Hindawi Publishing Corporation
EURASIP Journal on Audio, Speech, and Music Processing
Volume 2010, Article ID 546047, 11 pages
doi:10.1155/2010/546047

Research Article

Automatic Recognition of Lyrics in Singing

Annamaria Mesaros and Tuomas Virtanen (EURASIP Member)

Department of Signal Processing, Tampere University of Technology, 33720 Tampere, Finland

Correspondence should be addressed to Annamaria Mesaros, annamaria.mesaros@tut.fi

Received 5 June 2009; Revised 6 October 2009; Accepted 23 November 2009

Academic Editor: Elmar Nöth

The paper considers the task of recognizing phonemes and words from a singing input by using a phonetic hidden Markov model recognizer. The system is targeted to both monophonic singing and singing in polyphonic music. A vocal separation algorithm is applied to separate the singing from polyphonic music. Due to the lack of annotated singing databases, the recognizer is trained using speech and linearly adapted to singing. Global adaptation to singing is found to improve singing recognition performance. Further improvement is obtained by gender-specific adaptation. We also study adaptation with multiple base classes defined by either phonetic or acoustic similarity. We test phoneme-level and word-level *n*-gram language models. The phoneme language models are trained on the speech database text. The large-vocabulary word-level language model is trained on a database of textual lyrics. Two applications are presented. The recognizer is used to align textual lyrics to vocals in polyphonic music, obtaining an average error of 0.94 seconds for line-level alignment. A query-by-singing retrieval application based on the recognized words is also constructed; in 57% of the cases, the first retrieved song is the correct one.

1. Introduction

Singing is used to produce musically relevant sounds by the human voice, and it is employed in most cultures for entertainment or self-expression. It consists of two main aspects: melodic (represented by the time-varying pitch) and verbal (represented by the lyrics). The sung lyrics convey the semantic information, and both the melody and the lyrics allow us to identify the song.

Thanks to the increased amount of music playing devices and available storage and transmission capacity, consumers are able to find plenty of music in the forms of downloadable music, internet radios, personal music collections, and recommendation systems. There is need for automatic music information retrieval techniques, for example, for efficiently finding a particular piece of music from a database or for automatically organizing a database.

The retrieval may be based not only on the genre of the music [1], but as well on the artist identity [2] (artist does not necessarily mean singer). Studies on singing voice have developed methods for detecting singing segments in polyphonic music [3], detecting solo singing segments in music [4], identifying the singer [5, 6] or identifying two simultaneous singers [7].

As humans can recognize a song by its lyrics, information retrieval based on the lyrics has a significant potential. There are online lyrics databases which can be used for finding the lyrics of a particular piece of music, knowing the title and artist. Also, knowing part of the textual lyrics of a song can help identify the song and its author by searching in lyrics databases. Lyrics recognition from singing would allow searching in audio databases, ideally automatically transcribing the lyrics of a song being played. Lyrics recognition can also be used for automatic indexing of music according to automatically transcribed keywords. Another application for lyrics recognition is finding songs using query-by-singing, based on the recognized sung words to find a match in the database. Most of the previous audio-based approaches to query-by-singing have used only the melodic information from singing queries [8] in the retrieval. In [9], a music retrieval algorithm was proposed which used both the lyrics and melodic information. The lyrics recognition grammar was a finite state automaton constructed from the lyrics in the queried database. In [10] the lyrics grammar was constructed for each tested song. Other works related to retrieval using lyrics include [11, 12].

Recognition of phonetic information in polyphonic music is a barely touched domain. Phoneme recognition in individual frames in polyphonic music was studied in [13], but there has been no work done using large vocabulary recognition of lyrics in English.

Because of the difficulty of lyrics recognition, many studies have focused on a simpler task of audio and lyrics alignment [14–18], where the textual lyrics to be synchronized with the singing are known. The authors of [15] present a system based on Viterbi forced alignment. A language model is created by retaining only vowels for Japanese lyrics converted to phonemes. In [16], the authors present LyricAlly, a system that aligns first the higher-level structure of a song and then within the boundaries of the detected sections performs a line-level alignment. The line-level alignment uses only a uniform estimated phoneme duration, rather than a phoneme-recognition-based method. The system works by finding vocal segments but not recognizing their content. Such systems have applications in automatic production of material for entertainment purposes, such as karaoke.

This paper deals with recognition of the lyrics, meaning recognition of the phonemes and words from singing voice, both in monophonic singing and polyphonic music, where other instruments are used together with singing. We aim at developing transcription methods for query-by-singing systems where the input of the system is a singing phrase and the system uses the recognized words to retrieve the song in a database.

The basis for the techniques in this paper is in automatic speech recognition. Even though there are differences between singing voice and spoken voice (see Section 2.1), experiments show that it is possible to use the speech recognition techniques on singing. Section 2.2 presents the speech recognition system. Due to the lack of large enough singing databases to train a recognizer for singing, we use a phonetic recognizer trained on speech and adapt it to singing, as presented in Section 2.3. We use different settings: adapt the models to singing, to gender-dependent models, and to singer-specific models, using different number of base classes in the adaptation. Section 2.4 presents phoneme- and word-level n-gram language models that will be used in the recognition. Experimental evaluation and results are presented in Section 3. Section 4 presents two applications: automatic alignment of audio and lyrics in polyphonic music and a small-scale query-by-singing application. The conclusions and future work are presented in Section 5.

2. Singing Recognition

This section describes the speech recognition techniques used in the singing recognition system. We first review the main differences between speech and singing voice and present the basic structure of the phonetic recognizer architecture, and then the proposed singing adaptation methods and language models.

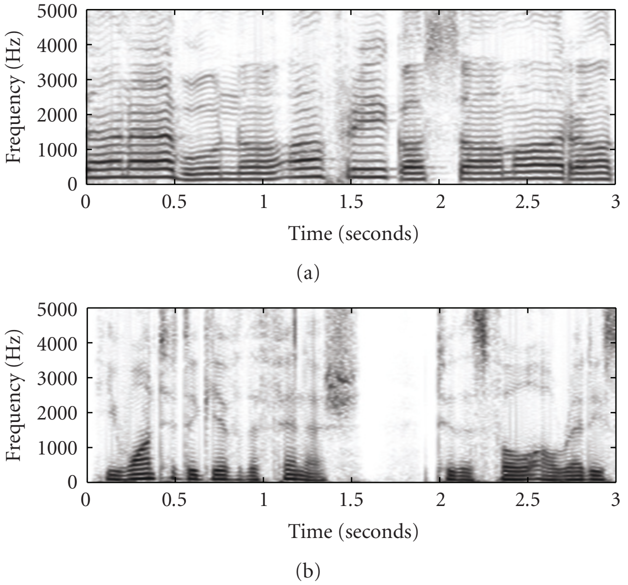


Figure 1: Example spectrograms of male singing (a) and male speech (b). In singing the voice is more continuous, whereas in speech the pitch and formants vary more rapidly in time. In speech the amount of unvoiced segments is higher.

2.1. Singing Voice. Speech and singing convey the same kind of semantic information and originate from the same production physiology. In singing, however, the intelligibility is often secondary to the intonation and musical qualities of the voice. Vowels are sustained much longer in singing than in speech and independent control of pitch and loudness over a large range is required. The properties of the singing voice have been studied in [19].

In normal speech, the spectrum is allowed to vary freely, and the pitch or loudness changes are used to express emotions. In singing, the singer is required to control the pitch, loudness, and timbre.

The timbral properties of a sung note depend on the frequencies at which there are strong and weak partials. In vowels this depends on the formant frequencies, which can be controlled by the length and shape of the vocal tract and the articulators. Different individuals tune their formant frequencies a bit differently for each vowel, but skilled singers can control the pitch and the formant frequencies more accurately.

Another important part of the voice timbre differences between male and female voices seems to be the voice source: major difference is primarily in the amplitude of the fundamental. The voice spectrum of a male voice has a weaker fundamental than the voice spectrum of a female voice.

The pitch range in a singing phrase is usually higher than in a spoken sentence. In speech the pitch varies all the time, whereas in singing it stays approximately constant during a note (with vibrato being used for artistic singing), as illustrated in Figure 1. Therefore in singing the variance of the spectrum of a phoneme with a note is smaller compared to speech, while difference between phonemes sung at different pitches can be much larger. The acoustic features used in the recognizer try to make the representation invariant to pitch changes.

2.2. HMM Phonetic Recognizer. Despite of the above-mentioned differences between speech and singing, they still have many properties in common and it is plausible that singing recognition can be done using the standard technique in automatic speech recognition, a phonetic hidden Markov model (HMM) recognizer. In HMM-based speech recognition it is assumed that the observed sequence of speech feature vectors is generated by a hidden Markov model. An HMM consists of a number of states with associated observation probability distributions and a transition matrix defining transition probabilities between the states. The emission probability density function of each state is modeled by a Gaussian mixture model (GMM).

In the training process, the transition matrix and the means and variances of the Gaussian components in each state are estimated to maximize the likelihood of the observation vectors in the training data. Speaker-independent models can be adapted to the characteristics of a target speaker, and similar techniques can be used for adapting acoustic models trained on speech to singing, as described further in Section 2.3.

Linguistic information about the speech or singing to be recognized can be used to develop language models. They define the set of possible phoneme or word sequences to be recognized and associate probabilities for each of them, improving the robustness of the recognizer.

The singing recognition system used in this work consists of 39 monophone HMMs plus silence and short-pause models. Each phoneme is represented by a left-to-right HMM with three states. The silence model is a fully connected HMM with three states and the short pause is a one-state HMM tied to the middle state of the silence model. The system was implemented using HTK (The Hidden Markov Model Toolkit (HTK), http://htk.eng.cam.ac.uk/).

As features we use 13 mel-frequency cepstral coefficients (MFCCs) plus delta and acceleration coefficients, calculated in 25 ms frames with a 10 ms hop between adjacent frames.

Figure 2 illustrates an example of MFCC features calculated from a descending scale of notes from G#4 to F#3 (fundamental frequency from 415 Hz to 208 Hz) sung by a male singer with the phoneme /m/. Looking at the values of different order MFCCs, we can see that the pitch affects differently the different order coefficients. Using only the low-order cepstral coefficients aims to represent the rough shape of the spectrum (i.e., the formants and phonemes), while making the representation pitch independent. The system also uses cepstral mean normalization [20].

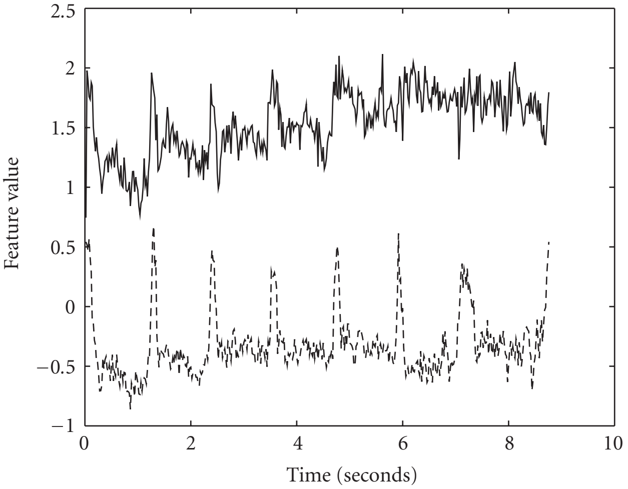


Figure 2: An example of MFCC features calculated from a descending scale of notes from G#4 to F#3 (fundamental frequency from 415 Hz to 208 Hz) sung by a male singer with the phoneme /m/. The solid line represents the 3rd MFCC and the dashed line the 7nd MFCC. The 3rd MFCC is clearly affected by the variation in pitch.

2.3. Adaptation to Singing. Due to the lack of a large enough singing database for training the acoustic models of the recognizer, we first train models for speech and then adapt them linearly to singing.

The acoustic material used for the adaptation is called the *adaptation data*. In speech recognition the data is typically a small amount of speech from the target speaker. The adaptation is done by finding a set of transforms for the model parameters in order to maximize the likelihood that the adapted models have produced the adaptation data. When the phoneme sequence is known, the adaptation can be done in supervised manner.

Maximum linear likelihood regression [21] (MLLR) is a commonly used technique in speaker adaptation. Given mean vector $\boldsymbol{\mu}$ of a mixture component of a GMM, the MLLR estimates a new mean vector as

$$\hat{\boldsymbol{\mu}} = \mathbf{A}\boldsymbol{\mu} + \mathbf{b}, \tag{1}$$

where $\mathbf{A}$ is a linear transform matrix and $\mathbf{b}$ is a bias vector. In constrained MLLR (CMLLR) [22], the covariance matrix $\boldsymbol{\Sigma}$ of the mixture component is also transformed as

$$\hat{\boldsymbol{\Sigma}} = \mathbf{A}\boldsymbol{\Sigma}\mathbf{A}. \tag{2}$$

The transform matrix and bias vector are estimated using the EM algorithm. It has been observed that MLLR can compensate the difference in the lengths of the vocal tract [23].

The same transform and bias vector can be shared between all the Gaussians of all the states (global adaptation). If enough adaptation data is available, multiple transforms can be estimated separately for sets of states or Gaussians. The states or Gaussians can be grouped by either their phonetic similarity or their acoustic similarity. The groups are called *base classes*.

We do the singing adaptation using a two-pass procedure. The usual scenario in speaker adaptation for speech recognition is to use a global transform followed by a second transform with more classes constructed in a data-driven manner, by means of a regression tree [24]. One global

Table 1: Divisions of phonemes into classes by phonetic similarity.

Number of classes	classes
3	vowels, consonants, silence/noise
8	monophthongs, diphthongs, approximants, nasals, fricatives, plosives, affricates, silence/noise
22	one class/vowel, approximants, nasals, fricatives, plosives, affricates, silence/noise

adaptation is suitable best when we have a small amount of training data or when we need a robust transform [25].

Speech and singing are composed from 3 large classes of sounds: vowels, consonants, and nonspeech. As nonspeech sounds we consider the silence and pause models of the speech recognizer. According to this rationale, we use 3 classes for adaptation. We also consider 7 broad phonetic classes: monophthongs, diphthongs, approximants, nasals, fricatives, plosives, and affricates, plus nonspeech, and use 8 classes for adaptation. The singing sounds can be grouped also in 22 classes: each vowel mapped to a separate class, one class for each consonant type (approximants, nasals, fricatives, plosives, affricates) and one for the nonspeech category. These 3 grouping methods are summarized in Table 1.

A second pass of the adaptation uses classes determined by acoustic similarity, by clustering the Gaussians of the states. We use clusters formed from the speech models and from the models after one-pass adaption to singing.

In the initial adaptation experiments [26] we observed that CMLLR performs better than MLLR, and therefore we restrict ourselves to CMLLR in this paper.

2.4. N-Gram Language Models. The linguistic information in the speech or singing to be recognized can be modeled using language models. A language model restricts the possible unit sequences into a set defined by the model. The language model can also provide probabilities for different sequences, which can be used together with the likelihoods of the acoustic HMM model to find the most likely phonetic sequence for an input signal. The language model consists of a *vocabulary* and a set of rules describing how the units in the vocabulary can be connected into sequences. The units in the vocabulary can be defined at different abstraction levels, such as phonemes, syllables, letters, or words.

An n-gram language model can be used to model probabilities of unit sequences in the language model. It associates a probability for each subsequence of length n: given $n - 1$ previous units $w_{i-1}, w_{i-2}, \ldots, w_{i-n}$, it defines the conditional probability $P(w_i \mid w_{i-1}, w_{i-2}, \ldots, w_{i-n})$ [27]. The probability of a whole sequence can be obtained as the product of above conditional probabilities over all i units in the sequence. An n-gram of size one is referred to as a *unigram*; size two is a *bigram*; size three is a *trigram*, while those of higher order are referred to as n-grams. Bigrams and trigrams are commonly used in automatic speech recognition.

It is not possible to include all possible words in a language model. The percentage of out of vocabulary (OOV) words affects the performance of the language model, since the recognition system cannot output them. Instead, the system will output one or more words from the vocabulary that are acoustically close to the word being recognized, resulting in recognition errors. While the vocabulary of the speech recognizer should be as large as possible to ensure low OOV rates, increasing the vocabulary size increases the acoustic confusions and does not always improve the recognition results.

A language model can be assessed by its *perplexity*, which measures the uncertainty in each word based on the language model. It can be viewed as the average size of the word set from which each word recognized by the system is chosen [28, pages 449–450]. The lower the perplexity, the better the language model is able to represent the text. Ideally, a language model should have a small perplexity and a small OOV rate on an unseen text.

The actual recognition is based on finding the most likely sequence of units that has produced the acoustic input signal. The likelihood consists of the contribution of the language model likelihood and the acoustic model likelihood. The influence of the language model can be controlled by the *grammar factor*, which multiplies the log likelihood of the language model. The number of words output by the recognizer can be controlled by the *word insertion penalty* which penalizes the log likelihood by adding a cost for each word. The values of these parameters have to be tuned experimentally to optimize the recognizer performance.

In order to test phoneme recognition with a language model, we built unigram, bigram and trigram language models. As n-grams are used for language recognition, we assume that a phoneme-level language model is characteristic to the English language and cannot differ significantly if estimated from general text or from lyrics text. For this reason, as training data for the phoneme-level language models we used the phonetic transcriptions of the speech database that was used for training the acoustic models.

To construct word language models for speech recognition we have to establish a vocabulary chosen as the most frequent words from the training text data. In large vocabulary recognition it is important to choose training text with similar topic to have a good coverage of vocabulary and words combinations. For our work we chose to use song lyrics text, with the goal of keeping a 5 k vocabulary.

2.5. Separation of Vocals from Polyphonic Music. In order to enable the recognition of singing in polyphonic music, we employ the vocal separation algorithm [29]. The algorithm separates the vocals using the time-varying pitch and enhances them by subtracting a background model. In our study on singer identification from polyphonic music, an earlier version of the algorithm led to a significant improvement [6]. The separation algorithm consists of the following processing steps.

(1) Estimate the notes of the main vocal line using the algorithm [30]. The algorithm has been trained using singing material, and it is able to distinguish between singing and solo instruments, at least to some degree.

(2) Estimate the time-varying pitch of the main vocal line by picking the prominent local maxima in the pitch salience spectrogram near the estimated notes and interpolate between the peaks.

(3) Predict the frequencies of the overtones by assuming perfect harmonicity and generate a binary mask which indicates the predicted vocal regions in the spectrogram. We use a harmonic mask where a ±25 Hz bandwidth around each predicted partial in each frame is marked as speech.

(4) Learn a time-varying background model by using the nonvocal regions and nonnegative spectrogram factorization (NMF) on the magnitude spectrogram of the original signal. A hamming window and absolute values of the frame-wise DFT is used to calculate the magnitude spectrogram. We use a specific NMF algorithm which estimates the NMF model parameters using the nonvocal regions only. The resulting model represents the background accompaniment but not vocals.

(5) The estimated NMF parameters are used to predict the amplitude of the accompaniment in the vocal regions, which is then subtracted from the mixture spectrogram.

(6) The separated vocals are synthesized by assigning the phases of the mixture spectrogram for the estimated magnitude spectrogram of vocals and generating time-domain signal by inverse DFT and overlap add.

More detailed description of the algorithm is given in [29].

The algorithm has been found to produce robust results on realistic music material. It improved the signal-to-noise ratio of the vocals on average by 4.9 dB on material extracted from Karaoke DVDs and on average by 2.1 dB on material synthesized by mixing vocals and MIDI background. The algorithm causes some separation artifacts because of erroneously estimated singing notes (insertions or deletions), and some interference from other instruments. The spectrogram decomposition was found to perform well in learning the background model, since the sounds produced by musical instruments are well represented with the model [31].

Even though the harmonic model for singing used in the algorithm does not directly allow representing unvoiced sounds, the use of mixture phases carries some information about them. Furthermore, the rate of unvoiced sounds in singing is low, so that the effect of voiced sounds dominates the recognition.

3. Recognition Experiments

We study the effect of above recognition techniques for recognition of phonemes and words in both clean singing and singing separated from polyphonic music. Different adaptation approaches and both phoneme-level and word-level language models are studied.

3.1. Acoustic Data. The acoustic models of the recognizer were trained using the CMU Arctic speech database (CMU ARCTIC databases for speech synthesis: http://festvox.org/cmuarctic/). For testing and adaptation of the models we used a database containing monophonic singing recordings, 49 fragments (19 male and 30 female) of popular songs, which we denote as *vox_clean*. The lengths of the sung phrases are between 20 and 30 seconds and usually consist in a full verse of a song. For the adaptation and testing on clean singing we used m-fold cross-validation, with m depending on the test case. The total amount of singing material is 30 minutes, and it consists of 4770 phoneme instances.

To test the recognition system on polyphonic music we chose 17 songs from commercial music collections. The songs were manually segmented into structurally meaningful units (verse, chorus) to obtain sung phrases having approximately the same durations with the fragments in the monophonic database. We obtained 100 fragments of polyphonic music containing singing and instrumental accompaniment. We denote this database as *poly*_100. For this testing case, the HMMs were adapted using the entire *vox_clean* database. In order to suppress the effect of the instrumental accompaniment, we applied the vocal separation algorithm described in Section 2.5.

The lyrics of both singing databases were manually annotated for reference. The transcriptions are used in the supervised adaptation procedure and in the evaluation of the automatic lyrics recognition.

3.2. Evaluation. We measure the recognition performance using correct recognition rate and accuracy of the recognition. They are defined in terms of the number of substitution errors S, deletion errors D, and insertion errors I, reported for the total number of tested instances N. The correct rate is given as

$$\text{correct (\%)} = \frac{N - D - S}{N} \times 100 \tag{3}$$

and the accuracy as

$$\text{accuracy (\%)} = \frac{N - D - S - I}{N} \times 100. \tag{4}$$

The two measures differ only in the number of insertions. In speech recognition usually the reported results are word error rates. The error rate is defined as

$$\text{error rate (\%)} = \frac{D + S + I}{N} \times 100. \tag{5}$$

3.3. Adaptation of Models to Singing Voice. For adapting the models to singing voice, we use a 5-fold setting for the *vox_clean* database, with one fifth of the data used as test data at a time and the rest for adaptation. As each song

Table 2: Phoneme recognition rates (39 + sil) for clean singing for systems adapted with different number of base classes in the first pass and 8 classes in the second pass.

First pass	Correct	Accuracy	Second pass	Correct	Accuracy
nonadapted	33.3%	−6.4%	–		
G	41.2%	20.0%	GT8	41.3%	18.9%
G3	40.4%	19.9%	G3T8	41.4%	20.0%
G8	40.3%	18.7%	G8T8	41.1%	18.9%
G22	38.4%	18.7%	G22T8	40.7%	19.5%

was sung by multiple singers, splitting into folds was done so that the same song appeared either in the test or in the adaptation set, not in both. The same singer was allowed in both adaptation and testing sets. We adapt the models using supervised adaptation procedure, providing the correct transcription to the system in the adaptation process.

Evaluation of the recognition performance was done without a language model; the number of insertion errors was controlled by the insertion penalty parameter with value fixed to $p = -10$ (for reasons explained in Section 3.6).

Table 2 presents recognition rates for systems adapted to singing voice using different number of classes in the first adaptation pass, as presented in Table 1. A single global transform (G) improves clearly the performance of the nonadapted system. A larger number of base classes in the first adaptation pass improves the performance in comparison with the nonadapted system, but the performance decreases from the single class adaptation. Using all the information to estimate a global transform provides a more reliable adaptation than splitting the available information between different classes. In case of multiple classes it can happen that for some class there might not be enough data for estimating a robust enough transform.

A second pass was added, using classes defined by acoustic similarity. Different numbers of classes were clustered using the speech models and also the models already adapted to singing in the first pass. Figure 3 presents the average recognition results for 2 to 20 classes in the two cases. The differences in the adaptation are not statistically significant (the maximum 95% confidence interval for the test cases is 2%), but this might be due to the limited amount of data available for the adaptation. The adaptation classes constructed from the singing-adapted models reflect better the characteristics of the signal and lead to more reliable adaptation. Still, the performance does not vary much as a function of the number of base classes.

Table 2 also presents recognition performance of systems adapted with different number of classes in the first pass and 8 classes in the second pass using the clustering to form these 8 classes. The second pass improve slightly the correct rate of systems where multiple classes were used in the first adaptation pass.

For a better understanding of the adaptation process effect, in Table 3 we present the phoneme recognition rates of the nonadapted models and one set of models adapted to singing, using as test data the speech database and the clean singing database. The system adapted to singing has much lower recognition rates on speech. The adaptation process modifies the models such that the adapted models do not represent the speech vowels anymore.

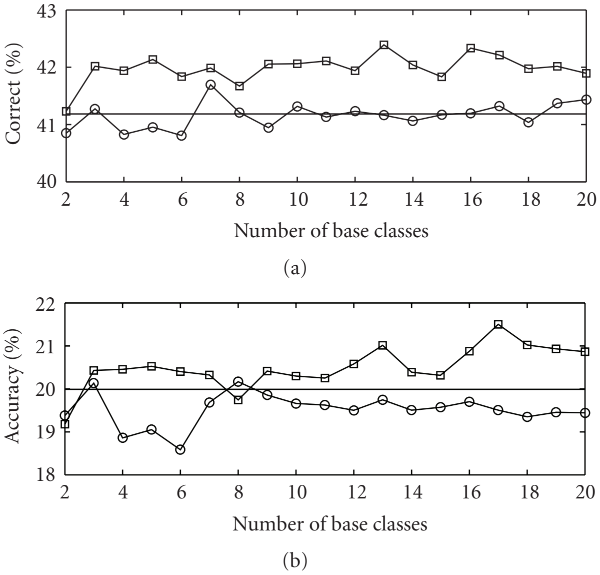


Figure 3: Correct and accuracy rates of systems adapted to singing using global adaptation in the first pass and 2 to 20 base classes in the second pass; the classes are determined by clustering acoustically similar Gaussians from speech (marked with circle) and adapted to singing (marked with square) models. The baseline of global adaptation is marked with a constant line.

Table 3: Phoneme recognition rates (correct % / accuracy %) for speech and singing using nonadapted and singing-adapted models.

Test data	Nonadapted models	Models adapted to singing (G3T8)
speech	41.6/30.4	9.8/9.3
singing	33.3/−6.4	41.4/20.0

3.4. Gender-Dependent Adaptation. The gender differences in singing voices are much more evident than in speech, because of different singing techniques explained in Section 2.1. Gender-dependent models are used in many cases in speech recognition [32].

Adaptation to male singing voice is tested on four different male voices. The fragments belonging to the same voice were kept as test data, while the rest of the male singing in *vox_clean* was used to adapt the speech models using a one-pass global adaptation. We do the same for adaptation to female singing voice, using as adaptation data all the female singing fragments except the tested one.

The results for individual voices for male and female gender-adapted recognition systems are presented in Table 4, together with the averages over genders. The other column in the table represents recognition results for the same test data using the nonadapted system.

Table 4: Phoneme recognition rates (correct % / accuracy %) for nonadapted and gender-adapted models for 4 different male and female sets.

Data set	Test fragments	Nonadapted	Gender-adapted
Male 1	3	39.4/8.1	59.3/28.8
Male 2	5	33.0/7.8	46.4/25.2
Male 3	3	36.4/6.6	52.2/28.2
Male 4	5	32.9/−10.7	48.0/1.9
Male average		35.4/2.9	51.5/21.0
Fem 1	3	31.2/−16.1	52.3/10.9
Fem 2	5	33.4/−21.2	58.7/21.2
Fem 3	3	41.4/−11.7	60.7/9.9
Fem 4	4	29.5/−11.2	51.6/24.0
Fem average		33.9/−15.0	55.8/16.5

The gender-specific adaptation improves the recognition performance for all the singers. Especially the recognition performance for female singers is improved, from negative values in the case of the nonadapted system. Negative accuracy means over 100% error rate, rendering recognition results unusable. The testing in this case was also done without a language model, using the fixed insertion penalty $p = -10$ (see Section 3.6 for explanation).

3.5. Singer-Specific Adaptation. The models adapted to singing voice can be further adapted to a target singer. We tested singer adaptation for three male and three female voices. The adaptation to singing was carried out in a one-pass global step, using all the singing material from *vox_clean* except the target voice. After this, the adapted models were adapted using another one-pass global adaptation and tested in 3-fold (for male 1, male 3, Fem 1, and Fem 3) or 5-fold (for male 2 and Fem 2), so that one fragment at a time was used in testing and the rest as adaptation data.

Table 5 presents the recognition rates for the six target voices, for nonadapted, adapted to singing, and adapted to target voice systems. On average, the recognition performance of singer-specific adapted systems is lower than that of the systems adapted to singing in general. The first adaptation estimates a transform from speech to singing voice, but its advantage is lost by trying to adapt the models further for a target singer. This situation may be due to the very small amount of adaptation data in the attempt to overfit it [25].

There are significant differences between male and female singers, which explains the fact that a gender-dependent recognizer performs better than gender-independent recognizer. The gender-adapted systems have lower accuracy than the systems adapted to singing, but higher correct rate. The two situations may need different tuning of the recognition step parameters (here only p) in order to maximize the accuracy of the recognition, but we kept the same value for comparison purposes.

3.6. Language Models and Recognition Results. The phoneme-level language models were trained using the phonetic transcriptions of the speech database that was used for training the acoustic models. The database contains 1132 phonetically balanced sentences, over 48000 phoneme instances.

Table 5: Phoneme recognition rates (correct % / accuracy %) for 3 male and 3 female voice adapted systems.

Voice	Nonadapted	Adapted to singing	Adapted to voice
Male 1	39.4/8.1	47.1/41.0	30.5/20.4
Male 3	36.4 /6.6	42.9/33.1	28.7/12.5
Male 4	32.9/−10.7	36.0/16.2	35.0/20.4
Fem 1	31.2/−16.1	39.3/20.6	27.1/12.5
Fem 2	33.4/−21.2	37.7/22.0	36.4/12.5
Fem 3	41.4/−11.7	47.3/13.0	29.9/1.9

Table 6: Perplexities of bigram and trigram phoneme and word level language models on the training text (speech database transcriptions for phonemes, lyrics text for words) and on test lyrics texts.

Language model	Training text	*vox_clean*	*poly_*100
Phoneme bigram	11.5	11.8	11.3
Phoneme trigram	6.4	8.4	8.3
Word bigram	90.2	147.1	97.8
Word trigram	53.7	117.8	77.5
OOV %	5.9	2.2	2.5

To test the modeling capabilities of the constructed language models, we evaluate perplexities on the test text in comparison with their perplexity on the training text. The perplexities of the phoneme bigram, and trigram models on the speech training text and on the lyrics text from *vox_clean* and *poly_*100 databases are presented in Table 6. For a phoneme language model there is no concern over OOV, since all the phonemes are included in the vocabulary of the LM and of the recognizer. According to the perplexities, our assumption is correct, and the phoneme language model built using speech represents well the lyrics text too.

For constructing a word language model we used the lyrics text of 4470 songs, containing over 1.2 million word instances, retrieved from http://www.azlyrics.com/. From a total of approximately 26000 unique words, a vocabulary of 5167 words was chosen by keeping the words that appeared at least 5 times. The perplexities of bigram and trigram word level language models evaluated on the training data and on the lyrics text of *vox_clean* and *poly_*100 databases are also presented in Table 6. The percentage of OOV words on the training text represents mostly words in languages other than English, also the words that appeared too few times and were removed when choosing the vocabulary. The perplexities of the language models on *poly_*100 are not much higher than the ones on the training text, meaning that the texts are similar regarding the used words. The *vox_clean* text is less well modeled by this language model. Nonetheless, in almost 4500 songs we could only find slightly over 5000

Table 7: Phoneme recognition rates (correct % / accuracy %) for monophonic singing with no language model, unigram, bigram or trigram, using gender-adapted models.

LM	Male	Female
No language model	51.5/21.0	55.8/16.5
Phoneme unigram	44.8/32.8	47.8/29.3
Phoneme bigram	45.3/34.9	50.6/32.0
Phoneme trigram	49.3/16.3	56.2/16.6

Table 8: Phoneme recognition rates (correct / accuracy %) for vocal line extracted from polyphonic music, with no language model, unigram, bigram or trigram, using singing-adapted models.

LM	Correct%	Accuracy%
No language model	23.5	5.8
Phoneme unigram	18.7	16.7
Phoneme bigram	21.8	19.2
Phoneme trigram	30.2	13.8

Table 9: Word recognition for clean singing and vocal line extracted from polyphonic music, with bigram and trigram language models.

	LM	Correct%	Accuracy%
Clean singing	Word bigram	23.9	12.4
	Word trigram	21.0	−1.4
Polyphonic	Word bigram	6.8	5.5
	Word trigram	6.5	3.9

words that appear more than 5 times; thus, the language model for lyrics of mainstream songs is quite restricted vocabulary-wise.

In the recognition process, the acoustic models provide a number of hypotheses as output. The language model provides complementary knowledge about how likely those hypotheses are. The balance of the two components in the Viterbi decoding can be controlled using the grammar scale factor s and the insertion penalty parameter p. These parameters are usually set experimentally to values where the number of insertion errors and deletion errors in recognition is nearly equal. We fixed the values to $s = 5$ and $p = -10$, where deletion and insertion errors for phoneme recognition using bigrams were approximately equal. No tuning was done for the other language models to maximize accuracy of the recognition results.

3.6.1. Phoneme Recognition. The average phoneme recognition rates for clean singing voice using different language models are presented in Table 7. The systems used in the test use the gender-adapted models, with adaptation steps and test settings described in Section 3.4. The parameters s and p are 5 and -10, respectively.

When there is no prior information about the phoneme probabilities (no language model), the rate of recognized phonemes is quite high, but with a low accuracy. Including phoneme probabilities into the recognition process (unigram language model), the accuracy of the recognition improves significantly. The bigram language models gives more control over the output of the recognizer, yielding better rates than the unigram. For the trigram language model we obtained higher recognition rate but with lower accuracy of the recognition. This case might also need different tuning of the language model control parameters to maximize the accuracy of the recognition, but we kept the same values for comparison purposes.

The 2000 NIST evaluation of Switchboard corpus automatic speech recognition systems [33] reports error rates of 39%–55% for phoneme recognition in speech, while the lowest error rate (100-accuracy) in Table 7 is approximately 65%. Even though our singing recognition performance results are clearly lower, we find our results encouraging, considering that singing recognition has not been studied before.

Phoneme recognition results for vocal lines separated from polyphonic music are presented in Table 8. We did not use the gender information about the polyphonic material, therefore we used the systems adapted to singing using the entire material from the *vox_clean* database. The separated vocal line is more difficult to recognize, because of some interference of other sources which have not been properly separated, and also artifacts caused by the separation algorithm. In some cases parts of the singing are missing, for example, consonants being removed at the beginning of the word by the separation algorithm, resulting in recognition errors.

3.6.2. Word Recognition. Word recognition of monophonic singing was tested on the *vox_clean* database in the 5-fold setup presented in Section 3.3. We use the word language models presented in Section 3.6, with a vocabulary size of 5167. The recognition results for bigram and trigram language models are presented in Table 9. Again, the language model and insertion penalty parameters were kept fixed. In this case, the use of the bigram language model offers better results than the trigram. The trigram language model results to negative accuracy for the female test case, meaning there are too many insertion errors. The best results obtained are the correct recognition of one fifth of the words, using the bigram language model. Recognition rate of singing extracted from polyphonic music using the same vocabulary and language models is presented in the same table.

If a closed vocabulary language model can be constructed from the lyrics of the songs in the database, then such knowledge gives an important advantage for recognition [9]. For example, in the case of the *vox_clean* database, a bigram language model constructed from the lyrics text of database has a vocabulary of only 185 words (compared to the vocabulary size of 5167 of the previously used language model) and a perplexity of 2.9 on the same text, offering a recognition rate of 55% with 40% accuracy for the singing-adapted models in the 5-fold test case.

The word recognition results are low, with even lower accuracy, and as a speech recognition tool, this system fails. Still, thinking about information retrieval purposes, even highly imperfect transcriptions of the lyrics can be useful. By maximizing the rate of correctly recognized words, even with producing a lot of insertion errors, the results may be useful. In the next section we present two applications for lyrics recognition.

4. Applications

4.1. Automatic Singing-to-Lyrics Alignment. Alignment of singing to lyrics refers to finding the temporal relationship between a possibly polyphonic music audio and the corresponding textual lyrics. We further present the system developed in [17].

A straightforward way to do alignment is by creating a phonetic transcription of the word sequence comprising the text in the lyrics and aligning the corresponding phoneme sequence with the audio using the HMM recognizer. For alignment, the possible paths in the Viterbi search algorithm are restricted to just one string of phonemes, representing the input text.

The polyphonic audio from the *poly*_100 database was preprocessed to separate the singing voice. The text files are processed to obtain a sequence of words with optional silence, pause, and noise between them. An optional short pause is inserted between each two words in the lyrics. At the end of each line we insert optional silence or noise event, to account for the voice rest and possible background accompaniment. An example of resulting grammar for one of the test songs is

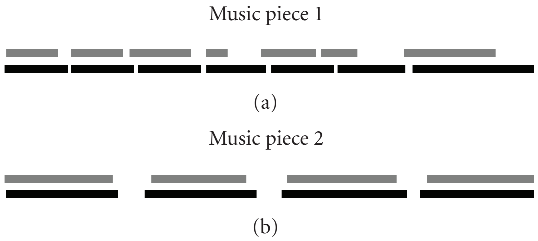
> [sil | noise] I [sp] BELIEVE [sp] I [sp] CAN [sp] FLY [sil | noise] I [sp] BELIEVE [sp] I [sp] CAN [sp] TOUCH [sp] THE [sp] SKY [sil | noise] I [sp] THINK [sp] ABOUT [sp] IT [sp] EVERY [sp] NIGHT [sp] AND [sp] DAY [sil | noise] SPREAD [sp] MY [sp] WINGS [sp] AND [sp] FLY [sp] AWAY [sil | noise]

where the [ ] encloses options and | denotes alternatives. This way, the alignment algorithm can choose to include pauses and noise where needed. The noise model was separately trained on instrumental sections from different songs, other than the ones in the test database.

The text input contains a number of lines of text, each line corresponding roughly to one singing phrase.

The timestamps for beginning and end of each line in the lyrics were manually annotated. As a performance measure of the alignment we use the average of the absolute alignment errors in seconds at the beginning and at the end of each lyric line. The absolute alignment errors range from 0 to 9 seconds. The average errors for different adapted systems are presented in Table 10. The best achieved performance is 0.94 seconds average absolute alignment error. Examples of alignments are presented in Figure 4.

One main reason for misalignments is a faulty output of the vocal separation stage. Some of the songs are from pop-rock genre, featuring loud instruments as an accompaniment, and the melody transcription (step 1 in Section 2.5) fails to pick the voice signal. In this case, the output contains a mixture of the vocals with some instrumental sounds, but the voice is usually too distorted to be recognizable. In other cases, the errors appear when the transcribed lyrics do not have the lines corresponding to singing phrases, so there are breathing pauses in the middle of a text line. In these cases even the manual annotation of the lyrics can have ambiguity.


Music piece 1

(a)

Music piece 2

(b)

Figure 4: Automatic alignment examples. The black line represents the manual annotation and the gray line the automatic alignment output. The errors are calculated at the ends of each black segment.

Table 10: Average alignment errors for different sets of singing-adapted models.

Adapted system	Avg. error (s)	Adapted system	Avg. error (s)
G3	1.27	G3T8	0.97
G8	1.31	G8T8	0.94
G22	1.31	G22T8	1.07

The relatively low average alignment error indicates that this approach could be used to produce automatic alignment of lyrics for various applications such as automatic lyrics display in karaoke.

4.2. Query-by-Singing Based on Word Recognition. In query-by-humming/singing, the aim is to identify a piece of music from its melody and lyrics. In a query-by-humming application, the search algorithm will transcribe the melody sung by the user and will try to find a match of the sung query with a melody from the database. For large databases, the search time can be significantly long. Assuming that we also have the lyrics of the songs we are searching through, the words output from a phonetic recognizer can be searched for in the lyrics text files. This will provide additional information and narrow down the melody search space. Furthermore, lyrics will be more reliable than the melody in the case of less skilled singers.

The output of the recognition system offers sometimes words that are acoustically very similar with the correct ones, sometimes cases with different spelling but same phonetic transcription. For recognition performance evaluation they count as errors, but for music information retrieval purpose, we do not need perfect transcription of the lyrics. Some examples representing typical recognition results can be found in Table 11.

We built a retrieval system based on sung queries recognized by the system presented in Table 9 (23.93% correct recognition rate), which that uses a bigram language model to recognize the clean singing voice in the presented 5-fold experiment. For this purpose, we constructed a lyrics database consisting of the text lyrics of *poly*_100 and

Table 11: Examples of errors in recognition.

correct transcription	recognized
yesterday	yes today
seemed so far away	seem to find away
my my	mama
finding the answer	fighting the answer
the distance in your eyes	from this is in your eyes
all the way	all away
cause it's a bittersweet symphony	cause I said bittersweet symphony
this life	this our life
trying to make ends meet	trying to maintain sweetest
you're a slave to the money	ain't gettin' money
then you die	then you down
I heard you crying loud	I heard you crying alone
all the way across town	all away across the sign
you've been searching for that someone	you been searching for someone
and it's me out on the prowl	I miss me I don't apologize
as you sit around	you see the rhyme
feeling sorry for yourself	feelin' so free yourself

Table 12: Query-by-singing retrieval results.

	Top 1	Top 5	Top 10
recognized (%)	57%	67%	71%

vox_clean databases. We used as test queries the 49 singing fragments of the *vox_clean* database. The recognized words for each query will be matched to the content of the lyrics database to identify the queried song.

For retrieval we use a bag-of-words approach, simply searching for each recognized word in all the text files and ranking the songs according to the number of matched words. We consider a song being correctly identified when the queried fragment appears among the first N-ranked lyrics files. Table 12 presents the retrieval accuracy for N being 1, 5, and 10. The application shows promising results, the first retrieved song being correct in 57% of the cases.

5. Conclusions

This paper applied speech recognition methods to recognize lyrics from singing voice. We attempt to recognize phonemes and words in singing voice from monophonic singing input and from polyphonic music. In order to suppress the effect of the instrumental accompaniment, a vocal separation algorithm was applied.

Due to the lack of large enough singing databases to train a singing recognizer, we used a phonetic recognizer that was trained on speech and applied speaker adaptation techniques to adapt the models to singing voice. Different adaptation setups were considered. A general adaptation to singing using a global transform was found to provide a system with much higher performance in recognizing sung phonemes than the nonadapted one. Different numbers of base classes in the setup of the adaptation did not have very much importance for the system performance. Separate adaptation using male and female data led to gender-dependent models, producing the best performance in phoneme recognition. More specific speaker-dependent adaptation did not improve the results, but this may be due to the limited amount of speaker-specific adaptation data in comparison with the adaptation data used to adapt the speech models to singing in general.

The recognition results are also influenced by the language models used in the recognition process. Phoneme bigram language model built on phonetically balanced speech data was found to increase the accuracy of the recognition with up to 13% both for clean singing test cases and for vocal line extracted from polyphonic music. Word recognition in clean singing using a bigram language model built from lyrics text allows recognition of approximately one fifth of the sung words in clean singing. In polyphonic music, the results are lower.

Even though the results are far from being perfect, they have potential in music information retrieval. Our query-by-singing experiment indicates that a song might be retrieved based on words that are correctly recognized from a user query. We also demonstrated the capability of the recognition methods in automatic alignment of singing from polyphonic audio and text, where an average alignment error of 0.94 seconds was obtained.

The constructed applications prove that even such low recognition results can be useful in particular tasks. Still, it is important to find methods for improving the recognition rates. Ideally, a lyrics recognition system should be trained on singing material. We lack a large enough database with monophonic recordings, but we do have at our disposal plenty of polyphonic material. One approach could be using vocals separated from polyphonic music for training of the models. Also, considering that there are millions of songs out there, we know that we only selected a small amount of information to build the word language model. A better selection of the vocabulary can be obtained by using more text in the construction of the language model.

Acknowledgment

This research has been funded by the Academy of Finland.

References

[1] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 5, pp. 293–302, 2002.

[2] S. Downie, K. West, A. Ehmann, and E. Vincent, "The 2005 music information retrieval evaluation exchange (MIREX 2005): preliminary overview," in *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR '05)*, 2005.

[3] S. Z. K. Khine, T. L. New, and H. Li, "Singing voice detection in pop songs using co-training algorithm," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '08)*, pp. 1629–1632, 2008.

[4] C. Smit and D. P. W. Ellis, "Solo voice detection vio optimal cancelation," in *Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2007.

[5] W.-H. Tsai and H.-M. Wang, "Automatic singer recognition of popular music recordings via estimation and modeling of solo vocal signals," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, no. 1, pp. 330–341, 2006.

[6] A. Mesaros, T. Virtanen, and A. Klapuri, "Singer identification and polyphonic music using vocal separation and pattern recognition methods," in *Proceedings of International Conference on Music Information Retrieval (ISMIR '07)*, 2007.

[7] W.-H. Tsai, S.-J. Liao, and C. Lai, "Automatic identification of simultaneous singers in duet recordings," in *Proceedings of International Conference on Music Information Retrieval (ISMIR '08)*, 2008.

[8] R. Typke, F. Wiering, and R. C. Veltkamp, "Mirex symbolic melodic similarity and query by singing/humming," in *International Music Information Retrieval Systems Evaluation Laboratory(IMIRSEL)*, http://www.music-ir.org/mirex/2006/index.php/Main_Page.

[9] M. Suzuki, T. Hosoya, A. Ito, and S. Makino, "Music information retrieval from a singing voice using lyrics and melody information," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, Article ID 38727, 8 pages, 2007.

[10] A. Sasou, M. Goto, S. Hayamizu, and K. Tanaka, "An autoregressive, non-stationary excited signal parameter estimation method and an evaluation of a singing-voice recognition," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '05)*, vol. 1, pp. 237–240, 2005.

[11] H. Fujihara, M. Goto, and J. Ogata, "Hyperlinking lyrics: a method for creating hyperlinks between phrases in song lyrics," in *Proceedings of the 9th International Conference on Music Information Retrieval (ISMIR '08)*, 2008.

[12] M. Müller, F. Kurth, D. Damm, C. Fremerey, and M. Clausen, "Lyrics-based audio retrieval and multimodal navigation in music collections," in *Proceedings of European Conference on Research and Advanced Technology for Digital Libraries*, pp. 112–123, 2007.

[13] M. Gruhne, K. Schmidt, and C. Dittmar, "Phoneme recognition in popular music," in *Proceedings of the 8th International Conference on Music Information Retrieval (ISMIR '07)*, 2007.

[14] H. Fujihara and M. Goto, "Three techniques for improving automatic synchronization between music and lyrics: fricative detection, filler model, and novel feature vectors for vocal activity detection," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '08)*, pp. 69–72, 2008.

[15] H. Fujihara, M. Goto, J. Ogata, K. Komatani, T. Ogata, and H. G. Okuno, "Automatic synchronization between lyrics and music CD recordings based on Viterbi alignment of segregated vocal signals," in *Proceedings of the 8th IEEE International Symposium on Multimedia (ISM '06)*, pp. 257–264, 2006.

[16] M.-Y. Kan, Y. Wang, D. Iskandar, T. L. Nwe, and A. Shenoy, "LyricAlly: automatic synchronization of textual lyrics to acoustic music signals," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 16, no. 2, pp. 338–349, 2008.

[17] A. Mesaros and T. Virtanen, "Automatic alignment of music audio and lyrics," in *Proceedings of the 11th International Conference on Digital Audio Effects (DAFx '08)*, 2008.

[18] C. H. Wong, W. M. Szeto, and K. H. Wong, "Automatic lyrics alignment for Cantonese popular music," *Multimedia Systems*, vol. 12, no. 4-5, pp. 307–323, 2007.

[19] J. Sundberg, *The Science of Singing Voice*, Northern Illinois University Press, 1987.

[20] B. S. Atal, "Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification," *Journal of the Acoustical Society of America*, vol. 55, no. 6, pp. 1304–1312, 1974.

[21] C. J. Leggetter and P. C. Woodland, "Flexible speaker adaptation using maximum likelihood linear regression," in *Proceedings of the ARPA Spoken Language Technology Workshop*, 1995.

[22] M. J. F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer Speech and Language*, vol. 12, no. 2, pp. 75–98, 1998.

[23] M. Pitz and H. Ney, "Vocal tract normalization equals linear transformation in cepstral space," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 5, pp. 930–944, 2005.

[24] D. Pye and P. C. Woodland, "Experiments in speaker normalisation and adaptation for large vocabulary speech recognition," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '97)*, 1997.

[25] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models," *Computer Speech and Language*, vol. 9, no. 2, pp. 171–185, 1995.

[26] A. Mesaros and T. Virtanen, "Adaptation of a speech recognizer to singing voice," in *Proceedings of 17th European Signal Processing Conference*, 2009.

[27] D. Jurafsky and J. H. Martin, *Speech and Language Processing*, Prentice-Hall, Upper Saddle River, NJ, USA, 2000.

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

[29] T. Virtanen, A. Mesaros, and M. Ryynänen, "Combining pitch-based inference and non-negative spectrogram factorization in separating vocals from polyphonic music," in *Proceedings of the ISCA Tutorial and Research Workshop on Statistical and Perceptual Audition SAPA*, 2008.

[30] M. P. Ryynänen and A. P. Klapuri, "Automatic transcription of melody, bass line, and chords in polyphonic music," *Computer Music Journal*, vol. 32, no. 3, pp. 72–86, 2008.

[31] T. Virtanen, "Monaural sound source separation by non-negative matrix factorization with temporal continuity and sparseness criteria," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 3, 2007.

[32] P. C. Woodland, J. J. Odell, V. Valtchev, and S. J. Young, "Large vocabulary continuous speech recognition using HTK," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP '94)*, 1994.

[33] S. Greenberg, S. Chang, and J. Hollenback, "An introduction to the diagnostic evaluation of the switchboard-corpus automatic speech recognition systems," in *Proceedings of the NIST Speech Transcription Workshop*, 2000.