Technische Universität Ilmenau
Fakultät für Elektrotechnik und Informationstechnik

Application of Speech Recognition Algorithms to Singing

PhD Thesis at Fraunhofer Institute for Digital Media Technology

Submitted by: Anna Marie Kruspe

Submitted on:

Course of study: Media Technology

Matriculation Number: 39909

Advisor: Prof. Dr.-Ing. Dr. rer. nat. h.c. mult. Karlheinz Brandenburg

Abstract

The Higgs boson or Higgs particle is an elementary particle initially theorised in 1964,[6][7] and tentatively confirmed to exist on 14 March 2013.[8] The discovery has been called "monumental"[9][10] because it appears to confirm the existence of the Higgs field,[11][12] which is pivotal to the Standard Model and other theories within particle physics. In this discipline, it explains why some fundamental particles have mass when the symmetries controlling their interactions should require them to be massless, and?linked to this?why the weak force has a much shorter range than the electromagnetic force.

Kurzfassung

Das Higgs-Teilchen gehört zum Higgs-Mechanismus, einer schon in den 1960er Jahren vorgeschlagenen Theorie, nach der alle fundamentalen Elementarteilchen (beispielsweise das Elektron) ihre Masse erst durch die Wechselwirkung mit dem allgegenwärtigen Higgs-Feld erhalten. Als einziges Teilchen des Standardmodells ist das Higgs-Boson experimentell noch nicht vollständig gesichert.

Acknowledgements

Thanks to Leonard Hofstadter and thanks to my mee-maw.

Table of Contents

1 Introduction

This is my introduction...

2 State of the art

2.1 From speech to singing

Singing presents a number of challenges for speech recognition when compared to pure speech [1] [2] [3]. The following factors make speech recognition on singing more difficult than on speech, and make it necessary to adapt existing algorithms.

Larger pitch fluctuations A singing voice varies its pitch to a much higher degree than a speaking voice. It often also has very different spectral properties.

Larger changes in loudness In addition to pitch, loudness also fluctuates much more in singing than in speech.

Higher pronunciation variation Singers are often forced by the music to pronounce certain sounds and words differently than if they were speaking them.

Larger time variations In singing, sounds are often prolonged for a certain amount of time to fit them to the music. Conversely, they can also be shortened or left out completely.

Different vocabulary In musical lyrics, words and phrases often differ from normal conversation texts. Certain words and phrases have different probabilities (e.g. higher focus on emotional topics in singing).

Background music This is the biggest interfering factor when considering polyphonic recordings. Harmonic and percussive instruments add a huge amount of spectral components to the signal, which lead to confusion in speech recognition algorithms. Ideally, these components should be removed or suppressed in a precursory step. This could be achieved, for example, by employing source separation algorithms. However, such algorithms add additional artifacts to the signal, and may not even be sufficient for this purpose at the current state of research.
Voice activity detection (VAD) could be used as a non-invasive first step to at the very least discard segments of songs that do not contain voice. However,

such algorithms often make mistakes in the same cases that are problematic for speech recognition algorithms (e.g. instrumental solos)[].
For these reasons, most of the experiments in this work were performed on unaccompanied singing. The integration of the mentioned pre-processing algorithms would be a very interesting next step of research.
The exception are the lyrics-to-singing alignment algorithms presented in Chapter 8. Those were also tested on polyphonic music, and the algorithms appear to be largely robust to these influences.

2.2 Phoneme recognition

Due to the factors mentioned above in section 2.1, phoneme recognition on singing is more difficult than on clean speech. It has only been a topic of research for a few years, and there are few publications.
In 2007, Gruhne et al. presented a classical approach that employs feature extraction and various machine learning algorithms to classify singing into 15 phoneme classes [5] [6]. The specialty of this approach lies in the pre-processing: At first, fundamental frequency estimation is performed on the audio input, using a Multi-Resolution Fast Fourier Transform [7]. Based on the estimated fundamental frequency, the harmonic partials are retrieved from the spectrogram. Then, a sinusoidal re-synthesis is performed, using only the detected fundamental frequency and partials. Feature extraction is then performed on this re-synthesis instead of the original audio. Extracted features include MFCCs, PLPs [8], LPCs, and WLPCs [9]. MLP, GMM, and SVM models are trained on the resulting feature vectors. The re-synthesis idea comes from a singer identification approach by Fujihara [10].
The approach is tested on more than 2000 separate, manually annotated phoneme instances from polyphonic recordings. Only one feature vector per phoneme instance is calculated. The phoneme set is reduced down to 15 classes. Using SVM models, 56% of the tested instances were classified correctly. This is significantly better than the best result without the re-synthesis step (34%).
In [11] (2010), the approach is expanded by testing a larger set of perceptually motivated features, and more classifiers. No significant improvements are found when using more intricate features, and the best-performing classifier remains SVM.
Fujihara et al. described an approach based on spectral analysis in 2009 [4]. The underlying idea is that spectra of polyphonic music can be viewed as the weighted sum of two types of spectra: One for the singing voice, and one for the background music.

This approach then models these two spectra as probabilistic spectral templates. The singing voice is modeled by multiplying a vocal envelope template, which represents the spectral structure of the singing voice, with a harmonic filter, which represents the harmonic structure of the produced sound itself. This is analogous to the source-filter model of speech production []. For recognizing vowels, five such harmonic filters are prepared (a - e - i - o - u). Vocal envelope templates are trained on voice-only recordings, separated by gender. Templates for background music are trained on instrumental tracks. In order to recognize vowels, the probabilities for each of the five harmonic templates are estimated. As a side product, the algorithm also estimates the fundamental frequency of the singing voice.
As described, the phoneme models are gender-specific and only model five vowels, but also work for singing with instrumental accompaniment. The approach is tested on 10 Japanese-language songs. The best result is 65% correctly classified frames, compared to the 56% with the previous approach by this team, based on GMMs.
Also in 2009, Mesaros et al. presented another classical approach to phoneme recognition in singing which is based on MFCC features and GMM-HMMs for acoustic modeling [12]. The models are trained on the CMU ARCTIC speech corpus [1]. Then, different Maximum Likelihood Linear Regression (MLLR) techniques for adapting the models to singing voices are tested [13].
The adaptation and test corpus consists of 49 voice-only fragments from 12 pop songs with durations between 20 and 30 seconds. The best results are achieved when both the means and variances of the Gaussians are transformed with MLLR. The results improved slightly when not just a single transform was used for all phonemes, but when they were grouped into base classes beforehand, each receiving individual transformation parameters. The best result is around .79 Phoneme Error Rate on the test set.
In [14] and [15], language modeling is added to the presented approach. Phoneme-level language models are trained on the CMU ARCTIC corpus as unigrams, bigrams, and trigrams, while word-level bigram and trigram models are trained on actual song lyrics in order to match the application case. The output from the acoustic models is then refined using these language models. The approach is tested on the clean singing corpus mentioned above, and on 100 manually selected fragments of 17 polyphonic pop songs. To facilitate recognition on polyphonic music, a vocal separation algorithm is introduced [16].
Using phoneme-level language modeling, the phoneme error rate on clean singing is

[1] `http://festvox.org/cmuarctic/`

reduced to .7. On polyphonic music, it is .81. For the word recognition approach, the word error rate is .88 on clean singing, and .94 on the polyphonic tracks.
A more detailed voice adaptation strategy is tested in [17]. Instead of adapting the acoustic models with mixed singing data, they are adapted gender-wise, or to specific singers. With the gender-specific adaptations, the average phoneme error rate on clean singing is lowered to .81 without language modeling, and .67 with language modeling. Singer-specific adaptation does not improve the results, probably because of the very small amount of adaptation data in this case.
In [**?**] (2014), McVicar et al. build on a very similar baseline system, but also exploit repetitions of choruses to improve transcription accuracy. This has been done for other MIR tasks, such as chord recognition, beat tracking, and source separation. They propose three different strategies for combining individual results: Feature averaging, selection of the chorus instance with the highest likelihood, and combination using the Recogniser Output Voting Error Reduction (ROVER) algorithm [18]. They also employ three different language models, two of which were matched to the test songs (and therefore not representative for general application). 20 unaccompanied, English-language songs from the RWC database [19] were used for testing; chorus sections were selected manually. The best-instance selection and the ROVER strategies improve results significantly; with the ROVER approach and a general-purpose language model, the Phoneme Error Rate is at .74 (versus .76 in the baseline experiment), while the Word Error Rate is improved from .97 to .9. Interestingly, cases with a low baseline result benefit the most from exploiting repetition information.
The final system was proposed by Hansen in 2012 [20]. It also employs a classical approach consisting of a feature extraction step and a model training step. Extracted features are MFCCs and TRAP (TempoRAl Pattern) features. Then, Multilayer Perceptrons (MLPs) are trained separately on both feature sets. The assumption is that each feature models different properties of the considered phonemes: Short-term MFCCs are good at modeling the pitch-independent properties of stationary sounds, such as sonorants and fricatives. On the flip-side, TRAP features are able to model temporal developments in the spectrum, forming better representations for sounds like plosives or affricates.
The results of both MLP classifiers are combined via a fusion classifier, also an MLP. Then, Viterbi decoding is performed on its output.
The approach is trained and tested on a data set of 12 vocal tracks of pop songs, which were manually annotated with a set of 27 phonemes. The combined system achieves a recall of .48, compared to .45 and .42 for the individual MFCC and TRAP classifiers

respectively. This confirms the assumption that the two features complement each other. The phoneme-wise results further corroborate this.

2.3 Forced alignment and retrieval

2.4 Language identification

2.4.1 Language identification in speech

Language identification has been extensively researched in the field of Automatic Speech Recognition since the 1980's. A number of successful algorithms have been developed over the years. An overview over the fundamental techniques is given by Zissman in [21].
Fundamentally, four properties of languages can be used to discriminate between them:

Phonetics The unique sounds that are used in a given language.

Phonotactics The probabilities of certain phonemes and phoneme sequences.

Prosody The "melody" of the spoken language.

Vocabulary The possible words made up by the phonemes and the probabilities of certain combinations of words.

Even modern system mostly focus on phonetics and phonotactics as the distinguishing factors between languages. Vocabulary is sometimes exploited in the shape of language models.
Zissman mentions Parallel Phone Recognition followed by Language Modeling (PPRLM) as one of the basic techniques. It requires audio data, language annotations, and phoneme annotations for each utterance. In order to make use of vocabulary characteristics, full sentence annotations and word-to-phoneme dictionaries are also necessary. Using the audio and phoneme data, acoustic models are trained. They describe the probabilities of certain sound and sound sequences occurring. This is done separately for each considered language. Similarly, language models are generated using the sentence annotations and the dictionary. These models describe the probabilities of certain words and phrases. Again, this is done for each language.
New audio examples are then run through all pairs of acoustic and language models, and the likelihoods produced by each model are retained. The highest acoustic likelihood, the highest language likelihood, or the highest combined likelihood are then

considered to determine the language. This approach achieves up to 79% accuracy for ten languages [22].
Another approach uses the idea to train Gaussian Mixture Models for each language. This technique can be considered a "bag of frames" approach, i.e. the single data frames are considered to be statistically independent of each other. The generated GMMs then describe probability densities for certain characteristics of each language. Using these, the language of new audio examples can be easily determined.
GMM approaches used to perform worse than their PPRLM counterparts, but the development of new features has made the difference negligible [23]. They are in general easier to implement since only audio examples and their language annotations are required. Allen et al. [24] report results of up to 76.4% accuracy for ten languages. Different backend classifiers, such as Multi-Layer Perceptrons (MLPs) and Support Vector Machines (SVMs) [25] have also been used successfully instead of GMMs.

2.4.2 Language identification in singing

A first approach for language identification in singing was proposed by Tsai and Wang in 2004 [27]. At its core, the algorithm is similar to Parallel Phoneme Recognition (PPR). However, instead of full phoneme modeling, they employ an unsupervised clustering algorithm to the input feature data and tokenize the results to form language-specific codebooks (plus one for background music). Following this, the results from each codebook are run through a matching language models to determine the likelihood that the segment was performed in this language. Prior to the whole process, vocal/non-vocal segmentation is performed. This is done by training GMMs on segments of each language, and on non-vocal segments. MFCCs are used as features.
The approach is tested on 112 English- and Mandarin-language polyphonic songs each, with 32 songs performed in both languages. A classification accuracy of .8 is achieved on the non-overlapping songs. On the overlapping songs, it is only at .7, suggesting some influence of the musical material (as opposed to the actual language characteristics). Misclassifications occur more frequently on the English-language songs, possibly because of accents of Chinese singers performing in English, and because of louder background music.
A second, simpler approach was presented by Schwenninger et al. in 2006 [26]. They also extract MFCC features, and then use these to directly train statistical models for each language. Three different pre-processing strategies are also tested: Automatic vocal/non-vocal segmentation, distortion reduction, and azimuth discrimina-

tion. Vocal/non-vocal segmentation is performed by thresholding the energy in high-frequency bands as an indicator for voice presence over 1 second windows. This leaves a relatively small amount of material per song. Distortion reduction is employed to discard strong drum and bass frames where the vocal spectrum is masked by using a mel-based approach. Finally, azimuth discrimination attempts to detect and isolate singing voice panned to the center of the stereo scene.
The approach is tested on three small data sets of speech, unaccompanied singing, and polyphonic music. Without pre-processing steps, the accuracy is .84, .68, and .64 respectively, highlighting the increased difficulty of language identification on singing versus speech, and on polyphonic music versus pure vocals. On the polyphonic corpus, the pre-processing steps do not improve the result.
In 2011, Mehrabani and Hansen presented a full Parallel Phoneme Recognition followed by Language Modeling (PPRLM) approach for singing language identification. MFCC features are run through phoneme recognizers for Hindi, German, and Mandarin; then, the results are scored by individual language models for each considered language. In addition, a second system is employed which uses prosodic instead of phonetic tokenization. This is done by modeling pitch contours with Legendre polynomials, and then quantizing these vectors with previously trained GMMs. The results are then again used as inputs to language models.
The approach is trained and tested on a corpus containing 12 hours of unaccompanied singing and speech in Mandarin, Hindi, and Farsi. The average accuracy for singing is .78 and .43 for the phoneme- and prosody-based systems respectively, and .83 for a combination of both.
Also in 2011, Chandrasekhar et al. presented a very interesting approach for language identification on music videos, taking into account both audio and video features [29]. On the audio side, the spectrogram, volume, MFCCs, and perceptually motivated Stabilized Auditory Images (SAI) are used as inputs. One-vs-all SVMs are trained for each language. The approach is trained and tested on 25,000 music videos, taking 25 languages into consideration. Using audio features only, the accuracy is .45; combined with video features, it rises to .48. It is interesting to note that European languages seem to achieve much lower accuracies than Asian and Arabic ones. English, French, German, Spanish and Italian rank below .4, while languages like Nepali, Arabic, and Pashto achieve accuracies above .6. It is possible that the language characteristics of European languages make them harder to discriminate (especially against each other) than others.

2.5 Keyword spotting

Keyword spotting in singing was first attempted in 2008 by Fujihara et al. [**?**]. Their approach employs a phoneme recognition step first, which is again based on the vocal re-synthesis method first described in [10]. MFCCs and power features are extracted from the re-synthesized singing and used as inputs to a phoneme model, similar to Gruhne's phoneme recognition approach mentioned above in **??**. Three phoneme models are compared: One trained on pure speech and adapted with a small set of singing recordings, one adapted with all recordings, and one trained directly on singing. Viterbi decoding is then performed using keyword-filler HMMs (see **??**) to detect candidate segments where keywords may occur. These segments are then re-scored through the filler HMM to verify the occurrence.
The method is tested on 79 unaccompanied Japanese-language songs from the RWC database [19]. The Phoneme Error Rate is .73 for the acoustic models trained on speech, .67 for the adapted models, and .49 for the models trained on singing (it should be mentioned that the same songs were used for training and testing, although a cross-validation experiment appears to show that the effect is negligible). The employed evaluation measure is "link success rate", describing the percentage of detected phrases that were linked correctly to other occurrences of the phrase in the data set. In that sense, it is a sort of accuracy measure. The link success rate for detecting the keywords is .3. The authors show that the result depends highly on the number of phonemes in the considered keyword, with longer keywords being easier to detect.
In 2012, Mercado et al. presented an approach to keyword spotting in singing based on a different principle: Dynamic Time Warping (DTW) between a sung query and the requested phrase in the song recording. In particular, Statistical Sub-sequence DTW is the algorithm employed for this purpose. MFCCs are used as feature inputs, then the costs of the warping paths are calculated from all possible starting points to obtain candidate segments, which are then further refined to find the most likely position.
The approach is tested on a set of vocal tracks of 19 pop songs (see Section **??**) as the references, and phrase recordings by amateur singers as the queries, but no quantitative results are given. The disadvantage of this approach lies in the necessity for audio recordings of the key phrases, which need to have at least similar timing and pitch as the reference phrases.
Finally, Dzhambazov et al. developed a score-aided approach to keyword spotting in 2015 [**?**]. A user needs to select a keyword phrase and a single recording in which this phrase occurs. The keyword is then modeled acoustically by concatenating record-

ings of the constituent phonemes (so-called acoustic keyword spotting). Similar to Mercado's approach, Sub-Sequence DTW is then performed between the acoustic template and all starting positions in the reference recording to obtain candidate segments. These segments are then refined by aligning the phonemes to the score in these positions to model their durations. This is done by using Dynamic Bayesian Network HMMs. Then, Viterbi decoding is performed to re-score the candidate segments and obtain the best match.
The approach is tested on a small set of unaccompanied Turkish-language recordings of traditional Makam music. The Mean Average Precision (MAP) for the best match is .08 for the DTW approach only, and .05 for the combined approach. For the top-6 results, the MAP is .26 and .38 respectively.

3 Technical Background

3.1 General processing chain

3.2 Audio features

3.2.1 Perceptive Linear Predictive features (PLPs)

PLP features, first introduced in [30], are among the most frequently used features in speech processing. They are based on the idea to use knowledge about human perception to emphasize important speech information in spectra while minimizing the differences between speakers. We use a model order of 13 in two experiments and one of 32 in another. Deltas and double deltas between frames are also calculated. We test PLPs with and without RASTA pre-processing [31].

3.2.2 Mel-Frequency Cepstral Coefficients (MFCCs)

Just like PLPs, MFCCs are frequently used in all disciplines of automatic speech recognition [21]. We kept 20 cepstral coefficients for model training. Additionally, we calculated deltas and double deltas.

3.2.3 Shifted Delta Cepstrum (SDCs)

Shifted Delta Cepstrum features were first described in [32] and have since been successfully used for speaker verification and language identification tasks on pure speech data [33] [25] [24]. They are calculated on MFCC vectors and take their temporal evolution into account. Their configuration is described by the four parameter $N - d - P - k$, where N is the number of cepstral coefficients for each frame, d is the time context (in frames) for the delta calculation, k is the number of delta blocks to use, and P is the shift between consecutive blocks. The delta cepstrals are then calculated as:

$$\Delta c(t) = c(t + iP + d) + c(t + iP - d), 0 <= i <= k \tag{3.1}$$

with $c \in [0, N-1]$ as the previously extracted cepstral coefficients. The resulting k delta cepstrals for each frame are concatenated to form a single SDC vector of the length kN. We used the common parameter combination $N = 7, d = 1, P = 3, k = 7$.

3.2.4 TempoRal Patterns (TRAP)

3.3 Machine learning algorithms

This section describes the various machine learning algorithms employed throughout this thesis. Gaussian Mixture Models (GMMs), Hidden Markov Models (HMMs), and Support Vector Machines (SVMs) are three traditional approaches that are used as the basis of many new approaches, and were used for several starting experiments. i-Vector processing is a relatively new, more sophisticated approach that bundles several other machine learning techniques.
In recent years, Deep Learning has become the standard for machine learning applications []. This chapter also describes two of those new approaches that were used in this work: Deep Neural Networks (DNNs) and Deep Belief Networks (DBNs).

3.3.1 Gaussian Mixture Models

3.3.2 Hidden Markov Models

3.3.3 i-Vector processing

I-Vector (identity vector) extraction was first introduced in [34] and has since become a state-of-the-art technique for various speech processing tasks, such as speaker verification, speaker recognition, and language identification [35]. To our knowledge, it has not been used for any Music Information Retrieval tasks before.
The main idea behind i-vectors is that all training utterances contain some common trends, which effectively add irrelevance to the data in respect to training. Using i-vector extraction, this irrelevance can be filtered out, while only the unique parts of the data relevant to the task at hand remain. The dimensionality of the training data is massively reduced, which also makes the training less computationally expensive. As a side effect, all feature matrices are transformed to i-vectors of equal length, eliminating problems that are caused by varying utterance lengths.
Mathematically, this assumption can be expressed as:

$$M(u) = m + Tw \tag{3.2}$$

In this equation, $M(u)$ is the GMM supervector for utterance u. The supervector approach was first presented in [36] and has since been successfully applied to a number of speech recognition problems. A music example can be found in [37]. m represents the language- and channel-independent component of u and is estimated using a Universal Background Model (UBM). T is a low-rank matrix modeling the relevant language- and channel-related variability, the so-called Total Variability Matrix. Finally, w is a normally distributed latent variable vector: The i-vector for utterance u.

Step 1: UBM training A Universal Background Model (UBM) is trained using Gaussian Mixture Models (GMMs) from all utterances. This UBM models the characteristics that are common to all of them.

Step 2: Statistics extraction 0th and 1st order Baum-Welch statistics are calculated for each of the utterances from the UBM according to:

$$N_c(u) = \sum_{t=1}^{L} P(c|y_t, \Omega) \tag{3.3}$$

$$\widetilde{F}_c(u) = \sum_{t=1}^{L} P(c|y_t, \Omega)(y_t - m_c) \tag{3.4}$$

where $u = y_1, y_2, ..., y_L$ denotes an utterance with L frames, $c = 1, ..., C$ denotes the index of the Gaussian component, Ω denotes the UBM, m_c is the mean of the UBM mixture component c, and $P(c|y_t, \Omega)$ denotes the posterior probability that the frame y_t was generated by mixture component c. As the equation shows, the 1st order statistics are centered around the mean of each mixture component.

Step 3: T matrix training Using the Baum-Welch statistics for all utterances, the Total Variability Matrix T is now trained iteratively according to:

$$w = (I + T^t \Sigma^{-1} N(u) T)^{-1} T^t \Sigma^{-1} \widetilde{F}(u) \tag{3.5}$$

using Expectation Maximization.

Step 4: Actual i-vector extraction Finally, an i-vector w can be extracted for each utterance using equation 3.5 again. This can also be done for unseen utterances, using a previously trained T.

3.3.4 Artificial Neural Networks

3.3.4.1 Deep Neural Networks

3.3.4.2 Deep Belief Networks

3.4 Evaluation

3.4.1 Evaluation of phoneme recognition and alignment tasks

3.4.2 Evaluation of language identification tasks

3.4.3 Evaluation of keyword spotting tasks

3.5 Speech recognition systems

3.5.1 Phoneme recognition

3.5.2 Forced alignment

3.5.3 Language identification

3.5.4 Keyword spotting

4 Data sets

This chapter contains descriptions of all the data sets (or corpora) used over the course of this thesis. They are grouped into speech-only data sets, data sets of unaccompanied (=a-capella) singing, and data sets of full musical pieces with singing ("real-world" data sets).

4.1 Speech data sets

4.1.1 TIMIT

TIMIT is, presumably, the most widely used corpus in speech recognition research [38]. It was developed in 1993 and consists of 6300 English-language audio recordings of 630 native speakers with annotations on the phoneme, word, and sentence levels. The corpus is split into a training and a test section, with the training section containing 4620 utterances, and the test section containing 1680. Each of those utterances has a duration of a few seconds.
The phoneme annotations follow a model similar to ARPABET and contain 61 different phonemes [39].

4.1.2 NIST Language identification corpora

4.1.3 OGI Language identification corpus

For comparison, the algorithms in this work were also tested on the *OGI Multi-language Telephone Speech Corpus (OGIMultilang)* [40], using all recordings for the three previously mentioned languages. There are 3,177 utterances in sum with more varying durations (1-60 seconds). For experiments on longer recordings, results on these individual utterances were aggregated for each speaker, producing 118 documents per language (354 in sum).

Table 4.1: Amounts of data in the three used data sets: Sum duration on top, number of utterances in italics.

hh:mm:ss ***#Utterances***	**NIST2003LRE**	**OGIMultilang**	**YTAcap**
English	00:59:08 *240*	05:13:17 *1912*	08:04:25 *1975*
German	00:59:35 *240*	02:52:27 *1059*	04:18:57 *1052*
Spanish	00:59:44 *240*	03:05:45 *1151*	07:21:55 *1810*

4.2 A-Capella singing data sets

4.2.1 YouTube data set

As opposed to the speech case, there are no standardized corpora for sung language identification. For the sung language identification experiments, A-capella audio files were therefore extracted from *YouTube*[1] videos. This was done for three languages: English, German, and Spanish. The author collected between 116 (258min) and 196 (480min) examples per language. These were mostly videos of amateur singers freely performing songs without accompaniment. Therefore, they are of highly varying quality and often contain background noise. Most of the performers contributed only a single song, with just a few providing up to three. In this way, we aim to avoid effects where the classifier recognizes the singer's voice instead of the language.
Special attention was paid to musical style. Rap, opera singing, and other specific singing styles were excluded. All the songs performed in these videos were pop songs. Different musical styles can have a high impact on language classification results. The author tried to limit this influence as much as possible by choosing recordings of pop music instead of language-specific genres (such as latin american music).

4.2.2 Hansen's vocal track data set

This is one of the data sets used for keyword spotting and phoneme recognition. It was first presented in [20]. It consists of the vocal tracks of 19 commercial English-

[1] `http://www.youtube.com`, Last checked: 05/16/13

language pop songs. They are studio quality with some post-processing applied (EQ, compression, reverb). Some of them contain choir singing. These 19 songs are split up into ??? clips that roughly represent lines in the song lyrics.
Twelve of the songs were annotated with time-aligned phonemes. The phoneme set is the one used in CMU Sphinx[2] and TIMIT [38] and contains 39 phonemes. All of the songs were annotated with word-level transcriptions. This is the only one of the singing data sets that has full manual annotations, which are assumed to be reliable and can be used as ground truth.
For comparison, recordings of spoken recitations of all song lyrics were also made. These were all performed by the same speaker (the author).

4.2.3 DAMP data set

As described, Hansen's data set is very small and therefore not suited to training phoneme models for singing. As a much larger source of unaccompanied singing, the *DAMP* data set, which is freely available from Stanford University[3][41], was employed. This data set contains more than 34,000 recordings of amateur singing of full songs with no background music, which were obtained from the *Smule Sing!* karaoke app. Each performance is labeled with metadata such as the gender of the singer, the region of origin, the song title, etc. The singers performed 301 English-language pop songs. The recordings have good sound quality with little background noise, but come from a lot of different recording conditions.
No lyrics annotations are available for this data set, but the textual lyrics can be obtained from the *Smule Sing!* website[4]. These are, however, not aligned in any way. Such an alignment was performed automatically on the word and phoneme levels (see section **??**).

4.2.4 Aji's synthesized singing data set

Since it was not feasible to hand-annotate a large data set over the course of this work, another approach was the automatic generation of sung audio. The advantage of this approach is that the results can be assumed to be perfectly aligned to the given phonemes.

[2] `http://cmusphinx.sourceforge.net/`
[3] `https://ccrma.stanford.edu/damp/`
[4] `http://www.smule.com/songs`

For the generation of this data set, ??? recordings from the previously described *DAMP* data set were selected. Their phonemes were automatically aligned, and an automatic transcription of the melody was performed. These two sources of data were then aligned to each other. This alignment did not need to be perfect, it just needed to produce a plausible combination of melody line and phonemes.
The result of this step was then fed into the *Sinsy*[][5] singing synthesizer to generate new singing recordings. This synthesizer provided one female singing voice. The resulting recordings are good in quality and relatively natural sounding, but appear to have a slight accent.
The whole generation process of this data set was performed in collaboration with Adam Aji.

4.2.5 Choosing keywords

4.3 "Real-world" data sets

4.3.1 QMUL Expletive data set

This data set consists of 80 popular songs which were collected at Queen Mary University, most of them Hip Hop. 711 instances of 48 expletives were annotated on these songs. In addition, the matching textual, unaligned lyrics were retrieved from the internet.

4.3.2 "69 Love Songs" data set

"69 Love Songs" is a 3-CD album by the band "The Magnetic Fields", which was released in 1999 and named one of the *Rolling Stone*'s 500 Greatest Albums of All Time in 2012 [42]. It contains 69 songs in various musical styles and instrumentations, performed by a variety of musicians, including five vocalists. The total duration is 2 hours and 52 minutes. The data set is interesting for the purposes of this work because the songs' lyrics all cover a similar theme - namely, love. A word count on the lyrics shows, for example, that the word "love" itself occurs $\tilde{2}25$ times in these songs.
Unaligned lyrics were retrieved from `http://stephinsongs.wiw.org`. A thorough semantic analysis can be found in [43].

[5]

5 Singing phoneme recognition

5.1 Phoneme recognition using models trained on speech

5.2 Phoneme recognition on synthesized singing

5.3 Phoneme recognition using models trained on "songified" speech

5.4 Phoneme recognition using models trained on a-capella singing

5.5 Conclusion

6 Language identification

6.1 LID in singing using GMMs

6.1.1 Processing chain

6.1.2 Results

6.2 LID in singing using i-Vectors and GMMs

6.2.1 i-Vector implementation

6.2.2 i-Vector processing chain

6.2.3 Results

6.3 LID in singing using phoneme recognition posteriors

6.3.1 Phoneme recognition for LID

6.3.2 Post-processing

6.3.3 Results

6.4 Conclusion

7 Sung keyword spotting experiments and results

7.1 Keyword spotting using keyword-filler HMMs

7.1.1 Phoneme posterior extraction and further processing

7.1.2 Implementation of keyword-filler HMMs

7.1.3 Results on speech and music

7.2 Keyword spotting using duration-informed keyword-filler HMMs

7.2.1 Duration modeling approaches

7.2.2 Implementation of duration modeling approaches for keyword-filler HMMs

7.2.3 Results on speech and music

7.3 Improving keyword spotting using specified phoneme models

7.3.1 Improving phoneme models

7.3.2 Post-processing

7.3.3 Results on speech and music

7.4 Conclusion

8 Lyrics Retrieval and Alignment

8.1 HMM-based lyrics-to-audio alignment

8.2 Posteriorgram-based retrieval and alignment

8.3 Phoneme-based retrieval and alignment

8.4 Application: Expletive detection

9 Conclusion

10 Future work

Bibliography

[1] A. Loscos, P. Cano, and J. Bonada, "Low-delay singing voice alignment to text," in *Proceedings of the ICMC*, 1999.

[2] H. Fujihara and M. Goto, *Multimodal Music Processing*, chapter Lyrics-to-audio alignment and its applications, Dagstuhl Follow-Ups, 2012.

[3] A. M. Kruspe, "Keyword spotting in a-capella singing," in *15th International Conference on Music Information Retrieval (ISMIR)*, Taipei, Taiwan, 2014.

[4] H. Fujihara, M. Goto, and H. G. Okuno, "A novel framework for recognizing phonemes of singing voice in polyphonic music.," in *WASPAA*. 2009, pp. 17–20, IEEE.

[5] Matthias Gruhne, Konstantin Schmidt, and Christian Dittmar, "Phoneme recognition in popular music," *ISMIR*, 2007.

[6] Matthias Gruhne, Konstantin Schmidt, and Christian Dittmar, "Detecting phonemes within the singing of polyphonic music," *Proceedings of ICoMCS ...*, , no. December, pp. 60–63, 2007.

[7] Karin Dressler, "Sinusoidal Extraction using an efficient implementation of a multi-resolution {FFT}," in *Proc. of the 9th Int. Conference on Digital Audio Effects (DAFx-06)*, sep 2006, pp. 247–252.

[8] Hynek Hermansky, Nelson Morgan, Aruna Bayya, and Phil Kohn, "Rasta-plp speech analysis technique," in *Proceedings of the 1992 IEEE International Conference on Acoustics, Speech and Signal Processing - Volume 1*, Washington, DC, USA, 1992, ICASSP'92, pp. 121–124, IEEE Computer Society.

[9] Aki Härmä and Unto K. Laine, "A comparison of warped and conventional linear predictive coding," *IEEE Trans. Speech and Audio Processing*, vol. 9, pp. 579–588, 2001.

[10] Hiromasa Fujihara, Tetsuro Kitahara, Masataka Goto, Kazunori Komatani, Tetsuya Ogata, and Hiroshi G. Okuno, *Singer identification based on accompaniment sound reduction and reliable frame selection*, pp. 329–336, 2005.

[11] G. Szepannek, M. Gruhne, B. Bischl, S. Krey, T. Harczos, F. Klefenz, C. Dittmar, and C. Weihs, *Classification as a tool for research*, chapter Perceptually Based Phoneme Recognition in Popular Music, Springer, Heidelberg, 2010.

[12] Annamaria Mesaros and Tuomas Virtanen, "Adaptation of a speech recognizer for singing voice," *European Signal Processing Conference*, , no. 1, pp. 1779–1783, 2009.

[13] M.J.F. Gales and P.C. Woodland, "Mean and variance adaptation within the mllr framework," *Computer Speech & Language*, vol. 10, pp. 249–264, 1996.

[14] A Mesaros and T Virtanen, "Recognition of phonemes and words in singing," *Acoustics Speech and Signal ...*, pp. 1–4, 2010.

[15] Annamaria Mesaros and T Virtanen, "AUTOMATIC UNDERSTANDING OF LYRICS FROM SINGING," *Akustiikkapäivät*, pp. 1–6, 2011.

[16] Tuomas Virtanen, Annamaria Mesaros, and Matti Ryynänen, "Combining pitch-based inference and non-negative spectrogram factorization in separating vocals from polyphonic music," in *ISCA Tutorial and Research Workshop on Statistical and Perceptual Audition, SAPA 2008, Brisbane, Australia, September 21, 2008*, 2008, pp. 17–22.

[17] Annamaria Mesaros and Tuomas Virtanen, "Automatic Recognition of Lyrics in Singing," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2010, pp. 1–11, 2010.

[18] Jonathan G. Fiscus, "A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (rover)," 1997, pp. 347–352.

[19] Masataka Goto, Hiroki Hashiguchi, Takuichi Nishimura, and Ryuichi Oka, "Rwc music database: Popular, classical, and jazz music databases," in *In Proc. 3rd International Conference on Music Information Retrieval*, 2002, pp. 287–288.

[20] J K Hansen, "Recognition of Phonemes in A-cappella Recordings using Temporal Patterns and Mel Frequency Cepstral Coefficients," in *9th Sound and Music Computing Conference (SMC)*, Copenhagen, Denmark, 2012, pp. 494–499.

[21] M. A. Zissman, "Comparison of four approaches to automatic language identification of telephone speech," *IEEE Transactions on Speech and Audio Processing*, vol. 4, no. 1, pp. 31–44, Jan. 1996.

[22] Y K Muthusamy, E Barnard, and R A Cole, "Reviewing automatic language identification," *IEEE Signal Procesing Magazine*, vol. 11, no. 4, pp. 33–41, oct 1994.

[23] E Singer, P A Torres-Carrasquillo, T P Gleason, W M Campbell, and D A Reynolds, "Acoustic, phonetic, and discriminative approaches to automatic language identification," in *Proceedings of Eurospeech*, Geneva, Switzerland, 2003, pp. 1345–1348.

[24] F Allen, E Ambikairajah, and J Epps, "Language identification using Warping and the Shifted Delta Cepstrum," in *2005 IEEE 7th Workshop on Multimedia Signal Processing*, Shanghai, China, 2006, pp. 1–4.

[25] W M Campbell, J P Campbell, D A Reynolds, E Singer, and P A Torres-Carrasquillo, "Support vector machines for speaker and language recognition," *Computer Speech and Language*, vol. 20, pp. 210–229, 2006.

[26] J Schwenninger, R Brueckner, D Willett, and M E Hennecke, "Language Identification in Vocal Music," in *7th International Conference on Music Information Retrieval (ISMIR)*, Victoria, Canada, 2006, pp. 377–379.

[27] W.-H. Tsai and H.-M. Wang, "Towards Automatic Identification Of Singing Language In Popular Music Recordings," in *5th International Conference on Music Information Retrieval (ISMIR)*, Barcelona, Spain, 2004, pp. 568–576.

[28] M Mehrabani and J H L Hansen, "Language identification for singing," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Prague, Czech Republic, 2011, pp. 4408–4411.

[29] V Chandraskehar, M E Sargin, and D A Ross, "Automatic language identification in music videos with low level audio and visual features," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Prague, Czech Republic, 2011, pp. 5724–5727.

[30] H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech," *J. Acoust. Soc. Am.*, vol. 57, no. 4, pp. 1738–52, Apr. 1990.

[31] H Hermansky, N Morgan, A Bayya, and P Kohn, "{RASTA-PLP} Speech Analysis," Tech. Rep. TR-91-069, ICSI, 1991.

[32] B Bielefeld, "Language identification using shifted delta cepstrum," in *Fourteenth annual speech research symposium*, Baltimore, MD, USA, 1994.

[33] P A Torres-Carrasquillo, E Singer, M A Kohler, R J Greene, D A Reynolds, and Jr. J. R. Deller, "Approaches to language identification using Gaussian mixture models and shifted delta cepstral features," in *International Conference on Spoken Language Processing (ICSLP)*, Denver, CO, USA, 2002, pp. 89–92.

[34] Najim Dehak, Patrick J Kenny, Reda Dehak, Pierre Dumouchel, and Pierre Ouellet, "Front-End Factor Analysis for Speaker Verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, may 2011.

[35] D. Martinez, O. Plchot, and L. Burget, "Language Recognition in iVectors Space," in *Interspeech*, Florence, Italy, August 2011, pp. 861–864.

[36] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models," in *Digital Signal Processing*, 2000, p. 2000.

[37] C Charbuillet, D Tardieu, and G Peeters, "GMM Supervector for content based music similarity," *... Conference on Digital ...*, , no. 1, pp. 1–4, 2011.

[38] J. S. Garofolo et al., "TIMIT Acoustic-Phonetic Continuous Speech Corpus," Tech. Rep., Linguistic Data Consortium, Philadelphia, 1993.

[39] Victor Zue, Stephanie Seneff, and James Glass, "Speech database development at MIT: Timit and beyond," *Speech Communication*, vol. 9, no. 4, pp. 351–356, 1990.

[40] R. Cole and Y. Muthusamy, "OGI Multilanguage Corpus," Tech. Rep., Linguistic Data Consortium, Philadelphia, 1994.

[41] J. C. Smith, *Correlation analyses of encoded music performance*, Ph.D. thesis, Stanford University, 2013.

[42] "500 Greatest Albums of All Time (465. The Magnetic Fields, '69 Love Songs')," *Rolling Stone*, May 2012.

[43] LD Beghtol, *Magnetic Fields' 69 Love Songs: A Field Guide (33 1/3)*, Bloomsbury Academic, 2006.

List of Figures

List of Tables

A Appendix

A lot of stuff that didn't fit into the main part ...

B Eigenständigkeitserklärung

Die vorliegende Arbeit habe ich selbstständig ohne Benutzung anderer als der angegebenen Quellen angefertigt.

Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten Quellen entnommen wurden, sind als solche deutlich kenntlich gemacht. Die Arbeit ist in gleicher oder ähnlicher Form oder auszugsweise im Rahmen einer oder anderer Prüfungen noch nicht vorgelegt worden.

Ilmenau, 17.12.2013

Sheldon Cooper

Thesis Summary

1. Scissors cuts paper, paper covers rock, rock crushes lizard, lizard poisons Spock, Spock smashes scissors, scissors decapitates lizard, lizard eats paper, paper disproves Spock, Spock vaporizes rock, and as it always has, rock crushes scissors.
2. I'm not insane, my mother had me tested!
3. All I need is a healthy ovum and I can grow my own Leonard Nimoy!

Thesen

1. These 1
2. These 2
3. These 3