can provide user precise and brief spoken information can benefit from limited length TTS. For example, there is a training application in Nokia 5500 Sport, which provides distance, speed, energy consumption and duration information using TTS during physical exercising [P9][85].

The application may also present specific requirements regarding normalization and pre-processing. Such requirements are mainly set by the vocabulary. Financial applications must provide correct handling and expansion of currencies, and e-mail readers must filter out the unnecessary header information and unpronounceable characters. In some cases, generic normalization rules must be overridden with domain or application specific rules. For example, "AC" may indicate alternating current, air conditioning or time (ante Christum) depending on the context.

Applications may also need to offer a certain level of personalization. There could be a selection of different synthetic voices available to the user or another voice could be generated using so called voice conversion techniques [1][20]. Voice conversion modifies the original TTS voice to resemble the target voice that has been provided to the system. Some applications may also provide the user some parameters to modify the voice and other related parameters such as speaking rate and average pitch.

5.1.2. Speech quality

The quality of synthetic speech is difficult to quantify. It may be viewed as a combination of several attributes such as intelligibility, ease of listening and pleasantness [93]. The perceived quality might also vary according to expectations. Based on the listener feedback it can be assumed that more mistakes in pronunciation and prosody are tolerated with a machine-like synthesis than with synthesis having highly natural and human sounding voice. If the general voice quality is human-like, all the deviations and mistakes stand out. There have been several suggestions for objective evaluation methods of TTS systems but they are not necessarily comprehensive. Modified and diagnostic rhyme tests [38] or minimal pairs may reveal something about intelligibility but other methods are also needed to get the complete picture. One simple way for rating intelligibility is to measure reception threshold as suggested by Vainio et al. [109]. Mean opinion score (MOS) rating has also been used in quality assessment especially in concatenative synthesis where natural speech can be used as reference. Comparisons of different systems or comparison of different system parameters can provide important information about the quality. If the TTS system is composed of several functional modules it is possible to evaluate the performance of each block by keeping the other variables constant. For example, the impact of prosody modeling, sentence segmentation and pronunciation modeling can be isolated using such method in listening evaluation tests.

The required level of speech quality depends on the application. Simple and short utterances can be synthesized with lower speech quality especially if the utterances are from a limited set known by the user. This is the case when TTS is used as a feedback mechanism for automatic speech recognition and repeats the recognition result. For arbitrary and lengthy utterances, higher speech quality is needed to provide higher intelligibility and to prevent listener's fatigue. The selection of synthesis technology also has an impact on quality. A high-quality concatenative synthesis can sound more natural and human-like than formant synthesis. However, a well tuned formant synthesis might still be more intelligible especially in noisy environment. A balance between the naturalness, pleasantness, intelligibility and ease of listening has to be determined based on the requirements of the application.

5.1.3. Language coverage

There are 4000-8000 languages in the world but only 5% of them are spoken by 94% of the population. The five largest languages are Mandarin Chinese, Spanish, English, Arabic and Hindi [40]. The languages differ by their historical origin and/or their linguistic characteristics (typology). The linguistic characteristics include morphology, word order, phonetics, phonotactics, prosody, and the writing system [56]. Any piece of software that needs to function in many countries using different languages needs to be internationalized. The usual internationalization mechanism is to present language dependent information such as keymaps, fonts, numbers and currencies, UI messages and text, and language rules separately from the actual software [41]. In practice this can be done using, for example, resource or data files. The same applies also for TTS software.

The application using TTS can either be monolingual or multilingual. Ideally the same TTS engine should be used in every language configuration so that only language specific data is modified. The dialects and the sociolects must also in many cases be treated as separate languages in TTS system development. The requirements and the development effort for a new TTS language depend on the synthesis technology, the complexity of the system and the language itself. Concatenative systems using segments from speech database naturally require the recording and annotation of several spoken utterances. The amount of recorded data for unit selection synthesis is considerably larger (many hours of speech) than for diphone synthesis. The annotation includes usually marking of sentence, word and syllable boundaries, syllable stress and intonation. Some of the annotation can be done automatically using for example ASR technology. Even LPC, HMM and formant synthesizers require recorded and annotated speech data for new languages. Parameters for synthesis and prosody can be extracted from real speech.

In addition to recording of speech data, usually language dependent lexicons are needed. Lexicons are required to provide information about part of speech, syllabification, lexical stress and pronunciation. The lexicon or dictionary is often the most straightforward way of getting the required information. The use of commercially available lexicons might be restricted and license payments are often mandatory. It is also possible that suitable lexicons might not be readily available for all the languages. This makes the language development process more time consuming and requires more effort in data collection.

Systems which are capable of handling full sentences need more complex methods for syntactic analysis and text normalization. On the other hand, systems, which support only isolated words, can be tuned for new languages faster because there is no need to embed sophisticated language models in the system. The prosody and pronunciation modeling for single words is much simpler than for full sentences. Languages are also different. Some of the languages are very regular in syntax and pronunciation whereas some others are very complex. For example, languages such as Finnish or Korean (Hangul) have a fairly straightforward correspondence between the orthographic representation and the pronunciation. However in Arabic the written text usually does not contain any vowels so they need to be generated during the pronunciation modeling.

Another language dependent issue relates to the representation of text. Some languages use logographic characters (e.g. Mandarin Chinese) whereas others use phonographic ones. There are ways to transliterate non-western characters but nowadays it is more attractive to use the Unicode character representation [108]. Unicode encoding allows the use of multiple languages with a "single" character set on a single device.

5.1.4. Portability

The term portability denotes the ability to transfer the software implementation to another platform. The porting may take place between different CPUs, different operating system (e.g. from Windows to Symbian or from Windows to Linux), or between different operating system versions. The porting is carried out either using executable format or in source code. Porting executable code (binary compatibility) requires that the software is written using a suitable binary interface or the target platform uses a virtual machine that mimics the source platform. The source code porting requires re-compilation of the software. There are also several other forms of porting between these two extremes [13][50].

The most portable system implementation is the one which does not have any dependencies on the actual platform. However, this is quite impossible in practice. The best feasible solution usually is to minimize the dependencies by presenting most of the data in platform independent format. This is usually done by creating an abstraction layer in the software. The software above the layer sees only the facilities provided by the abstraction layer. Only the code beneath the layer has dependencies to the implementation platform [13][50].

Portability has to be taken into account also in TTS system development if the software needs to run on different platforms and if the lifecycle of the system is to span over several operating system versions. This is important especially for the data that is frequently updated or varied. Although the program code needs to be ported once for every platform, the data packages should be compatible with all the platforms. Thus the data does not have to be recompiled during porting but can be handled separately. This also allows the managing of different language configurations as long as it is done using data packages.

5.2. FRAMEWORK AND SYNTHESIS TECHNOLOGY

The specific requirements of embedded platforms such as mobile devices should be taken into account when the system framework is designed. Some requirements are also application specific. For example, the speech quality for longer, arbitrary utterances needs to be higher than for the short utterances of known content. To meet the different requirements of different applications and platforms, there is a need to support different TTS system configurations within the same framework as described in [S2][73]. The given approach introduces two different TTS engines. In both cases the applications use the same application programming interface (API) to interact with the TTS system. The unit selection synthesizer with compressed unit database can be used in applications requiring TTS support for longer utterances provided that the platform allows the higher memory consumption. The low-footprint solution for isolated words and short utterances is optimized for low memory consumption and large language coverage. The low-footprint TTS uses formant synthesis and rules to control the synthesis parameters. The low-footprint synthesis system and the related rule language are described in [P5][82]. The overview of the two-engine framework is shown in Figure 9.

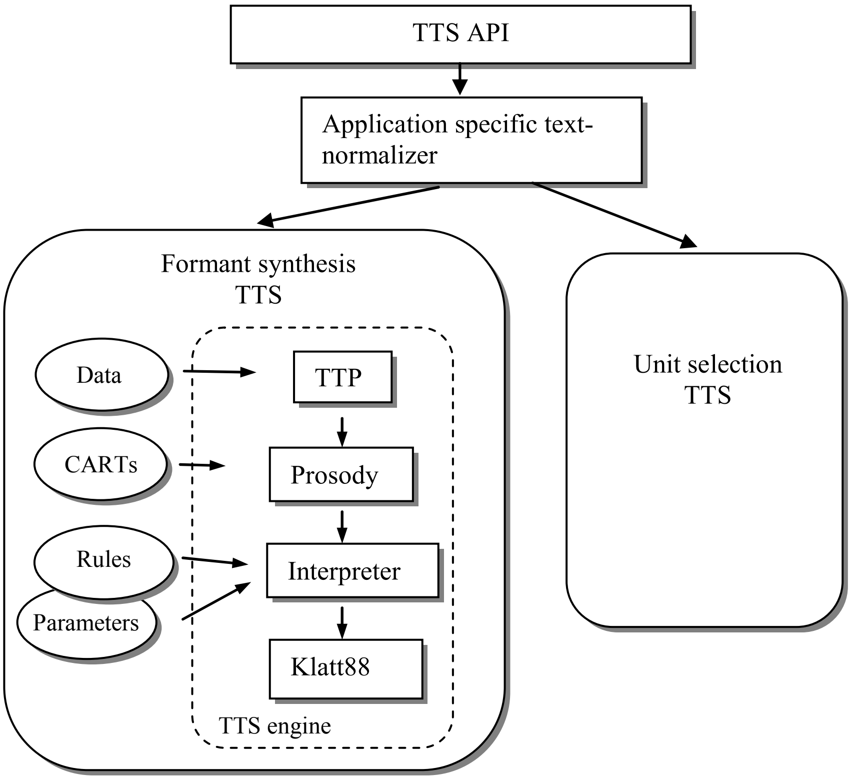

Figure 9. TTS framework for hand-held devices.

Both engines may be present in a device at the same time or one of them can be left out to optimize the configuration. In some applications it is also possible to use low-footprint engine as a backup synthesis method for languages which are not available in unit selection system. The following section provides some further information about the solutions made in the development of the low-footprint TTS engine.

5.3. LOW-FOOTPRINT TTS ENGINE

The low memory budget and the need for support of multiple languages (about 40 at the time of the development phase) have been the two main drivers during the system design. Some shortcuts and optimizations, which are presented in the following sections, were applied since the TTS system was targeted only to support short utterances and isolated words. The system was made truly multilingual by separating the program code and language dependent data. A similar architectural approach has previously been used in the SVOX system [99] offering scalability and engine independent language development. The separation of the engine code and language specific data also allows easy system configurability for different languages as also shown in ETI-Eloquence system [42][44]. The language dependent data in our solution included the rules needed for text normalization, phonemes of a given language, prosody models, and rules for controlling synthesis parameters and parameter transitions.

5.3.1. Text normalization

The text normalization for the low-footprint TTS system was designed for a limited set of applications. However, the rule framework was made expandable to other normalization tasks too. It was realized that the published prior-art normalization frameworks were mainly developed from the English language point of view. Due to this reason, number, gender and case specific inflections were overlooked in the normalization task [114].

The main requirement was to provide normalization for numbers in multiple languages. In addition to basic numeric expansion in nominative case, the gender-specific inflexions had to be supported (e.g. *un* and *une* in French). The same framework was also to provide expansion for abbreviations especially when they are present with a number (e.g. 45 km). The rule framework should also be so easy to use that translators without technical expertise could write the number and abbreviation rules with proper tools.

The previously published normalization techniques were based on simple look-up tables, weighted finite-state transducers [91], or on some hand-written rules. Most of the documented rule frameworks were created for financial applications so they did not cover all the requirements of a TTS system. One of the most compact rule-based normalization frameworks that has been published was developed by IBM Corporation [39]. However, the framework still suffered from the lack of support for context-based word inflections. Because there was no normalization framework that was meeting the requirements available, it had to be created.

The rules were developed to provide expansion for integer numerals from 0 to 999999, for a limited set of units (m, yd, mi, km, kcal, kJ, min/km, min/mi, h, s, min, km/h, mph) and for time expressions. The information about the gender and inflections was also embedded into the rules as they have an impact on spelling and pronunciation of related numbers (e.g. 1 m → "one meter", 5 m → "five meter**s**"). Other types of normalization were not needed nor included as it was assumed that applications provide only properly formatted text strings for the TTS system. The complete description of the rule framework and verification of its suitability for the task is given in [P7][72].

5.3.2. Optimization of phonetic representation

The low-footprint TTS system takes advantage of the text-to-phoneme conversion originally developed for automatic speech recognition purposes [P8][54]. The conversion is based on statistical models (decision trees), hand-written rules or look-up tables depending on the language. The number of phones needed to represent all sounds of a given language varies from one language to another. Despite the differences in languages, there are sounds that are somewhat similar between languages. This fact can be utilized in simplifying phone descriptions in a multilingual system. The total number of phones can be reduced by sharing the same phone definition between multiple languages. Obstruents, which are described by a common IPA notation, are likely to be close to each other irrespective of the language although allophonic variations are quite common (e.g. stop with and without aspiration). Sonorants such as vowels have a lot more language dependent and perceptually significant variance. This is applicable for both ASR and TTS. The ASR technology can tolerate phonetic generalizations quite well but for TTS the phonetic mismatch is audible. However, there are use cases and applications where the approximation of pronunciation is adequate when it provides savings in memory consumption and easier support for multiple languages.

Another TTS specific method for reducing the number of phone parameters of a given language is to combine several sounds to form another sound. For example, an affricate sound [tS] can be approximated with combined [t] and [S]. A similar approach can be used to form diphthongized vowels such as [OI], [aU], [eI] and so on. This method is described and evaluated in [P1][75] and it is shown that some memory reduction can be obtained if the TTS system has to store phone specific data (e.g. parameters, database units).

The language specific information stored in a multilingual TTS system can be further reduced by approximating language specific sounds with resembling sounds from another language. This technique is used in the so called cross-lingual phoneme mapping, which allows fast development of new TTS languages [P2][3][17][74]. A new TTS language can be created using an existing TTS language as long as the phonetic input matches the phoneme set of the existing TTS language. The phoneme mapping takes care of the conversion of non-supported phonemes into supported ones. This way, the sounds of a new language can be approximated by the sounds of the existing synthesis language. The drawback of this solution is the phonetic inaccuracy in pronunciation and possibly degradation in intelligibility. The mismatch is heard as a foreign accent. This is because the phones and prosody come from the

TTS language used in approximation. Despite the shortcomings, the approach is attractive when the language portfolio needs to be rapidly expanded with limited development resources.

5.3.3. Prosody modeling

The full-scale prosody modeling usually requires natural language processing to detect phrases, to assign stress and pauses, and to produce proper intonation. The analysis often requires a large lexicon where morphological information, part of speech information and syllabification can be obtained. The syntactic analysis requires language expertise and significant effort when multiple languages need to be supported. The full lexicon, albeit very useful, consumes memory that can be scarce on hand-held devices. Based on this rationale, the prosody modeling can be simplified to match the application requirements with given resources.

The language specific modeling of prosody in a low-footprint TTS system could be carried out for example using rules with simple mathematical models or trained classification and regression trees (CARTs [14][102]). It has been shown that both CART approach and mathematical model (Fujisaki [33][34][35]) can produce intonation contours whose perceived quality in formant synthesis is comparable to natural, human intonation [P3][70]. Tuning the parameters for a mathematical model requires some work for each language whereas CARTs can be automatically generated with properly annotated recordings. Another driver for such a statistical approach is the availability of tools and automated methods for model generation [8]. The CARTs, especially when coded, are also memory efficient and thus quite suitable for a multilingual low-footprint system.

There have been some suggestions for automatic methods for extraction of Fujisaki parameters [69][76][88] but controlling the parameters during synthesis remains challenging. The linking of Fujisaki parameters to linguistic features is not always straightforward and it may vary from one language to another.

5.3.4. Rules for controlling synthesis parameters

The use of formant synthesis provides a very compact solution for low-footprint TTS engine. Different rules are needed in controlling the parameter transitions caused by co-articulation and other properties of human physiology. In systems where waveform concatenation is used, the context dependent selection of each synthesis unit takes care of the smooth boundaries between different phonemes.

The lowest memory consumption is achieved using phones as basic units in formant synthesis. Each phone can be defined using a set of 30-40 Klatt-parameters [58][59]. Phones or phonemes in more abstract form were used also to have common units with the in-house ASR system. Due to the properties of human speech, the phones rarely appear as segments of steady-state parameters. For this reason, additional rules are needed to modify the parameters, formants in particular, which are defined in the formant synthesizer. The stored parameters are used as target values for parameter contours. The contours are formed by interpolating the values between the consecutive targets. Human perception defines the resolution needed in modeling formant parameter contours [77]. It has been shown that linear interpolation together with two dynamically positioned control points for each phoneme are adequate in formant contour modeling [P4][43][83]. The low number of points used in interpolation also simplifies the required computations.

In a multilingual synthesis system, language specific rules need to be separated from the TTS engine software. The framework that separates the data and the code also allows the separation of language development from the software (SW) development. This way, linguists can concentrate on their special know-how without being concerned about implementation issues. A good example of an abstract language made especially for creating formant synthesis rules from the phonetics point of view is the Delta language [42][44]. A somewhat similar approach has been taken in [P5][82]. A special script language was developed to allow the creation of language rules. The rules are efficiently coded into language packages to reduce memory consumption and loaded when needed by the synthesis engine. Thus the same engine can support multiple languages through the use of language packages. Furthermore, expressing the language rules and parameters in language packages rather than in program code reduces the memory consumption.

5.3.5. Development and maintenance

A considerable amount of effort in TTS work goes into synthesis language development. The required effort varies depending on technology, language, and also on desired level of quality. Language universal components need to be ideally created only once. New languages or new voices can be developed from scratch or based on some existing work. The development work may consist of speech database design, speech recordings, annotations, study of linguistic properties and their formulation, rule creation and many rounds of iterations and optimizations [11]. If tens of new languages were needed in a short period of time, it would require considerable effort or some other means to reduce the work and speed up the development process.

Some properties of low-footprint TTS can be utilized in making the development of new TTS languages faster. This is true especially when the TTS system is designed for short utterances and isolated words. The data-driven techniques should be used as much as possible to make parts of the development process automated. Such steps, for example, are the generation of models for text-to-phoneme conversion, intonation, segment duration, syllabification and stress assignment [8][98][102]. Segmentation and annotation of speech data can be done, at least initially, by automated tools using for example ASR technology [78]. For tasks, which can not be fully automated, there should be easy-to-use tools. Tools are typically needed for annotating speech data [18], and for detailed analysis of speech spectrogram [63][92]. The rules required for controlling the formant synthesis parameters are created manually. A software tool was created to allow language specific rule development, debugging and data analysis. Tools and their user interfaces may also be useful in hiding the technical details of the implementation from the language developers [P6][71].

5.4. UNIT SELECTION TTS

The unit selection system which is integrated in the embedded TTS framework can be optimized for memory footprint. The full system would require tens of megabytes of memory which is usually too large for handheld devices. The memory size of the unit selection system can be reduced by decreasing the sampling rate, decreasing the number of units in the database or by applying compression to the units. Of course, all of these methods can be used in parallel. The removal of units can be carried out, for example, based on their relative frequency; the less common units are the first ones to be left out. The reduction may also take into account the application domain and the vocabulary that is needed. More intelligent methods such as pruning using different quantization or clustering methods can be used to scale down the unit database [6][10][55].

Chapter 6

Conclusions

MODERN voice-based services and applications are common in everyday use. Automatic speech recognition (ASR) and speech synthesis are frequently used together in various voice dialogue systems. Users are able to interact with the system using normal speech. Such voice-based user interfaces are frequently encountered in modern telephone services. The services may provide, for example, flight schedule information or allow ticket reservation. The advancement of technology has brought ASR and text-to-speech (TTS) applications also to smaller, personal devices. The computational load and memory consumption of such applications no longer require a large server computer but instead a hand-held device can provide a sufficient platform for several useful applications.

Text-to-speech technology has come a long way since the first computerized synthesis system in 1980's. The most commercially used TTS systems are based on waveform concatenation, which produces synthetic speech by connecting speech segments retrieved from a recorded database. The concatenation and especially unit selection systems are capable of providing highly natural speech. The drawbacks of this technology are the effort of database recording and preparation for new languages and high memory consumption. The speech database of such systems may require tens of megabytes of memory for a single language. Of course, the size of the multilingual system increases accordingly. Formant synthesis, although old and traditional as a synthesis technology, can still provide attractive alternative for some applications and systems. The speech quality is often machine-like but still intelligible. In some situations like in noisy environment it may even offer better intelligibility than systems using more natural, human-like speech. Furthermore, formant synthesis does not require a speech database; thus the memory footprint is smaller.

The thesis investigates the requirements of a multilingual low-footprint TTS system for hand-held devices. Several novel techniques were developed to meet the requirements. The main motivations behind the contributions were to reduce memory consumption, support multiple languages and allow easy configurability and maintenance.

A memory efficient, multilingual rule framework was developed for number and abbreviation expansion. This text-normalization task was aimed for the so called training UI to provide voice information during physical exercise (speed, duration, energy consumption etc.) Unlike some previously published rule frameworks, the method described in this thesis supports inflections due to number and gender. The framework was also made expandable for other purposes such as time expressions. Usability and functionality of the system were verified by implementing the number, unit and time rules for 42 languages using mostly professional translators. The average number of rule lines for number, unit and time rules were 87, 49 and 13, respectively. The main shortcoming of the presented method was the finite numeric range.

Enhancements were also made at the phonetic level. After applying the text-to-phoneme conversion, the phonetic representation of the utterance is modified. Memory consumption of Klatt formant synthesis parameters was reduced 7% by decreasing the number of phones needed to present the sounds of a given language. This was carried out for example by constructing affricates from two separate sounds or by creating a long vowel through prolonging the shorter one. The reduction in memory consumption was relatively small but provided a way to simplify and optimize the system. The rapid expansion of the language coverage was done through cross-lingual phoneme mapping where several sounds were approximated with others that were already available in the system. The support for new languages was quickly obtained with the cost of approximated phonetics and foreign intonation. However, listening evaluations indicated that for similar languages and for certain applications the described approach was adequate.

Some ways to improve the handling of the formant synthesis parameters were also investigated. It was shown that a fairly simple intonation model can be successfully used when short utterances such as names or other isolated words are synthesized. The synthetic intonation using either Classification And Regressions Trees (CART) or Fujisaki model was seen comparable with natural intonation in listening evaluation tests when formant synthesis was used. In addition to fundamental frequency F0, other formant parameters and their control mechanisms were studied. It was shown that it is adequate to model formant contours using only two control points per phoneme and fill in the missing points with linear interpolation. The location of control points must be set dynamically for each phoneme to obtain desired results.

In addition to functional improvements the system architecture and implementation methods were improved. Support for high quality unit selection system was added to the overall TTS framework for applications such as message reader. For the low-footprint TTS, a script language was developed for expressing language dependent synthesis rules. The use of an abstract script language separated synthesis language development from actual programming. The rules controlled the co-articulation and manipulated synthesis parameters to mimic natural transitions between adjacent sounds. A binary coding method was also applied to store all the language rules in a memory efficient format. It was shown that the use of the script language for rules reduces memory consumption in average by 69% for each synthesis language compared with DLL-based approach. The software engineering and the linguistic tasks were further separated by having a special tool for creating language rules and testing them. A platform independent integrated development environment was created using Java. The environment allowed editing of the rule files and converting of each rule file into binary format. It was also possible to listen and debug the results of the rule file modifications. The script language, the separation of code and data, and the development environment enabled rapid and memory efficient synthesis language development and easy configurability for a multilingual TTS system.

Future work includes improving speech quality, creating new data-driven methods for parameter and rule development, and expanding the application domains. Quality improvement may result in enhancing the synthesis of languages which are currently implemented using cross-lingual phoneme mapping. On the other hand, more intensive use of native speakers as well as linguists in language development could also be helpful. Furthermore, the use of another parametric model such as LPC would allow easier ways to automatically search speaker parameters, and even perform voice conversion. Further research is also needed in natural language processing to provide better prosody models and support for arbitrary length sentences.

This thesis shows that commercially viable multilingual TTS-based applications can be created using the following four main techniques. First, the limitations of the TTS technology can be hidden or alleviated by limiting the scope of the application. Second, optimization in memory consumption and performance makes the TTS technology more attractive for handheld devices. Third, multilingualism and rapid development of new synthesis languages are enabled through system design and proper development methods and tools. Finally, the separated TTS engine software and language dependent data make it possible to hide the software engineering details by providing language developers an interface with a higher level of abstractness.

Bibliography

[1] M. Abe, S. Nakamura, K. Shikano and H. Kuwabara, "Voice conversion through vector quantization", *Proceedings of ICASSP 1998*, New York, NY, USA 1988, vol 1, pp. 655-658.

[2] J. Allen, M. S. Hunnicutt, D. H. Klatt, R. C. Armstrong and D. B. Pisoni, *From text to speech, The MITalk system*, Cambridge: Cambridge University Press, 1987.

[3] L. Badino, C. Barolo and S. Quazza, "Language Independent Phoneme Mapping For Foreign TTS", *Proceedings of 5th ISCA speech synthesis workshop*, Pittsburgh, PA, USA, 2004, pp. 217-218.

[4] M. J. Ball and J. Rahilly, *Phonetics, the Science of Speech*, London, Great Britain: Arnold, 1999.

[5] A. Black and N. Campbell, "Optimising selection of units from speech databases for concatenative synthesis", *Proceedings of Eurospeech 1995*, Madrid, Spain, 1995, pp. 581–584.

[6] A. Black and P. Taylor (1997), "Automatically clustering similar units for unit selection in speech synthesis", *Proceedings of Eurospeech 1997*, Rhodes, Greece, 1997, pp. 601-604.

[7] A. Black, K. Lenzo and V. Pagel, "Issues in building general letter to sound rules". *Proceedings of ESCA/COSCODA international workshop on speech synthesis*, Jenolan Caves, Australia, 1998, pp.77-80.

[8] A. Black, P. Taylor, R. Caley (1999, Jun.), The Festival speech synthesis system, system documentation, Centre for Speech Technology Research, University of Edinburgh. [Online], Available: http://www.cstr.ed.ac.uk/projects/festival/manual/.

[9] A. Black and K. Lenzo (2003, Jan.), "Building synthetic voices", Festvox system documentation, [Online]. Available: http://www.festvox.org/bsv/

[10] A. Black and K. Lenzo (2003, Oct.), "Optimal Utterance Selection for Unit Selection Speech Synthesis Databases". *International Journal of Speech Technology* [Online]. vol 6, issue 4, pp. 357-363, Available: http://www.springerlink.com/content/r86n47248t077721/

[11] A. Black, "Multilingual speech synthesis", T. Schultz and K. Kirchhoff (Eds.), *Multilingual Speech Processing*, MA, USA: Academic Press, 2006, pp. 207-231.

[12] M. Beutnagel, A. Conkie, J. Schroeter, Y. Stylianou, and A. Syrdal, "The AT&T Next-Gen TTS system", *Joint Meeting of ASA, EAA, and DAGA*, Berlin, Germany, 1999, pp. 18-24.

[13] C. Brand (2005, Aug.), "How to Achieve Application Software Portability", *COTS journal* [Online]. Available: http://www.cotsjournalonline.com/home/article.php?id=100378

[14] L. Breiman, J. Friedman, C. J. Stone and R. A. Olshen, *Classification and Regression Trees*. Belmont, CA, USA: Wadsworth Inc., 1984.

[15] E. Brill, "A simple rule-based part of speech tagger", *Proceedings of the Third Conference on Applied Natural Language Processing, ACL (Association for Computational Linguistics).* Trento, Italy, 1992, pp. 152-155.

[16] E. Brill, "Some advances in transformation-based part of speech tagging", *Proceedings of 12th national conference on artificial intelligence*, Seattle, Washington, USA, 1994, pp. 722-727.

[17] N. Campbell, "Talking foreign. Concatenative Speech Synthesis and Language Barrier", *Proceedings of Eurospeech 2001*, Aalborg, Denmark, 2001, pp. 337-340.

[18] S. Cassidy (2004), EMU speech database system documentation. Centre for Language Technology, Macquarie University [Online]. Available: http://emu.sourceforge.net/

[19] F.J. Charpentier and M.G. Stella, "Diphone synthesis using an overlap-add technique for speech waveforms concatenation", *Proceedings of IEEE ICASSP 1986*, Tokyo, Japan, 1986, vol. 11, pp. 2015- 2018.

[20] D. Childers, B. Yegnanarayana and Ke Wu, "Voice conversion: Factors responsible for quality", *Proceedings of ICASSP 1985*, Gainesville, FL, USA, 1985, vol 10, pp. 748-751.

[21] L. A. Christovich and V. V. Lublinskaya, The "center of gravity" effect in vowel spectra and critical distance between the formants: Psychoacoustical study of the perception of vowel-like stimuli. *Hearing Research*, vol. 1, issue 3, pp. 185-195, 1979.

[22] J. S. Coleman, Unification Phonology: Another look at "synthesis-by-rule". H. Karlgren (ed.), *Proceedings of the 13th International Conference on Computational Linguistics* (COLING 90), 1990, Helsinki, Finland, vol. 2, pp. 79-84.

[23] R. M. Dauer, "Stress-timing and syllable-timing reanalyzed", *Journal of Phonetics*, 11, pp. 51-62, 1983.

[24] S. DeRose, "Grammatical category disambiguation by statistical optimization", *Computational Linguistics*, 14, pp. 31-39, 1988.

[25] R. Donovan and P. Woodland, "Improvements in an HMM-based speech synthesizer", *Proceedings of Eurospeech 1995*, Madrid, Spain, 1995, vol. 1, pp. 573-576.

[26] B. Doval and C. d'Alessandro, Spectral correlates of glottal waveform models: an analytic study, *Proceedings of IEEE ICASSP 97*, Munich, Germany, 1997, pp. 1295-1298.

[27] T. Dutoit and H. Leich, "MBR-PSOLA : Text-To-Speech Synthesis based on an MBE Re-Synthesis of the Segments Database", *Speech Communication*, vol. 13, issue 3-4, pp. 435-440, Nov. 1993.

[28] T. Dutoit, *An Introduction to Text-to-Speech Synthesis*, Dordrecht, The Netherlands: Kluwer Academic Publishers, 1997.

[29] T. Dutoit, V. Pagel, N. Pierret, F. Bataille and O. van der Vrecken, "The MBROLA project: Towards a set of high quality speech synthesizers free of use for non commercial purposes", *Proceedings of ICSLP 1996*, Philadelphia, PA, USA, 1996, vol. 3, pp. 1393-1396.

[30] G. Fant, *Acoustic Theory of Speech Production*. Hague, The Netherlands: Mouton de Gruyter, 1970.

[31] G. Fant, "The source filter concept in voice production", *STL-QPSR*, vol. 22, issue 1, pp. 21-37, 1981.

[32] G. Fant, J. Liljencrants and Q. Lin, "A four-parameter model of glottal flow", *STL-QPSR*, vol. 26, issue 4, pp. 1-13, 1985.

[33] H. Fujisaki and K. Hirose, "Analysis of voice fundamental frequency contours for declarative sentences of Japanese", *Journal of Acoustical Society of Japan*, vol. 5, issue 4, pp. 233-241, 1984.

[34] H. Fujisaki, "From information to intonation", *Proceedings of the 1993 International Symposium on Spoken Dialogue*, Tokyo, Japan, 1993, pp. 7-18.

[35] H. Fujisaki, and S. Ohno, 1995. "Analysis and modeling of Fundamental Frequency Contours of English Utterances", *Proceedings of Eurospeech 1995*, Madrid, Spain, 1995, vol. 2, pp. 985-988.

[36] B. Gabioud, "Articulatory Models in Speech Synthesis", E. Keller (Ed.), *Fundamentals of speech synthesis and speech recognition*, West Sussex, England: John Wiley & Sons Ltd, 1994, pp. 215-230.

[37] R. Garside, G. Leech, and G. Sampson, *The computational analysis of English*, London, United Kingdom: Longman, 1987.

[38] D. Gibbon, R. Moore and R. Winski, (Eds.), *Handbook of standards and resources for spoken language systems, volume III, Spoken language system assessment*, New York, NY, USA: Mouton de Gruyter, 1998.

[39] R. Gillam, (1998), "A Rule-Based Approach to Number Spellout". Presented at the 1998 12th International Unicode/ISO 10646 Conference in Tokyo, Japan [Online], Available: http://www.concentric.net/ ~rtgillam/pubs/NumberSpellout.htm

[40] R. Gordon (ed.), *Ethnologue: Languages of the world*, Dallas, TX, USA: SIL International, 2005.

[41] P. A. V. Hall, R. Hudson, *Software without frontiers: A multi-platform, multi-cultural, multi-national approach*, New York, NY, USA: John Wiley & Sons, 1997.

[42] S. R. Hertz, "The Delta programming language: An integrated approach to non-linear phonology, phonetics and speech synthesis", *Papers in Laboratory Phonology I, Between the Grammar and Physics of Speech*, J. Kingston and M. E. Beckman (Eds.), Cambridge: Cambridge University Press, 1990, pp. 215-257.

[43] S. R. Hertz, "A modular approach to multi-dialect and multi-language speech synthesis using the delta system", *Proceedings of ESCA 1990 workshop on speech synthesis*, Autrans, France, 1990, pp. 225-228.

[44] S. R. Hertz, R. J. Younes and N. Zinovieva, "Language-Universal and Language-Specific components in the Multi-Language ETI-Eloquence Text-To-Speech System", *Proceedings of XIV International Congress of Phonetic Sciences*, San Francisco, CA, USA, 1999, vol. 3, pp. 2283-2286.

[45] S. R. Hertz. "Integration of rule-based formant synthesis and waveform concatenation: a hybrid approach to text-to-speech synthesis", *Proceedings of IEEE 2002 workshop on speech synthesis*, Santa Monica, CA, USA, 2002, pp. 87-90.

[46] J. Hirschberg, "Pitch accent in context: Predicting intonational prominence from text", *Artificial Intelligence*, vol. 63, issue 1-2, pp. 305–340, 1993.

[47] J. N. Holmes, "Influence of glottal waveforms on the naturalness of synthetic speech from a parallel-formant synthesizer", *IEEE Transactions on Audio and Electro-acoustics*, vol. 21, issue 3, pp. 298-305, 1973.

[48] J. N. Holmes, "Synthesis of natural-sounding speech using a formant synthesizer". B. Lindblom and S. Ohman (Eds.), *Frontiers of speech communication research*, London, England: Academic Press, 1979, pp. 275-285.

[49] J. Holmes and W. Holmes. *Speech Synthesis and Recognition*, 2nd edition, London, England: Taylor & Francis, 2001.

[50] B. Hook, *Write Portable Code: An Introduction to Developing Software for Multiple Platforms*, 1st edition, San Francisco, CA, USA: No Starch Press, 2005.

[51] C-H. Hu and J-H. Chen (1999), "Template-driven generation of prosodic information for Chinese concatenative synthesis", *Proceedings of ICASSP 1999*, Phoenix, AZ, USA, 1999, vol. 1, pp. 65-68.

[52] A. Hunt and A. Black, "Unit selection in a concatenative speech synthesis system using a large speech database", *Proceedings of ICASSP 1996*, Atlanta, GA, USA, 1996, pp 373-476.

[53] International Phonetic Association, *Handbook of the International Phonetic Association; A Guide to Use of the International Phonetic Alphabet*, Cambridge: Cambridge University Press, 1999.

[54] J. Iso-Sipilä, O. Viikki and M. Moberg, "Multi-Lingual Speaker-Independent Voice User Interface for Mobile Devices", *Proceedings of ICASSP 2006*, vol 1, Toulouse, France, 15-19 May, 2006, pp. 1081-1084.

[55] S. Kim, Y. Lee and K. Hirose, "Pruning of Redundant Synthesis Instances Based on Weighted Vector Quantization", *Proceedings of Eurospeech 2001*, Aalborg, Denmark, 2001, vol. 3, pp. 2231-2234.

[56] K. Kirchhoff, "Language characteristics", T. Schultz and K. Kirchhoff (Eds.), *Multilingual speech processing,* MA, USA: Academic Press, 2006. pp 5-31.

[57] E. Klabbers, J. P. H. van Santen and A. Kain, " The contribution of various sources of spectral mismatch to audible discontinuities in a diphone database", *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, issue 3, pp. 949-956, 2007.

[58] D. H. Klatt, “Software for a cascade/parallel formant synthesizer”, *Journal of the Acoustical Society of America*, vol. 67, issue 3, pp. 971-995, Mar. 1980.

[59] D. H. Klatt and L. C. Klatt, “Analysis, synthesis, and perception of voice quality variations among female and male talkers”, *Journal of the Acoustical Society of America*, vol. 87, issue 2, pp. 820-857, Feb. 1990.

[60] G. Kochanski and C. Shih, "Prosody modeling with soft templates", *Speech Communication*, vol. 39, issue 3-4, pp. 311-352, 2003.

[61] I. Kruijff-Korbayova, S. Ericsson, K. J. Rodriguez and E. Karagjosova, "Producing contextually appropriate intonation in an information-state based dialogue system", *Proceedings of 10th Conference of the European Chapter of the Association for Computational Linguistics*, Budabest, Hungary, 2003, pp. 227-234.

[62] J. Kupiec, "Robust part-of-speech tagging using a hidden Markov model", *Computer Speech and Language*, 6, pp. 225-242, 1992.

[63] Y. Laprie, “Snorri, a software for speech sciences”, *In MATISSE (Method and Tool Innovations for Speech Science Education)*, London, United Kingdom, 1999, pp. 89-92.

[64] I. Lehiste, *Suprasegmentals*, Cambridge, MA, USA: The MIT Press, 1970.

[65] K. Lenzo and A. Black (2000) "Diphone collection and synthesis", *Proceedings of ICSLP 2000*, Beijing, China, 2000.

[66] J. C. Liljencrants, "The OVE III speech synthesizer", *IEEE Transactions on Audio and Electroacoustics*, vol. 16, issue 1, pp. 137-140, 1968.

[67] J. Local, "Phonological structure, parametric phonetic interpretation and natural-sounding synthesis". E. Keller (Ed.), *Fundamentals of speech synthesis and speech recognition*, West Sussex, England: John Wiley & Sons Ltd, 1994, pp. 254-270.

[68] I. R. A. MacKay, *Phonetics: the Science of Speech Production*, 2nd edition, Boston, MA, USA: Allyn and Bacon, 1987.

[69] H. Mixdorff, "A novel approach to the fully automatic extraction of Fujisaki model parameters", *Proceedings of ICASSP 2000*, Istanbul, Turkey, 2000, vol. 3, pp. 1281-1284.

[70] M. Moberg and K. Pärssinen, "Comparing CART and Fujisaki Intonation Models for Synthesis of US-English Names", *Proceedings of Speech Prosody 2004*, Nara, Japan, 23-26 March, 2004, pp. 439-442.

[71] M. Moberg and K. Pärssinen, "Integrated Development Environment for a Multilingual Data Configurable Synthesis System", *Proceedings of Specom 2005*, Patras, Greece, 17-19 October, 2005, pp. 155-158.

[72] M. Moberg and K. Pärssinen, "Multilingual Rule-Based Approach to Number Expansion: Framework, Extensions and Application", *International Journal of Speech Technology*, Published on-line 22 September 2006 (Online First™).

[73] M. Moberg and K. Pärssinen, "Using Text-to-Speech in Mobile Phones", Fonetiikan päivät 2006 - The Phonetics Symposium 2006, Reijo Aulanko, Leena Wahlberg & Martti Vainio (Eds.), Publications of the Department of Speech Sciences, University of Helsinki, 53, Helsinki, Finland, 30-31 August, 2006, pp. 125-133.

[74] M. Moberg, K. Pärssinen and J. Iso-Sipilä, "Cross-Lingual Phoneme Mapping for Multilingual Synthesis Systems", *Proceedings of ICSLP 2004*, Jeju, Korea, 4-8 October, 2004, pp. 1029-1032.

[75] M. Moberg and O. Viikki, "Optimizing Speech Synthesizer Memory Footprint through Phoneme Set Reduction", *Proceedings of IEEE-SP Workshop on Speech Synthesis 2002*, Santa Monica, CA, 11-13 September, 2002, pp. 171-174.

[76] E. Navas, I. Hernaez, B. Etxebarria and J. Salaberria, "Modelling Basque intonation using Fujisaki's model and carts", *Proceedings of IEE Seminar on State of the Art in Speech Synthesis*, London, UK, 2000, vol. 3, pp. 3/1-3/6.

[77] A. T. Neel, "Formant detail needed for vowel identification", *Acoustics Research Letters Online (ARLO)*, vol. 5, issue 4, pp. 125-131, 2004.

[78] M. Ostendorf and I. Bulyko, "The use of speech recognition technology in speech synthesis", S. Narayanan and A. Alwan (Eds.), *Text-to-Speech Synthesis: New Paradigms and Advances*, New Jersey, USA: Prentice hall PTR, 2004, pp. 109-133.

[79] B. Pfister and C. Traber, "Text-to-speech synthesis: An introduction and a case study". E. Keller (Ed.), *Fundamentals of speech synthesis and speech recognition*, West Sussex, England: John Wiley & Sons Ltd, 1994, pp. 87-107.

[80] J. Pierrehumbert, *The phonology and phonetics of English intonation*, Ph. D. dissertation, MIT, 1980.

[81] K. L. Pike, *The intonation of American English*, Ann Arbor, Michigan: University of Michigan Press, 1945.

[82] K. Pärssinen and M. Moberg, "Multilingual Data Configurable Text-to-Speech System for Embedded Devices", *Proceedings of Multiling 2006*, Stellenbosch, South Africa, 9-11 April, 2006.

[83] K. Pärssinen and M. Moberg, "Evaluation of Perceptual Quality of Control Point Reduction in Rule Based Synthesis", *Proceeding of ICSLP 2006*, Pittsburgh, PA, USA, 17-21 September, 2006, pp. 2070-2073.

[84] K. Pärssinen, M. Moberg, M. Harju, and O. Viikki, "Development Challenges of a Text-to-Speech System for Multiple Languages", *Internationalizing W3C's Speech Synthesis Markup Language Workshop II*, 30-31 May 2006, Heraklion, Crete.

[85] K. Pärssinen, M. Moberg and M. Gabbouj, "Reading text messages using a text-to-speech system in Nokia Series 60 mobile phones: Usability study and application," *Technical report, Tampere University of Technology, Report (3):2006*, Tampere, Finland.

[86] S. Riis, M. Pedersen and J. Kåre, "Multilingual text-to-phoneme mapping", *Proceedings of Eurospeech 2001*, Aalborg, Denmark, 2001, pp. 1441-1444.

[87] R. L. Rivest, Learning decision lists, *Machine learning*, vol. 2, issue 3, pp. 229-246, 1987.

[88] P. Salvo Rossi, F. Palmieri and F. Cutugno. "A Method for Automatic Extraction of Fujisaki-Model Parameters", *Proceedings of Speech prosody 2002*, Aix-en-Provence, France, 2002, pp. 615-618.

[89] SAMPA (Speech Assessment Methods Phonetic Alphabet) homepage. University College London, Department of phonetics and linguistics. [Online]. Available: http://www.phon.ucl.ac.uk/home /sampa/home.htm

[90] T. Sejnowski and C. Rosenberg, "Parallel networks that learn to pronounce English text", *Complex Systems*, vol. 1, pp. 145-168, 1987.

[91] C. Shih and R. Sproat, "Issues in text-to-speech conversion for Mandarin". *Computational Linguistics and Chinese Language Processing*, vol. 1, issue 1, pp. 37–86, 1996.

[92] K. Sjölander and J. Beskow, "WaveSurfer - an Open Source Speech Tool", *Proceedings of ICSLP 2000*, Beijing, China, 2000, vol.4, pp. 464-467.

[93] A. Sluijter, E. Bosgoed, J. Kerkhoff, E. Meier, T. Rietveld, M. Swerts and J. Terken, "Evaluation of speech synthesis systems for Dutch in telecommunication applications", Proceedings of the 3rd ESCA Workshop on Speech Synthesis, Jenolan Caves, Australia, 1998, pp. 213-218.

[94] R. Sproat (ed.). *Multilingual Text-to-Speech Synthesis – The Bell Labs Approach*. Norwell, MA, USA: Kluwer Academic Publishers, 1998.

[95] R. Sproat, A. Black, S. Chen, S. Kumar, M. Ostendorf and C. Richards (1999), Normalization of non-standard words: WS'99 final report. An NSF workshop: Language engineering for students and professionals integrating research and education, Final technical report [Online]. Available: http://www.clsp.jhu.edu/ws99/projects/normal/

[96] M. Steedman, "Information structure and the syntax-phonology interface", *Linguistic Inquiry*, vol. 31, issue 4, pp. 649–689, 2000.

[97] K. N. Stevens, *Acoustic Phonetics*, Cambridge, Massachusetts, USA: The MIT Press, 1998.

[98] J. Suontausta and J. Tian, "Low memory decision tree method for text-to-phoneme mapping", *Proceedings of ASRU 2003*, Virgin Islands, USA, pp. 135-140.

[99] SVOX Ag, SVOX architecture description [Online]. Available: http://www.svox.com/SVOX-Architecture.aspx, 2006.

[100] A. Syrdal, G. Möhler, K. Dusterhoff, A. Conkie and A. Black, 1998. "Three methods of intonation modeling", *Proceedings of the 3rd ESCA Workshop on Speech Synthesis*, Jenolan Caves, Australia, 1998, pp. 305-310.

[101] M. Tartham and K. Morton. *Developments in Speech Synthesis*, Chichester, England: John Wiley & Sons Ltd., 2005, pp. 28.

[102] P. Taylor, A. Black and R. Caley, "The architecture of the Festival speech synthesis system", *Proceedings of the 3rd ESCA Workshop on Speech Synthesis*, Jenolan Caves, Australia, 1998, pp. 147-151.

[103] P. Taylor, "Analysis and synthesis of intonation using the Tilt model", *The Journal of the acoustical society of America*, vol. 107, issue 3, pp. 1697-1714, 2000.

[104] Text-to-Speech Accuracy Testing – 2005, *ASRNews*, Voice Information Associates. Available: http://www.asrnews.com/accuracyt.htm

[105] J. Tian, "Efficient Compression Method for Pronunciation Dictionaries", *Proceedings of ICSLP 2004*, Jeju Island, South Korea, 2004, pp. 2733-2736.

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

[107] K. Tokuda, H. Zen and A. Black. "An HMM-based approach to multilingual speech synthesis". S. Narayanan and A. Alwan (Eds.), *Text to speech synthesis,* New Jersey, USA: Prentice Hall, 2005, pp. 135-153.

[108] The Unicode consortium (2003), Conformance, 3.9 Unicode encoding forms. *The Unicode Standard, Version 4.0*. Boston, MA: Addison-Wesley [Online]. Available: http://www.unicode.org/versions/Unicode4.0.0/ch03.pdf#G31703

[109] M. Vainio, A. Suni, H. Järveläinen, J. Järvikivi, and V-V. Mattila. "Developing a speech intelligibility test based on measuring speech reception thresholds in noise for English and Finnish", *Journal of the Acoustical Society of America*, vol. 118, issue 3, pp. 1742-1750, Sep. 2005.

[110] O. Viikki, I. Kiss and J. Tian, "Speaker- and language independent speech recognition in mobile communication systems", *Proceedings of ICASSP 2001*, Salt Lake City, UT, USA, 2001, vol. 1, pp. 5-8.

[111] A. J. Viterbi. "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm", *IEEE Transactions on Information Theory*, vol. 13, issue 2, pp. 260–269, Apr. 1967.

[112] W3C (2004, September), Speech Synthesis Markup Language (SSML), version 1.0, D. C. Burnett, M. R. Walker, A. Hunt (Eds.), *W3C Recommendation*. [Online]. Available: http://www.w3.org/TR/speech-synthesis/

[113] J. Wouters and M. W. Macon, " Control of spectral dynamics in concatenative speech synthesis", *IEEE Transactions on Speech and Audio Processing*, vol 9, issue 1, pp. 30-38, 2001.

[114] G. Xydas, G. Karberis and G. Kouroupetroglou (2004). "Text normalization for the pronunciation of non-standard words in an inflected language", G.A. Vouros and T. Panayiotopoulos (Eds.), *Methods and Applications of Artificial Intelligence: Third Hellenic Conference on AI, SETN 2004*, Samos, Greece, May 5–8, 2004, *Proceedings / Lecture Notes in Artificial Intelligence (LNAI)*: Vol. 3025. Berlin Heidelberg: Springer-Verlag, pp. 390–399.

[115] G. Yule, *The study of language*, 2nd edition. Cambridge, United Kingdom: Cambridge University Press, 1996.

[116] D. Öhlin and R. Carlson. "Data-driven formant synthesis", *Proceedings of FONETIK 2004*, Department of Linguistics, Stockholm University. [Online]. Available: http://www.ling.su.se/fon/fonetik_2004/ohlin_carlson_fonetik2004.pdf

Publications

Tampereen teknillinen yliopisto
PL 527
33101 Tampere

Tampere University of Technology
P.O. Box 527
FIN-33101 Tampere, Finland