Universität Bielefeld
Faculty of Technology
Applied Informatics Group

Implementation and Evaluation of Acoustic Distance Measures for Syllables

Master Thesis

in Computer Science in the Natural Sciences
at the Faculty of Technology
Bielefeld University

Author: Christian Munier
cmunier@techfak.uni-bielefeld.de

Supervisors: Dipl.-Inf. Lars Schillingmann
Dr.-Ing. Britta Wrede

March 2, 2011

Contents

Abstract

In this work, several acoustic similarity measures for syllables are motivated and successively evaluated. The Mahalanobis distance as local distance measure for a dynamic time warping approach to measure acoustic distances is a measure that is able to discriminate syllables and thus allows for syllable classification with an accuracy that is common to the classification of small acoustic units (60% for a nearest neighbor classification of a set of ten syllables using samples of a single speaker).

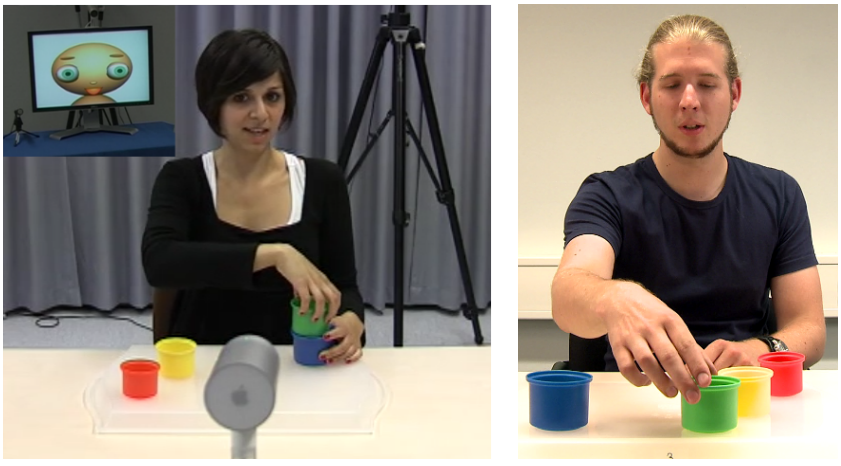
This measure can be improved using several techniques that however impair the execution speed of the distance measure (usage of more mixture density components for the estimation of covariances from a Gaussian mixture model, usage of fully occupied covariance matrices instead of diagonal covariance matrices). Through experimental evaluation it becomes evident that a decently working syllable segmentation algorithm allowing for accurate syllable border estimations is essential to the correct computation of acoustic distances by the similarity measures developed in this work. Further approaches for similarity measures which are motivated by their usage in timbre classification of music pieces do not show adequate syllable discrimination abilities.

1 Motivation

An ambitious goal in robotics is the creation of robots that can assist humans in various situations, having learned their behavior and actions from humans on their own. The process of learning in general can be identified with three levels of complexity:

1. learning from reflection of previously gained knowledge,
2. learning from reception and observation of processes in the environment,
3. learning from an explicit demonstration of a specific action or fact.

It is easily imaginable that the third level is both the distinctest in terms of identifying what is to be learned as well as the easiest to accomplish when aiming at the imitation of a specific action. This way of learning is called *tutoring scenario* [BS02] or *imitation learning*. From the point of view of the speech recognition involved in processing of the robot sensor data the tutoring scenario is particularly convenient, because the speech employed for explanations tends to have beneficial acoustic attributes. In the special case of *child-directed speech (CDS)* [Bat+08], also referred to as *motherese* [Kit03], explanatory speech can be characterized by showing exaggerative prosody, hyper-articulation, raised pitch, broader pitch range, slower speech rate [Bat+08] and longer pauses [Kit03].



(a) (b)

Figure 1.1: Tutors demonstrate how to stack cups in interaction with a robot simulation; (b) (source: [Sch+09]).

Let us now imagine a tutoring scenario, in which the tutor (human) explains and shows the learner (robot) how to stack four cups of different size and color (blue, green, yellow, red) (see Figure 1.1, cf. [NR07; Roh+06]).

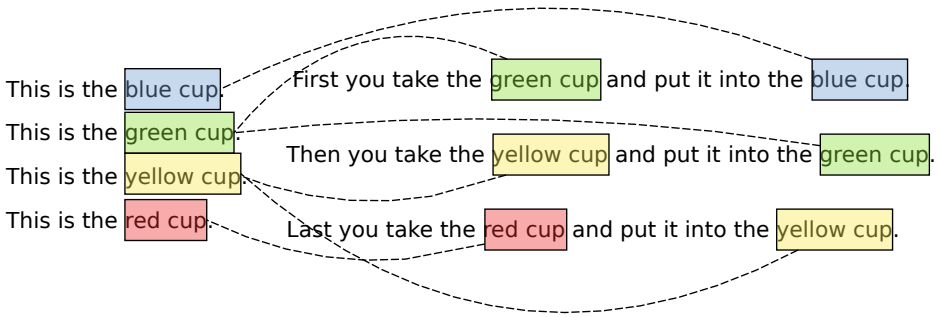


Figure 1.2: Example of an explanation in the cup stacking scenario. The dashed lines indicate the linkage between utterance parts.

The explanation could for example look like this:

> "This is the blue cup. This is the green cup. This is the yellow cup. First you take the green cup and put it into the blue cup. Then you take the yellow cup and put it into the green cup. Last you take the red cup and put it into the yellow cup."

This explanation exhibits two types of sentences. The first four sentences create an abstract representation of the objects through a linkage of visual (images) and acoustic information (speech signal of the tutor). The remaining sentences describe actions using these objects, referring to their abstract representation through the context of visual and acoustic information (see Figure 1.2).

Hence to identify the linkage between action and abstract object representation it is necessary for the acoustic processing in the robot to compare the acoustic representation of the cups from the speech signals. In general, in the tutoring scenario there are parts of utterances referring to parts of previously uttered sentences.

A promising approach is to perform the acoustic similarity test needed to identify corresponding utterance parts on syllable level. This is motivated by the fact that syllables form a perceptually and acoustically coherent unit, moreover facilitating the consideration of pronunciation variations [Gan+97].

This work explores acoustic distance measures for syllables, aiming at a measure which is particularly well suited for application in the tutoring scenario. The following chapter gives an introduction to established speech recognition methods in order to provide a general understanding of the course of action that needs to be taken when transforming an acoustic speech signal to a meaningful representation in a computer.

2 Introduction

In this chapter an introduction to common methods for speech recognition is given, in order to provide a general understanding of the proceeding for the transformation of an acoustic speech signal to a semiotic representation in a computer.

The ability to recognize and understand speech plays an important role in human society, because a large part of human communication is conveyed through speech production and speech recognition in humans. The ability of correctly uttering and recognizing spoken word sequences is naturally learned in the early years of a human's life and thereafter capable of recognizing speech under almost any circumstances. It is thus desirable to equip computers and robots with the ability to recognize human speech. This facilitates the usage strongly: Users can then interact with a computer or robot in a by far more natural way than by interaction via a keyboard or mouse, so that they are ideally able to interact without previous learning of how to use the system.

2.1 Automatic Speech Recognition

Automatic speech recognition (ASR) aims at transforming a spoken utterance into a symbolic representation. First, the speaker articulates a sequence w of words, producing an acoustic speech signal. In automatic speech recognition theory this is called *coding*. The speech signal is recorded and digitalized; then a sequence X of feature vectors is computed as representation (section 2.2).

Formally, speech recognition tries to find the optimal word sequence $\hat{w}$, given a sequence of acoustic observations $X = x_1, x_2, \ldots, x_T$ with T being the observation length or the number of feature vectors. This leads to the fundamental equation of speech recognition:

$$\hat{w} = \arg\max_{w} P(w|X),$$

where $P(w|X)$ describes the probability of the word sequence w being uttered given the acoustic observation sequence X. When Bayes' theorem is applied, the equation can be written as:

$$\hat{w} = \arg\max_{w} \frac{P(X|w)P(w)}{P(X)}.$$

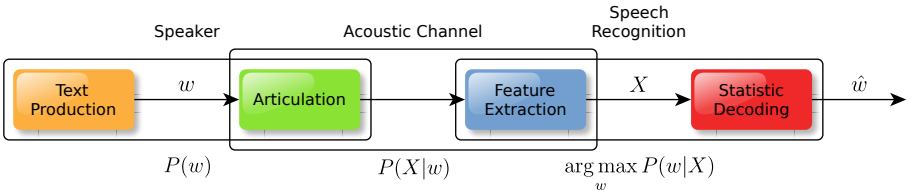


Figure 2.1: Channel model of speech production and recognition (cf. [Fin07; Jel98]).

Since $P(X)$ is constant when searching for the argument w that maximizes the equation, it can be omitted:

$$\hat{w} = \arg\max_{w} P(X|w)P(w).$$

This equation is commonly referred to as the basic speech recognition equation, describing the separation between acoustic model and language model which is common for most speech recognition systems. The process of finding the most probable word sequence is called *decoding*.

$P(X|w)$ is estimated by the acoustic model. The acoustic model denotes the probability of an acoustic observation X given a word sequence w. Commonly used acoustic models are *hidden Markov models* (*HMMs*; section 2.3) which are able to model the statistical relation between word and observation sequence. $P(w)$ is estimated by the language model, denoting the probability of producing a specific sequence w of words. A widely used language model is the *n-gram model* (section 2.4).

Together the aforementioned steps form the channel model of speech production and recognition (Figure 2.1). The amount of combinatorially possible word sequences depends on the size of the lexicon of possible words and grows exponentially with the sequence length. So evaluation of the speech recognition equation by means of an exhaustive search is not feasible. To reduce the solution space to an acceptable size, conventional graph traversal algorithms like A*, beam search and dynamic programming are typically used (see also [Fin07; ST95]).

2.2 Feature Extraction

To be digitally represented in a computer the signal to be recorded is *sampled* with a specific frequency, reducing the infinite amount of analogue information to a finite amount of information in its digital representation. A widely used sample frequency is 16 kHz, because it is assumed that the essential amount of information in speech concentrates in a range of 8 kHz [ST95; HAH01]. The sampled data is then *quantized* to be stored as integer values of usually 8 or 16 bit. In theory this basic representation could already be used as acoustic features for speech recognition. Because of the high complexity and the huge amount of information contained in the representation this is in practice not operable.

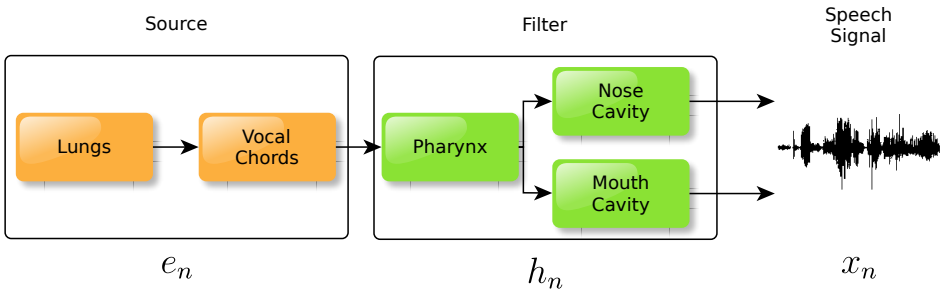


Figure 2.2: Source-filter model of speech production.

2.2.1 Short Term Analysis

It is necessary to reduce the amount of information to get useful and tractable acoustic features. One approach is to rely on models for speech production, i.e. the *source-filter model* [Fan60] which identifies fundamental parameters that define how a speech signal is produced. The source-filter model divides human speech production in two components, *source* and *filter*. The source is anatomically represented by the lungs and the vocal chords, generating an acoustic excitation signal. This signal is then modified by the filter, represented by the vocal tract (i.e. the pharynx, the mouth and the nose cavities). In the vocal tract the excitation signal leads to resonances which temporally change while speaking. The application of a filter to the excitation signal is mathematically represented by a convolution. If e_n denotes the excitation signal and h_n the filter, the final speech signal is defined as

$$x_n = e_n * h_n.$$

For a diagram of this separation of source and filter see Figure 2.2.

[HAH01] shows that the components e_n and h_n can be separated using a homomorphic transformation $\hat{x}_n = D(x_n)$ that converts the convolution into a sum $\hat{x}_n = \hat{e}_n + \hat{h}_n$. The *cepstrum* is introduced as one such homomorphic transformation allowing separation of source from filter. This separation is useful because the information gained from the filter coefficients is more meaningful to the information conveyed by the speech signal rather than the characteristics of the glottal excitation. It is possible because in the cepstral representation there exists a N so that $\hat{h}_n \approx 0$ for $n \geq N$ and $\hat{e}_n \approx 0$ for $n < N$. The *real cepstrum* of a signal x_n is defined as

$$c_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} \ln |X(e^{i\omega})| e^{i\omega n} \, d\omega,$$

$X(e^{i\omega})$ being the Fourier transform of the signal x_n. The discrete Fourier transformation is defined only for periodic signals. Speech however is clearly not periodic. It can yet be assumed that the signal is approximatively stationary for short time periods, so it is possible to extract useful characteristics from small windows in the signal [ST95].

The speech segment marked by a window is called *frame*. The windowing process is identified by the width of the window (*frame size*), the offset between successive windows (*frame shift*) and the window shape. Each frame is multiplied with a window function. A basic approach would be to use a rectangular window function. This however would cause problems because the signal would be abruptly cut off at the window boundaries, creating discontinuities that would make the Fourier transformation not applicable. Instead, a window function is needed that shrinks the signal amplitude toward zero at the window boundaries so that the signal can be periodically continued. One such window function that is commonly used is the *Hamming window*, defined as $w_n = 0.54 - 0.46\cos(\frac{2\pi n}{T})$ for $0 \leq n \leq T-1$ and $w_n = 0$ otherwise. Since the signal is damped at the boundaries of each window a frame shift value is chosen that effects an overlap between successive windows. Often, a frame size of 20 ms and a frame shift of 10 ms are used. The short time Fourier transform of the m-th frame is defined as

$$X_m(e^{i\omega}) = \sum_{n=-\infty}^{\infty} w_n^m x_n e^{-i\omega n}$$

if w_n^m designates the window function for frame m.

2.2.2 Mel-Frequency Cepstral Coefficients

The short term analysis method presented in the previous section shows still potential of improvement. Human hearing is not equally sensitive in respect to different signal frequencies. In fact, humans are less sensitive to small differences at high frequencies than at low frequencies with the perceptual sensitivity being approximately logarithmic above a limit frequency of about 1000 Hz. Therefore the spectrum as output of the Fourier transformation is warped onto the *mel scale* [SVN37]. The *mel frequency* $B(f)$ is computed from the acoustic frequency f by $B(f) = 1127\ln(1 + \frac{f}{700})$. To be even more accurate it can be observed that the sensitivity in human hearing is organized in frequency bands. A common approach is to model this behavior through a bank of triangular filters that are equally spaced on the mel scale (thus perceptually equidistant) [HAH01; ST95]. The filters calculate the average spectrum around their center frequencies while increasing in bandwidth as the center frequencies increase. In [HAH01], the M Filters ($m = 1, \ldots, M$) H_k^m are defined as

$$H_k^m = \begin{cases} \frac{2(k-f_{m-1}))}{(f_{m+1}-f_{m-1})(f_m-f_{m-1})} & f_{m-1} \leq k \leq f_m \\ \frac{2(f_{m+1}-k))}{(f_{m+1}-f_{m-1})(f_{m+1}-f_m)} & f_m \leq k \leq f_{m+1} \\ 0 & k < f_{m-1}, k > f_{m+1} \end{cases},$$

$$f_m = \frac{N}{F_s} \cdot B^{-1}\left(B(f_l) + \frac{m}{M+1}(B(f_h) - B(f_l))\right),$$

with f_m the center frequencies, f_l the lowest, f_h the highest frequency of the filter bank, f_s the sampling frequency, N the size of the fast Fourier transform, B the frequency projection onto the mel scale and B^{-1} its inverse. The discrete cosine transform of the filter outputs S_m is called *mel-frequency cepstrum*. The *mel-frequency*

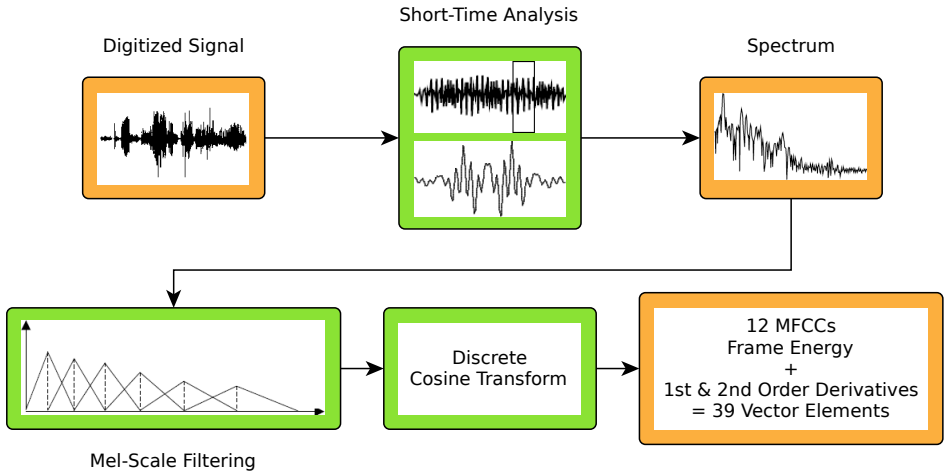


Figure 2.3: Diagram of the steps necessary to compute the mel-frequency cepstral coefficients (MFCCs).

cepstral coefficients (MFCCs) c_n are then given by

$$c_n = \sum_{m=0}^{M-1} S_m \cos\left(\pi n \frac{2m-1}{2M}\right) \quad 0 \leq n < M,$$
$$S_m = \ln\left(\sum_{k=0}^{N-1} |X(k)|^2 H_k^m\right) \quad 0 < m \leq M.$$

In [DM80] it is shown that incorporating the human auditory system in this way leads to an improvement in speech recognition performance. The mel-frequency cepstrum is no longer a homomorphic transformation like the basic cepstrum, yet approximately homomorphic for filters with smooth transfer function. More recently, in [TSB05] Terasawa, Slaney, and Berger show that MFCC representations of speech decently match their perceptual representation. The variance for different cepstral coefficients shows the useful property of being roughly uncorrelated which is a beneficial when building models from a database of feature vectors. In most implementations the number of triangular filters in the filter bank is between 24 and 40.

Usually only the first 12 cepstral coefficients are taken for the final feature vectors because they represent information solely from the vocal tract, cleanly separated from the excitation characteristics of the glottal source (cf. source-filter model). The first 12 cepstral coefficients are often supplemented by the *energy* of the corresponding speech frame. The energy is the sum of the power values of the frame over time. Moreover, since the speech signal is not constant from frame to frame, an useful cue for reasonable feature vectors is the modeling of temporal dynamics. In common applications, the 12 cepstral coefficients and the frame energy are complemented by their first-order derivatives (velocity) and their second-order derivatives (acceleration) so in total then there are 39 MFCC features. The steps necessary to compute the mel-frequency cepstral coefficients are visualized in Figure 2.3.

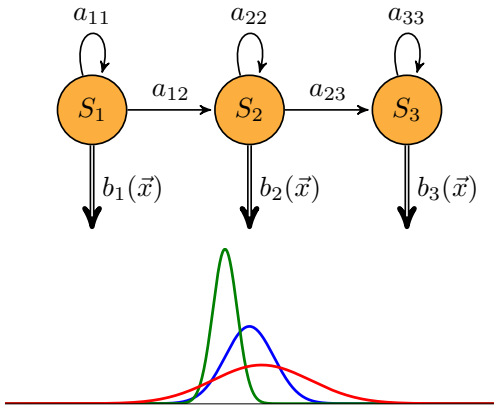


Figure 2.4: Diagram of a typical semi-continuous linear hidden Markov model.

2.3 Acoustic Modeling with Hidden Markov Models

Hidden Markov models (HMMs) are statistical models which are able to describe speech samples via a discrete time series of observed data. In speech recognition this discrete time series is represented by the sequence of feature vectors generated for a speech signal as preprocessing. Formally, a hidden Markov model describes a two-tiered stochastic process.

The first tier is a discrete Markov chain that distinguishes itself by a state sequence $s_1, s_2, \ldots, S_T$, resulting from the transition probabilities on a finite state set $S = \{S_1, S_2, \ldots, S_N\}$ with $s_t \in S$. The transition probabilities form a matrix $\mathbf{A} = (a_{ij})$ with $a_{ij} = P(s_{t+1} = S_j | s_t = S_i)$. The stochastic process is *stationary* because the state transitions do not depend on the time t. It is *causal* because the probability distribution of the random variable s_t only depends on states in the past. In most applications of hidden Markov models it even depends on only on the immediate predecessor state (*simple process*). The starting state of the process originates from a probability distribution $\boldsymbol{\pi} = (\pi_i)$ with $\pi_i = P(q_1 = S_i)$.

The second tier comes into existence through the concept that a hidden Markov model generates an emission in every state, resulting in an emission sequence of $x_1, x_2, \ldots x_T$. The emissions x_t come from a finite emission space $X = \{X_1, X_2, \ldots X_M\}$ with $x_t \in X$. In speech recognition, this emission space is provided by the feature vectors of the speech signals that are modeled by this HMM. The probability to emit a specifc feature vector $\mathbf{x}$ in the state S_i is described by a distribution $\mathbf{B} = (b_i(\mathbf{x}))$ with $b_i(\mathbf{x}) = P(\mathbf{x}_t = x | s_t = S_i)$.

The emission space is often modeled by mixture densities since they are capable of modeling probability distributions with multiple agglomeration centers arbitrarily well given an infinite number of normal distributions. A linear combination of those normal distributions forms then the mixture density. The number of normal distributions is limited to a certain number resulting in a suitable approximation of the intrinsic distribution of the data. Often *semicontinuous HMMs* are used in speech recognition. In such HMMs not each individual state is assigned a complete mixture density. Instead, a single mixture density is shared between all states in the model. A mixture

density is defined by $b_i(\mathbf{x}) = \sum_{k=1}^{M} c_{ik}\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$. Each of the involved normal distributions is represented by a mean vector $\boldsymbol{\mu}_k$ and a covariance matrix $\boldsymbol{\Sigma}_k$.

A hidden Markov model can thus be described by a tuple $\lambda = (\mathbf{A}, \mathbf{B}, \boldsymbol{\pi})$. The state sequence cannot be observed (*hidden*), as opposed to the simulation of a basic Markov chain. So the goal is to draw conclusions from knowing the emission sequence to identify the most probable state sequence. For a diagram of a typical linear HMM see Figure 2.4.

In speech recognition hidden Markov models are widely used to define acoustic models. The functionality of HMMs is characterized by their structure (i.e. the number of states and their connection topology) and their statistic parameters (i.e. the transition probabilities and the mean vectors and covariance matrices of the emission distributions). Being in fact initially unknown, these parameters can be estimated with a transliterated training set of sample speech signals. This principle of automated learning from training data yields the capability of adapting a speech recognition system arbitrarily well to for example specific speakers, dialects or lexicons given the availability of comprehensive training data. Often, a three-state linear HMM is created for every phoneme. The main idea behind this is that the first state then models the influence of the previous phoneme to the current one; the second state is to describe the stable part of the phoneme and the third state models the influence of the next phoneme. Continuative and more extensive descriptions can be found in [Fin07; ST95].

2.4 Language Modeling

The distribution $P(w)$ from the speech recognition equation in section 2.1 forms the statistical model of restrictions to possible word sequences in a grammatical sense. It describes statistically how individual words are combined to form sentences. Most approaches to handle the distribution $P(w)$ are based on a factorization in conditional probabilities

$$\begin{aligned} P(w) &= P(w_1, \ldots, w_T) \\ &= P(w_1) \cdot P(w_2|w_1) \cdot P(w_3|w_1 w_2) \cdot \prod_{t=4}^{T} P(w_t|w_1 \cdots w_{t-1}) \\ &= \prod_{t=1}^{T} P(w_t|w_1 \cdots w_{t-1}) \approx \prod_{t=1}^{T} P(w_t|w_{t-n+1} \cdots w_{t-1}) \end{aligned}$$

Evaluation of the unapproximated factorization formula would mean the next word probability had to be evaluated for every possible sequence $w_1 \cdots w_{t-1}$. This is not operable since the number of possible combinations explodes and the number of occurrences in the training data diminishes.

N-grams however approximate the probability by only considering the $n-1$ previous words. In practice, *bigram* probabilities $P(w_t|w_{t-1})$ and *trigram* probabilities

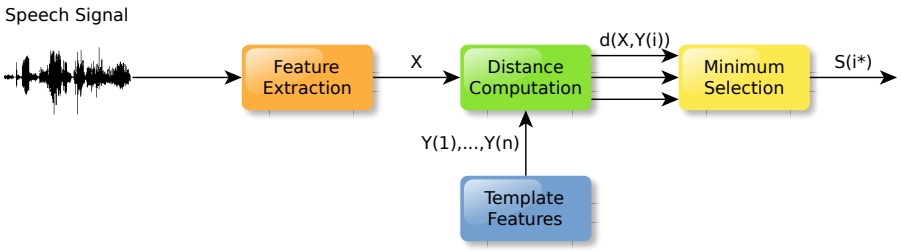


Figure 2.5: Diagram of a template based speech recognition approach where the best matching template from a database for a reference speech sample is to be found.

$P(w_t|w_{t-2}w_{t-1})$ are often used because their evaluation is still combinatorially feasible. However in general, even for a small n not every of the word sequences denoted by the conditional probabilities involved will occur even in a very large set of training data. Normally the model would then assign them zero probability. To be robust, the model has to be adapted so that probability mass is shifted from seen events to those unseen events. Two commonly used approaches are *discounting* and *backing-off* (both explained in [ST95]).

2.5 Template Based Recognition

In contrast to speech recognition strategies based on hidden Markov models, template based speech recognition (TBSR) does not use statistical models. Instead of replacing each acoustic unit with a HMM the preprocessed data itself can be seen as model. So instead of using HMMs as acoustic models, speech signals are directly compared to examples of the relevant acoustic units (e.g. words, syllables or phonemes) from a database. These acoustic units are called *templates*. Figure 2.5 shows a diagram of a template based speech recognition approach where the best matching template from a database for a reference speech sample is to be found.

The standard algorithm to perform the comparison of input and template data is the *dynamic time warping algorithm* (DTW). It allows to measure similarity between two sequences of acoustic features with different length. To compare the elements (i.e. feature vectors) from both sequences it needs a local distance measure. With distances of all element combinations a distance matrix is built, which is then subject to finding an optimum warping path through it. Dynamic time warping is described in detail in the following section.

2.6 Dynamic Time Warping

When attempting to compute a distance between two speech samples (i.e. two feature vector sequences each representing a syllable) in the general case they do not have the

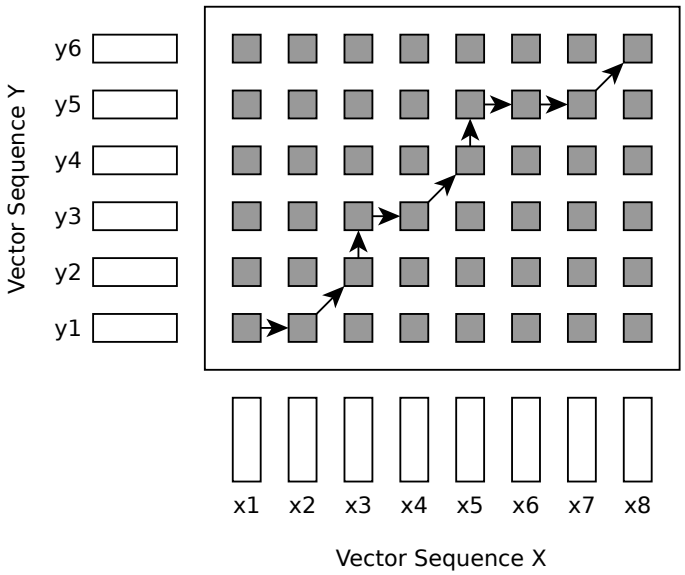


Figure 2.6: Alignment of two sequences of feature vectors in dynamic time warping. The arrows indicate the warping path with minimum accumulated distances.

same length. This is particularly true even for two samples that are both representatives of the same syllable. Due to pronunciation variations, this also results in an inherent variation of the speaking rate during utterance of a sample causing a nonlinear distortion of the speech time axis. Normalizing this fluctuation temporally in order to eliminate this distortion has been a problem of major interest in research for isolated speech segment recognition. In the beginning linear normalization techniques that eliminated timing differences by linear transformation of the time axis were examined. However these approaches were insufficient for the often highly fluctuating speaking rate. In consequence a dynamic programming approach was originally proposed by Sakoe and Chiba in 1971 [SC78]. This dynamic programming approach was later called *dynamic time warping algorithm (DTW)*. Variations to the original approach have been described for example by Itakura in [Ita75] and by Myers, Rabiner, and Rosenberg in [MRR80].

2.6.1 Idea

Formally we define a reference speech sample $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_{L_\mathbf{x}})$ with length $L_\mathbf{x}$ and a test speech sample $\mathbf{y} = (\mathbf{y}_1, \ldots, \mathbf{y}_{L_\mathbf{y}})$ with length $L_\mathbf{y}$, both consisting of a sequence of feature vectors $\mathbf{x}_i$ $(1 \leq i \leq L_\mathbf{x})$ and $\mathbf{y}_j$ $(1 \leq j \leq L_\mathbf{y})$ respectively. The distance between these sequences can be computed through searching for the lowest accumulated distance on a path through a local distance matrix, as shown in Figure 2.6.

The elements of the matrix represented in this Figure are the local distances between individual feature vectors of both sequences. The distance between $\mathbf{x}$ and $\mathbf{y}$ is then a sum of the distances encountered on a path starting in $(\mathbf{x}_i, \mathbf{y}_1)$ and ending in

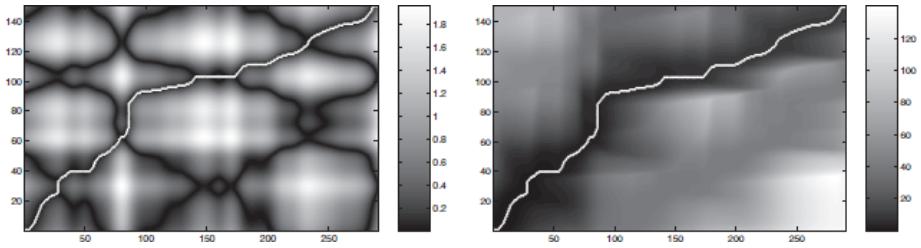


Figure 2.7: (a) Local dynamic time warping distances between individual feature vectors and (b) accumulated dynamic time warping distance, each including optimum warping path (source: [MÖ07]).

$(\mathbf{x}_{L_\mathbf{x}}, \mathbf{y}_{L_\mathbf{y}})$. This concept of an accumulated distance $D_{\psi_\mathbf{x},\psi_\mathbf{y}}$ can be formulated as

$$D_{\psi_\mathbf{x},\psi_\mathbf{y}}(\mathbf{x},\mathbf{y}) = \sum_{k=1}^{L} d(\mathbf{x}_{\psi_\mathbf{x}(k)}, \mathbf{y}_{\psi_\mathbf{y}(k)}) \cdot \frac{m_k}{M_\psi},$$

$\psi_\mathbf{x}(k)$ and $\psi_\mathbf{y}(k)$ being warping functions that represent the position at step k in the sequences $\mathbf{x}$ and $\mathbf{y}$ respectively; m_k being a weighting coefficient, M_ψ a normalization factor and L the number of total steps in the resulting alignment. The warping functions have to be chosen so that they result in the minimum accumulated distance. The distance D between $\mathbf{x}$ and $\mathbf{y}$ is therefore

$$D(\mathbf{x},\mathbf{y}) = \min_{\psi_\mathbf{x},\psi_\mathbf{y}} D_{\psi_\mathbf{x},\psi_\mathbf{y}}(\mathbf{x},\mathbf{y}),$$

which is called *dynamic time warping distance (DTW distance)*. Figure 2.7 visualizes the local DTW distance for an example alignment of two samples and the resulting accumulated DTW distance.

2.6.2 Local Path Constraints

The warping function models the fluctuation of the time axis of a speech segment. It must therefore preserve linguistically essential structures like continuity and monotonicity. In order to construct a valid distance measure, the warping path has to be constrained by several properties:

- **Boundary Constraints:** The path through the distance matrix has to start at the first frame and end at the last frame of the samples being compared. This implies that it is essential that the underlying speech segmentation that defines the syllable boundaries in the speech stream has to be correct for the dynamic time warping to work correctly. The warping path is constrained by

$$\begin{aligned} \psi_\mathbf{x}(1) &= 1, & \psi_\mathbf{x}(L) &= L_\mathbf{x}, \\ \psi_\mathbf{y}(1) &= 1, & \psi_\mathbf{y}(L) &= L_\mathbf{y}. \end{aligned}$$

- **Monotonicity:** The path can only move forward through the distance matrix.

It is constrained by:

$$\psi_{\mathbf{x}}(k+1) \geq \psi_{\mathbf{x}}(k),$$
$$\psi_{\mathbf{y}}(k+1) \geq \psi_{\mathbf{y}}(k).$$

- **Local Continuity:** Important information in both compared speech samples should be preserved. It should therefore be impossible to skip multiple frames in a row. There were several approaches to define these constraints. Among others, Sakoe and Chiba [SC78] proposed:

$$\psi_{\mathbf{x}}(k+1) - \psi_{\mathbf{x}}(k) \leq 1,$$
$$\psi_{\mathbf{y}}(k+1) - \psi_{\mathbf{y}}(k) \leq 1.$$

  A different variant was proposed by Itakura [Ita75]:

$$\psi_{\mathbf{x}}(k+1) - \psi_{\mathbf{x}}(k) = 1,$$
$$0 \leq \psi_{\mathbf{y}}(k+1) - \psi_{\mathbf{y}}(k) \leq 2,$$
$$\psi_{\mathbf{y}}(k+2) > \psi_{\mathbf{y}}(k).$$

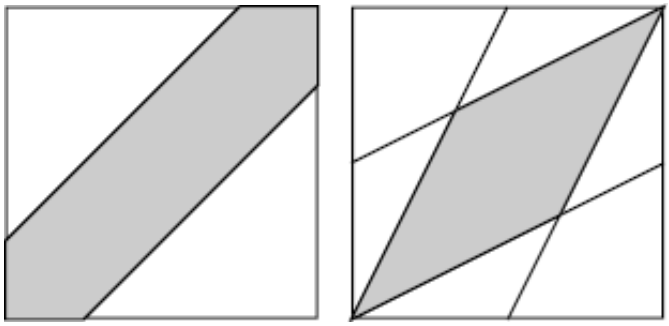
Together, these local constraints define the trajectory of the warping path through the distance matrix.

2.6.3 Distance Computation

The naive approach to compute the dynamic time warping distance is to compute the accumulated distance $D_{\psi_{\mathbf{x}},\psi_{\mathbf{y}}}(\mathbf{x},\mathbf{y})$ for every possible warping function $\psi_{\mathbf{x}}$ and $\psi_{\mathbf{y}}$ and then to take the minimum. However, more efficient approaches exist to compute the DTW distance by using dynamic programming techniques, as proposed by Sakoe and Chiba [SC78]. *Dynamic programming* is a method to solve complex problems by dividing them into smaller subproblems which have to exhibit properties of so-called overlapping subproblems. A problem has *overlapping subproblems* if it can be divided into smaller subproblems that can be reused several times in order to solve the main problem.

In the case of the dynamic time warping distance computation between speech segments the overlapping subproblems are obvious. If P_{ij} denotes the partial optimum path in the local distance matrix up to position (i,j) and (i,j) is in the global optimum path P, then P_{ij} is also part of P. This taken into account, the distance problem can easily be written as a recursive formulation. For the constraints proposed by Sakoe and Chiba this recursion is given by

$$D(\mathbf{x}^{\psi_{\mathbf{x}}(t)},\mathbf{y}^{\psi_{\mathbf{y}}(t)}) = \min\left\{\begin{array}{ll} D(\mathbf{x}^{\psi_{\mathbf{x}}(t)-1},\mathbf{y}^{\psi_{\mathbf{y}}(t)}) & +\gamma_0 \cdot d(\mathbf{x}_{\psi_{\mathbf{x}}(t)},\mathbf{y}_{\psi_{\mathbf{y}}(t)}) \\ D(\mathbf{x}^{\psi_{\mathbf{x}}(t)-1},\mathbf{y}^{\psi_{\mathbf{y}}(t)-1}) & +\gamma_1 \cdot d(\mathbf{x}_{\psi_{\mathbf{x}}(t)},\mathbf{y}_{\psi_{\mathbf{y}}(t)}) \\ D(\mathbf{x}^{\psi_{\mathbf{x}}(t)},\mathbf{y}^{\psi_{\mathbf{y}}(t)-1}) & +\gamma_2 \cdot d(\mathbf{x}_{\psi_{\mathbf{x}}(t)},\mathbf{y}_{\psi_{\mathbf{y}}(t)}) \end{array}\right\},$$



(a) (b)

Figure 2.8: (a) Sakoe-Chiba band and (b) Itakura parallelogram to which the warping path is confied (source: [MÖ7]).

with starting condition

$$D(\mathbf{x}^1, \mathbf{y}^1) = d(\mathbf{x}_1, \mathbf{y}_1).$$

The coefficients γ_0, γ_1 and γ_2 allow for a different weighting of the path options. For the constraints proposed by Itakura the recursion is given by

$$D(\mathbf{x}^{\psi_\mathbf{x}(t)}, \mathbf{y}^{\psi_\mathbf{y}(t)}) = \min \left\{ \begin{array}{ll} D(\mathbf{x}^{\psi_\mathbf{x}(t)-1}, \mathbf{y}^{\psi_\mathbf{y}(t)}) & +\gamma_0 \cdot d(\mathbf{x}_{\psi_\mathbf{x}(t)}, \mathbf{y}_{\psi_\mathbf{y}(t)}) \\ D(\mathbf{x}^{\psi_\mathbf{x}(t)-1}, \mathbf{y}^{\psi_\mathbf{y}(t)-1}) & +\gamma_1 \cdot d(\mathbf{x}_{\psi_\mathbf{x}(t)}, \mathbf{y}_{\psi_\mathbf{y}(t)}) \\ D(\mathbf{x}^{\psi_\mathbf{x}(t)}, \mathbf{y}^{\psi_\mathbf{y}(t)-2}) & +\gamma_2 \cdot d(\mathbf{x}_{\psi_\mathbf{x}(t)}, \mathbf{y}_{\psi_\mathbf{y}(t)}) \end{array} \right\},$$

with starting conditions

$$\begin{aligned} D(\mathbf{x}^1, \mathbf{y}^1) &= d(\mathbf{x}_1, \mathbf{y}_1), \\ D(\mathbf{x}^1, \mathbf{y}^2) &= \gamma_2 \cdot d(\mathbf{x}_1, \mathbf{y}_2), \\ D(\mathbf{x}^1, \mathbf{y}^j) &= +\infty \, \forall j > 2. \end{aligned}$$

2.6.4 Further Considerations

The type of local continuity constraints including the respective weighting coefficients in the recursion formula for the warping path are subject to careful selection, since they directly influence the possible warping paths and thus the alignment of the two compared feature vector sequences.

In a large part of work related to dynamic time warping further constraints are discussed that globally constrain the warping function and thus confine the warping path to a specific region in the distance matrix. Two famous approaches were proposed by Sakoe and Chiba and Itakura.

In [SC78] Sakoe and Chiba impose further constraints in order to prohibit warping paths from deviating to much from a linear warping function (i.e. the diagonal path through the distance matrix) so that highly distorted alignments are inhibited. This constraint should correspond to the fact that in usual cases time axis fluctuations

never cause a too excessive timing difference [SC78]. They impose a so-called *adjustment window condition* which confines the warping path to band of a specific width that runs along the main diagonal of the distance matrix. The area to which the warping path is hereby restricted is called *Sakoe-Chiba band* (see Figure 2.8(a)).

In [Ita75] Itakura proposes a constraint to the slope of the warping path in order to prohibit to steep or to gentle gradients. The slope of the warping path is confined to lie between the values $\frac{1}{S}$ and S, S being a slope constant. The warping path is hereby confined to an area called *Itakura parallelogram* (see Figure 2.8(b)).

Aside from preventing warping paths that are highly distorted and hence thought to produce unrealistic alignments, confining the warping path to a specific region in the distance matrix effects that not all cells of the local distance matrix have to be evaluated. By this, the dynamic time warping algorithm can be sped up substantially. Nevertheless it is possible (e.g. for distorted data) that the global optimum warping path runs outside the region to which it is restricted by the algorithm and thus cannot be found anymore. Further considerations about modifications to the original dynamic time warping algorithm can be found in [MRR80; KP01; SC07; MÖ7; RJ93].

The next chapter presents several approaches from other work that are related to the development of acoustic distance measures allowing for determination of acoustic similarity.

3 Related Work

In this chapter several approaches from other works are presented that are related to the development of acoustic distance measures allowing for the determination of acoustic similarity. Mel-frequency cepstral coefficients (MFCCs) are the type of acoustic features that is most commonly used in applications for speech recognition, so this work confines itself to approaches that use them as representational basis for similarity computation and on that basis reviews several distance measures.

Most of the work that addresses acoustic similarity focuses either on the review and development of local distance measures for comparison of single feature vectors in a template based speech recognition setting or on distance measures for comparison of Gaussian distributions that model a feature vector sequence, aiming at comparison of timbre similarity in music classification. The development of distance measures specifically for syllable similarity was not discussed in any of the work found during research for this thesis. However, approaches that are meant for similarity measurement on either acoustic units of a different granularity level (e.g. phonemes, words) or even on content with acoustic characteristics different from speech (e.g. music) yield the opportunity of providing decent acoustic similarity measures that work on syllables as well. The following sections present several such approaches.

3.1 Local Mahalanobis Distance in Template Based Speech Recognition

In [De +07b] a framework for continuous template based speech recognition is introduced which employs dynamic time warping for comparing feature vector sequences on sample level and several local distance measures to compare individual vectors on frame level. The templates used in this framework are at least on phoneme level and scalable to higher levels such as syllables or words via concatenation. Dynamic time warping is combined with hidden Markov model techniques in the overall framework to avoid disadvantages from both dynamic time warping (search space explosion in continuous recognition and poor speaker independent performance) and hidden Markov models (discarded information about time dependencies and over-generalization). For the exploration of acoustic distance measures for syllables the local distance measures presented in this paper are of particular interest. In most applications DTW is based on a distance metric that is global and symmetric between compared frames where in contrast hidden Markov models employ a local probability density function that is state specific. The class-dependent probability density functions of HMMs are one aspect of their good performance and wide usage so the paper presents several

approaches to transfer this aspect to the local distance measures for dynamic time warping.

A general weighted frame-based distance measure for feature vectors $\mathbf{x}$ and $\mathbf{y}$ is given by

$$d(\mathbf{x}, \mathbf{y}) = (\mathbf{x} - \mathbf{y})^T \mathbf{\Lambda} (\mathbf{x} - \mathbf{y})$$

with $\mathbf{\Lambda} \in \mathbb{R}^{M \times M}$ the weights. With $\mathbf{\Lambda}$ the identity matrix this is the *Euclidean distance*. When employing the inverse covariance matrix of the data $\mathbf{\Lambda} = \mathbf{\Sigma}^{-1}$ this is the *Mahalanobis distance*. In the next step the weights can be made dependent on the class $k(\mathbf{y})$ of feature vector $\mathbf{y}$, dropping the symmetry and the triangular inequality properties. An adaption of the Mahalanobis distance is introduced as such a local distance measure, given by

$$d_{\mathrm{LM}}(\mathbf{x}, \mathbf{y}) = (\mathbf{x} - \mathbf{y})^T \mathbf{\Sigma}_{k(\mathbf{y})}^{-1} (\mathbf{x} - \mathbf{y}) + \ln |\mathbf{\Sigma}_{k(\mathbf{y})}|,$$

with an extra bias term compensating for the transformations towards different classes. This measure was originally presented by De Wachter et al. in [De +04]. In both [De +04] and [De +07b] a version for diagonal covariance matrices is used which can be computed faster than for full covariance matrices. It is shown that such a distance measure combined with dynamic time warping leads to a natural hidden Markov model interpretation of the recognition system. From the comparison to Parzen density estimation (cf. [Sil86; DDV07]) the idea of using adaptive kernel estimates to cope with the poor performance of basic Parzen density estimation in the tails of the distributions can be applied to the local Mahalanobis distance mentioned above. The modified local distance measure for diagonal covariances then becomes

$$d_{\mathrm{LBM}}(\mathbf{x}, \mathbf{y}) = \sum_{l=1}^{M} \left(\frac{\mathbf{x}_l - \mathbf{y}_l}{\alpha_{\mathbf{y}} \hat{\sigma}_{k(\mathbf{y}),l}} \right) + \ln \left(\prod_{l=1}^{M} (\alpha_{\mathbf{y}} \hat{\sigma}_{k(\mathbf{y}),l})^2 \right),$$

with $\alpha_{\mathbf{y}}$ the so-called local bandwidth calculated from Gaussian mixture models (GMMs) with diagonal covariance matrices. This distance measure was originally introduced in [DDV07] as *adaptive kernel local Mahalanobis distance*. In [DDV07] apart from adding local bandwidth factors to the kernel interpretation of the reference vectors, a second technique called *data sharpening* is used. The idea is to replace each reference vector with an average of its neighborhood from the recognition database. Both data sharpening and adaptive kernel estimation are intended to compensate for outliers (i.e. samples in the tails of the class distribution) by adjusting the distance measure based on the position of the sample vector within its class.

[De +07b] compares the Euclidean distance, the local Mahalanobis distance and the local Mahalanobis distance with variable bandwidth on the DARPA Resource Management Database for Continuous Speech Recognition (cf. [Pri+88]) using 24 MFCC coefficients and their first and second derivatives that have been transformed via linear discriminant analysis to only keep the 25 most meaningful dimensions. The distance measures are evaluated on phoneme level. The local Mahalanobis distance shows a relative improvement of 14% over the Euclidean distance in word error rate (WER) where the addition of the variable bandwidth factor gives a relative improvement of

21% WER over the Euclidean distance. [De +07a] shows that the data sharpening method yields a substantial improvement in overall recognition rates on phone level while narrowing down the relative improvement of the three distance measures among one another.

3.2 Discriminative Locally Weighted Mahalanobis Distance in Template Based Speech Recognition

In [Mat+04] another locally weighted distance measure is used for template based speech recognition which was first presented in [PV99]. This measure was designed for k nearest neighbor classification, modeling for each frame the nearest neighbor in the relevant class while discriminating it from the other classes. The distance measure is defined as

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{j=1}^{D} \lambda_{c,j}^2 (\mathbf{x}_j - \mathbf{y}_j)}$$

with $\lambda_{c,j}$ the weights which are estimated using a discriminative iterative procedure. For estimation the criterion index $\sum_{\mathbf{x}} \frac{d(\mathbf{x}_{\in}, \mathbf{x})}{d(\mathbf{x}_{\notin}, \mathbf{x})}$ is minimized via gradient descent leading to a set of iterative update equations, $\mathbf{x}_{\in}$ denoting the nearest neighbor of $\mathbf{x}$ in the same class as $\mathbf{x}$ and $\mathbf{x}_{\notin}$ denoting the nearest neighbor of $\mathbf{x}$ that is not in the same class. Experiments which are performed with the framework described in [De +04] also using the Resource Management benchmark (cf. [Pri+88]) give a mean relative improvement in recognition error rate of 14% over the Euclidean distance.

3.3 Kullback-Leibler Divergence in Timbre Matching for Music Genre Classification

In [Jen+09] Jensen et al. present an approach where a nearest neighbor classifier using the Kullback-Leibler divergence between Gaussian mixture models of mel-frequency cepstral coefficients is used to compare the timbre of music pieces. In music, *timbre* is the quality that distinguishes a sound (i.e. a musical note or tone) from another sound with identical pitch and loudness [Moo03]. For timbre matching, in [Jen+09] for the MFCC features of each song a separate Gaussian mixture model is trained and then compared with the Kullback-Leibler divergence. The approach of interpreting a sequence of feature vectors as Gaussian model for consecutive comparison to the model of another vector sequence is often referred to as *"bag of frames" approach* (cf. [ADP07]; see section 5.3). The *Kullback-Leibler divergence* is an information theoretic measure that models the dissimilarity of two probability distributions. In general, the probability density function for a random variable $\mathbf{x}$ when described by a Gaussian mixture model is given by

$$p(\mathbf{x}) = \sum_{k=1}^{K} c_k \frac{1}{\sqrt{|2\pi\boldsymbol{\Sigma}_k|}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k)\right)$$

where K is the number of mixtures and $\boldsymbol{\mu}_k$, $\boldsymbol{\Sigma}_k$ and c_k are the mean, covariance matrix and weight of the k-th Gaussian. In its original form, the Kullback-Leibler divergence of two density functions $p_1(\mathbf{x})$ and $p_2(\mathbf{x})$ is an asymetric measure, formally denoted by

$$d_{\mathrm{KL}}(p_1, p_2) = \int p_1(\mathbf{x}) \ln \frac{p_1(\mathbf{x})}{p_2(\mathbf{x})}\, d\mathbf{x}$$

For simple multivariate Gaussian distributions (i.e. Gaussian mixture model with $K = 1$) there exists a closed form expression to compute the Kullback-Leibler divergence (see section 5.4.1). For general Gaussian mixtures however, a closed form does not exist and it must be estimated via approximation methods. In [Jen+09] a symmetric version of the Kullback-Leibler distance is used which is formally obtained by

$$d_{\mathrm{SKL}}(p_1, p_2) = d_{\mathrm{SKL}}(p_2, p_1) = d_{\mathrm{KL}}(p_1, p_2) + d_{\mathrm{KL}}(p_2, p_1).$$

The approach described in the paper was originally presented in [AP02]; similar approaches can be found in [LS01] and [LH00]. Experiments from [Jen+09] are carried out for simple multivariate Gaussian models to classify the instruments playing in synthesized MIDI files. The experiments in detail are not relevant from the point of view of acoustic similarity measures for syllables since this application is substantially different. The paper concludes that the Kullback-Leibler divergence on multivariate Gaussian models is indeed able to recognize instrumentation (with certain limitations). Jensen et al. state that this approach won the genre classification contest of the International Conference on Music Information Retrieval (ISMIR) 2004. Altogether this might hold a promising approach to measure acoustic similarity of speech since here timbre dissimilarity is being measured on MFCCs that are an established method in speech processing as well.

In [Jen+07] several distance measures between Gaussian mixture models are compared in terms of usefulness for timbre similarity measurement in music classification. Jensen et al. evaluate the symmetric Kullback-Leibler divergence, the Earth Movers distance and the normalized L2 distance. It is emphasized that a distance measure satisfying the triangle inequality is beneficial because nearest neighbor classification can be sped up by precomputing a number of distances. The Kullback-Leibler divergence does not satisfy the triangle inequality.

The *Earth Movers distance* allows for approximation of the Kullback-Leibler divergence of Gaussian mixture models which is necessary since there is no closed form expression for the exact solution. It describes the minimum cost of transforming one mixture into another when the cost of shifting probability mass between them is given [LS01]. The cost chosen in [Jen+07] is the symmetric Kullback-Leibler divergence due to which the Earth Movers distance here does not satisfy the triangle inequality. The *normalized L2 distance* is defined as $d_{NL2}(p_1, p_2) = \int (p_1'(\mathbf{x}) - p_2'(\mathbf{x}))^2\, d\mathbf{x}$ with p_1 and p_2 scaled to unit L2 norm, being a continuous version of the cosine distance. Closed form expressions can be derived for an arbitrarily sized Gaussian mixture model (cf. [Ahr05]). Furthermore, it satisfies the triangle inequality.

For experiments the Kullback-Leibler divergence for Gaussian mixture models was approximated via stochastic integration. The evaluation was carried out for a single

Gaussian (i.e. a multivariate Gaussian model) and a mixture of ten Gaussians (i.e. a regular Gaussian mixture model). Jensen et al. find that when using a single Gaussian all measures perform approximately equally well. For a mixture of Gaussians the Kullback-Leibler divergence is slightly better in accuracy than the normalized L2 distance that on the other side satisfies the triangle inequality which the other measures do not.

3.4 Comparison of Model Parameters in Timbre Matching for Music Genre Classification

In [LS06] lightweight measures for similarity of timbre in music are reviewed. Again, all similarity measures are based on Gaussian mixture models of mel-frequency cepstral coefficients (except for one measure). Levy and Sandler strive to find lightweight measures that perform equally well in respect to established methods that have high computational requirements. Music classification usually operates on large collections of data so similarity measures that are expensive to compute are impractical. This is comparable to the situation of speech classification in a real-time application as it is targeted by this work. The first method presented in the paper is to use a vector quantization algorithm to partition the global space of MFCCs for each sample into indexed regions that are each identified by a single vector in a codebook. The similarity measure is then a distance between codebook index sequences for two samples. Secondly the paper examines the symmetric Kullback-Leibler divergence on a single Gaussian from the feature vectors both for diagonal and full covariance matrices. Diagonal covariances bear the advantage that the computation of the measure is sped up significantly because matrix inversion becomes obsolete. Thirdly Levy and Sandler present a version of the Mahalanobis distance that operates directly on the parameters (i.e. mean and covariance) of the Gaussian densities $p_1(\mathbf{x})$ and $p_2(\mathbf{x})$, given by

$$\begin{aligned} d(p_1, p_2) =& (\boldsymbol{\mu}(p_1) - \boldsymbol{\mu}(p_2))^T \boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{-1} (\boldsymbol{\mu}(p_1) - \boldsymbol{\mu}(p_2)) \\ &+ (\boldsymbol{\Sigma}(p_1) - \boldsymbol{\Sigma}(p_2))^T \boldsymbol{\Sigma}_{\boldsymbol{\Sigma}}^{-1} (\boldsymbol{\Sigma}(p_1) - \boldsymbol{\Sigma}(p_2)) \end{aligned}$$

where $\boldsymbol{\mu}(p_i)$ and $\boldsymbol{\Sigma}(p_i)$ denote the mean and covariance of the sample Gaussian distribution and $\boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{-1}$ and $\boldsymbol{\Sigma}_{\boldsymbol{\Sigma}}^{-1}$ denote the variances of the feature means and covariances. This comparison could be carried out as simple Euclidean distance respectively.

Experiments show that the vector quantization index based measure is outperformed by the Gaussian model based measures. Kullback-Leibler divergence and Mahalanobis distance on single Gaussians both perform well compared to a reference GMM-based measure (with a loss of relatively only 2,5% and 5% in classification rates). When diagonal covariances are used instead of full covariance matrices the loss in classification rate stays small (relatively 2,5% for the Kullback-Leibler divergence) while there is a 10-fold gain in both speed and memory requirements. Moreover it is mentioned that in contrast to the Kullback-Leibler divergence the Mahalanobis distance

Author	Overall System Application Area	Measure Target	Distance Measure
De Wachter et al.	Continuous speech recognition w. variable granularity templates	MFCC vectors	Local Mahalanobis distance Local bandwidth Mahalanobis distance
Matton et al.	Template based speech recognition	MFCC vectors	Discriminative locally weighted Mahalanobis distance
Jensen et al.	Timbre matching for NN music genre classification	GMMs/ SGMS of MFCCs	Symmetric KL divergence Earth Movers distance Normalized L2 distance
Levy and Sandler	Music genre classification via timbre	MFCC vectors SGMs of MFCCs Parameters of MFCC SGMs	Distances on VQ codebook index sequences Symmetric KL divergence Mahalanobis distance

Table 3.1: Distance Measures of related work.

is a metric, potentially enabling a further speed-up when using indexing structures in nearest neighbor classification.

The property that the Kullback-Leibler divergence on multivariate Gaussian distributions and the Mahalanobis distance of parameter vectors (i.e. the concatenation of mean and covariance as a vector) of single Gaussians (here called *MFCC statistics*) perform comparably well for music classification was also shown by Mandel and Ellis in [ME05].

3.5 Synopsis

The reviewed papers summon interesting methods to measure acoustic similarity on mel-frequency cepstral coefficients (MFCCs). Table 3.1 provides an overview of the distance measures presented in the related work.

The next chapter investigates the requirements of an acoustic similarity measure for syllables targeting an application area that matches the necessities of the tutoring scenario (cf. chapter 1). Subsequently it discusses the methods referred to in the related work in terms of usefulness for the targeted syllable similarity measure. Finally the best matching methods are selected for implementation and evaluation in this work.

4 Requirements

In order to develop an acoustic similarity measure for syllables it is necessary to identify the requirements that such a distance measure must meet. To this end, in the following section this chapter first revisits the tutoring scenario that was originally introduced in chapter 1. In sections 4.2 and 4.3 requirements and further desirable properties are formulated which emerge from the conditions the tutoring scenario implies. The methods presented in the related work (see chapter 3) are then in section 4.4 reviewed and discussed in order to select measures that are best for being transferred to a tutoring scenario application. Finally the best matching methods are selected for implementation and evaluation in this work.

4.1 The Tutoring Scenario Revisited

As stated in chapter 1, in an application like the tutoring scenario the robot respectively the speech recognition system needs to map certain utterance parts to each other in order to identify and relate the acoustic representation of corresponding concepts, like e.g. objects that are referred to by the speaker.

A previously discussed example was that in one sentence uttered by the speaker an object is firstly presented to the robot (e.g. "This is the *blue cup*."). In a later sentence the tutor might again refer to the object initially presented and for example now describe an action he or she is demonstrating with it (e.g. "You take the green cup and put it into the *blue cup*."). If the robot was now able to map both acoustic representations of the object (e.g. "*blue cup*") to each other while maintaining a temporally coherent linkage of speech and vision (e.g. a video stream) it could actually gain a concept of this object consisting of an acoustic and a visual representation. So the cue that could enable this mapping is *acoustic similarity*.

Another cue to identify the importance of certain parts in the continuous speech stream other than the acoustic similarity of utterance parts is for example *stress*, i.e. the relative emphasis that may be given to certain syllables of words in the speech stream. To allow an overall speech recognition system in a tutoring application to incorporate this cue it is beneficial for the acoustic similarity test to take place on syllable level as well. Also, this diminishes the significance of acoustic pronunciation variations so that the compound acoustic similarity of whole words is more consistent than by performing the acoustic similarity measurement on a higher granularity level (e.g. words) [Gan+97].

The relevant task in the tutoring scenario is hence a *classification task*. It is necessary to determine the acoustic similarity of the syllables uttered during a session in the tutoring scenario in order to match equal syllables.

4.2 Conceptual Properties

An acoustic similarity measure for syllables bears certain requirements. In order to assess methods presented in related work it is first necessary to formally identify these requirements so that the best matching measures can be selected for investigation in this thesis. Moreover there are properties of a syllable distance measure which are not necessarily required but are nevertheless beneficial to be met. The requirements and further desirable properties are identified subsequently.

The most substantial and trivial requirement for the distance measure is the measurement of similarity on syllable samples itself. Therefore an order of the pairs between which distance is measured is imposed, small values indicating that a pair of samples is *similar* and large values indicating that a sample pair is *dissimilar* respectively. In the following the distance measure is constituted as a mathematical function d that measures similarity between to samples $\mathbf{x}$ and $\mathbf{y}$.

Requirement 1. ***Measurement of Similarity*** *The distance measure is required to be able to compute the similarity of two samples, i.e. provide a value that is smaller the more similar two samples are and larger the more dissimilar they are.*

In the domain of the distance measure there has to be a smallest value so that the similarity term can actually be defined. Consequently, zero is defined as smallest possible resulting value.

Requirement 2. ***Non-Negativeness*** *The distance measure is required to only produce non-negative values, i.e.* $d(\mathbf{x}, \mathbf{y}) \geq 0 \; \forall \mathbf{x}, \mathbf{y}$.

With defining the term of similarity comes the need to have the distance measure produce the smallest possible value (i.e. zero) if and only if two compared samples are exactly similar.

Requirement 3. ***Identity of Indiscernibles*** *The distance measure is required to produce the smallest possible value if and only if the samples are identical, i.e.* $d(\mathbf{x}, \mathbf{y}) = 0 \Leftrightarrow \mathbf{x} = \mathbf{y} \; \forall \mathbf{x}, \mathbf{y}$.

Requirements 2 and 3 together being satisfied imposes complying with *positive definiteness* as well. To have an applicable distance measure, it needs to be symmetric so that it is equivalent to measure the distance to a specific sample from the point of view of another certain sample or vice versa.

Requirement 4. ***Symmetry*** *The distance measure is required to be symmetric, i.e.* $d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x}) \; \forall \mathbf{x}, \mathbf{y}$.

For the distance measure to work reliably in the ideal case distances between samples within the sample class are always to be smaller than the distance of the currently examined sample to any sample of any other class. Due to pronunciation variations and distortion caused by noise this property will not always be met. However the ambition to fulfill this ideal condition is desirable.

Desirable Property 1. ***Preference of Class Affiliation*** *It is desirable for the distance between a sample and another sample from the same class always to be smaller than the distance between the sample and a sample of a different class, i.e.* $d(\mathbf{x}, \mathbf{y}) < d(\mathbf{x}, \mathbf{z})\ \forall \mathbf{x}, \mathbf{y} \in C_i \wedge \mathbf{z} \notin C_i$.

To efficiently handle a nearest neighbor search (cf. tutoring scenario as classification task) it is conducive if the distance measure satisfies the triangle inequality as this allows for the search to be sped up via precomputation of a few distances. Assume the nearest neighbor to $\mathbf{x}$ is to be searched and the distance to $\mathbf{y}$ was just computed. If the distance between $\mathbf{y}$ and $\mathbf{z}$ is already known, the distance to $\mathbf{z}$ is bounded by $d(\mathbf{x}, \mathbf{z}) \geq d(\mathbf{y}, \mathbf{z}) - d(\mathbf{y}, \mathbf{x})$. The candidate $\mathbf{z}$ can now be discarded without computation of $d(\mathbf{x}, \mathbf{z})$ if the current best candidate is already smaller than $d(\mathbf{y}, \mathbf{z}) - d(\mathbf{y}, \mathbf{x})$. It is thus desirable for the distance measure to satisfy the triangular inequality. If in addition to the requirements 2, 3 and 4 this property is complied with as well, the distance measure becomes a *metric*.

Desirable Property 2. ***Triangle Inequality*** *It is desirable for the distance measure to satisfy the triangle inequality, i.e.* $d(\mathbf{x}, \mathbf{z}) \leq d(\mathbf{x}, \mathbf{y}) + d(\mathbf{y}, \mathbf{z})\ \forall \mathbf{x}, \mathbf{y}, \mathbf{z}$.

Lastly, a further convenient property of the distance measure is to be consistent with the percepted similarity in humans. This allows for an easier assessment and increased plausibility of the distance measure. As an example the syllables “dog” and “hog” are perceptually more similar than the syllables “dog” and “cat”.

Desirable Property 3. ***Consistency with Perceptual Similarity*** *It is desirable for the distance measure to reproduce the order of perceptual similarity so that the distance of a sample to another sample is smaller than the distance to yet another sample if and only if they are perceptually more similar, i.e.* $d(\mathbf{x}, \mathbf{y}) < d(\mathbf{x}, \mathbf{z}) \Leftrightarrow d_P(\mathbf{x}, \mathbf{y}) < d_P(\mathbf{x}, \mathbf{z})\ \forall \mathbf{x}, \mathbf{y}, \mathbf{z}$, d_P *denoting the perceptual similarity.*

4.3 Properties Implied by Application in a Tutoring Scenario

The application of the distance measure in a tutoring scenario yields several further requirements and desirable properties that must be thought of. Firstly, the distance measure is required to be independent of initially knowing what data will occur in a specific session. In template based recognition systems there is a template database that defines possible candidates to which the current sample of interest is compared. The overall characteristics of such a template database is in such systems often used to enhance the distance measure. However this is not applicable here since the actual

data that will occur during a specific classification session is unknown. As a side note it is very well possible and conceivable that knowledge gained during a session can be integrated in the distance computation to provide an improvement of the measure quality.

Requirement 5. ***Independence of the Data Actually Occurring*** *The distance measure is required to be independent of initially knowing the data that is actually occurring during a session.*

Moreover it is desirable that the distance measure works equally well for any individual speaker, for any dialect, idiolect or gender and for any environment the recognition system is used in. This can clearly not be guaranteed or even easily assessed since statistically there will be variations in the measure quality among different conditions. Nevertheless striving to comply with this property is beneficial.

Desirable Property 4. ***Independence of Setting Characteristics*** *It is desirable for the distance measure to work equally well for any individual speaker, dialect, idiolect or gender and for any environment the recognition system is used in.*

Furthermore, the recognition system in which the distance measure is incorporated needs to respond sufficiently quickly in order to allow for classification to be performed continuously while the speaker generates acoustic input to the system, so that a robot in the tutoring scenario can react and interact without delay. It is thus desirable to select a distance measure with low complexity that provides good results nevertheless.

Desirable Property 5. ***Minimum Complexity*** *It is desirable for the distance measure to yield minimal complexity while still providing good classification results so that it can performantly be applied in an online classification task in continuous speech recognition.*

4.4 Discussion of Related Work Methods

In the following the methods presented in related work (see chapter 3) are discussed with respect to the requirements and desirable properties that were postulated in the preceding sections. The aim is to select methods that are useful in an acoustic similarity measure for syllables that targets an application area matching the necessities of the tutoring scenario (cf. chapter 1 and section 4.1).

In subsection 4.4.1 the local Mahalanobis distance, the local bandwidth Mahalanobis distance and a technique called data sharpening are discussed. These are distance measures that were in the original work used in a dynamic time warping setting. Subsection 4.4.2 discusses measures that operate on statistical models (Gaussian distributions) that model the feature vectors sequence for the individual samples being compared. It discusses the symmetric Kullback-Leibler divergence and the Mahalanobis distance on Gaussian model parameters.

Vector Quantization Index Sequences In [LS06] Levy and Sandler presented a method that uses a vector quantization algorithm to partition the global space of MFCCs for each sample into indexed regions that are each identified by a single vector in a codebook, the similarity measure then being a distance between codebook index sequences for two samples. Since experiments from Levy and Sandler showed that this measure is clearly outperformed by the Gaussian model based measures from the same paper (cf. subsection 4.4.2), this approach is not considered for evaluation in this thesis.

4.4.1 Dynamic Time Warping Based Methods

Subsequently, methods from the related work are discussed that were used as local distance measures in a dynamic time warping setting.

Local Mahalanobis Distance The local Mahalanobis distance as presented by De Wachter et al. in [De +07b; DDV07; De +07a; De +04] (cf. section 3.1) uses a covariance matrix that is dependent on the actual data encountered in the application of the overall distance measure. These are templates from a template database. However, as required for the measure targeted in this work, the measure cannot depend on previously knowing the actual data since in an application like the tutoring scenario the occurring data is initially unknown (see requirement 5). Moreover the distance measure incorporates only the covariance of the class belonging to the feature vector the reference vector is compared to, resulting in an asymmetric measure. This objects to requirement 4. Requirements 2 and 3 are obviously met. Nevertheless the Mahalanobis distance is a measure worth evaluating since it incorporates the covariance in order to diminish the influence of features that bear higher fluctuations than others. Instead of the covariance from the actual data covariances from a statistical model of the language that is targeted by the recognition system could be used. If the covariance matrix is made dependent on both feature vectors compared this would again result in a symmetric distance measure. Having replaced the biased covariance by one that describes the characteristics of both feature vectors, the additional bias term can be omitted since it previously compensated for the drift to the class of the feature vector to that the reference vector is compared. An interpretation of the Mahalanobis distance that satisfies all requirements and thus is adequate to be evaluated in this work is presented in section 5.2.2.

Local Bandwidth Mahalanobis Distance The local bandwidth Mahalanobis distance (also called adaptive kernel local Mahalanobis distance) as presented by De Wachter et al. (cf. section 3.1) in [De +07b; DDV07; De +07a] adds an additional adaptive scaling (called local bandwidth) to the individual components of the covariance matrix. The local bandwidth parameters are computed from Gaussian mixture models fitted to the actual data encountered in the application using the distance measure. Again, since the measure targeted in this work cannot depend on the actual data, this contradicts requirement 5. If the local bandwidth would instead be

computed on a Gaussian mixture that generally models the overall feature vectors in the target language this would instead result in a measure similar to a Mahalanobis distance that uses covariance models that come from a Gaussian mixture modeling the entire language.

Data Sharpening The data sharpening technique presented by De Wachter, Demuynck, and Van Compernolle in [DDV07] (cf. section 3.1) is used to compensate for outliers (i.e. samples in the tails of the class distribution) by adjusting the distance measure based on the position of the sample vector within its class. The idea is to replace each feature vector being compared with an average of its neighborhood from the recognition database. Technically this would as well disagree with requirement 5 since other than in the framework evaluated in [DDV07] there is no database of templates available from which feature vectors can be drawn to inquire neighborhoods.

Discriminative Locally Weighted Mahalanobis Distance The discriminative locally weighted Mahalanobis distance presented by Matton et al. in [Mat+04] (cf. section 3.2) models for each frame the nearest neighbor in the relevant class while discriminating it from the other classes. The weights are estimated using a discriminative iterative procedure to minimize a criterion index that considers the relation of the distance to the next neighbor in the same class to the distance to the next neighbor from a different class. However, the application targeted by the distance measure to be designed in this thesis does not operate on a template database. Consequently nearest neighbors to data that is already present cannot be evaluated when computing the distance so that this measure is as well impractical for use and evaluation in this thesis (contradicting requirement 5).

4.4.2 Temporal Statistics Based Methods

In the following measures are discussed which operate on Gaussian distributions (i.e. statistical models) modeling the sequence of feature vectors of the individual samples being compared.

Symmetric Kullback-Leibler Divergence The symmetric Kullback-Leibler divergence as presented by Jensen et al. in [Jen+09; Jen+07] (cf. section 3.3) and Levy and Sandler in [LS06] (cf. section 3.4) measures the dissimilarity of the Gaussian distributions of the feature vectors from the samples between which the distance is computed. If a Gaussian mixture is used to model the sample feature vectors the Kullback-Leibler divergence is not computable via a closed term expression. Instead it has to be approximated via stochastic methods. In [Jen+07] this was done by using the Earth Movers distance. If instead a single Gaussian is used as model the Kullback-Leibler divergence becomes computable by a closed term (see section 5.4.1). As experiments in [LS06] show the Kullback-Leibler divergence performs comparably well when using single Gaussians rather than Gaussian mixtures while there is a significant improvement in time complexity which panders to requirement 5. The

Kullback-Leibler divergence is always non-negative, i.e. $d(p_1, p_2) \geq 0$ for two probability densities p_1 and p_2, and thus satisfies requirement 2 (non-negativeness). It is moreover zero if and only if both p_1 and p_2 describe the exact same distribution (i.e. $d(p_1, p_2) = 0 \Leftrightarrow p_1 = p_2$), so it complies with requirement 3 (identity of indiscernibles) as well. Similarly [ME05] and [Pam06] showed that a single multivariate Gaussian distribution with a fully occupied covariance matrix can be used instead of mixture densities with diagonal covariance matrices without significantly worsen the results. See section 5.4.1 for a further description.

Mahalanobis Distance on Gaussian Model Parameters The Mahalanobis distance on Gaussian model parameters as described by Levy and Sandler in [LS06] (cf. section 3.4) compares the sample means and covariances that are computed from the feature vectors of the samples compared, thus describing each a single multivariate Gaussian distribution. It can be argued that by interpreting a feature vector sequence as a statistical distribution the temporal order of the individual feature vectors is lost and thus becomes unimportant for the distance measure. In essence, this violates requirement 3 if the samples compared are seen as temporal sequence of feature vectors. If however they are formally interpreted as mathematical set, the requirement still holds. Because of syllables being rather short acoustic units, the chronology of feature might altogether not be essential for discriminating two samples. In addition the Mahalanobis distance needs the covariance of the feature vector means and the covariance of the feature vector covariances. This can be provided by computing them on a large speech corpus of utterances in the target language. Since a large corpus would be used rather than the actual samples occurring during a classification session in the application, this would not offend requirement 5. In [LS06] it was also shown in experiments that when diagonal covariances are used in lieu of full covariance matrices the classification rate is only slightly worse while there is a huge gain in speed and memory consumption. This would benefit the capability to apply the distance measure in a scenario where classification is to be executed continuously. It is unclear if this observation is true as well for speech classification of syllables rather than music genre classification via timbre. This is evaluated as well in this thesis. Moreover this version of the Mahalanobis distance satisfies the triangle inequality (desirable property 2), enabling a potential further speed-up when using indexing structures in nearest neighbor classification. See section 5.4.2 for a further description.

5 Architecture

In this chapter the methods to measure acoustic syllable similarity that were implemented and evaluated in the context of this thesis are presented in respect to their concept and their detailed computation, including their prerequesites where applicable. Figure 5.1 gives a schema of the distance computation which is performed on the basis of two sequences of feature vectors, each describing an input syllable speech sample. The distance measures were selected with the prospect of processing vectors of mel-frequency cepstral coefficients (MFCCs), although technically they could process any kind of feature vectors which however is not evaluation subject to this thesis.

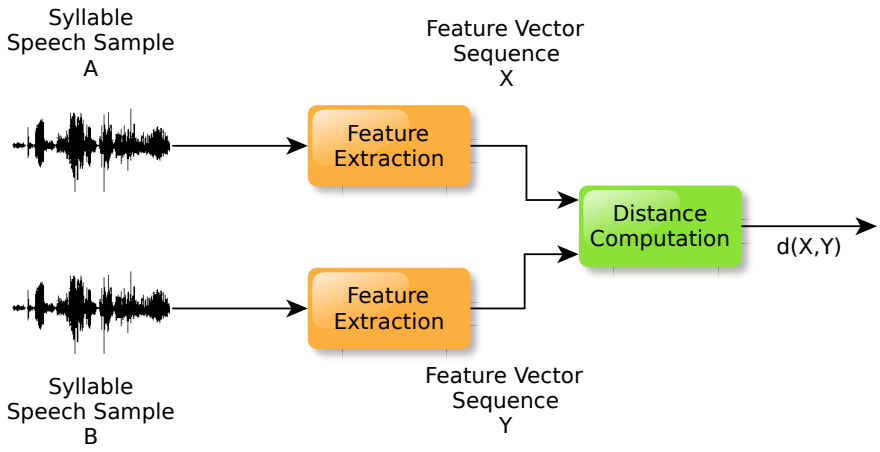


Figure 5.1: Computation of the acoustic distance $d(X, Y)$ between feature vector sequences X and Y for two syllable speech samples.

Section 5.1 explains a dynamic time warping approach that uses the Euclidean Distance and a variant of the Mahalanobis Distance as local distance measures, which are explained in section 5.2. Section 5.3 presents an approach that first estimates a statistical model of the sets of feature vectors from both speech samples and then compares the probability distributions (section 5.4) from the statistical model in order to measure the acoustic similarity.

5.1 Dynamic Time Warping

In this thesis a similarity measure for syllables is to be developed. To this end, when considering local distance measures for dynamic time warping, it is reasonable to use

a very rudimentary form of the dynamic time warping algorithm so that the space of possible alignments is not confined, since the usage of specific global constraints is subject to further evaluation itself.

However in an application that is intended to provide a similarity measure that can be computed sufficiently fast this would be a decent starting point to make an acceptable trade-off of speed vs. quality of the measure. In the implementation for this thesis, the original constraints as proposed by Sakoe and Chiba (see section 2.6.2) are used with an equal weighting of the three possible path options in the recursion formula.

Let $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_{L_\mathbf{x}})$ and $\mathbf{y} = (\mathbf{y}_1, \ldots, \mathbf{y}_{L_\mathbf{y}})$ be the feature vector sequences of two speech samples that are to be compared via dynamic time warping, $L_\mathbf{x}$ being the length of sequence $\mathbf{x}$ and $L_\mathbf{y}$ being the length of sequence $\mathbf{y}$. The accumulated dynamic time warping distance is then given as recursion formula by

$$
\begin{aligned}
D(\mathbf{x}^i, \mathbf{y}^j) &= \min \left\{ \begin{array}{ll} D(\mathbf{x}^{i-1}, \mathbf{y}^j) & +d(\mathbf{x}_i, \mathbf{y}_j) \\ D(\mathbf{x}^{i-1}, \mathbf{y}^{j-1}) & +d(\mathbf{x}_i, \mathbf{y}_j) \\ D(\mathbf{x}^i, \mathbf{y}^{j-1}) & +d(\mathbf{x}_i, \mathbf{y}_j) \end{array} \right\} && \forall i > 1 \wedge j > 1, \\
D(\mathbf{x}^i, \mathbf{y}^j) &= \infty && \forall (i = 1 \vee j = 1) \wedge i \neq j, \\
D(\mathbf{x}^1, \mathbf{y}^1) &= d(\mathbf{x}_1, \mathbf{y}_1),
\end{aligned}
$$

where $d(\mathbf{x}, \mathbf{y})$ is a local distance measure that measures the distance between individual feature vectors. The accumulated dynamic time warping distance of the optimum warping path is determined by $D(\mathbf{x}^{L_\mathbf{x}}, \mathbf{y}^{L_\mathbf{y}})$. For determining the final dynamic time warping score for both samples, the accumulated distance of the optimum path has to be normalized with repect to its length. As a consequence the optimum path has to be determined from the previously computed accumulated warping distances via backtracking.

Let the optimum warping path p^* be composed as sequence of tuples (i_l, j_l) with $1 \leq i \leq L_\mathbf{x}$ and $1 \leq j \leq L_\mathbf{y}$. It is then given by $p^* = (p_1, \ldots, p_L)$ with $p_L = (L_\mathbf{x}, L_\mathbf{y})$ and $p_1 = (1, 1)$ and then recursively by

$$
p_{l-1} = \begin{cases} (1, j_l - 1) & \text{if } i_l = 1, \\ (i_l - 1, 1) & \text{if } j_l = 1, \\ \arg\min \left\{ \begin{array}{l} D(i_l - 1, j_l - 1) \\ D(i_l - 1, j_l) \\ D(i_l, j_l - 1) \end{array} \right\} & \text{otherwise.} \end{cases}
$$

The final dynamic time warping score is then

$$
D(\mathbf{x}, \mathbf{y}) = \frac{1}{L} D(\mathbf{x}^{L_\mathbf{x}}, \mathbf{y}^{L_\mathbf{y}}).
$$

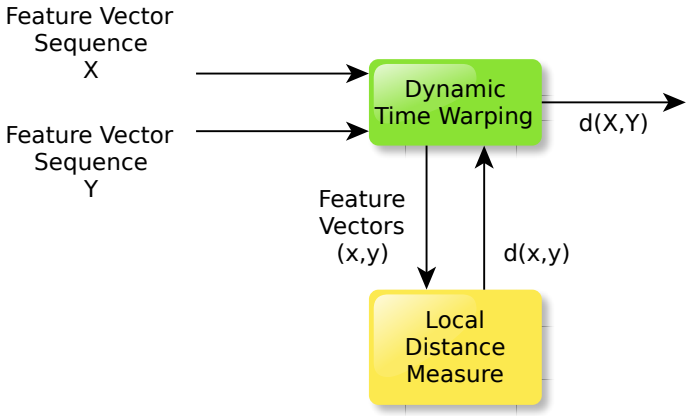


Figure 5.2: Computation of the acoustic distance $d(X, Y)$ via dynamic time warping of feature vector sequences X and Y for two syllable speech samples.

5.2 Local Distance Measures for Dynamic Time Warping

The dynamic time warping algorithm needs a local distance measure for comparison of individual feature vectors, while combining the local distances globally for the entire feature vector sequence of a speech sample. In the following subsections several local distance measures that were selected and implemented in this thesis are presented. Figure 5.2 shows a diagram of the computation of the acoustic distance via dynamic time warping.

5.2.1 Euclidean Distance

As a baseline local distance the *Euclidean distance* is used. It is one of the most fundamental measures for vector spaces and prevalent in a vast set of applications as a naive approach for distance measurement. In this work it serves as a reference frame for comparison with the Mahalanobis distance.

Let $\mathbf{x}$ and $\mathbf{y}$ be feature vectors of length N with $\mathbf{x} = (x_1, \ldots, x_N)^T$ and $\mathbf{y} = (y_1, \ldots, y_N)^T$. Then the Euclidean distance between $\mathbf{x}$ and $\mathbf{y}$ is defined as

$$d(\mathbf{x}, \mathbf{y}) = ||\mathbf{x} - \mathbf{y}||_2 = \sqrt{(\mathbf{x} - \mathbf{y})^T (\mathbf{x} - \mathbf{y})} = \sqrt{\sum_{i=1}^{N} (x_i - y_i)^2}.$$

The Euclidean distance obviously satisfies the properties of non-negativeness, identity of indiscernibles and symmetry (cf. requirements 2, 3 and 4). It also complies with the triangle inequality (cf. desirable property 2) and is thus a metric.

5.2.2 Mahalanobis Distance

The *Mahalanobis distance* uses a covariance matrix to compensate for the different fluctuations (i.e. different standard deviation) of individual feature vector components. This intends to prevent features with small fluctuation from being concealed by features with high fluctuations leading to an interpretation of the distance where individual feature vectors are statistically of equal importance. The Mahalanobis distance was originally introduced by Prasanta C. Mahalanobis in 1936 [Mah36].

Let $\mathbf{x}$ and $\mathbf{y}$ be feature vectors of length N with $\mathbf{x} = (x_1, \ldots, x_N)^T$ and $\mathbf{y} = (y_1, \ldots, y_N)^T$ and let $\mathbf{\Sigma}(\mathbf{x}, \mathbf{y})$ be a covariance matrix with $\mathbf{\Sigma}(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{N \times N}$. Then the Mahalanobis distance between $\mathbf{x}$ and $\mathbf{y}$ is defined as

$$\begin{aligned} d(\mathbf{x}, \mathbf{y}) &= \sqrt{(\mathbf{x} - \mathbf{y})^T \mathbf{\Sigma}^{-1}(\mathbf{x}, \mathbf{y})(\mathbf{x} - \mathbf{y})} \\ &= \sqrt{\sum_{i=1}^{N}(x_i - y_i) \sum_{j=1}^{N} \Sigma_{ij}^{-1}(\mathbf{x}, \mathbf{y})(x_j - y_j)}. \end{aligned}$$

If the covariance matrix is diagonal, i.e. $\mathbf{\Sigma}(\mathbf{x}, \mathbf{y}) = \mathrm{diag}(\sigma_1^2, \ldots, \sigma_N^2)$, the Mahalanobis distance is simplified to

$$\begin{aligned} d(\mathbf{x}, \mathbf{y}) &= \sqrt{\sum_{i=1}^{N}(x_i - y_i)\mathbf{\Sigma}_{ii}^{-1}(\mathbf{x}, \mathbf{y})(x_i - y_i)} \\ &= \sqrt{\sum_{i=1}^{N} \frac{1}{\Sigma_{ii}(\mathbf{x}, \mathbf{y})}(x_i - y_i)^2} = \sqrt{\sum_{i=1}^{N} \frac{1}{\sigma_i^2}(x_i - y_i)^2}. \end{aligned}$$

In this case the Mahalanobis distance is also called *normalized Euclidean distance*. The Mahalanobis distance satisfies the properties of non-negativeness and identity of indiscernibles (requirements 2 and 3). Since covariance matrices are symmetric, i.e. $(\mathbf{\Sigma} = \mathbf{\Sigma}^T \Leftrightarrow \mathbf{\Sigma}^{-1} = (\mathbf{\Sigma}^T)^{-1} \Leftrightarrow \mathbf{\Sigma}_{ij}^{-1} = \mathbf{\Sigma}_{ji}^{-1})$, the Mahalanobis distance is also symmetric (requirement 4). The Mahalanobis distance also complies to the triangle inequality (cf. desirable property 2) and is hence a metric.

Since $\mathbf{x}$ and $\mathbf{y}$ are two sample feature vectors, their covariance $\mathbf{\Sigma}(\mathbf{x}, \mathbf{y})$ is not known. It is therefore necessary to estimate this covariance from a general statistical model of the feature vectors in the target application area. In the next subsection several approaches to estimate this covariance which were evaluated in this work are presented.

5.2.3 Estimation of Covariance Matrices

The covariance that is used in the formula for the Mahalanobis distance has to be estimated from a general statistical model of the feature vectors in the target application area. This can for instance be the application target language if this statistical model is built on a unilingual speech corpus.

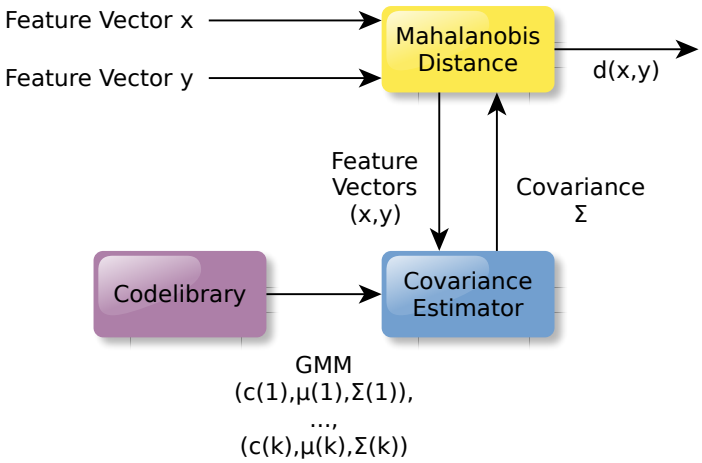


Figure 5.3: Computation of the Mahalanobis distance $d(\mathbf{x}, \mathbf{y})$ between two feature vectors $\mathbf{x}$ and $\mathbf{y}$ using a covariance estimated from a Gaussian mixture model.

The following approaches estimate covariance matrices from a Gaussian mixture model (cf. Figure 5.3). In general, the probability density function for a random variable $\mathbf{x}$ when described by a Gaussian mixture model is given by

$$p(\mathbf{x}) = \sum_{k=1}^{K} c_k \cdot \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \sum_{k=1}^{K} c_k \frac{1}{\sqrt{|2\pi\boldsymbol{\Sigma}_k|}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1}(\mathbf{x}-\boldsymbol{\mu}_k)\right),$$

where K is the number of mixtures and $\boldsymbol{\mu}_k$, $\boldsymbol{\Sigma}_k$ and c_k are the mean, covariance matrix and weight of the k-th Gaussian. A probability density function of a specific random variable is thus a linear combination of all Gaussians in the mixture model. The individual Gaussians can be interpreted as a set of *classes*, each described by a mean vector $\boldsymbol{\mu}_k$, a covariance matrix $\boldsymbol{\Sigma}_k$ and a prior probability c_k. The prior probability is usually determined by the quota of samples from the training set belonging to the relevant class in respect to the total number of training samples used to estimate the Gaussian mixture model, if the GMM estimation is realized by a classification algorithm (e.g. k-means clustering).

Analog to the probability density function from a Gaussian mixture model which is described as a linear combination of individual Gaussians, a covariance matrix that is estimated from a Gaussian mixture model can be defined as a linear combination of multiple covariance matrices. Such a combined covariance matrix is called *pooled covariance matrix*. In general, pooled covariance matrices are given by

$$\boldsymbol{\Sigma} = \sum_k w_k \Sigma_k \quad \text{with} \quad \sum_k w_k = 1.$$

The individual Gaussians of the Gaussian mixture model can be interpreted as a set of classes that each have a certain likelihood for the compared samples (i.e. a certain probability to produce the respective samples). Based on this consideration, a pooled covariance matrix can be composed as incorporation of the covariance matrices from

the Gaussian mixture model with respect to their likelihood for the sample feature vector in question. For this work, three different approaches were implemented and evaluated that each use a different method for incorporation of the classes from the model.

Linear Combination of Covariances from Single Best Matching Classes: One approach is to only incorporate the respective one best matching class for both compared feature vectors (i.e. with the highest likelihood) which is weighted by its a-priori probability. Let $\mathbf{\Sigma}_1(\mathbf{x}, \mathbf{y})$ denote the covariance matrix generated from this approach, let $k^*_{\boldsymbol{\xi}}$ denote the index of the class from the Gaussian mixture model that has the highest likelihood for $\boldsymbol{\xi}$ in respect to all other classes and let $p_k(\boldsymbol{\xi})$ denote the likelihood of class k for $\boldsymbol{\xi}$. $\mathbf{\Sigma}_1(\mathbf{x}, \mathbf{y})$ is then given by

$$\begin{aligned}
\mathbf{\Sigma}_1(\mathbf{x}, \mathbf{y}) &= \frac{1}{c_{k^*_\mathbf{x}} + c_{k^*_\mathbf{y}}} \cdot \left(c_{k^*_\mathbf{x}} \mathbf{\Sigma}_{k^*_\mathbf{x}} + c_{k^*_\mathbf{y}} \mathbf{\Sigma}_{k^*_\mathbf{y}} \right), \\
k^*_{\boldsymbol{\xi}} &= \arg\max_k p_k(\boldsymbol{\xi}), \\
p_k(\boldsymbol{\xi}) &= c_k \cdot \mathcal{N}(\boldsymbol{\xi} | \boldsymbol{\mu}_k, \mathbf{\Sigma}_k).
\end{aligned}$$

This comprises only a very limited amount of information from the mixture model. On the other hand the computational complexity is also limited (inuring to the benefit of desirable property 5).

Linear Combination of Covariances from Complete Mixture: Another approach is to incorporate all classes from the Gaussian mixture model and weigh them each according to their individual likelihood for both feature vectors. Let $\mathbf{\Sigma}_2(\mathbf{x}, \mathbf{y})$ denote the covariance matrix generated from this approach and let $p_k(\boldsymbol{\xi})$ denote the likelihood of class k for $\boldsymbol{\xi}$. $\mathbf{\Sigma}_2(\mathbf{x}, \mathbf{y})$ is then given by

$$\begin{aligned}
\mathbf{\Sigma}_2(\mathbf{x}, \mathbf{y}) &= \frac{1}{\sum_k p_k(\mathbf{x}) + \sum_k p_k(\mathbf{y})} \cdot \left(\sum_k p_k(\mathbf{x}) \mathbf{\Sigma}_k + \sum_k p_k(\mathbf{y}) \mathbf{\Sigma}_k \right), \\
&= \frac{1}{\sum_k (p_k(\mathbf{x}) + p_k(\mathbf{y}))} \cdot \sum_k (p_k(\mathbf{x}) + p_k(\mathbf{y})) \mathbf{\Sigma}_k, \\
p_k(\boldsymbol{\xi}) &= c_k \cdot \mathcal{N}(\boldsymbol{\xi} | \boldsymbol{\mu}_k, \mathbf{\Sigma}_k).
\end{aligned}$$

This approach includes information about every class in the final covariance matrix. The computational complexity on the other hand is quite high compared to the computation of $\mathbf{\Sigma}_1(\mathbf{x}, \mathbf{y})$.

Linear Combination of Covariances from N Best Matching Classes: A further approach is to incorporate a set of the N best matching classes (i.e. with highest likelihood) for both feature vectors, weighing them again each according to their individual likelihood for the respective feature vectors. Let $\mathbf{\Sigma}_3(\mathbf{x}, \mathbf{y})$ denote the covariance matrix generated from this approach, let $K^*_{\boldsymbol{\xi}}$ denote the set of class indices from the GMM for that the classes have a likelihood for the feature vectors that is not smaller

than for any class not in this set and let $p_k(\boldsymbol{\xi})$ denote the likelihood of class k for $\boldsymbol{\xi}$. $\boldsymbol{\Sigma}_2(\mathbf{x}, \mathbf{y})$ is then given by

$$\begin{aligned}
\boldsymbol{\Sigma}_3(\mathbf{x}, \mathbf{y}) &= \frac{1}{\sum_{k \in K^*_{\mathbf{x}}} p_k(\mathbf{x}) + \sum_{k \in K^*_{\mathbf{y}}} p_k(\mathbf{y})} \cdot \left(\sum_{k \in K^*_{\mathbf{x}}} p_k(\mathbf{x}) \boldsymbol{\Sigma}_k + \sum_{k \in K^*_{\mathbf{y}}} p_k(\mathbf{y}) \boldsymbol{\Sigma}_k \right), \\
K^*_{\boldsymbol{\xi}} &= \left\{ k \,|\, k \in K, p_k(\boldsymbol{\xi}) \geq p_{k'}(\boldsymbol{\xi}) \, \forall k' \notin K^*_{\boldsymbol{\xi}}, k' \in K, |K^*_{\boldsymbol{\xi}}| = n \right\}, \\
p_k(\boldsymbol{\xi}) &= c_k \cdot \mathcal{N}(\boldsymbol{\xi} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k).
\end{aligned}$$

This comprises a dynamically limitable amount of information about the classes in the Gaussian mixture model. This approach originates from the idea that a small set of classes dominates the relevance for the generated feature vector significantly in respect to all other classes.

Consecutively, measures are presented that operate on Gaussian distributions which model the sequence of feature vectors of the individual samples compared by the measure. Thus, this approach is substantially different from dynamic time warping.

5.3 Temporal Statistics

In this section distance measures are presented that are based on similarity measurement on Gaussian distributions. To this end it is necessary to construct a Gaussian model from a speech sample, i.e. from a sequence of feature vectors. This approach is often also referred to as *"bag of frames" approach* (cf. [ADP07]) or *MFCC statistics* (cf. [ME05]).

A multivariate Gaussian distribution is parametrized by a mean vector $\boldsymbol{\mu}$ and a covariance matrix $\boldsymbol{\Sigma}$. The probability density function of a Gaussian distribution for a random variable $\mathbf{x}$ is defined as

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{|2\pi\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right).$$

A Gaussian mixture model is a set of Gaussian distributions, each parametrized by separate mean vectors $\boldsymbol{\mu}_k$, covariance matrices $\boldsymbol{\Sigma}_k$ and weights c_k, its probability density function being defined as

$$p(\mathbf{x}) = c_k \cdot \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k).$$

A single Gaussian distribution that is to model a sequence of feature vectors can be estimated by simply computing the sample mean and the sample covariance for the data. A Gaussian mixture model on the other hand is estimated by using more complex approaches as for example the k-means algorithm and the expectation maximization algorithm.

In the implementation for this work, single Gaussians are estimated for the data (see Figure 5.4). This simplifies the estimation itself on the one hand and furthermore it substantially simplifies the computation of the distance measure, since for the

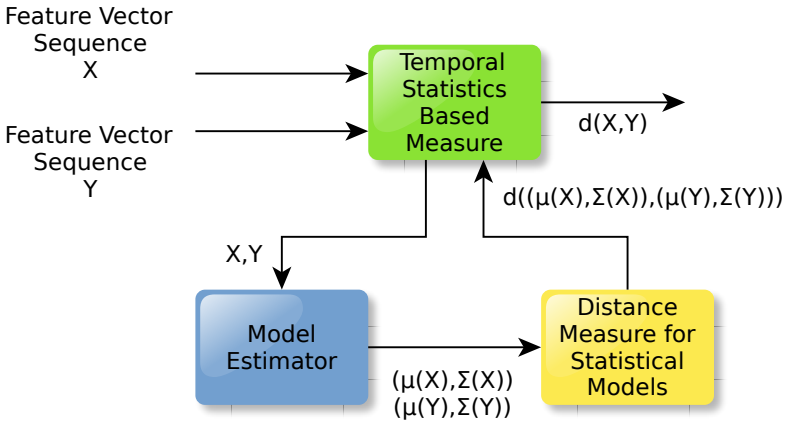


Figure 5.4: Computation of a distance distance $d(X, Y)$ between two feature vectors sequences X and Y based on multivariate Gaussian distributions estimated for both sequences.

Kullback-Leibler divergence (see section 5.4.1) there exists no closed term expression so that for GMMs it had to be approximated via stochastic methods (e.g. stochastic integration or Earth Movers distance, cf. [Jen+07]). The performance of distance measures that restrain on single Gaussians usually only drops insignificantly compared to the usage of GMMs with multiple Gaussians while there is a substantial gain in speed, especially for the Kullback-Leibler divergence (cf. [LS06; Pam06; ME05]; see section 4.4.2).

Let $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_K\}$ be a set of K feature vectors for a specific speech sample. The sample mean $\boldsymbol{\mu}_X$ and the sample covariance $\boldsymbol{\Sigma}_X$ are defined as

$$\boldsymbol{\mu}_X = \frac{1}{|X|} \sum_{\mathbf{x} \in X} \mathbf{x},$$
$$\boldsymbol{\Sigma}_X = \frac{1}{|X| - 1} \sum_{\mathbf{x} \in X} (\mathbf{x} - \boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu})^T.$$

The individual entries $(\boldsymbol{\mu}_X)_i$ and $(\boldsymbol{\Sigma}_X)_{ij}$ are then given by

$$(\boldsymbol{\mu}_X)_i = \frac{1}{K} \sum_{k=1}^{K} (\mathbf{x}_k)_i,$$
$$(\boldsymbol{\Sigma}_X)_{ij} = \frac{1}{K - 1} \sum_{k=1}^{K} ((\mathbf{x}_k)_i - \boldsymbol{\mu}_i)((\mathbf{x}_k)_j - \boldsymbol{\mu}_j).$$

The next section presents several distance measures which are used to determine the similarity of the Gaussian models estimated from the data.

5.4 Distance Measures for Temporal Statistics

In this section the similarity measures selected for implementation and evaluation in this thesis for comparison of Gaussian distributions are presented.

5.4.1 Kullback-Leibler Divergence

The *Kullback-Leibler divergence* is an information theoretic measure modeling the dissimilarity of two probability distributions. The Kullback-Leibler divergence of two density functions $p_1(\mathbf{x})$ and $p_2(\mathbf{x})$ is an asymmetric measure, formally defined as

$$d_{\mathrm{KL}}(p_1, p_2) = \int p_1(\mathbf{x}) \ln \frac{p_1(\mathbf{x})}{p_2(\mathbf{x})} \, d\mathbf{x}$$

It was originally introduced by Solomon Kullback and Richard Leibler in 1951 [KL51]. If the density probability distributions are multivariate Gaussian distributions, p_1 being parametrized by a mean vector $\boldsymbol{\mu}_1 \in \mathbb{R}^N$ and a covariance matrix $\boldsymbol{\Sigma}_1 \in \mathbb{R}^{N \times N}$ and p_2 being parametrized by $\boldsymbol{\mu}_2 \in \mathbb{R}^N$ and $\boldsymbol{\Sigma}_2 \in \mathbb{R}^{N \times N}$ respectively, there exists a closed term expression to compute the Kullback-Leibler divergence which is given by (cf. [Jen+09])

$$d_{\mathrm{KL}}(p_1, p_2) = \frac{1}{2} \left((\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T \boldsymbol{\Sigma}_1^{-1} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) + \mathrm{tr}(\boldsymbol{\Sigma}_1^{-1} \boldsymbol{\Sigma}_2) + \ln \left(\frac{|\boldsymbol{\Sigma}_1|}{|\boldsymbol{\Sigma}_2|} \right) - N \right).$$

The similarity measure that is to be developed needs to be symmetric, i.e. $d(p_1, p_2) = d(p_2, p_1)$ (see requirement 4). This can be achieved by including the asymmetric Kullback-Leibler divergence in both possible orientations. The symmetric Kullback-Leibler divergence $d_{\mathrm{SKL}}(p_1, p_2)$ is then given by

$$\begin{aligned}
d_{\mathrm{SKL}}(p_1, p_2) = \; & d_{\mathrm{SKL}}(p_2, p_1) = d_{\mathrm{KL}}(p_1, p_2) + d_{\mathrm{KL}}(p_2, p_1) \\
= \; & \frac{1}{2} \left(\begin{array}{c} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T \boldsymbol{\Sigma}_1^{-1} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) + (\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)^T \boldsymbol{\Sigma}_2^{-1} (\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1) \\ + \mathrm{tr}(\boldsymbol{\Sigma}_1^{-1} \boldsymbol{\Sigma}_2) + \mathrm{tr}(\boldsymbol{\Sigma}_2^{-1} \boldsymbol{\Sigma}_1) + \ln \left(\frac{|\boldsymbol{\Sigma}_1|}{|\boldsymbol{\Sigma}_2|} \right) + \ln \left(\frac{|\boldsymbol{\Sigma}_2|}{|\boldsymbol{\Sigma}_1|} \right) - 2N \end{array} \right) \\
= \; & \frac{1}{2} \left((\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T (\boldsymbol{\Sigma}_1^{-1} + \boldsymbol{\Sigma}_2^{-1}) (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) + \mathrm{tr}(\boldsymbol{\Sigma}_1^{-1} \boldsymbol{\Sigma}_2 + \boldsymbol{\Sigma}_2^{-1} \boldsymbol{\Sigma}_1) \right) - N.
\end{aligned}$$

Let p_X and p_Y be Gaussian probability densities describing two sets of feature vectors X and Y and let p_X be identified by a mean vector $\boldsymbol{\mu}_X \in \mathbb{R}^N$ and a covariance matrix $\boldsymbol{\Sigma}_X \in \mathbb{R}^{N \times N}$ and let p_X be identified by $\boldsymbol{\mu}_Y \in \mathbb{R}^N$ and $\boldsymbol{\Sigma}_Y \in \mathbb{R}^{N \times N}$ respectively. The symmetric Kullback-Leibler divergence is then given by

$$\begin{aligned}
d_{\mathrm{SKL}}(p_X, p_Y) = \; & \frac{1}{2} \left((\boldsymbol{\mu}_X - \boldsymbol{\mu}_Y)^T (\boldsymbol{\Sigma}_X^{-1} + \boldsymbol{\Sigma}_Y^{-1}) (\boldsymbol{\mu}_X - \boldsymbol{\mu}_Y) + \mathrm{tr}(\boldsymbol{\Sigma}_X^{-1} \boldsymbol{\Sigma}_Y + \boldsymbol{\Sigma}_Y^{-1} \boldsymbol{\Sigma}_X) \right) - N \\
= \; & \frac{1}{2} \left(\begin{array}{l} \displaystyle\sum_{i=1}^{N} ((\boldsymbol{\mu}_X)_i - (\boldsymbol{\mu}_Y)_i) \sum_{j=1}^{N} ((\boldsymbol{\Sigma}_X^{-1})_{ij} + (\boldsymbol{\Sigma}_Y^{-1})_{ij}) ((\boldsymbol{\mu}_X)_j - (\boldsymbol{\mu}_Y)_j) \\ \displaystyle + \sum_{i=1}^{N} \sum_{j=1}^{N} \left((\boldsymbol{\Sigma}_X^{-1})_{ij} (\boldsymbol{\Sigma}_Y)_{ji} + (\boldsymbol{\Sigma}_Y^{-1})_{ij} (\boldsymbol{\Sigma}_X)_{ji} \right) \end{array} \right) - N
\end{aligned}$$

5.4.2 Comparison of Model Parameters

Another approach to measure the similarity of two multivariate Gaussian distributions is to compare the model parameters (i.e. the sample means and covariances) which were previously computed from the feature vectors of the samples being compared (see [LS06]).

This comparison can be carried out for example by computing the Euclidean distance for between the sample mean vectors and sample covariance matrices of the speech samples. Analog to the Mahalanobis distance as local distance measure for dynamic time warping (see section 5.2.2) this comparison can also incorporate information about the variance of the data which can be estimated from a statistical model of the language.

Let X and Y be two sets of feature vectors which are modeled by two Gaussian densities that are identified each by mean vectors $\boldsymbol{\mu}_X \in \mathbb{R}^N$ and $\boldsymbol{\mu}_Y \in \mathbb{R}^N$ and covariance matrices $\boldsymbol{\Sigma}_X \in \mathbb{R}^{N\times N}$ and $\boldsymbol{\Sigma}_Y \in \mathbb{R}^{N\times N}$ respectively. The Euclidean distance $d_E(X,Y)$ between X and Y is then given by

$$\begin{aligned} d_E(X,Y) &= \sqrt{(\boldsymbol{\mu}_X - \boldsymbol{\mu}_Y)^T(\boldsymbol{\mu}_X - \boldsymbol{\mu}_Y) + (\boldsymbol{\Sigma}_X - \boldsymbol{\Sigma}_Y)^T(\boldsymbol{\Sigma}_X - \boldsymbol{\Sigma}_Y)} \\ &= \sqrt{\sum_{i=1}^{N}((\boldsymbol{\mu}_X)_i - (\boldsymbol{\mu}_Y)_i)^2 + \sum_{i=1}^{N}\sum_{j=1}^{N}((\boldsymbol{\Sigma}_X)_{ij} - (\boldsymbol{\Sigma}_Y)_{ij})^2}. \end{aligned}$$

Let $\boldsymbol{\Sigma}_{\boldsymbol{\mu}} \in \mathbb{R}^{N\times N}$ be the "covariance of means", denotig the covariance describing the distribution of the mean vectors in the speech sample space (as space of sets of feature vectors) and let $\boldsymbol{\Sigma}_{\boldsymbol{\Sigma}} \in \mathbb{R}^{N\times N}$ be the "covariance of covariances", denoting the covariance that describes the distribution of the covariance matrices in the speech sample space. The Mahalanobis distance $d_M(X,Y)$ between X and Y is then given by

$$\begin{aligned} d_M(X,Y) &= \sqrt{(\boldsymbol{\mu}_X - \boldsymbol{\mu}_Y)^T\boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{-1}(\boldsymbol{\mu}_X - \boldsymbol{\mu}_Y) + (\boldsymbol{\Sigma}_X - \boldsymbol{\Sigma}_Y)^T\boldsymbol{\Sigma}_{\boldsymbol{\Sigma}}^{-1}(\boldsymbol{\Sigma}_X - \boldsymbol{\Sigma}_Y)} \\ &= \sqrt{\begin{aligned} &\sum_{i=1}^{N}((\boldsymbol{\mu}_X)_i - (\boldsymbol{\mu}_Y)_i)\sum_{j=1}^{N}(\boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{-1})_{ij}((\boldsymbol{\mu}_X)_j - (\boldsymbol{\mu}_Y)_j) \\ &+ \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N}((\boldsymbol{\Sigma}_X)_{ki} - (\boldsymbol{\Sigma}_Y)_{ki})\sum_{l=1}^{N}(\Sigma_{\boldsymbol{\Sigma}}^{-1})_{kl}((\boldsymbol{\Sigma}_X)_{lj} - (\boldsymbol{\Sigma}_Y)_{lj}). \end{aligned}} \end{aligned}$$

The covariance of means $\boldsymbol{\Sigma}_{\boldsymbol{\mu}}$ and the covariance of covariances $\boldsymbol{\Sigma}_{\boldsymbol{\Sigma}}$ as a statistical model of the feature vectors in the target application area both have to be estimated on a large data set desribing the characterstics of the application (e.g. the language). In order to be able to compute them as sample covariances, the respective sample means have to be determined as prerequisites. Let $\boldsymbol{\mu}_{\boldsymbol{\mu}} \in \mathbb{R}^N$ be the "mean of means", denoting the mean describing the distribution of the mean vectors in the speech sample space; let $\boldsymbol{\mu}_{\boldsymbol{\Sigma}} \in \mathbb{R}^N$ be the "mean of covariances", denoting the covariance that describes the distribution of the covariance matrices in the speech sample space and let Ξ be a set of the sets X of feature vectors per speech sample in the data

estimation set. The mean of means $\boldsymbol{\mu}_{\boldsymbol{\mu}}$ and the mean of covariances $\boldsymbol{\mu}_{\boldsymbol{\Sigma}}$ are then given by

$$\boldsymbol{\mu}_{\boldsymbol{\mu}} = \frac{1}{|\Xi|} \sum_{X \in \Xi} \boldsymbol{\mu}_X,$$
$$\boldsymbol{\mu}_{\boldsymbol{\Sigma}} = \frac{1}{|\Xi|} \sum_{X \in \Xi} \boldsymbol{\Sigma}_X.$$

The covariance of means $\boldsymbol{\Sigma}_{\boldsymbol{\mu}}$ and the covariance of covariances $\boldsymbol{\Sigma}_{\boldsymbol{\Sigma}}$ can then be computed by

$$\boldsymbol{\Sigma}_{\boldsymbol{\mu}} = \frac{1}{|\Xi| - 1} \sum_{X \in \Xi} (\boldsymbol{\mu}_X - \boldsymbol{\mu}_{\boldsymbol{\mu}})(\boldsymbol{\mu}_X - \boldsymbol{\mu}_{\boldsymbol{\mu}})^T,$$
$$\boldsymbol{\Sigma}_{\boldsymbol{\Sigma}} = \frac{1}{|\Xi| - 1} \sum_{X \in \Xi} (\boldsymbol{\Sigma}_X - \boldsymbol{\mu}_{\boldsymbol{\Sigma}})(\boldsymbol{\Sigma}_X - \boldsymbol{\mu}_{\boldsymbol{\Sigma}})^T.$$

The individual entries $(\boldsymbol{\mu}_{\boldsymbol{\mu}})_i$, $(\boldsymbol{\mu}_{\boldsymbol{\Sigma}})_{ij}$, $(\boldsymbol{\Sigma}_{\boldsymbol{\mu}})_{ij}$, $(\boldsymbol{\Sigma}_{\boldsymbol{\Sigma}})_{ij}$ of the respective means and covariances are given by

$$(\boldsymbol{\mu}_{\boldsymbol{\mu}})_i = \frac{1}{|\Xi|} \sum_{X \in \Xi} (\boldsymbol{\mu}_X)_i,$$
$$(\boldsymbol{\mu}_{\boldsymbol{\Sigma}})_{ij} = \frac{1}{|\Xi|} \sum_{X \in \Xi} (\boldsymbol{\Sigma}_X)_{ij},$$
$$(\boldsymbol{\Sigma}_{\boldsymbol{\mu}})_{ij} = \frac{1}{|\Xi| - 1} \sum_{X \in \Xi} ((\boldsymbol{\mu}_X)_i - (\boldsymbol{\mu}_{\boldsymbol{\mu}})_i)((\boldsymbol{\mu}_X)_j - (\boldsymbol{\mu}_{\boldsymbol{\mu}})_j),$$
$$(\boldsymbol{\Sigma}_{\boldsymbol{\Sigma}})_{ij} = \frac{1}{|\Xi| - 1} \sum_{X \in \Xi} \sum_{k=1}^{N} ((\boldsymbol{\Sigma}_X)_{il} - (\boldsymbol{\mu}_{\boldsymbol{\Sigma}})_{il})((\boldsymbol{\Sigma}_X)_{jl} - (\boldsymbol{\mu}_{\boldsymbol{\Sigma}})_{jl}).$$

5.5 Used Software

For some of the functionality that is inherent to an implementation of the selected acoustic similarity measures external software packages were used. These packages are presented in this section, with respect to the functionality used.

5.5.1 ESMERALDA

ESMERALDA[1] ("Environment for Statistical Model Estimation and Recognition on Arbitrary Linear Data Arrays") is a set of applications that allow for setting up a system for automated speech recognition which was developed mainly by Gernot A. Fink[2] and Thomas Plötz[3] at Bielefeld University.

ESMERALDA is a toolkit for building statistical recognizers that operate on sequential data (e.g. speech, handwriting and biological sequences). It focuses primarily on support for continuous density hidden Markov models of different topologies and

[1] `http://www.irf.tu-dortmund.de/cms/en/IS/Research/ESMERALDA1`
[2] TU Dortmund University, Robotics Research Institute, Department of Intelligent Systems
[3] Newcastle University (UK), Culture Lab

definable internal structure. Moreover it supports incorporation of Markov chain models (as statistical n-gram models) for long-term sequential restrictions and Gaussian mixture models for general classification tasks. Methods supported in relation to generating and operating on mixture densities are k-means and LBG-based unsupervised mixture estimation, expectation maximization based model training, maximum a-posteriori adaptation and estimation of linear feature space transforms (PCA/LDA). It allows feature extraction with mel-frequency cepstral coefficients (MFCCs).

In the implementation for this thesis, methods to read the data format that stores a so-called codelibrary (i.e. a Gaussian mixture model) are used. This is convenient because by using the ESMERALDA codelibrary format for GMMs it is easily possible to provide general language information to distance measures (e.g. the Mahalanobis distance) that was estimated with ESMERALDA. Further descriptions of the ESMERALDA framework are given by [FP08; Fin99].

5.5.2 LAPACK

LAPACK[4] ("Linear Algebra PACKage") is a software package that provides methods for solving systems of linear equations, least-squares solutions for linear systems of equations, eigenvalue problems and singular value problems. It also supports the matrix factorizations (LU, Cholesky, QR, SVD, Schur / generalized Schur) associated to the above methods as well as other related computations as factorization reordering and condition number estimation. The package handles matrices as either dense or banded, but not as sparse in general. All methods are provided for either single or double precision values and for either real and complex matrices.

As underlying basis it uses the *BLAS*[5] ("Basic Linear Algebra Subprograms") which provides various routines for matrix multiplications and for solving triangular systems with multiple right-hand sides. LAPACK is written in Fortran 90 and is available as linkable library to implementations in C or C++ (which is relevant to this thesis).

In the implementation for this thesis, methods to perform matrix inversions (`SGETRI`) and to compute determinants of matrices (`SGETRF`) are used in order to compute the Mahalanobis distance (cf. section 5.2.2), to estimate covariance matrices via a Gaussian mixture model for calculating the Mahalanobis distance (cf. section 5.2.3), to compute the Kullback-Leibler divergence of multivariate Gaussian models (cf. section 5.4.1) and for the comparison of statistical model parameters via the Mahalanobis distance (cf. section 5.4.2). A further description of LAPACK is given by [And+99].

5.6 System Architecture

Altogether the methods to measure acoustic syllable similarity presented afore form a system which allows for the employment of several different ways to compute the distance between two sequences of feature vectors each representing a syllable speech

[4]The Netlib Repository: LAPACK – Linear Algebra PACKage (`http://netlib.org/lapack`).
[5]The Netlib Repository: BLAS – Basic Linear Algebra Subprograms (`http://netlib.org/blas`).

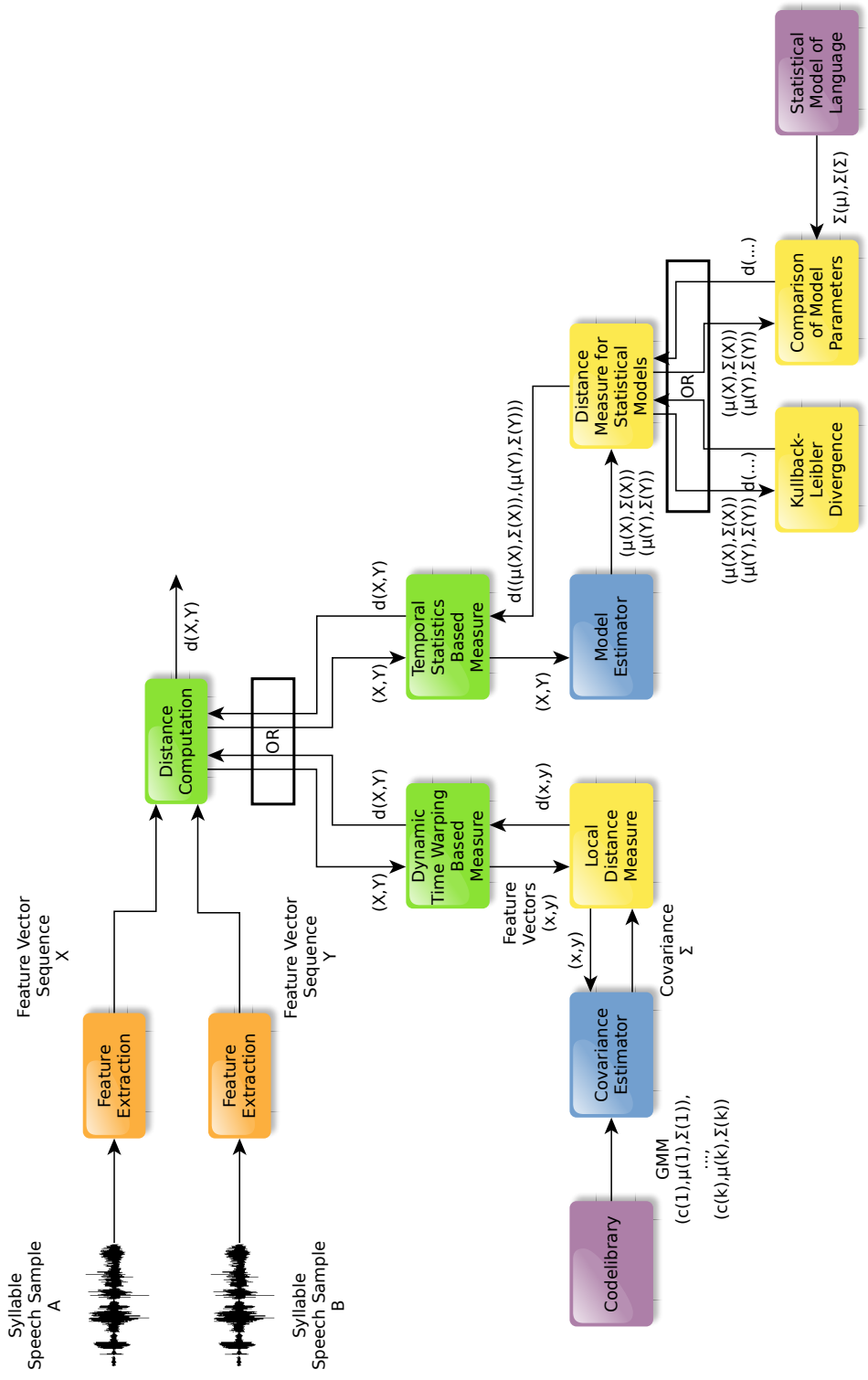


Figure 5.5: Overview of the computation for all similarity measures implemented and evaluated in this work. Orange: feature extraction; green: distance computation for feature vector sequences; yellow: secondary level distance computation for feature vectors and statistical models; blue: estimation of statistical model parameters; purple: statistical model representation which is estimated and stored as prerequisite.

signal. This comprises a set of conceptual modules which correspond to the methods presented in the preceding sections.

A chart that gives an overview of the computational flow for all distance measures is given by Figure 5.5. The distance computation for two syllables is performed on the basis of two sequences of feature vectors, each describing an input syllable speech sample. So prior to the actual computation of an acoustic distance features have to be extracted from both speech samples (represented in orange in the diagram). The distance computation for the feature vector sequences can be performed by either a dynamic time warping based measure or by a temporal statistics based measure (illustrated in green), formally mapping two sequences of feature vectors X and Y to a distance $d(X, Y)$.

The dynamic time warping approach uses a local distance measure which formally maps individual feature vectors $\mathbf{x}$ and $\mathbf{y}$ to a local distance $d(\mathbf{x}, \mathbf{y})$ (depicted in yellow). The local measure can either be the Euclidean distance or the Mahalanobis distance of which the latter incorporates a covariance matrix in the computation. This covariance matrix has to be estimated from a general statistical model of the features in the application area. The estimator (represented in blue) formally maps the feature vectors $\mathbf{x}$ and $\mathbf{y}$ to a covariance matrix $\mathbf{\Sigma}$ using a Gaussian mixture model $(c, \boldsymbol{\mu}, \mathbf{\Sigma})^K$ that is stored in the ESMERALDA codelibrary format (in purple). The computation of the Mahalanobis distance and the estimation of covariance matrices from a Gaussian mixture model involves the inversion of matrices (cf. sections 5.2.2 and 5.2.3) and in case of the covariance estimation as well the computation of matrix determinants (cf. section 5.2.3). For this, routines from the LAPACK software package are used (see section 5.5.2) in the implementation. The ESMERALDA codelibrary format used for the representation of the Gaussian mixture model which is needed for covariance estimation is processed with routines from the ESMERALDA software package (see section 5.5.1).

The temporal statistics based approach performs a distance computation for statistical models that were estimated on the feature vector sequences, in this approach interpreted as two mathematical sets of feature vectors without ordering. To this end, a model estimator (illustrated in blue) first estimates two multivariate Gaussian distributions on the input data. It hence formally maps two sets X and Y of feature vectors to statistical model parameters $\boldsymbol{\mu}_X$, $\mathbf{\Sigma}_X$, $\boldsymbol{\mu}_Y$ and $\mathbf{\Sigma}_Y$ representing the means and covariances of the Gaussians. The actual distance computation (depicted in yellow) can be provided by either the Kullback-Leibler divergence or a measure that compares the model parameters of the Gaussian distributions, formally mapping the model parameters to a distance $d((\boldsymbol{\mu}_X, \mathbf{\Sigma}_X), (\boldsymbol{\mu}_Y, \mathbf{\Sigma}_Y))$. The measure that compares the Gaussian model parameters uses either the Euclidean distance or a Mahalanobis distance, of which the latter needs a covariance matrix for the distribution of the feature means and the feature covariances in the application area. These covariance of means $\mathbf{\Sigma}_{\boldsymbol{\mu}}$ and covariance of covariances $\mathbf{\Sigma}_{\mathbf{\Sigma}}$ are taken from a statistical model for the sample features vector distributions in the application area (i.e. the language) which has to be generated as a prerequisite. The computation of the Kullback-Leibler divergence (cf. section 5.4.1) and the comparison of Gaussian model parameters (cf.

section 5.4.2) when carried out as Mahalanobis distance comprise the inversion of matrices, for which routines from the LAPACK software package are used (see section 5.5.2) in the implementation.

6 Evaluation

The development of suitable acoustic similarity measures for syllables that are able to reliably discriminate different syllables, thus allowing for application in a continuous syllable classification scenario, is subject to a comprehensive experimental evaluation.

In the first section, prerequisites and general conditions are presented which describe constraints and properties of the evaluation. The next section then motivates and in detail describes the methods used for the experimental evaluation. Subsequently several different evaluation tasks that were carried out for the different similarity measures are described in the ensuing sections. Lastly a conclusion is given that summerizes the most central observations from the experimental evaluation.

6.1 Prerequisites

There are certain prerequisites that need to be discussed prior to the acutal evaluation of the similarity measures. The following sections discuss what speech data is used for evaluation, which syllables are selected for being evaluated against, how the speech data which occurs as utterances in the speech corpus is segmented into syllables, which statistical models are used for estimation of the covariances used in the local Mahalanobis distance measure for the dynamic time warping approach and how the acoustic features processed by the similarity measures are computed.

6.1.1 Evaluation Speech Corpus

The speech data for the evaluation was taken from the german *Verbmobil* corpus. Verbmobil[1] was a long-term project for the recognition of spontaneous speech, the consecutive translation to a foreign language and the subsequent synthesis of the translated text, funded by the German federal ministry for education, science, research and technology (BMBF) and several industrial partners. For the evaluation in this work version 14.0 of the speech corpus was used, containing utterances from context of appointment negotiation.

Some relevant statistics about the data in the corpus are presented in Table 6.1. The corpus contains temporal annotations on word, syllable and phoneme level which is convenient to the evaluation of the acoustic similarity measures. The syllables are annotated in the German *Speech Assessment Methods Phonetic Alphabet (SAMPA)*[2].

[1] `http://verbmobil.dfki.de`; `http://dfki.de/web/research/iui/projects/base_view?pid=382`
[2] `http://www.phon.ucl.ac.uk/home/sampa/german.htm`

	Training Set	Evaluation Set
Utterances	13,567	343
Dialog Sessions	754	35
Speakers	1,345	43
Syllable Samples	462,662	9,943
Samples of Unique Syllables	3,619	925

Table 6.1: Statistics for characteristics of the German Vermobil corpus.

For the evaluation of this work, length, stress and tone marks which are accounted for in the SAMPA format were ignored so that syllable concepts which originally only differed by these modifiers in the annotation were merged. This reduced the number of syllable concepts from 5,330 to 3,619. An utterance in the Verbmobil corpus looks for example like this:

> "Hallo Herr Speyer gut dass ich Sie treffe wir wollten ja noch zu der Filiale AVBR nach Aachen hochfahren da sollten wir mal schauen ob wir in den nächsten zwei drei Monaten einen Termin finden".

If this utterance is represented by the reduced syllable concepts from the annotation it reads like this, the vertical bars denoting syllable borders:

> `hal|o|hE6|SpaI|6|gu|das|IC|zi|trEf|@|vi6|vOl|Qn|ja|nOx|tsu|`
> `de6|fil|ja|l@|Qa|faU|be|QE6|nax|Qa|x@n|hox|fa6n|da|zOl|tn|vi6|`
> `mal|SaUn|Op|vi6|n|den|neCs|tn|tsvaI|draI|mo|na|tn|aIn|n|tE6|`
> `min|fIn|hn`.

As it contains spontaneous speech from the context of appointment negotiation, the Verbmobil corpus does not contain highly distinctive emphasis of the syllables. As such it is in a way not optimal for the evaluation with particular respect to the tutoring scenario since in the tutoring scenario speech is uttered in the context of demonstration which causes a comparatively high amount of emphasized syllables. On the other hand Verbmobil provides a large set of syllable data for evaluation and comes with an accurate temporal annotation of the utterances, so that syllable speech samples can be extracted unreproachfully from complete utterances. Moreover spontaneous speech without exaggerated emphasis yields in fact a harder task for an acoustic similarity measure than demonstrative speech as occurring in the tutoring scenario but is nonetheless significant for assessing the measure. More information about Verbmobil can be found in [Wah00].

6.1.2 Selection of Syllables

The Verbmobil corpus contains a large set of different syllables, most of which with a huge set of representatives. In order to be able to perform a tractable evaluation a small subset of syllables is to be selected. A straightforward indicator to the qualification of syllables is their frequency in the corpus. For being able to correctly classify the syllables occurring in an application resembling the tutoring scenario a decent approach is to test the discrimination ability of the measure for those syllables that

Syllable	Abs. Frequency	Syllable	Abs. Frequency	Syllable	Abs. Frequency
n	8,385	bIs	3,684	tE6	2,736
IC	8,295	mi6	3,624	d@	2,666
ja	7,241	baI	3,572	am	2,548
das	6,792	van	3,563	b@	2,518
tn	6,414	zi	3,493	QEm	2,502
da	5,143	t@	3,454	QE	2,488
dan	4,843	vi6	3,450	n@	2,471
tak	4,836	@n	3,312	fo6	2,417
zo	4,801	mIt	3,295	min	2,371
@	4,562	b6	3,232	Un	2,350
d6	4,130	den	3,220	In	2,298
s	4,078	di	3,211	Uns	2,231
vi	3,979	vOx	3,095	a	2,199
tsen	3,912	de6	3,088	nOx	2,176
g@	3,790	t@n	2,957	Is	2,162

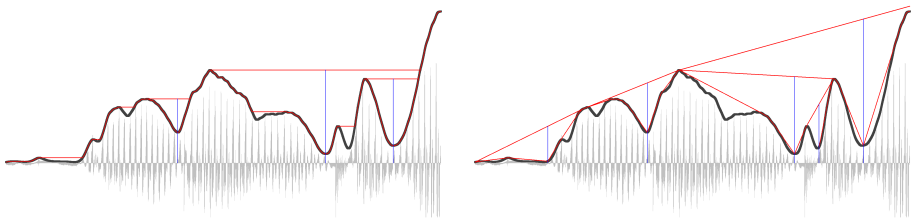
Table 6.2: Absolute frequencies of syllables in the Verbmobil corpus. The table displays the 45 most frequent syllables from the Verbmobil training set, sorted in descending order of their respective absolute frequency.

occur in the corpus most frequently. To this end, the ten most frequent syllables were selected for evaluation. Since the statistical models (i.e. the Gaussian mixture model) for estimation of the covariance matrices for the local Mahalanobis distance measure were estimated from the Verbmobil training set it is consistent to select the ten most frequent syllables from the training set as well, which provides that the covariances in the Gaussian mixture are estimated from a roughly commensurate and representative set of samples. Table 6.2 shows the 45 most frequent syllables from the Verbmobil training set.

6.1.3 Estimation of Syllable Borders

The Verbmobil corpus contains a temporal annotation on word, syllable and phoneme level that was created semi-automatically and which is highly accurate. Nevertheless the correct estimation of syllable borders in a scenario for continuous online speech classification like the tutoring scenario (cf. section 4.1) is a challenging task which is crucial to being able to correctly apply the distance measure and allow for a reasonable classification of syllables. Purely automatic syllable border estimation that operates on real-time data often does not work very reliably. To this end, the distance measures selected for implementation and evaluation in this work are to be assessed as well with syllable speech samples whose borders are estimated automatically.

In other work, a rudimentary syllable segmentation algorithm was implemented. This algorithm, presented by Villing et al. in [Vil+04], first computes the intensity envelope of the speech signal and then constructs a convex hull of the points that discretize the intensity envelope. The intensity envelope is computed by first band pass filtering the speech signal with a Butterworth filter and then low pass filtering the result. The envelope is then subtracted from the convex hull and a syllable boundary candidate is identified where the envelope has maximum distance from its convex hull. Successively



(a) (b)

Figure 6.1: Algorithm for automatic syllable segmentation. A speech signal which is represented by its amplitude over time is automatically segmented into syllables. Thick black line: intensity envelope; red lines: hull constructed from the envelope; blue vertical lines: estimated syllable borders. (a) Mermelstein hull; (b) convex hull (image source: Lars Schillingmann).

convex hulls for the subintervals determined by the syllable boundary candidates are computed and the algorithm is carried out recursively for the difference of the convex hulls for the subintervals and the intensity envelope. The selection of candidates is limited by two constraints. Firstly the difference between the intensity envelope and its hull must not under-run a certain threshold. Secondly the subintervals around a newly identified syllable border candidate must not fall below a certain size. This algorithm was originally presented by Paul Mermelstein in 1975 [Mer75]. In its original form the algorithm replaces the convex hull by a hull that is monotonically increasing from the beginning of the signal to the point with the highest amplitude and from there monotonically decreasing to the end. This hull is henceforth called Mermelstein hull. Figure 6.1 illustrates the syllable segmentation algorithm with using both the Mermelstein hull and the convex hull.

This basic strategy to estimate syllable borders generally oversegmentizes the input speech signal, even for optimized threshold values. To this end sophisticated strategies were developed to reject syllable candidates that would cause inappropriate segmentation. In [Vil+04] Villing et al. present such strategies, which are refined in their subsequent paper [VWT06]. However these syllable rejection strategies were not implemented in the rudimentary syllable segmentation algorithm that was used for the evaluation of this work. Consequently, the syllable segmentation generated by this means can be interpreted as being representative of an partly erroneous syllable segmentation in an application for continuous speech clustering resembling the tutoring scenario which imposes a challenging task to an acoustic similarity measure.

6.1.4 Statistical Models for Covariance Estimation

The statistical models (i.e. the Gaussian mixture model) used for the estimation of covariance matrices for the computation of the local Mahalanobis distance measure (see section 5.2.3) in the dynamic time warping approach have to be estimated from a representative set of feature vectors from the application domain.

They were thus estimated from the samples of the training set of the Verbmobil corpus (cf. section 6.1.1). For this evaluation, already prefabricated codelibraries

containing Gaussian mixtures estimated from the Verbmobil data with ESMERALDA were used.

Available to this evaluation were codelibraries that were estimated either by using a k-means algorithm to generate an initial codelibrary and then using an expectation maximization algorithm to successively adapt the estimation stronger to the training data or by using a LBG algorithm (also called Linde-Buzo-Gray algorithm). The codelibraries exist with both fully occupied covariance matrices and diagonal covariance matrices for the Gaussian mixture, each using 1024 classes (i.e. Gaussian mixture components).

6.1.5 Acoustic Features

The acoustic similarity measures presented in this work were developed in terms of operating on mel-frequency cepstral coefficients (MFCCs) as feature vectors. Consequently the feature vectors used in this evaluation are MFCCs as well. As statistical models used for estimation of covariance matrices already existing ESMERALDA codelibraries were used, so the computation of MFCCs from syllable speech segments is performed exactly as for the existing ESMERALDA code libraries. MFCC computation is performed with the ESMERALDA tool `dsp_fex` using the MFCC version "v1.4". Prior to the actual computation of the feature vectors for the entire syllable speech corpus a channel adaptation is performed on the entire corpus.

As discussed in section 2.2.2, the first 12 mel-frequency cepstral coefficients are taken for the feature vectors since they represent information solely from the vocal tract, ignoring the excitation of the glottal source (cf. source-filter model, Figure 2.2), supplemented by the signal energy of the corresponding speech frame. The feature vectors are completed by the first-order derivatives (velocity) and the second-order derivatives (acceleration) of the present coefficients and the energy so in total the feature vectors have 39 elements.

An interesting consideration is if the dynamic coefficients (i.e. the last 26 values of a feature vector) suffice for a decent similarity measure and the consequential discrimination ability for syllables. This is motivated by the idea that the essential information that allows for discrimination of different syllables is represented by the dynamics of the feature vectors rather than the stationary coefficients. In consequence would cut the time complexity of a local distance measure that operates on individual feature vectors by a third, thus enabling a gain in performance. Hence this is subsequently also subject to evaluation.

6.2 Methods

An appropriate evaluation of the acoustic similarity measures has to be conducted with deliberate evaluation methods. To this end the following sections present and motivate the methods used for evaluation of this work, of which the first one gives an

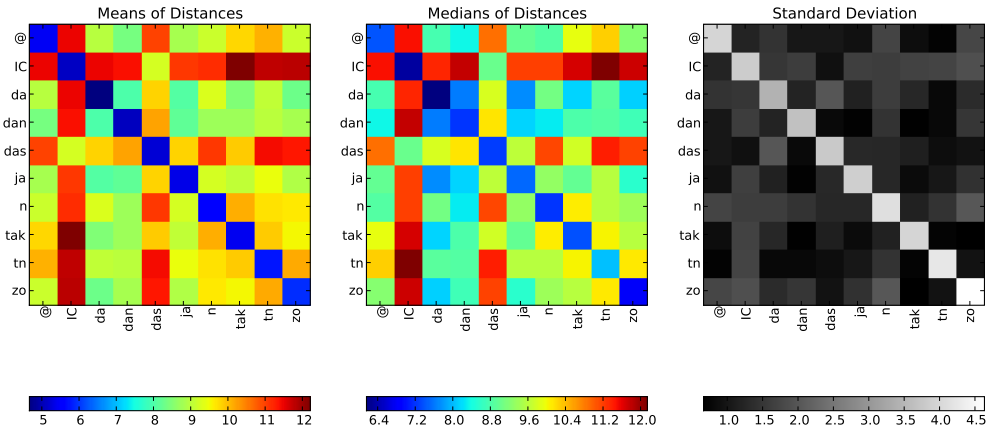


Figure 6.2: Example confusion matrices for a set of ten syllables. Arithmetic mean, median and standard deviation of acoustic distances computed for different pairs of sample representatives for each syllable concept. Each distance computation for two syllable concepts is carried out for five sample representatives each (result for DTW based measure with local Euclidean distance).

initial impression of the quality of a similarity measure, supplemented by a factual classification accuracy from experimental evaluation given by the second method.

6.2.1 Confusion Matrices

A straightforward method to assess the performance of the acoustic similarity measures presented in this work in a qualitative way is to display for a given set of syllables the average distance computed by the measure between all combinations of syllables in a matrix-like illustration.

This diagram can be called *confusion matrix* since it depicts the discrimination ability of the similarity measure for the occurring syllables. It illustrates which syllables are more likely to be confused with certain other syllables and which combinations of syllables can be kept apart reliably.

Figure 6.2 shows example confusion matrices for the arithmetic mean, the median and the standard deviation of acoustic distances computed for different pairs of sample representatives for each syllable concept in the test. Low values (blue) in the confusion matrices for the mean and median indicate that two syllables are in general identified as being similar by the measure since they have low distances on average. High values (red) indicate that two syllables are generally seen by the measure as being dissimilar since on average they induce high distances. In the standard deviation matrix low values (dark) indicate a low deviation respectively variance of the produced distances whereas high values (light) show a high deviation.

A decent measure consequentially yields low distance averages on the main diagonal of the confusion matrices and high distance averages outside of the main diagonal so that sample representatives of the one syllable concept are identified as being similar and sample representatives of different syllable concepts are identified as being

different. Moreover ideally the standard deviation is low for the distance values of any combination of syllables which means that the diagnosis made by the similarity measure is very homogeneous and there are by trend few outliers.

The confusion matrices for the mean and median acoustic distances allow for a qualitative assessment of the distance measure and give an impression if a similarity measure shows a tendency to perform well in practice. The arithmetic mean is accompanied by the median since it is more robust to outliers, so together these averages give an impression of how frequently outliers occur in the computed distances.

The evaluation of the individual measures is carried out each for the ten most frequent syllables concepts in the Verbmobil training set (see section 6.1.2), each represented by five random samples from the evaluation set of the corpus. The representation of each concept by multiple random samples is supposed to provide that samples which are unrepresentative (i.e. distorted or atypically pronounced) for the syllable concept in question become less meaningful to the resulting distance.

6.2.2 Nearest Neighbor Classification

When aiming at an evaluation method that resembles the application of the distance measure in the tutoring scenario (cf. section 4.1) a forthright approch is to use a 1-nearest neighbor classificator (1-NN) which is a special case of the widely used k-nearest neighbor algorithm (k-NN). This is a common method to classify test samples based on the closest training samples in the feature space.

By using k-NN classification, a test sample is assigned to the class (i.e. syllable concept) that is most common for its k nearest neighbors. In the case of the 1-NN classificator every test sample is assigned to the class of its nearest neighbor in the set of training samples. For the computation of the classification result of a given test sample this means that distances to every sample in the training set have to be computed. This evaluation method corresponds to an application resembling the tutoring scenario because in this scenario a newly uttered syllable is compared to every syllable previously captured to distinguish if they are likely to be representatives for the same syllable concept (see also chapter 1).

The nearest neighbor classification for the similarity measures proposed in this work is carried out for the ten most frequent syllable concepts from the Verbmobil training set (see section 6.1.2), each represented by one random training sample from the set. Ten random test samples for each syllable concept are then selected from the Verbmobil test set and are then classified. Since only one training sample is selected for each concept, the entire classification is repeated ten times to prevent unrepresentative classification results due to the selection of training samples that are atypical for their syllable concept (i.e. distorted or atypically pronounced). This results in 100 classification attempts per syllable and 1,000 classifications in total. For this 10,000 distances have to be computed. Table 6.3 shows the results of an example nearest neighbor classification, displaying the individual quotas and accuracies for the different syllables together with an overall accumulated accuracy which serves as

Syllable	Quota	Accuracy
@	22/100	0.22
zo	31/100	0.31
dan	46/100	0.46
n	35/100	0.35
tn	29/100	0.29
das	60/100	0.60
IC	79/100	0.79
da	41/100	0.41
ja	47/100	0.47
tak	16/100	0.16
Accumulated	406/1000	0.41

Table 6.3: Example nearest neighbor classification result. Individual quotas and accuracies for different syllable concepts together with an overall accumulated accuracy (result for DTW based measure with local Euclidean distance).

quantitative figure to assess the quality of the distance measures in the subsequent sections.

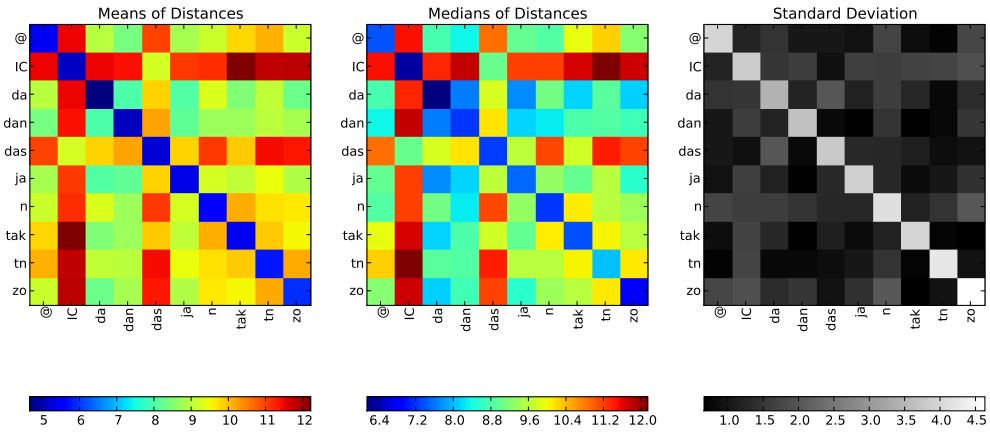
6.3 Dynamic Time Warping with Mahalanobis Distance

The following sections evaluate the dynamic time warping approach to measure acoustic similarity (see section 5.1) which uses a local distance measure to compute distances between individual feature vectors. As local distance measure the local Mahalanobis distance was proposed (cf. section 5.2.2) which is to be evaluated under several different aspects.

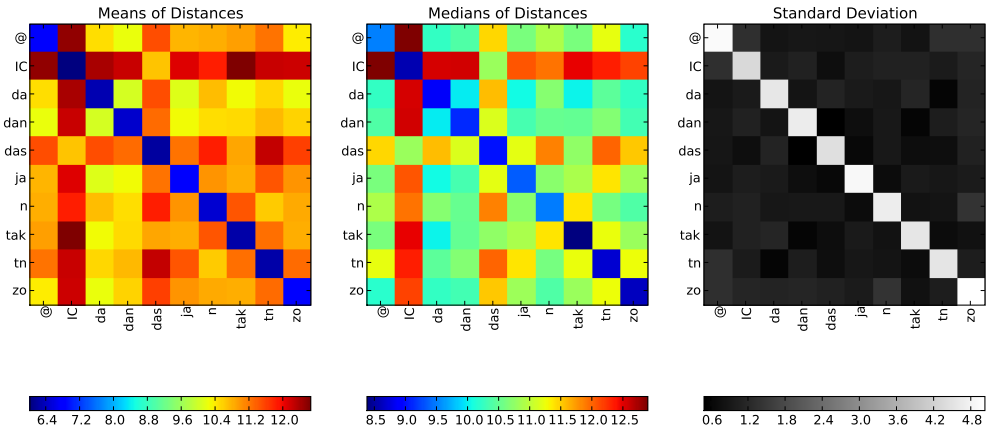
6.3.1 Mahalanobis Distance vs. Euclidean Distance

The Euclidean distance was imposed as a baseline local distance measure, serving as a reference frame for assessment of the Mahalanobis distance. It allows for a first impression of the performance gain that is achieved through the inherent property of the Mahalanobis distance, the variance normalization of the feature vectors for distance computation. It represents one of the most fundamental measures for vector spaces, being prevalent in a large set of applications as a naive approach for distance measurement. In this evaluation the Mahalanobis distance is compared to the Euclidean distance.

The Mahalanobis distance is computed using fully occupied covariance matrices which are estimated from a codelibrary that was optimized with the expectation maximization algorithm. The covariance estimation is based on a linear combination of the single best matching classes for the feature vectors in question. The syllable samples in the evaluation are randomly selected from utterances of arbitrary speakers in the corpus test set. The syllable segmentation is provided by the existing annoation that comes with the Verbmobil corpus.



(a) Euclidean Distance



(b) Mahalanobis Distance (fully occupied covariances, optimized codelibrary, estimation via single best classes)

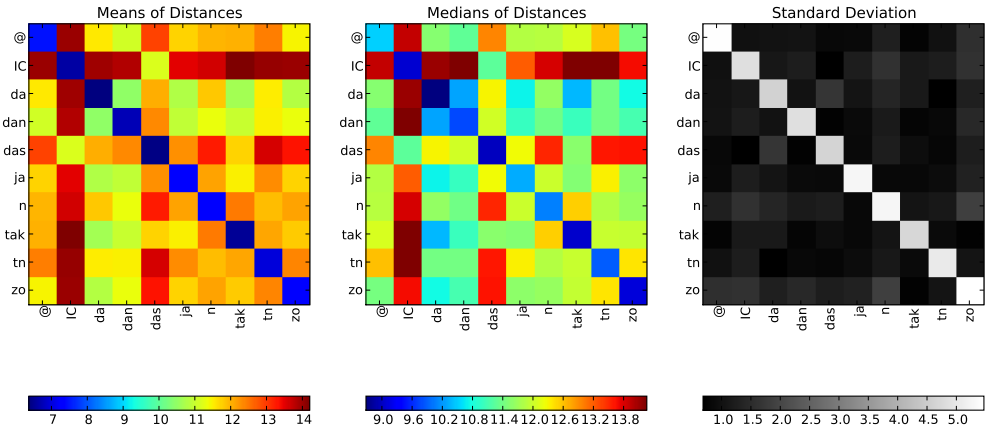
Figure 6.3: Mahalanobis Distance vs. Euclidean Distance: Confusion Matrices (annotated segmentation, arbitrary speakers)

Figure 6.3 shows confusion matrices for this evaluation task. The confusion matrices for the Mahalanobis distance (Figure 6.3(b)) show that the measure by trend is able to discriminate different syllables and correctly identify similar syllables since for the mean and median of the computed distances there are low values on the main diagonal and high values outside of the main diagonal. This contrast is higher for the arithmetic mean and lower for the median which indicates that there is a certain amount of outliers in the computed distances which systematically distort the arithmetic mean. It is apparent that some syllables (e.g. `IC`, `das`) are much less likely to be confused with other syllables and that some syllables (e.g. `da`, `dan`) are more likely to be confused with others. Moreover some syllable combinations (e.g. `IC/@`, `IC/tak`) are less likely to be confused than other combinations (e.g. `da/dan`). The standard deviation matrix shows that distances computed for sample representatives belonging to a common syllable concept have on average a higher standard deviation than distances computed for sample representatives of different concepts. This is because the distance means for common syllable concepts have systematically low values so that values dissenting from these means cause a higher standard deviation than for distances between samples of different syllable concepts which have a systematically smaller displacement from the corresponding mean values, which are high in comparison to the diagonal entries. The tendencies described afore propagate through the evaluation of all investigated aspects of the Mahalanobis distance.

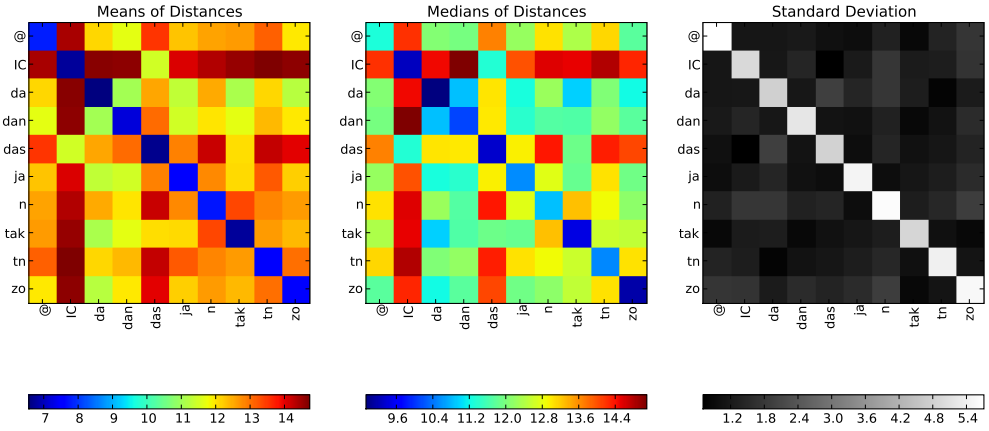
In comparison to the Euclidean distance, the Mahalanobis distance has clearly a better discrimination ability. For the arithmetic mean distance values outside of the main diagonal of the confusion matrices are systematically relatively higher than for the Euclidean distance whereas the diagonal entries are systematically lower. For the median this is not as obvious as for the mean. By trend, the distance values outside of the main diagonal increase but for some syllable combinations, they decrease. The standard deviation matrix shows that the variance of the computed distance values systematically decreases compared to the Euclidean distance, hence the Mahalanobis distance produces more homogeneous results. Altogether this promises a gain in performance which was evaluated by nearest neighbor classification, the results of which are presented in Table 6.4. The table confirms that there is indeed a substantial gain in performance.

Measure	Accuracy
Euclidean Distance	41%
Mahalanobis Distance	54%

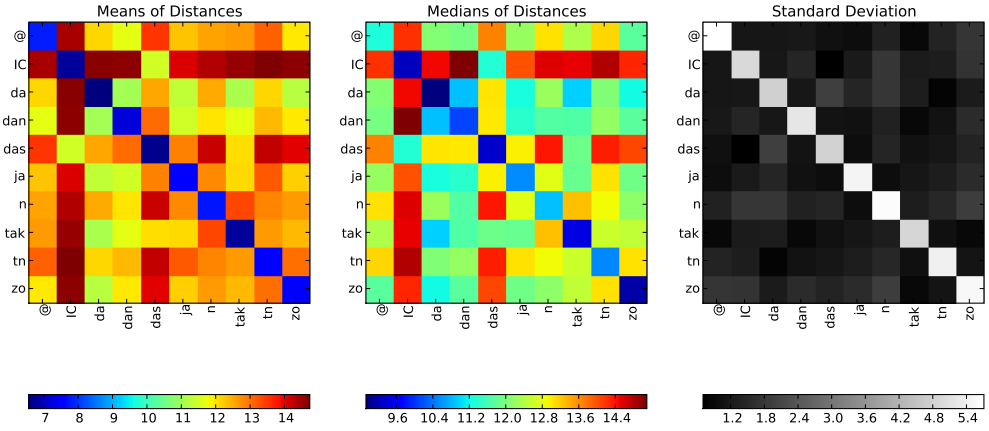
Table 6.4: Mahalanobis Distance vs. Euclidean Distance: NN-Classification (annotated segmentation, arbitrary speakers; Mahalanobis distance with fully occupied covariances, optimized codelibrary, estimation via single best classes)



(a) Combination from Single Best Matching Classes



(b) Combination from 50 Best Matching Classes



(c) Combination from Complete Mixture

Figure 6.4: Techniques for Covariance Estimation from a Gaussian Mixture Model: Confusion Matrices (annotated segmentation, arbitrary speakers, diagonal covariances, optimized codelibrary)

6.3.2 Techniques for Covariance Estimation from a Gaussian Mixture Model

The covariance that is used for the Mahalanobis distance has to be estimated from a general statistical model of the feature vectors in the target application area, i.e. a Gaussian mixture model from an ESMERALDA codelibrary. This work proposed several different methods to select Gaussian mixture components (classes) whose covariance matrices are combined as weighted sum, i.e. as linear combination (see section 5.2.3).

One approach is to only incorporate the respective one best matching class for both compared feature vectors which is weighted by its a-priori probability. Another approach is to incorporate all classes from the Gaussian mixture model and weigh them each according to their individual likelihood for both feature vectors. A third approach is to incorporate a set of the N best matching classes for both feature vectors, weighing them again each according to their individual likelihood for the respective feature vectors. This approach is evaluated for $N = 50$.

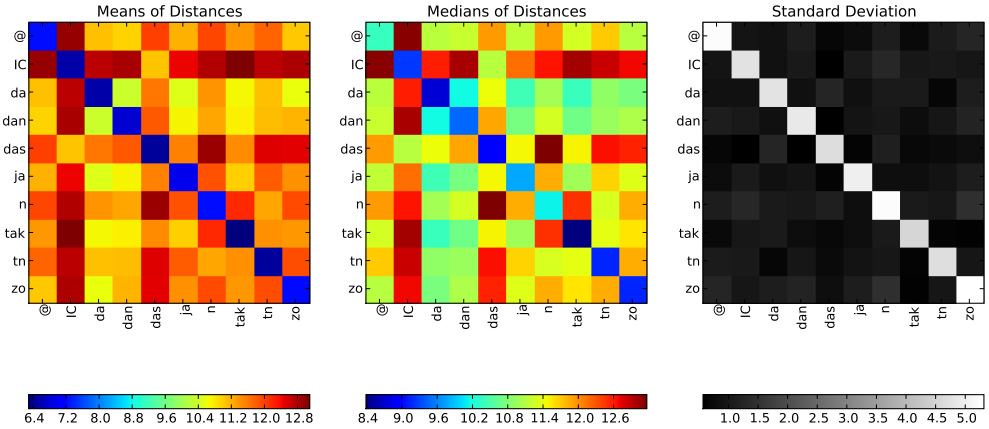
The Mahalanobis distance is computed using diagonal covariance matrices which are estimated again from a codelibrary that was optimized with the expectation maximization algorithm. The syllable samples in the evaluation are again randomly selected from utterances of arbitrary speakers in the corpus test set. The syllable segmentation is again provided by the existing annotation that comes with the Verbmobil corpus.

Figure 6.4 shows confusion matrices for this evaluation task. The confusion matrices show only very little differences for the different approaches to estimate the covariance matrices from Gaussian mixture model. In the confusion matrices for the median of the computed distances it is distinguishable that the distances for sample representatives of a common syllable concept decrease with increasing number of Gaussian mixture components for the covariance estimation whereas distances for representatives of different syllable concepts increase by trend. Interestingly, distances that engage syllables which are particularly well discriminable from others (e.g. `IC`) decrease with increasing number of mixture components. Results of a nearest neighbor classification which was performed in the context of this evaluation task are presented in Table 6.5.

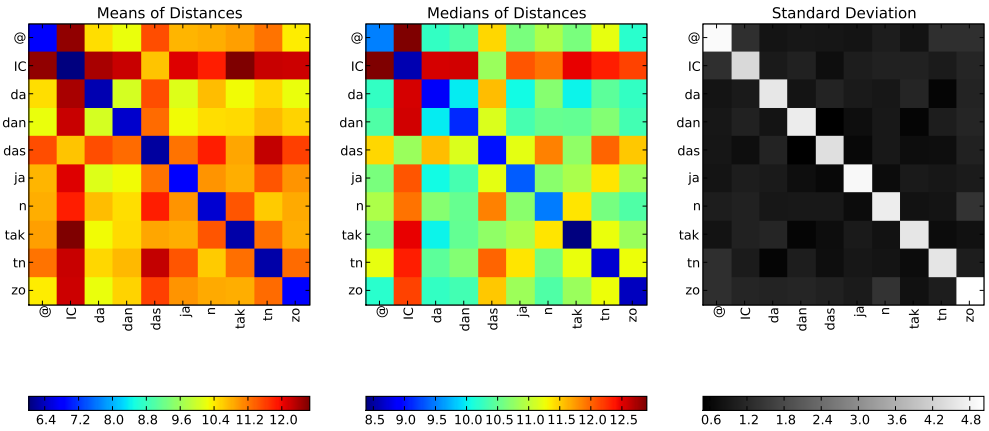
Method	Accuracy
Combination from Single Best Matching Classes	46%
Combination from 50 Best Matching Classes	48%
Combination from Complete Mixture	50%

Table 6.5: Techniques for Covariance Estimation from a Gaussian Mixture Model: NN-Classification (annotated segmentation, arbitrary speakers, diagonal covariances, optimized codelibrary)

It is evident that the measure accuracy increases with increasing number of mixture components. This is plausible because the Gaussian mixture model that is used for



(a) Diagonal Covariance Matrices



(b) Fully Occupied Covariance Matrices

Figure 6.5: Diagonal Covariance Matrices vs. Fully Occupied Covariance Matrices: Confusion Matrices (annotated segmentation, arbitrary speakers, optimized codelibrary, estimation via single best classes)

estimating the covariance matrix can only in its entirety accurately describe a feature vector since a Gaussian mixture model is motivated as linear combination of single multivariate Gaussian distributions in the first place. The approach to estimate the covariance matrix via a combination of the N best matching classes is in practice not operable since it is considerably too slow due to the necessary sorting algorithm. The approach that uses all mixture components (i.e. 1024) to estimate the covariance is in practice roughly twice as slow for the computation of a single distance compared to the approach that uses only one mixture component per sample while the gain in accuracy is small (4%).

6.3.3 Diagonal Covariance Matrices vs. Fully Occupied Covariance Matrices

The covariance matrices used in the Mahalanobis distance can either be diagonal or fully occupied which supposedly affects the accuracy of the distance measure while definitely yielding substantial impact to the time complexity of the computation, since for fully occupied covariance matrices more matrix elements have to be included in

the calculation than for diagonal matrices. Assessing the gain in accuracy through the usage of fully occupied covariances instead of diagonal ones motivates the evaluation thereof. The covariances for this evaluation task are again estimated from a codelibrary optimized via the EM algorithm as a linear combination of the single best matching classes for the feature vectors in question. Again the syllable samples are randomly selected from utterances of arbitrary speakers in the test set of the corpus. The syllable segmentation is again provided by the existing annotation that comes with the Verbmobil corpus.

Figure 6.5 shows confusion matrices for this evaluation task. The confusion matrices for the arithmetic mean of the distances reveal no significant difference. The median shows that for fully occupied covariance matrices distances between sample representatives of a common syllable concept decrease on average. The distances between samples of different syllable concepts also decrease by trend; however the displacement between the diagonal entries of the confusion matrix and the entries outside the main diagonal seems to rise as well. Together this promises a gain in accuracy through the usage of fully occupied covariance matrices instead of diagonal covariances. Table 6.6 presents results of a nearest neighbor classification which was performed in the context of this evaluation task.

Method	Accuracy
Diagonal Covariance Matrices	46%
Fully Occupied Covariance Matrices	54%

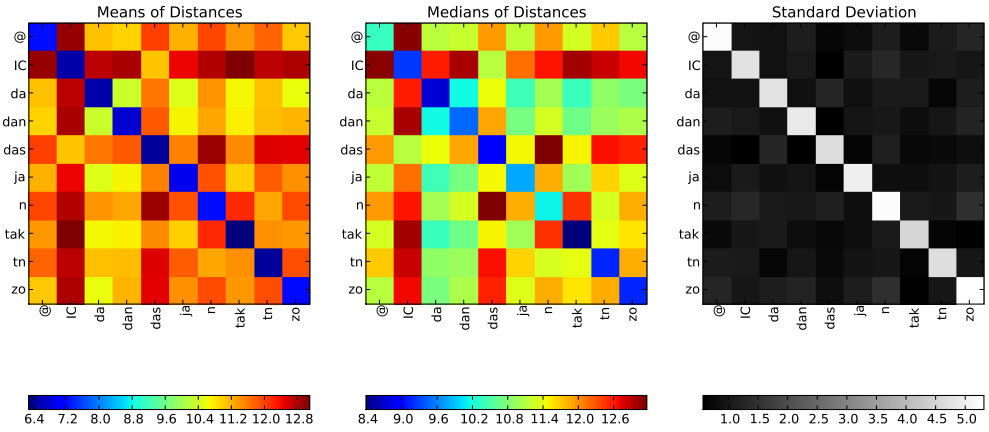
Table 6.6: Diagonal Covariance Matrices vs. Fully Occupied Covariance Matrices: NN-Classification (annotated segmentation, arbitrary speakers, optimized codelibrary, estimation via single best classes)

Obviously the accuracy substantially increases when using fully occupied covariance matrices instead of diagonal covariances. This can be explained with diagonal covariance matrices only containing the simple variances of the features, not incorporating the covariances which model dependencies between different features. This means that information is omitted when using diagonal covariances. For fully occupied covariances a distance computation takes roughly 7.8 s on an up-to-date personal computer[3] compared to approximately 0.2 s for diagonal the covariance[4]. Consequently, the usage of full covariance matrices is much to slow to be applied in online processing for a continuous classification task like the tutoring scenario.

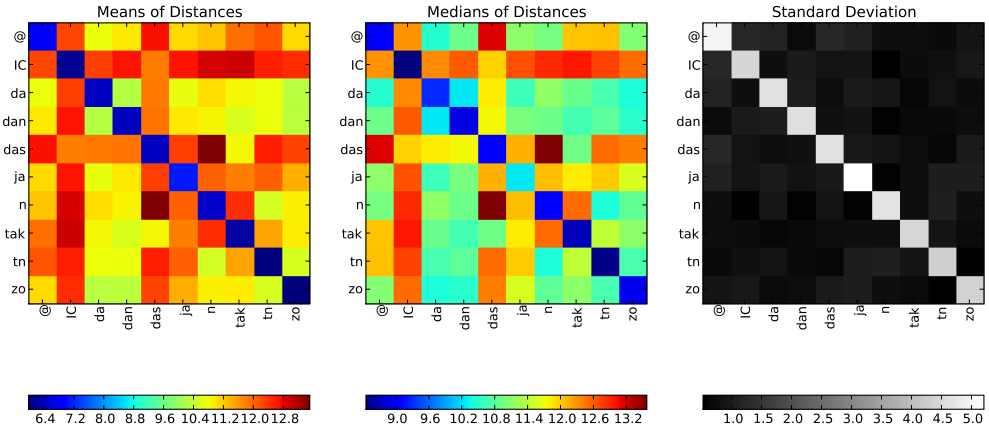
6.3.4 One Speaker vs. Arbitrary Speakers

In an application resembling the tutoring scenario (cf. section 4.1) a classification session concerns only the computation of acoustic distances for syllable samples that were uttered by one speaker, not arbitrary different speakers as in the other evaluation tasks. An interesting consideration is thus how the accuracy of the measure performs if only one speaker is used instead of multiple speakers.

[3] Intel Core2 Quad CPU Q9450 (2.66 GHz), 4 GB RAM
[4] This was evaluated by taking an average for 10,000 distance computations.



(a) Arbitrary Speakers



(b) One Speaker

Figure 6.6: One Speaker vs. Arbitrary Speakers: Confusion Matrices (annotated segmentation, diagonal covariances, optimized codelibrary, estimation via single best classes)

The Mahalanobis distance is again computed using diagonal covariance matrices which are estimated from a codelibrary that was optimized with the EM algorithm. The syllable segmentation is provided by the existing annotation that comes with the Verbmobil corpus. Figure 6.6 shows confusion matrices for this evaluation task. The confusion matrices show that for the median the displacement of the distance averages on the diagonal in respect to distances for non-diagonal values increases when using syllable samples of only one speaker, which promises a gain in accuracy. Table 6.7 shows the results of a nearest neighbor classification that was performed in context of this evaluation. The table indicates a substantial gain in measure accuracy when using only samples of a single speaker. This is plausible since pronunciation variations for syllables of a single speaker are less likely to be as high as for different speakers.

Method	Accuracy
Arbitrary Speakers	46%
One Speaker	60%

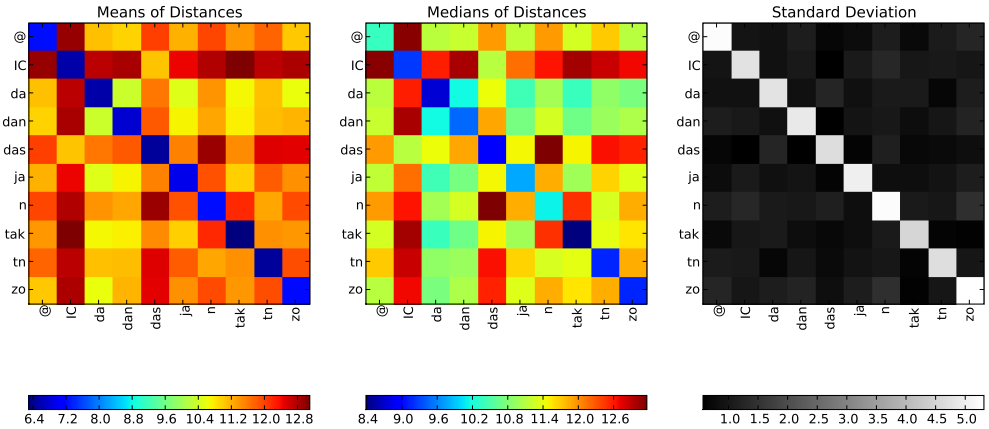
Table 6.7: One Speaker vs. Arbitrary Speakers: NN-Classification (annotated segmentation, diagonal covariances, optimized codelibrary, estimation via single best classes)

6.3.5 Automatic Segmentation vs. Annotated Segmentation

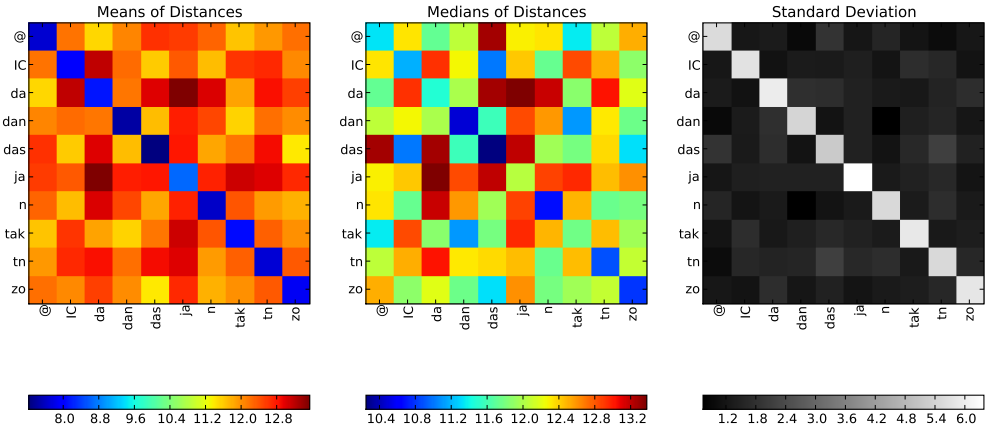
The correct estimation of syllable borders in a scenario for continuous online speech classification like the tutoring scenario (cf. section 4.1) is a challenging task which is crucial to being able to correctly apply the distance measure and allow for a reasonable classification of syllables. Purely automatic syllable border estimation that operates on real-time data often does not work very reliably. Consequently the Mahalanobis distance is to be assessed with syllable speech samples whose borders are estimated automatically as well in order to provide an impression of how the distance measure will perform in an actual application.

The evaluation is again carried out for diagonal covariance matrices which are estimated from a codelibrary post-processed with the EM algorithm after initialization. The covariance estimation is again based on a linear combination of the single best matching classes for the feature vectors in question. The syllable samples in the evaluation are again randomly selected from utterances of arbitrary speakers in the corpus test set.

Figure 6.7 shows confusion matrices for this evaluation task. The confusion matrices already reveal that the discrimination capabilities of the similarity measure are substantially worse than for the syllable segmentation taken from the Verbmobil corpus annotation. Results of a nearest neighbor classification which was performed in the context of this evaluation task are presented in Table 6.8. Even without analyzing the characteristics respectively the systematics of how syllable borders estimated via the automatic segmentation differ from a correct segmentation (i.e. the corpus annotation

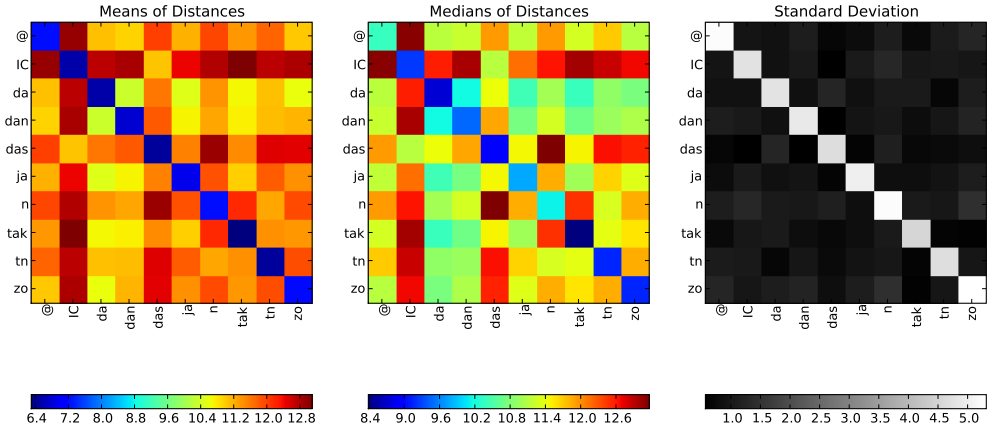


(a) Segmentation from Corpus Annotation

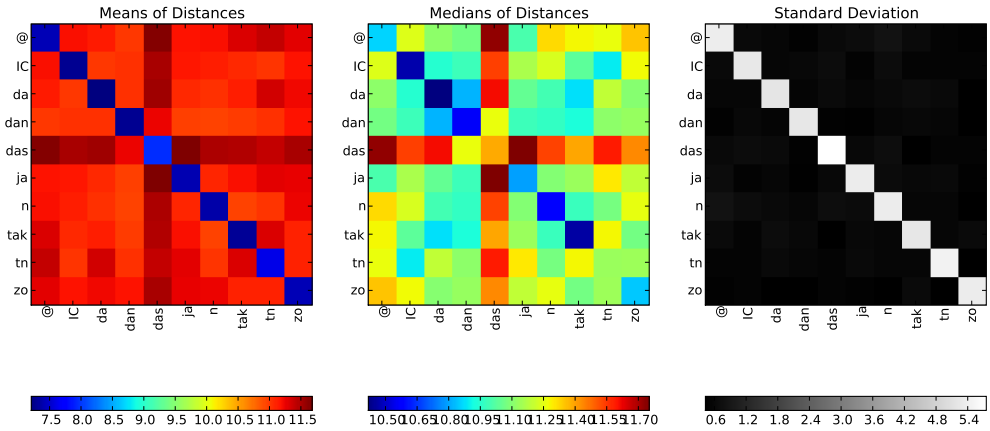


(b) Automatic Segmentation

Figure 6.7: Automatic Segmentation vs. Annotated Segmentation: Confusion Matrices (arbitrary speakers, diagonal covariances, optimized codelibrary, estimation via single best classes)



(a) No Context



(b) Left and Right Syllable

Figure 6.8: Consideration of Acoustic Context: Confusion Matrices (annotated segmentation, arbitrary speakers, diagonal covariances, optimized codelibrary, estimation via single best classes)

as reference) it can be derived that an accurate syllable segmentation is essential to the similarity measure. In a complete framework which also incorporates segmentation of speech into syllables this would be a starting point for an improvement of the overall performance of the system.

Method	Accuracy
Segmentation from Corpus Annotation	46%
Automated Segmentation	19%

Table 6.8: Automatic Segmentation vs. Annotated Segmentation: NN-Classification (arbitrary speakers, diagonal covariances, optimized codelibrary, estimation via single best classes)

6.3.6 Consideration of Acoustic Context

A question that investigates the implications of a systematic undersegmentation of the utterance speech signal is the addition of the features of both the syllable left from

the syllable in question and of its right syllable respectively. Thus syllables generally are represented by a broader aperture of the utterance feature sequence. The diagonal covariances for the Mahalanobis distance in this evaluation task are again estimated from a codelibrary optimized via the EM algorithm as a linear combination of the single best matching classes for the feature vectors in question. Again the syllable samples are randomly selected from utterances of arbitrary speakers in the test set of the corpus. The syllable segmentation is again provided by the existing annotation that comes with the Verbmobil corpus.

Figure 6.8 shows confusion matrices for this evaluation task. The displacement of the confusion matrices for the mean and the median in respect to each other indicate that there are substantially more outliers in the computed distances using this approach, which nevertheless result in low mean values on the diagonal and comparatively high mean values outside of the main diagonal. This is plausible since together the outliers roughly compensate for each other in the arithmetic mean. Table 6.9 shows results of a nearest neighbor classification that was performed in the context of this evaluation task. It is evident that due to adding the left and right syllable features to the respective distance computations the measure accuracy significantly worsens. This can be explained with the implicit adding of combinations of feature contexts left and right from a syllable which add to its characterization, thus causing a higher variance in the data. To this end, in can be concluded that an accurate estimation of syllable borders is immanently essential to the accuracy of the distance measure.

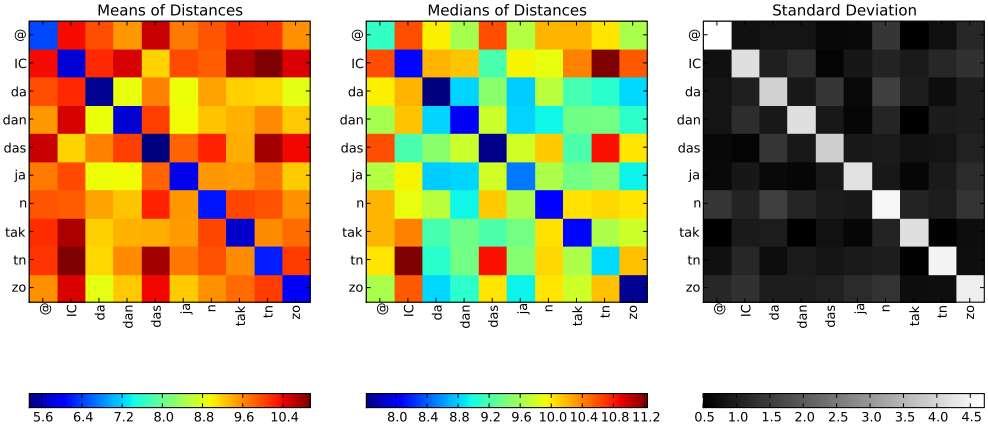
Method	Accuracy
No Context	46%
Context of Left and Right Syllable	13%

Table 6.9: Consideration of Acoustic Context: NN-Classification (annotated segmentation, arbitrary speakers, diagonal covariances, optimized codelibrary, estimation via single best classes)

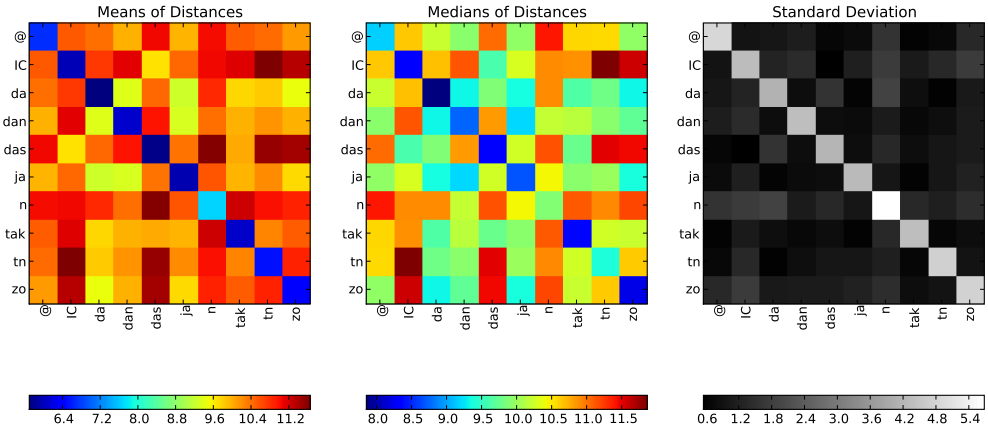
6.3.7 Consideration of Dynamic Features

An interesting question is if considering only the dynamic part of the feature vectors (i.e. the last 26 coefficients) leads to an improvement of the distance measure, causing a better discrimination ability for syllables. This is motivated by the idea that the essential information that allows for discrimination of different syllables is represented by the dynamics of the feature vectors rather than the stationary coefficients. A beneficial side effect would be a substantial gain in time complexity since the feature vector size is cut by a third.

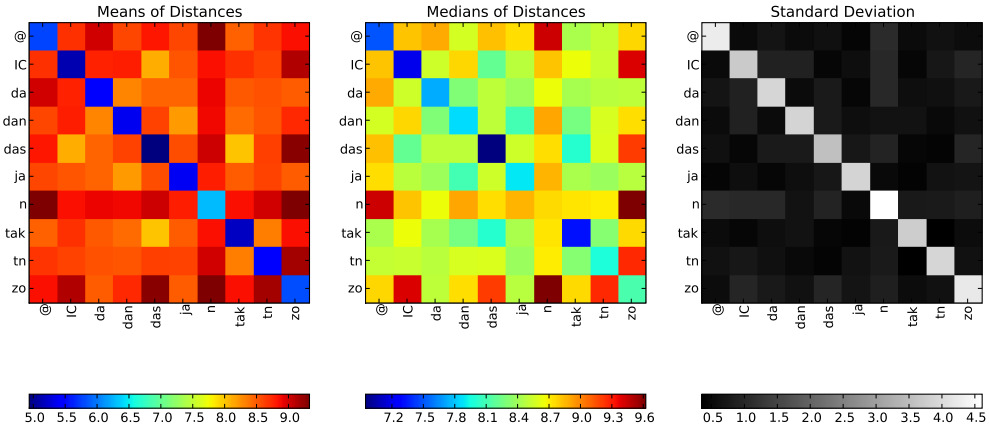
This evaluation task is carried out for diagonal covariance matrices, which have to be estimated from a codelibrary. A first approach is to take an already existing codelibrary which was estimated for feature vectors having 39 coefficients, thus also containing stationary information, and trim the needed mean vectors and covariance matrices. Strictly speaking this is illegitimate since the Gaussian mixture model in the



(a) Codelibrary estimated with k-means for stationary and dynamic features



(b) Codelibrary estimated with k-means for dynamic features



(c) Codelibrary estimated with LBG for dynamic features

Figure 6.9: Consideration of Dynamic Features: Confusion Matrices (annotated segmentation, arbitrary speakers, diagonal covariances, optimized codelibrary, estimation via single best classes)

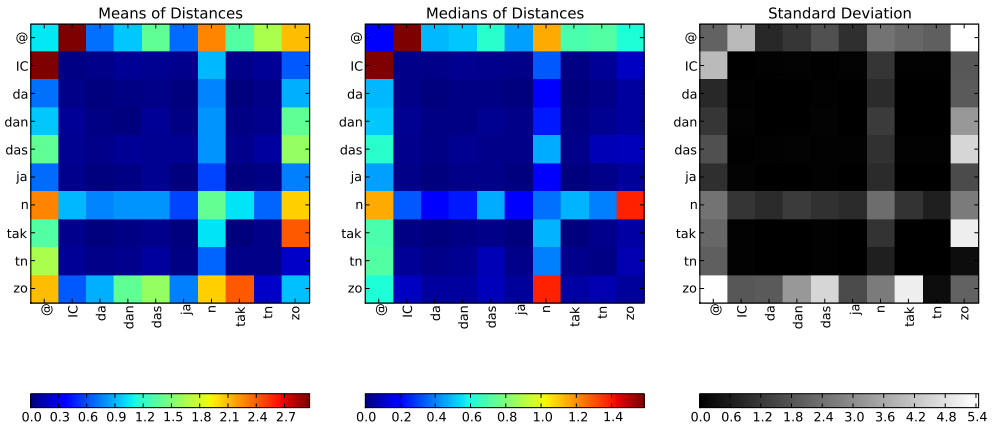


Figure 6.10: Kullback-Leibler Divergence on Gaussian Models: Confusion Matrices (annotated segmentation, arbitrary speakers)

codelibrary was originally estimated for vectors with stationary information as well and not only dynamic information as implicated by this approach. Another approach is to estimate a new codelibrary which is fitted for feature vectors containing only dynamic information. Using ESMERALDA, this can be performed by either using the k-means algorithm or the LBG algorithm. These three possibilities are hence subject to this evaluation task. Again the syllable samples are randomly selected from utterances of arbitrary speakers in the test set of the corpus. The syllable segmentation is again provided by the existing annotation that comes with the Verbmobil corpus.

Dynamic Features Method	Accuracy
CL estimated with k-means for stationary and dynamic features	38%
CL estimated with k-means for dynamic features	42%
CL estimated with LBG for dynamic features	38%

Table 6.10: Consideration of Dynamic Features: NN-Classification (annotated segmentation, arbitrary speakers, diagonal covariances, optimized codelibrary, estimation via single best classes)

Figure 6.9 shows confusion matrices for this evaluation task. Table 6.10 shows results of a nearest neighbor classification. In comparison to the usage of both dynamic and stationary features, the results are substantially worse (38%/42% compared to 46%; cf. also Table 6.6). This trend is also obvious when contemplating the displacement of the means and medians in the confusion matrices. Consequently the assumption that dynamic features alone cause a better discrimination ability in the distance measure has to be rejected. Interestingly the basic approach of taking a codelibrary that was not estimated for only dynamic features but instead for stationary and dynamic features combined is approximately equal to a re-estimation using either k-means or LBG, where k-means performs a little better than LBG.

6.4 Kullback-Leibler Divergence on Gaussian Models

An approach that is different from the Mahalanobis distance as local distance measure for a dynamic time warping approach is to first estimate a multivariate Gaussian model (i.e. a single Gaussian) for the sets of feature vectors representing both syllable speech samples that are to be compared and then to compute the Kullback-Leibler divergence as an information theoretic measure to describe the dissimilarity of the two Gaussian distributions (see section 5.4.1).

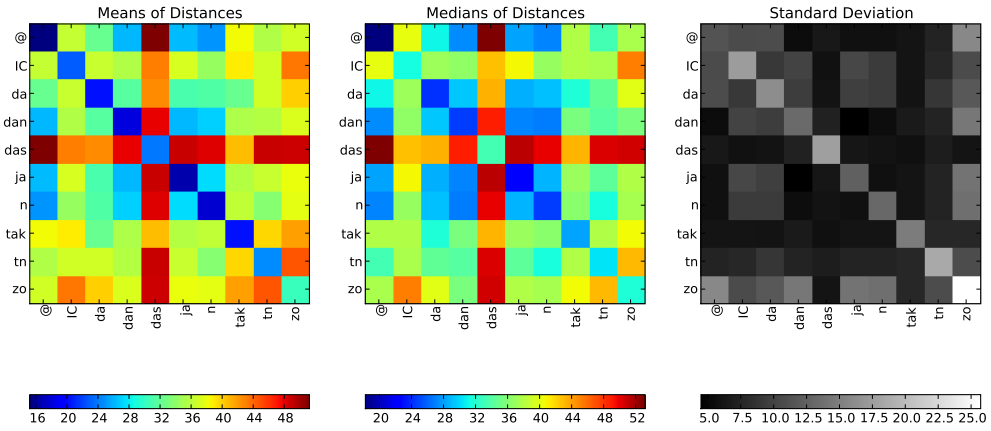
The Kullback-Leibler divergence is evaluated with syllable samples randomly selected from utterances of arbitrary speakers in the test set of the corpus, as in most other evaluation tasks. The syllable segmentation is also again provided by the existing annotation that comes with the Verbmobil corpus. Figure 6.10 shows confusion matrices for this evaluation task. Even from only considering these confusion matrices rather than an actual classification accuracy it is obvious that the Kullback-Leibler divergence is clearly not able to discriminate syllables. For the mean and median confusion matrices the distance averages on the main diagonal are low; however the distances outside of the diagonal are low as well. There are peculiar outliers in the syllables for which the distance to other syllables is particularly high (for `zo` and `n`), the reason for which is not easily apparent.

The significant malfunction of the Kullback-Leibler divergence as similarity measure can possibly be explained by a few reasons. The most obvious reason could be that the Kullback-Leibler divergence itself is not suitable as distance measure for the discrimination of syllables. Another reason could be that the estimation of models from feature vector sets for syllables is not a suitable approach, possibly because syllables consist of only few feature vectors (as opposed to the application area where this approach originated, i.e. the classification of complete music pieces; see section 3.3) or because the temporal order of the feature vectors is ignored by the approach. Two other reasons could be numerical problems due to implementation-related constraints, which is unlikely because there are for some syllables (i.e. `zo`/`n`) with distance values that are highly different from all other distance averages which are relatively homogeneous, or that there is a defect in the implementation itself which however could not be diagnosed despite exhaustive review.

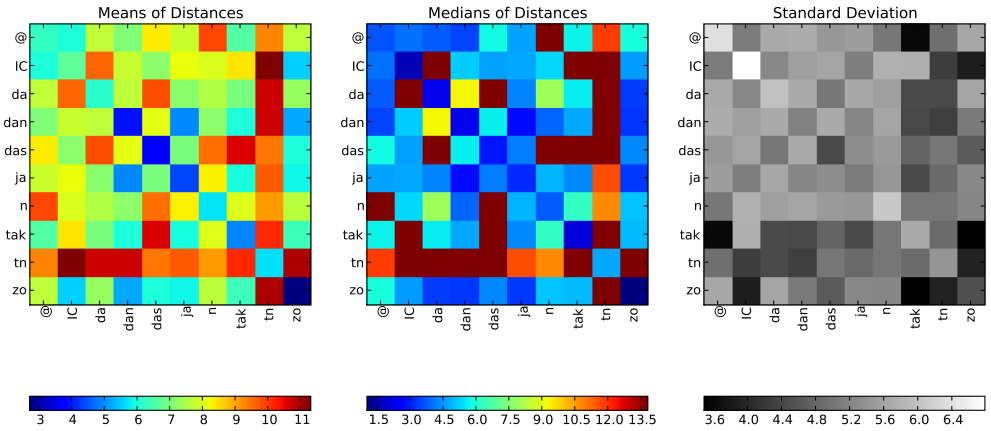
6.5 Comparison of Gaussian Model Parameters

An approach that also uses statistical models estimated for the sets of feature vectors of the speech samples compared is to directly compare the model parameters of the Gaussian distributions, i.e. the means and covariances (see section 5.4.2). This can be carried out either by using an Euclidean distance or by using a Mahalanobis distance (corresponding to the Mahalanobis distance as local distance measure for dynamic time warping).

This approach is again evaluated with syllable samples randomly selected from utterances of arbitrary speakers in the test set of the corpus. Also the syllable segmentation



(a) Euclidean Distance



(b) Mahalanobis Distance

Figure 6.11: Comparison of Gaussian Model Parameters: Confusion Matrices (annotated segmentation, arbitrary speakers)

Measure	Accuracy
Comparison of Model Parameters, Euclidean Distance	27%
Comparison of Model Parameters, Mahalanobis Distance	12%

Table 6.11: Comparison of Gaussian Model Parameters: NN-Classification (annotated segmentation, arbitrary speakers)

is again provided by the existing annotation that comes with the Verbmobil corpus. Figure 6.11 shows confusion matrices for this evaluation task. Results of a nearest neighbor classification which was performed in the context of this evaluation task are presented in Table 6.11. Considering the classification results and the trend that can be distinguished from the confusion matrices it is evident that the comparison of the Gaussian model parameters is better than the Kullback-Leibler divergence (cf. Figure 6.10) one the one hand, but significantly worse as compared to the Mahalanobis distance as local distance measure for dynamic time warping (27%/12% as opposed to 46%; cf. also Table 6.6) on the other hand. However the fundamental ability of being able to discriminate syllables can be observed.

The poor performance compared to the DTW-based measures can be explained either by the approach of directly comparing the model parameters being too simple or by the estimation of statistic models for the feature vector sets being no suitable approach at all (as corresponding to the explanation for the Kullback-Leibler divergence) since either syllables are concepts that are too short for the estimation of statistical models or the temporal order of the features is essential for their discrimination and may not be omitted. Apparently the model parameter comparison using the Mahalanobis distance is substantially worse than with using the Euclidean distance (12% as opposed to 27%). Since while being worse the tendency of a discrimination ability of the measure is still observable in the confusion matrices, a defect in the implementation of the Mahalanobis distance computation itself is unlikely. If there is an implementational defect it could at the most be in the estimation of the statistical models used for the variance normalization in the Mahalanobis distance which however could not be diagnosed despite exhaustive review. Another interpretation is that the deterioration caused by the variance normalization through the Mahalanobis distance indicates that the model parameters of the multivariate Gaussians alone are no suitable characteristics that allow for the discrimination of syllables.

6.6 Synopsis

Reconsidering the evaluation of the acoustic syllable similarity measures proposed in this work there are several central observations that can be made.

The Mahalanobis distance as local distance measure for the dynamic time warping based approach enables discrimination of syllables with a classification accuracy of about 60% for a single speaker, which is a common dimension for methods aiming at the classification of small acoustic units. This performance can be improved by either incorporating fully occupied covariance matrices instead of diagonal covariance

matrices in the distance computation or by using different estimation techniques for the covariances from a Gaussian mixture model, i.e. to use more mixture components. These modifications however are considerably slow and thus harder to apply in a continuous classification scenario where execution speed is an important figure. The use of an automatic syllable border estimation instead of an accurate segmentation from the corpus annotation causes a substantial loss in classification accuracy, which is due to the poor estimation of syllable borders by the automated method. Also, systematically using broader acoustic feature apertures from the utterance feature vectors sequence yields a significant accuracy loss as well. Providing a decent estimation of syllable borders is thus essential to a reliable quality the distance measure. When the distance measure is applied to a classification task concerning only a single speaker instead of arbitrary different speakers, the classification accuracy can be substantially improved. Moreover, the usage of only dynamic features bears worse classification results than using dynamic and stationary features as well.

Approaches that are based on estimation of statistical models on temporal statistics of feature vectors do not work out very well. The Kullback-Leibler divergence shows no syllable discrimination capabilities, when considering the respective confusion matrices. The comparison of Gaussian model parameters using either the Euclidean distance or the Mahalanobis distance are in principle able to discriminate different syllables, but are significantly inferior to the accuracy achieved through the DTW-based Mahalanobis distance.

7 Conclusion and Outlook

In this work, several acoustic similarity measures for syllables are motivated and successively evaluated. The Mahalanobis distance as local distance measure for a dynamic time warping approach to measure acoustic distances is a measure that is able to discriminate syllables and thus allows for syllable classification with an accuracy that is common to the classification of small acoustic units (60% for a nearest neighbor classification of a set of ten syllables using samples of a single speaker; see section 6.3 for details). This measure can be improved using several techniques that however impair the time complexity of the distance measure (usage of all mixture density components for the estimation of covariances from a Gaussian mixture model, see section 6.3.2; usage of fully occupied covariance matrices instead of diagonal covariances, see section 6.3.3). Moreover it is evident that a decently working syllable segmentation algorithm allowing for accurate syllable border estimations is essential to the correct computation of acoustic distances by the similarity measures evaluated. Further approaches which are motivated by their usage in timbre classification of music pieces (see sections 3.3 and 3.4) do not show adequate syllable discrimination abilities (see section 6.5).

There are a few good starting points to improve the acoustic syllable similarity measures developed in this work. Firstly sophisticated improvements to the dynamic time warping algorithm that narrow down the search space for an optimum distance alignment hold a promising gain in time complexity with the accuracy of the distance measure remaining constant (see section 2.6.4). Secondly, presupposing a faster dynamic time warping algorithm, more complex approaches for the local Mahalanobis distance measure can reasonably be applied (as for instance using all mixture density components for covariance estimation, see section 6.3.2, and using fully occupied covariance matrices, see section 6.3.3). Thirdly other measures to compare the Gaussian models estimated on the temporal statistics of the feature vectors could be investigated, which are possibly more appropriate than the Kullback-Leibler divergence and the basic comparison of the model parameters (see sections 6.4 and 6.5).

Reconsidering the general conditions of the development and evaluation of an acoustic similarity measure for syllables there emerge several interesting approaches for further investigation in this area. It would be interesting to conduct a study that applies the tutoring scenario to an interaction of a human and a robot using the Mahalanobis distance as local distance for a dynamic time warping measure as proposed in this work, in order to assess how the measure performs when being actually applied. A further question is if multiple distance measures (e.g. the DTW-based Mahalanobis distance together with a temporal statistics based measure) can be successfully combined, thus inducing a gain in the overall classification accuracy. Another consideration is

if mel-frequency cepstral coefficients (MFCCs) are the optimum acoustic features to measure syllable similarity in the first place or if there exists a better selection of the coefficients. To this end, other acoustic features or a different selection from the MFCCs could be evaluated with the similarity measure. In the tutoring scenario, outstandingly emphasized syllables occur frequently. Moreover it is more likely for emphasized syllables to correspond to a syllable already uttered in the current session. An interesting question is hence if this precondition allows incorporation in the similarity measure, enabling a consecutive improvement of the measure accuracy. A further consideration is if the accumulated information from previously processed syllables in a tutoring scenario session, providing characteristics and distributional information, allow for integration and successive improvement of the measure quality. In some cases syllables in the tutoring scenario are more likely to appear in the context of other certain syllables, which is caused by the semantic conditions of the dialog (e.g. `der grü|ne Be|cher`). A question in this regard is if selective inclusion of the acoustic feature context in such cases can improve the similarity measure.

Bibliography

[ADP07] J.-J. Aucouturier, B. Defreville, and F. Pachet. "The Bag-of-Frames Approach to Audio Pattern Recognition: A Sufficient Model for Urban Soundscapes but not for Polyphonic Music". In: *Journal of the Acoustical Society of America (JASA)* 122.2 (Aug. 2007), pp. 881–91.

[AP02] J.-J. Aucouturier and F. Pachet. "Music Similarity Measures: Whatâs the Use?" In: *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*. Paris, France 2002, pp. 157–163.

[Ahr05] P. Ahrendt. *The Multivariate Gaussian Probability Distribution*. Jan. 2005.

[And+99] E. Anderson et al. *LAPACK Users' Guide*. Third. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 1999.

[BS02] C. Breazeal and B. Scassellati. "Robots That Imitate Humans". In: *Trends in Cognitive Sciences* 6.11 (2002), pp. 481–487.

[Bat+08] A. Batliner et al. "Mothers, Adults, Children, Pets – Towards the Acoustics of Intimacy". In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Las Vegas, NV, USA 2008, pp. 4497–4500.

[DDV07] M. De Wachter, K. Demuynck, and D. Van Compernolle. "Outlier Correction for Local Distance Measures in Example Based Speech Recognition". In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Honolulu, Hawaii, USA 2007, pp. IV–433–IV–436.

[DM80] S. Davis and P. Mermelstein. "Comparison of Parametric Representations for Monosyllable Word Recognition in Continuously Spoken Sentences". In: *IEEE Transactions on Acoustics, Speech and Signal Processing* 28.4 (1980), pp. 357–366.

[De +04] M. De Wachter et al. "A Locally Weighted Distance Measure for Example Based Speech Recognition". In: *Proceedings of the IEEE International Conference on Acoustics, Speech, And Signal Processing (ICASSP)* 1.2 (May 2004), pp. I–181–4.

[De +07a] M. De Wachter et al. "Evaluating Acoustic Distance Measures for Template Based Recognition". In: *Proceedings of the European Conference on Speech Communication and Technology (Eurospeech)* x (Aug. 2007), pp. 874–877.

[De +07b] M. De Wachter et al. “Template-Based Continuous Speech Recognition”. In: *IEEE Transactions on Audio, Speech and Language Processing* 15.4 (May 2007), pp. 1377–1390.

[FP08] G. A. Fink and T. Plötz. “Developing Pattern Recognition Systems Based on Markov Models: The ESMERALDA Framework”. In: *Pattern Recognition and Image Analysis* 18.2 (June 2008), pp. 207–215.

[Fan60] G. Fant. *The Acoustic Theory of Speech Production*. The Hague, Netherlands: Mouton & Co., 1960.

[Fin07] G. A. Fink. *Markov Models for Pattern Recognition: From Theory to Applications*. Berlin, Germany: Springer, 2007.

[Fin99] G. A. Fink. “Developing HMM-Based Recognizers with ESMERALDA”. In: *Conference Proceedings of the International Workshop on Text, Speech and Dialogue (TSD)*. Vol. 1692. Plzen, Czech Republic: Springer Verlag, 1999, p. 229.

[Gan+97] A. Ganapathiraju et al. “Syllable – A Promising Recognition Unit for LVCSR”. In: *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding*. Santa Barbara, CA, USA 1997, pp. 207–214.

[HAH01] X. Huang, A. Acero, and H.-W. Hon. *Spoken Language Processing: A Guide to Theory, Algorithm And System Development*. New Jersey, NJ, USA: Prentice Hall, 2001.

[Ita75] F. Itakura. “Minimum prediction residual principle applied to speech recognition”. In: *IEEE Transactions on Acoustics, Speech and Signal Processing* 23.1 (Feb. 1975), pp. 67–72.

[Jel98] F. Jelinek. *Statistical Methods for Speech Recognition*. The MIT Press, 1998.

[Jen+07] J. Jensen et al. “Evaluation of Distance Measures Between Gaussian Mixture Models of MFCCs”. In: *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*. Vol. 2. x. Austria, Vienna 2007, pp. 107–108.

[Jen+09] J. Jensen et al. “Quantitative Analysis of A Common Audio Similarity Measure”. In: *IEEE Transactions on Audio, Speech, and Language Processing* 17.4 (May 2009), pp. 693–703.

[KL51] S. Kullback and R. A. Leibler. “On information and sufficiency”. In: *The Annals of Mathematical Statistics* 22.1 (1951), pp. 79–86.

[KP01] E.J. Keogh and M.J. Pazzani. “Derivative Dynamic Time Warping”. In: *Proceedings of the SIAM International Conference on Data Mining*. 2001, pp. 1–11.

[Kit03] C. Kit. “How Does Lexical Acquisition Begin? A Cognitive Perspective”. In: *Cognitive Science* 1.1 (2003), pp. 1–50.

[LH00] Z. Liu and Q. Huang. “Content-Based Indexing And Retrieval-By-Example in Audio”. In: *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*. Vol. 2. c. New York, NY, USA 2000, pp. 877–880.

[LS01] B. Logan and A. Salomon. “A Content-Based Music Similarity Function”. In: *Cambridge Research Labs-Tech Report*. June. 2001.

[LS06] M. Levy and M. Sandler. “Lightweight Measures for Timbral Similarity of Musical Audio”. In: *Proceedings of the ACM Workshop on Audio and music Computing Multimedia (AMCMM)*. New York, NY, USA 2006, pp. 27–35.

[MÖ7] M. Müller. *Information Retrieval for Music and Motion*. Berlin, Germany: Springer, 2007.

[ME05] M. Mandel and D. Ellis. “Song-Level Features And Support Vector Machines for Music Classification”. In: *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*. 2004. London, UK 2005, pp. 594–599.

[MRR80] C. Myers, L. Rabiner, and A. Rosenberg. “Performance Tradeoffs in Dynamic Time Warping Algorithms for Isolated Word Recognition”. In: *IEEE Transactions on Acoustics, Speech and Signal Processing* 28.6 (Dec. 1980), pp. 623–635.

[Mah36] P. C. Mahalanobis. “On the Generalized Distance in Statistics”. In: *Proceedings of the National Institute of Science, Calcutta*. Vol. 12. 1936, p. 49.

[Mat+04] M. Matton et al. “A Discriminative Locally Weighted Distance Measure for Speaker Independent Template Based Speech Recognition”. In: *Proceedings of the International Conference on Spoken Language Processing (Interspeech/ICSLP)*. Jeju Island, Korea 2004, pp. 429–432.

[Mer75] P. Mermelstein. “Automatic segmentation of speech into syllabic units”. In: *The Journal of the Acoustical Society of America (JASA)* 58.4 (1975), pp. 880–883.

[Moo03] B. Moore. *An Introduction to the Psychology of Hearing*. Academic Press, 2003.

[NR07] Y. Nagai and K. Rohlfing. “Can Motionese Tell Infants And Robots ‘What to Imitate’?” In: *Proceedings of the International Symposium on Imitation in Animals and Artifacts*. Newcastle upon Tyne, UK 2007, pp. 299–306.

[PV99] R. Paredes and E. Vidal. “A Nearest Neighbor Weighted Measure in Classification Problems”. In: *Proceedings of the Spanish Symposium of Pattern Recognition and Image Analysis (SNRFAI)*. Bilbao, Spain 1999, p. 44.

[Pam06] E. Pampalk. “Computational Models of Music Similarity and their Application in Music Information Retrieval”. PhD Dissertation. Vienna, Austria: Vienna University of Technology, 2006.

[Pri+88] P. Price et al. “The DARPA 1000-Word Resource Management Database for Continuous Speech Recognition”. In: *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. New York, NY, USA 1988, pp. 651–654.

[RJ93] L. Rabiner and B.-H. Juang. *Fundamentals of Speech Recognition*. Upper Saddle River, NJ, USA: Prentice Hall, 1993.

[Roh+06] K. Rohlfing et al. “How Can Multimodal Cues from Child-directed Interaction Reduce Learning Complexity in Robots?” In: *Advanced Robotics* 20.10 (2006), pp. 1183–1199.

[SC07] S. Salvador and P. Chan. “Toward accurate dynamic time warping in linear time and space”. In: *Intelligent Data Analysis* 11.5 (Oct. 2007), pp. 561–580.

[SC78] H. Sakoe and S. Chiba. “Dynamic programming algorithm optimization for spoken word recognition”. In: *IEEE Transactions on Acoustics, Speech and Signal Processing* 26.1 (Feb. 1978), pp. 43–49.

[ST95] E. Schukat-Talamazzini. *Automatische Spracherkennung*. Wiesbaden, Germany: Vieweg, 1995.

[SVN37] S. Stevens, J. Volkman, and E. Newman. “A Scale for the Measurement of the Psychological Magnitude of Pitch”. In: *Journal of the Acoustical Society of America (JASA)* 8.3 (1937), pp. 185–190.

[Sch+09] L. Schillingmann et al. “The Structure of Robot-Directed Interaction Compared to Adult- And Infant-Directed Interaction Using A Model for Acoustic Packaging”. In: *Proceedings of the IEEE Symposium on Robot and Human Interactive Communication (RO-MAN)*. Toyama, Japan 2009.

[Sil86] B. Silverman. *Density Estimation for Statistics And Data Analysis*. London, UK: Chapman and Hall, 1986.

[TSB05] H. Terasawa, M. Slaney, and J. Berger. “A Timbre Space for Speech”. In: *Proceedings of the European Conference on Speech Communication and Technology (Eurospeech)*. Lisboa, Portugal 2005, pp. 4–7.

[VWT06] R. Villing, T. Ward, and J. Timoney. “Performance Limits for Envelope Based Automatic Syllable Segmentation”. In: *Irish Signals and Systems Conference (ISSC)*. Dublin, Ireland 2006, pp. 521–526.

[Vil+04] R. Villing et al. “Automatic Blind Syllable Segmentation for Continuous Speech”. In: *Irish Signals and Systems Conference (ISSC)*. Belfast, Ireland 2004, pp. 41–46.

[Wah00] W. Wahlster. *Verbmobil: Foundations of Speech-to-Speech Translation*. Berlin, Heidelberg, New York, Barcelona, Hong Kong, London, Milan, Paris, Singapore, Tokyo: Springer, 2000.

List of Figures

List of Tables

List of Requirements

List of Desirable Properties