TOWARDS AUTOMATIC IDENTIFICATION OF SINGING LANGUAGE IN POPULAR MUSIC RECORDINGS

Wei-Ho Tsai and *Hsin-Min Wang*

Institute of Information Science, Academia Sinica
Nankang, 115, Taipei, Taiwan, Republic of China
{wesley,whm}@iis.sinica.edu.tw

ABSTRACT

The automatic analysis of singing from music is an important and challenging issue within the research target of content-based retrieval of music information. As part of this research target, this study presents a first attempt to automatically identify the language sung in a music recording. It is assumed that each language has its own set of constraints that specify which of the basic linguistic events present in a singing process are allowed to follow another. The acoustic structure of individual languages may, thus, be characterized by statistically modeling those constraints. To this end, the proposed method employs vector clustering to convert a singing signal from its spectrum-based feature representation into a sequence of smaller basic phonological units. The dynamic characteristics of the sequence are then analyzed by using bigram language models. Since the vector clustering is performed in an unsupervised manner, the resulting system does not use sophisticated linguistic knowledge and, thus, is easily portable to new language sets. In addition, to eliminate the interference of background music, we leverage the statistical estimation of a piece's music background so that the vector clustering is relevant to the solo singing voices in the accompanied signals.

1. INTRODUCTION

Recent advances in digital signal processing technologies, coupled with what are essentially unlimited data storage and transmission capabilities, have created an unprecedented growth of music material being produced, distributed, and made available universally. On the other hand, our ever-increasing appetite for music has provided a major impetus for the development of various new technologies. However, as the amount of music-related data and information continues to grow, finding the desired item from the innumerable options can, ironically, become more and more difficult. This problem has consequently motivated research into developing techniques for automatically extracting information from music. Specific topics such as melody spotting [1], instrument recognition [5], music score transcription [11], and genre classification [19], are being extensively studied within the overall context of content-based retrieval of music information. More recently, research in this area has made a foray into the problem of extracting singing information from music, such as lyric recognition [20] – decoding what is sung; and singer identification [10] – determining who is singing. In tandem with the above research, this study presents a first attempt to identify the singing language of a song. Specifically, it aims to determine which among a set of candidate languages is sung in a given music recording.

Singing Language IDentification (singing LID) is useful for organizing multilingual music collections that are unlabeled or insufficiently labeled. For instance, a song titled in English, but not sung in English, is commonplace in popular music, and very often it is not easy to infer the language of a song simply from its title. In such a case, singing LID can be deployed to categorize music recordings by language, without needing to refer to the lyrics or other information attached textually to the recordings. This function could support preference-based searches for music and may also be useful for assisting other techniques for classifying music, such as genre classification. Singing LID can also be used to distinguish between songs that have the same tune, but different languages. Such a case exists commonly in cover versions of songs, in which a singer performs a song written or made famous by a different artist. Since popular songs are often translated from one language to another and the titles are changed accordingly, singing LID could aid a melody-based music retrieval system for better handling of multilingual music documents.

Relatedly, copious amounts of research have been performed on spoken language identification (spoken LID) [12,16], which aims to identify the language being spoken from a sample of speech by an unknown speaker. Spurred by the market trend and the need to provide services to a wide public, spoken LID has been gaining in importance as a key step toward multilingual automatic systems such as multilingual speech recognition, information retrieval, and spoken language translation. Various methods [7,8,21] have been proposed in attempts to mimic the ability of humans to

distinguish between languages. From a linguistic standpoint, spoken languages can be distinguished from one another by the following traits.

- Phonology. Phonetic inventories are different from one language to another. Even when languages have nearly identical phones, the frequency of the occurrence of phones and the combinations of phones differ significantly across languages.
- Prosody. Significant differences exist in the duration of phones, speech rate and the intonation across different languages.
- Vocabulary. Each language has its own word roots and lexicons, and the process of word formation is also different from one language to another.
- Grammar. The syntactic and semantic rules which govern the concatenation of words into spoken utterances can vary greatly from language to language.

Although humans identify the language of a speech utterance by using one or a multiple of the traits described above, spoken-LID research to date has not exhaustively exploited all of these traits. Instead, it has developed methods which are reliable, computationally efficient, and easily portable to new language sets. In particular, phonological and prosodic information are the most prevalent cues exploited for spoken LID, since they are easily extracted from the acoustic signal without requiring too much language-specific knowledge. More particularly, a very promising and feasible way for spoken LID [8,14] is the stochastic modeling of the so-called *phonotactics*, i.e., the dependencies inherent in the phonetic elements of utterances. A spoken-LID system based on phonotactics commonly consists of a phonetic element recognizer, followed by a set of *n-gram*-based language models. There are also various modifications thereof [6,14,21]. Other combinations that use other language-discriminating information [2,4,7], and do not involve complex linguistic knowledge, are also being studied to improve spoken-LID performance.

Intuitively, singing LID might be performed using the methods for spoken LID. However, singing differs from speech in many ways, including various phonological modifications employed by singers, prosodic shaping to fit the overall melody, and the peculiar wordings used in lyrics. Moreover, interference caused by the background music is often inevitable in most popular songs. As a result, porting a well-developed technique of spoken LID to the singing LID may present its own set of problems. For example, in using phonotactic information for singing LID, it is rather difficult and cost prohibitive to build a phone recognizer capable of handling accompanied singing signals with satisfactory accuracy and reliability. In addition, existing spoken-LID methods based on prosodic information might fail in the singing-LID task, since the original prosodic structures of spoken language are largely submerged by the overall melody. Therefore, to better handle the singing-LID problem, this study attempts to develop a data-driven method for singing LID, which does not involve the cumbersome task of phone recognition and can be robust against the interference of the background music.

The rest of this paper is organized as follows. The overview of the proposed method is introduced in Section 2. The details of the singing-LID components, including vocal/non-vocal segmentation, language characteristic modeling, and stochastic matching, are presented in Sections 3, 4, and 5, respectively. Finally, the experimental results are discussed in Section 6, and conclusions are drawn in Section 7.

2. METHOD OVERVIEW

A singing-LID system takes as input a test music recording and produces as output the identity of the language sung in that music recording. Since the vast majority of music is a mixture of assorted sound sources, a prerequisite for designing a successful singing-LID system is to extract, model, and compare the characteristic features of language acoustics without interference from non-language features. Toward this end, a singing-LID process as shown in Figure 1 is proposed. It consists of two phases: training and testing.

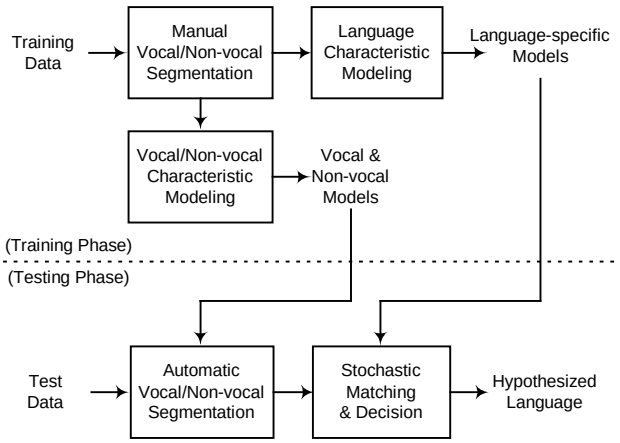


Figure 1. Illustration of the singing-LID process.

In the training phase, a music database containing all the languages of interest sung by plenty of singers must be acquired beforehand. The database is used to establish a characteristic representation of individual languages. Since singing language is irrelevant to accompaniment, the training procedure begins with a segmentation of each music recording into vocal and non-vocal regions, where a vocal region consists of concurrent singing and accompaniment, whereas non-vocal regions consist of accompaniment only. In our implementation, the vocal/non-vocal segmentation of the training data is performed manually. Then, the acoustic characteristics of the vocal and non-vocal regions are stochastically modeled in order to automate the segmentation procedure in the testing phase. On the other hand, a stochastic modeling technique is performed in an attempt

to extract the underlying characteristics of singing language in the vocal segments by specifically suppressing the characteristics of the background. After that, each language is represented by a language-specific parametric model.

During testing, the vocal and non-vocal segments of an unknown music recording are automatically located and marked as such. The vocal segments are then examined using each of the language-specific parametric models. Finally, the language of the model deemed best matching the observed vocal segments is taken as the language of that test recording.

3. VOCAL/NON-VOCAL CHARACTERISTIC MODELING AND SEGMENTATION

The basic strategy applied here follows our previous work [18], in which a stochastic classifier is constructed for distinguishing vocal from non-vocal regions. This classifier consists of a front-end signal processor that converts digital waveforms to spectrum-based feature vectors, e.g., cepstral coefficients, followed by a backend statistical processor that performs modeling and matching.

In modeling the acoustic characteristics of the vocal and non-vocal classes, a set of Gaussian mixture models (GMMs) is used. For each of the languages of interest, a GMM is created using the feature vectors of the manually-segmented vocal parts of music data sung in that language. Thus, L vocal GMMs Λ_1, Λ_2 ,…, Λ_L are formed for L languages. On the other hand, a non-vocal GMM Λ_N is created using the feature vectors of all the manually-segmented non-vocal parts of music data. Parameters of the GMMs are initialized via k-means clustering and iteratively adjusted via expectation-maximization (EM) [3]. When an unknown music recording is present, the classifier takes as input the T-length feature vectors $\mathbf{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_T\}$ extracted from that recording, and produces as outputs the frame likelihoods $p(\boldsymbol{x}_t|\Lambda_N)$ and $p(\boldsymbol{x}_t|\Lambda_l)$, $1 \le l \le L$, $1 \le t \le T$. Since singing tends to be continuous, classification can be made in a segment-by-segment manner. Specifically, a W-length segment is hypothesized as either vocal or non-vocal using

$$\max_{1\le l\le L}\left(\sum_{i=1}^{W}\log p(\boldsymbol{x}_{kW+i}\mid\Lambda_l)\right) \underset{\text{non-vocal}}{\overset{\text{vocal}}{\gtrless}} \sum_{i=1}^{W}\log p(\boldsymbol{x}_{kW+i}\mid\Lambda_N), \quad (1)$$

where k is the segment index.

4. LANGUAGE CHARACTERISTIC MODELING

This section presents a stochastic method for representing the characteristics of singing languages. The method can be implemented without involving complicated linguistic rules and pre-prepared phonetic transcriptions.

4.1. Vector Tokenization Followed by Grammatical Modeling

Our basic idea is to explore the phonotactics-related information of individual languages by examining the statistical dependencies of sound events present in a singing signal. In contrast to the conventional phonotactic modeling approach, which relies on phone recognition as a front-end operation, we use an unsupervised classification method to derive the basic phonological units inherently in a singing process. This allows us to circumvent the cumbersome task of segmenting singing into linguistically meaningful elements

Given a set of training data consisting of spectrum-based feature vectors computed from the manually-segmented vocal parts of music, the procedure for language characteristic modeling comprises two stages as shown in Figure 2. In the first stage, vector clustering is employed on all feature vectors pertaining to a particular language, and a language-specific codebook, consisting of several codewords for characterizing the individual clusters, is formed. Each of the feature vectors is then assigned a codeword index of its associated cluster. It is assumed that each of the clusters represents a certain vocal tract configuration corresponding to a fragment of a broad phonetic class, such as vowels, fricatives, or nasals. The concatenation of different codeword indices in a singing signal may follow some language-specific rules resembling phonotactics, and hence the characteristics of the singing languages may be extracted by analyzing the generated codeword index sequences.

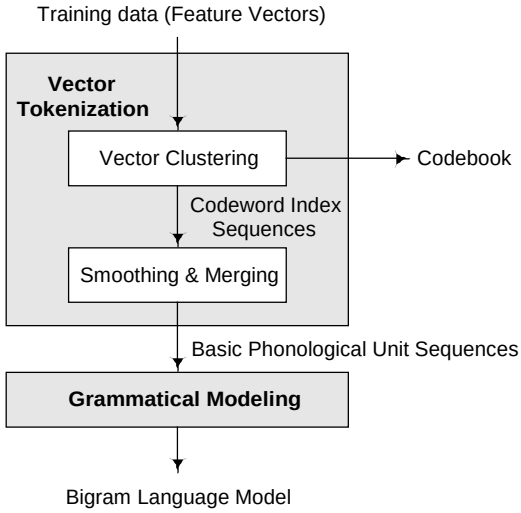


Figure 2. Language characteristic modeling.

To reflect the fact that a vocal tract configuration cannot change suddenly, the generated codeword index sequences are smoothed in the time domain. For smoothing, an index sequence is first divided into a series of consecutive, non-overlapping, fixed-length segments, and each segment is assigned the majority

index of its constituent vectors. After that, adjacent segments are further merged as a homogeneous segment if they have the same codeword index. Each homogeneous segment is regarded as a basic phonological unit. Accordingly, a vocal part of music is tokenized into a sequence of basic phonological units.

In the second stage, a grammatical model is used to characterize the dynamics of the generated basic phonological unit sequences. There are many choices to do this. In our implementation, bigram language models [9] are used. Parameters of a bigram language model, consisting of interpolated bigram probabilities, are estimated using the relative frequency method:

$$p(w_t = j \mid w_{t-1} = i) = \alpha \frac{n_{ij}}{\sum_{k=1}^{K} n_{ik}} + (1-\alpha)\frac{n_i}{\sum_{k=1}^{K} n_k}, \quad (2)$$

where w_t and w_{t-1} denote two successive basic phonological units, α is an interpolating factor subject to $0 \le \alpha \le 1$, K is the codebook size, n_i is the number of basic phonological units assigned as codeword i, and n_{ij} is the number of two successive basic phonological units assigned as codewords i and j, respectively. Note that the transition between two separate vocal regions in a music recording is not taken into account in the computation of bigram probabilities. In summary, a language-specific model consists of a codebook and a bigram language model.

4.2. Solo Voice Codebook Generation

The effectiveness of the language characteristic modeling described above crucially depends on whether the vector tokenization truly relates to the notion of phonology. Since the vast majority of popular music contains background accompaniment during most or all vocal passages, directly using conventional vector clustering methods, such as *k*-means algorithm on the accompanied singing signals, may cause that the generated clusters are not only related to the vocal tract configurations, but also to the instrumental types. To alleviate this problem, we develop a codebook generation method for vector clustering based on an estimation of the stochastic characteristics of the underlying solo voices from accompanied singing signals.

Let $\mathbf{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_T\}$ denote all the feature vectors computed from the vocal regions of music recordings. Due to the existence of accompaniment, $\mathbf{X}$ can be considered as a mixture of a solo voice $\mathbf{S} = \{\boldsymbol{s}_1, \boldsymbol{s}_2, ..., \boldsymbol{s}_T\}$ and a background music $\mathbf{B} = \{\boldsymbol{b}_1, \boldsymbol{b}_2, ..., \boldsymbol{b}_T\}$. More specifically, $\mathbf{S}$ and $\mathbf{B}$ are added in the time domain or linear spectrum domain, but both of them are unobservable. Our aim is to create a codebook for representing the generic characteristics of the solo voice signal $\mathbf{S}$, such that the vector tokenization can be performed on the basis of this codebook. Under the vector clustering framework, we assume that the solo voice signal and background music are, respectively, characterized by two independent codebooks $\mathbf{C}_s = \{\boldsymbol{c}_{s,1}, \boldsymbol{c}_{s,2}, \ldots, \boldsymbol{c}_{s,K_s}\}$ and $\mathbf{C}_b = \{\boldsymbol{c}_{b,1}, \boldsymbol{c}_{b,2}, \ldots, \boldsymbol{c}_{b,K_b}\}$, where $\boldsymbol{c}_{s,i}$, $1 \le i \le K_s$, and $\boldsymbol{c}_{b,j}$, $1 \le j \le K_b$, are the codewords. To better represent the acoustic feature space, each cluster is modeled by a Gaussian density function. Therefore, a codeword consists of a mean vector and a covariance matrix, i.e., $\boldsymbol{c}_{s,i} = \{\boldsymbol{\mu}_{s,i}, \boldsymbol{\Sigma}_{s,i}\}$ and $\boldsymbol{c}_{b,j} = \{\boldsymbol{\mu}_{b,j}, \boldsymbol{\Sigma}_{b,j}\}$, where $\boldsymbol{\mu}_{s,i}$ and $\boldsymbol{\mu}_{b,j}$ are mean vectors, and $\boldsymbol{\Sigma}_{s,i}$ and $\boldsymbol{\Sigma}_{b,j}$ are covariance matrices. The vector clustering can be formulated as a problem of how to best represent $\mathbf{X}$ by choosing and combining the codewords from $\mathbf{C}_s$ and $\mathbf{C}_b$. To measure how well the vector clustering is performed, we compute the following conditional probability:

$$p(\mathbf{X} \mid \mathbf{C}_s, \mathbf{C}_b) = \prod_{t=1}^{T} \left\{ \max_{i,j} p(\boldsymbol{x}_t \mid \boldsymbol{c}_{s,i}, \boldsymbol{c}_{b,j}) \right\}, \quad (3)$$

where $p(\boldsymbol{x}_t \mid \boldsymbol{c}_{s,i}, \boldsymbol{c}_{b,j})$ accounts for one of the possible combination of the solo voice and background music which can form an instant accompanied voice $\boldsymbol{x}_t$. If the accompanied signal is formed from a generative function $\boldsymbol{x}_t = f(\boldsymbol{s}_t, \boldsymbol{b}_t)$, $1 \le t \le T$, the probability $p(\boldsymbol{x}_t \mid \boldsymbol{c}_{s,i}, \boldsymbol{c}_{b,j})$ can be computed by

$$p(\boldsymbol{x}_t \mid \boldsymbol{c}_{s,i}, \boldsymbol{c}_{b,j}) = \iint_{\boldsymbol{x}_t = f(\boldsymbol{s}_t, \boldsymbol{b}_t)} G(\boldsymbol{s}; \boldsymbol{\mu}_{s,i}, \boldsymbol{\Sigma}_{s,i})\, G(\boldsymbol{b}; \boldsymbol{\mu}_{b,j}, \boldsymbol{\Sigma}_{b,j})\, d\boldsymbol{s}\, d\boldsymbol{b}, \quad (4)$$

where $G(\cdot)$ denotes a multi-variant Gaussian density function. In using such a measurement, vector clustering is considered as effective if the probability $p(\mathbf{X}|\mathbf{C}_s,\mathbf{C}_b)$ can be as large as possible.

In most popular music, substantial similarities exist between the non-vocal regions and the accompaniment of the vocal regions. Therefore, although the background music $\mathbf{B}$ is unobservable, its stochastic characteristics may be approximated from the non-vocal regions. This assumption enables us to estimate the background music codebook $\mathbf{C}_b$ directly, using the *k*-means clustering algorithm on the feature vectors from the non-vocal regions. Accordingly, from the available background music codebook $\mathbf{C}_b$ and the observable accompanied voice $\mathbf{X}$, it is sufficient to derive the solo voice codebook $\mathbf{C}_s$ via a maximum likelihood estimation as follows:

$$\mathbf{C}_s^* = \arg\max_{\mathbf{C}_s} p(\mathbf{X} \mid \mathbf{C}_s, \mathbf{C}_b). \quad (5)$$

Equation (5) can be solved using the EM algorithm, which starts with an initial codebook $\mathbf{C}_s$ and iteratively estimates a new codebook $\hat{\mathbf{C}}_s$ such that $p(\mathbf{X}|\hat{\mathbf{C}}_s,\mathbf{C}_b) \ge p(\mathbf{X}|\mathbf{C}_s,\mathbf{C}_b)$. It can be shown that the need of increasing the probability $p(\mathbf{X}|\hat{\mathbf{C}}_s,\mathbf{C}_b)$ can be satisfied by maximizing the auxiliary function

$$Q(\mathbf{C}_s, \hat{\mathbf{C}}_s) = \sum_{t=1}^{T} \sum_{i=1}^{K_s} \sum_{j=1}^{K_b} \delta(i = i^*, j = j^*) \log p(\boldsymbol{x}_t \mid \boldsymbol{c}_{s,i}, \boldsymbol{c}_{b,j}), \quad (6)$$

where $\delta(\cdot)$ denotes a Kronecker delta function, and

$$(i^*, j^*) = \arg\max_{i,j} p(\boldsymbol{x}_t \mid \boldsymbol{c}_{s,i}, \boldsymbol{c}_{b,j}). \quad (7)$$

Letting $\nabla Q_{(\mathbf{C}_s, \hat{\mathbf{C}}_s)} = 0$ with respect to each parameter to be re-estimated, we have

$$\hat{\boldsymbol{\mu}}_{s,i} = \frac{\sum_{t=1}^{T}\sum_{j=1}^{N} \delta(i=i^*, j=j^*)\, p(\boldsymbol{x}_t \mid \boldsymbol{c}_{s,i}, \boldsymbol{c}_{b,j}) \cdot E\{\boldsymbol{s}_t \mid \boldsymbol{x}_t, \boldsymbol{c}_{s,i}, \boldsymbol{c}_{b,j}\}}{\sum_{t=1}^{T}\sum_{j=1}^{N} \delta(i=i^*, j=j^*)\, p(\boldsymbol{x}_t \mid \boldsymbol{c}_{s,i}, \boldsymbol{c}_{b,j})}, \tag{8}$$

$$\hat{\boldsymbol{\Sigma}}_{s,i} = \frac{\sum_{t=1}^{T}\sum_{j=1}^{J} \delta(i=i^*, j=j^*)\, p(\boldsymbol{x}_t \mid \boldsymbol{c}_{s,i}, \boldsymbol{c}_{b,j}) \cdot E\{\boldsymbol{s}_t \boldsymbol{s}_t' \mid \boldsymbol{x}_t, \boldsymbol{c}_{s,i}, \boldsymbol{c}_{b,j}\}}{\sum_{t=1}^{T}\sum_{j=1}^{J} \delta(i=i^*, j=j^*)\, p(\boldsymbol{x}_t \mid \boldsymbol{c}_{s,i}, \boldsymbol{c}_{b,j})} - \boldsymbol{\mu}_{s,i}\boldsymbol{\mu}_{s,i}', \tag{9}$$

where the prime operator ($'$) denotes vector transpose, and $E\{\cdot\}$ denotes expectation. The details of the Equations (8) and (9) required for implementation can be found in [13,17,18]. Figure 3 summarizes the procedure for the solo voice codebook generation.

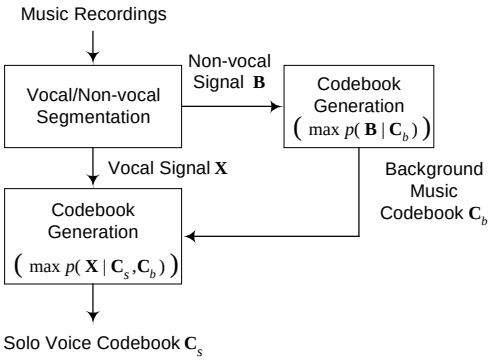


Figure 3. Procedure for a solo voice codebook generation.

5. STOCHASTIC MATCHING AND DECISION

This section is concerned with the testing phase of the proposed singing-LID system. As shown in Figure 4, a test music recording is first segmented into vocal and non-vocal regions, and the feature vectors from the non-vocal regions are used to form a codebook $\mathbf{C}_b$, which simulates the characteristics of the background accompaniment in the vocal regions. For each of the L candidate languages, the associated solo voice codebook $\mathbf{C}_{s,l}$, $1 \le l \le L$, along with the background music codebook $\mathbf{C}_b$, are used to tokenize the feature vectors of the vocal regions $\{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_T\}$ into a codeword index sequence $V^{(l)} = \{v_1^{(l)}, v_2^{(l)}, \ldots, v_T^{(l)}\}$, where T is the total length of the vocal regions, and $v_t^{(l)}$, $1 \le t \le T$, is determined by

$$v_t^{(l)} = \arg_i \left[\max_{i,j}\, p(x_t \mid \boldsymbol{c}_{s,i}^{(l)}, \boldsymbol{c}_{b,j}) \right]. \tag{10}$$

Each of the codeword index sequences $V^{(l)}$, $1 \le l \le L$, is then converted into a basic phonological unit sequence $W^{(l)} = \{w_1^{(l)}, w_2^{(l)}, \ldots, w_{N^{(l)}}^{(l)}\}$ by smoothing and merging the adjacent identical indices.

For each language l, the dynamics of the basic phonological unit sequence $W^{(l)}$ are examined using a bigram language model $\lambda^{(l)}$, in which a log-likelihood $\log p(W^{(l)}|\lambda^{(l)})$, that $W^{(l)}$ tests against $\lambda^{(l)}$, is computed using

$$\log p(W^{(l)} \mid \lambda^{(l)}) = \frac{1}{N^{(l)}} \sum_{t=1}^{N^{(l)}} \log p(w_t^{(l)} \mid w_{t-1}^{(l)}). \tag{11}$$

Note again that the transitions between vocal regions are not taken into account when computing Equation (11). According to the maximum likelihood decision rule, the identifier should decide in favor of a language satisfying

$$l^* = \arg\max_l \log p(W^{(l)} \mid \lambda^{(l)}). \tag{12}$$

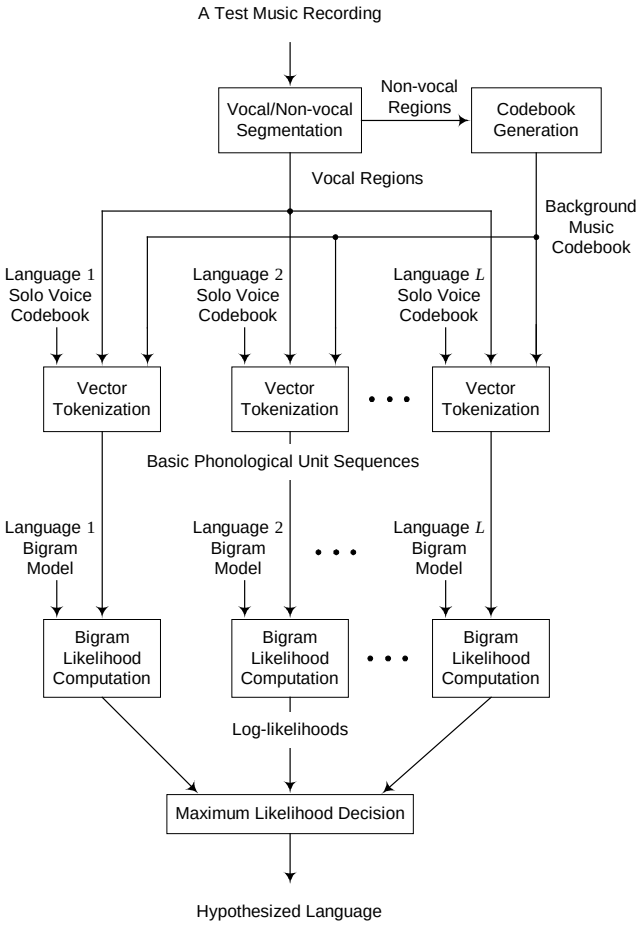


Figure 4. Procedure for hypothesizing the language of an unknown test music recording.

6. EXPERIMENTS

To test the validity of the proposed singing-LID method, computer simulations must be conducted with music data covering various languages, music styles, singers, and so on. However, during the initial development stage, the performance of our singing-LID system was only

evaluated using the task of distinguishing between English and Mandarin pop songs, due to the difficulty of collecting and annotating multilingual music data. In our experiments, emphasis was put on examining if the characteristics of individual languages can be extracted from the singing in a music recording.

6.1 Music Database

Our music database consisted of 224 tracks (112 per language) from pop music CDs. The average length of tracks was around three minutes. All the tracks were manually labeled with the language identity and the vocal/non-vocal boundaries. Among the 224 tracks, there were 32 pairs of tracks involving cover/original versions of songs, each pair of which contained two same-tune songs, one in English and one in Mandarin. These 32 pairs of tracks, denoted as a subset DB-C, were used for evaluating the performance of the proposed singing-LID system. Genders of the singers in DB-C were almost balanced, and 15 out of 32 pairs of tracks were performed by 15 bilingual singers, i.e., each of the 15 singers performed two same-tune songs, one in English and one in Mandarin. As DB-C was composed, we attempted to avoid the bias arising from tunes, singers, or music styles which may affect the objectivity of assessing a singing-LID system.

Aside from DB-C, the remaining 160 tracks (80 per language) in our database were mutually distinct in terms of tunes, lyrics, and singers. These 160 tracks were further divided into two subsets, respectively, denoted as DB-T and DB-R. The DB-T, containing 60 tracks per language, was used as training data for creating vocal/non-vocal models, language-specific codebooks, and bigram language models, while the DB-R, containing the rest 20 tracks per language, was used as another testing data besides DB-C. None of the singers in one of the three subsets appeared in another. All music data were down-sampled from the CD sampling rate of 44.1 kHz to 22.05 kHz, to exclude the high frequency components beyond the range of normal singing voices. Feature vectors, each consisting of 20 Mel-scale frequency cepstral coefficients, were computed using a 32-ms Hamming-windowed frame with 10-ms frame shifts.

6.2 Experimental Results

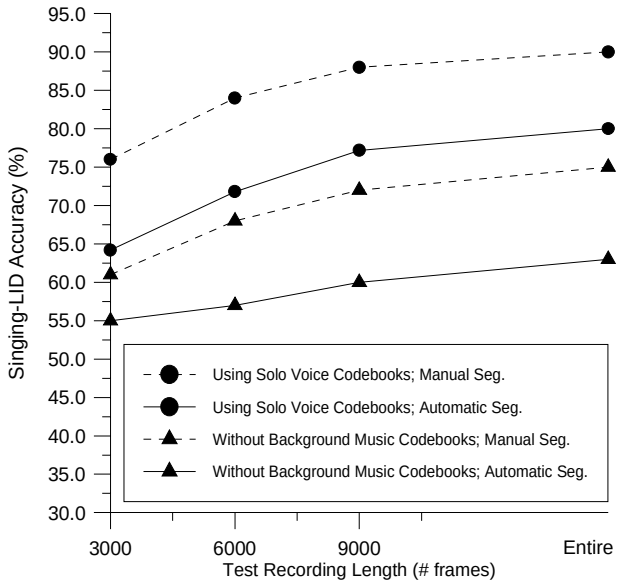
Our experiments began with an evaluation for the vocal/non-vocal segmentation of the music data in DB-C and DB-R. Segmentation performance was characterized by a frame-based accuracy computed as the percentage of correctly-hypothesized frames over the total number of test frames. In view of the limited precision with which the human ear detects vocal/non-vocal changes, all frames that occurred within 0.5 seconds of a perceived switch-point were ignored in the accuracy computation. Using 64 mixture components per GMM along with 60-frame analysis segments (empirically the most accurate configurations), the segmentation accuracies resulted on DB-R and DB-C were 78.1% and 79.8%, respectively.

Then, the singing-LID performance was evaluated with respect to different length of test recording. Each of the tracks in DB-R and DB-C was divided into several overlapping clips of T feature vectors. A 10-sec clip corresponds to 1000 feature vectors, and the overlap of two consecutive clips was 500 feature vectors. Each clip was treated as a separate music recording. The singing-LID experiments were conducted in a clip-by-clip manner, and the singing-LID accuracy was computed as the percentage of correctly-identified clips over the total number of test clips. In the training phase, the number of codewords used in each language-specific solo voice codebook and the background music codebook were empirically determined to be 32 and 16, respectively. In the testing phase, an online-created background music codebook was empirically set to have 4 codewords, if the number of the non-vocal frames exceeds 200; otherwise, no background music codebook was used. The segment length for smoothing the generated codeword index sequences was empirically set to be 5, and the interpolating factor α in Equation (2) was set to be 0.1. For performance comparison, we also performed singing LID without using any background music codebook.

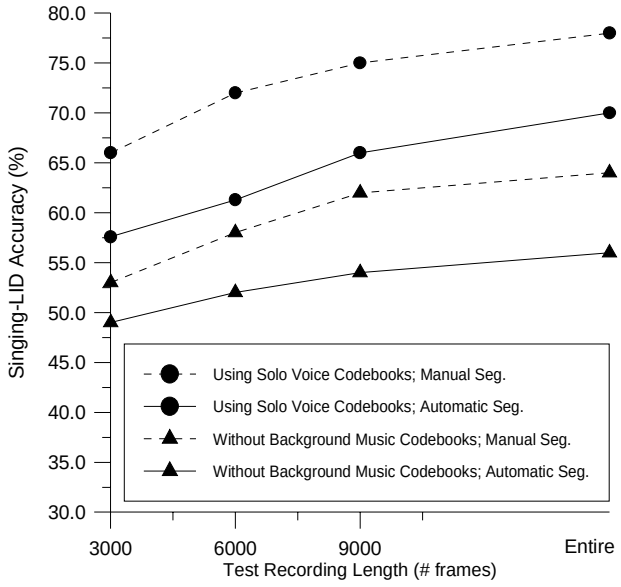
Figure 5 shows the singing-LID results with respect to T = 3000 (30 sec), 6000 (60 sec), 9000 (90 sec), and entire track, in which the T-length clips that fully labeled as non-vocal were not used for testing. Here, the singing-LID performance achieved with the manual vocal/non-vocal segmentation may serve as an upper bound for that obtained using automatic segmentation. We can see that as expected, the accuracy gains as the clip length increases. It is also clear that the performance of the singing LID based on the usage of solo voice codebooks is noticeably better than that of the method without taking background music into account. Such a performance gap is particularly visible when the test music recordings are long, mainly because more information about the background music can be exploited for assisting the construction of a reliable solo voice codebook. Using the automatic vocal/non-vocal segmentation, the best singing-LID accuracies of 80.0% and 70.0% were achieved when testing the entire tracks in DB-R and DB-C, respectively. The results indicate that the task of identifying the languages of the songs made originally in another language is more difficult.

Table 1 shows the confusion probability matrix from the best results of the singing LID based on the automatic vocal/non-vocal segmentation. The rows of the matrix correspond to the ground-truth of the tracks while the columns indicate the hypotheses. It can be found that the majority of errors are misidentifications of English songs. We speculate that such errors might be attributed to the louder background music usually existing in English pop songs, compared to Mandarin music, which often mix vocals louder to ensure that Mandarin syllables can be heard and understood semantically with the lack of tone information. The lower vocal-to-background ratio may cause the English model to be relatively ill-generated, and therefore, to poorly match the associated test music

recording. Another reason for the bias towards Mandarin in identifying the tracks in DB-C is likely because a large proportion of the singers in DB-C are Chinese. The accents of those Chinese singers might be different significantly from those of the singers in DB-T, who are mainly American, and hence the resulting discrepancy in phonological realizations may also lead the English model to match the test music recording poorly. One way to solve such problems is to use a wider variety of music data for training language-specific models, but this is not yet investigated at the initial stage of this study.



(a) Experiments on DB-R



(b) Experiments on DB-C

Figure 5. Singing-LID results.

The above experimental results indicate that though the singing-ID performance achieved with our proposed method still leaves much room for improving, a successful automatic singing-LID system should be feasible based on some possible extensions of the framework developed in this study.

	Hypothesized	
Actual	English	Mandarin
English	0.75	0.25
Mandarin	0.15	0.85

(a) Experiments on DB-R

	Hypothesized	
Actual	English	Mandarin
English	0.63	0.37
Mandarin	0.22	0.78

(b) Experiments on DB-C

Table 1. Confusion probability matrix of the discrimination of Mandarin and English songs.

7. CONCLUSIONS

This study has examined the feasibility of automatically identifying the singing language in a popular music recording. It has been shown that the acoustic characteristics of a language can be extracted from singing signals via grammatical modeling of the basic phonological unit sequences output from the vector tokenization of spectrum-based features. To eliminate the interference of background music, we have proposed a reliable codebook generation method for vector clustering based on an estimation of the solo voice characteristics.

Though this study showed that the language sung in a music recording could be distinguished from one another, the proposed method and the conducted experiments can only be regarded as a very preliminary investigation in the singing-LID problem. To further explore this problem, the essential work is to scale up the music database, which covers a large number of languages, singers with a wider variety of accents, and rich music styles or genres.

8. ACKNOWLEDGEMENT

This work was partially supported by National Science Council, Taiwan, under Grants NSC92-2422-H-001-093 and NSC93-2422-H-001-0004.

9. REFERENCES

[1] Akeroyd, M. A., Moore, B. C. J., and Moore, G. A. "Melody recognition using three types of

dichotic-pitch stimulus", *Journal of the Acoustical Society of America*, 110(3): 1498-1504, 2001.

[2] Cummins, F., Gers, F., and Schmidhuber, J. "Language identification from prosody without explicit features", *Proceedings of the European Conference on Speech Communication and Technology*, Budapest, Hungary, 1999.

[3] Dempster, A., Laird, N., and Rubin, D. "Maximum likelihood from incomplete data via the EM algorithm", *Journal of the Royal Statistical Society*, 39: 1-38, 1977.

[4] DuPreez, J. A., and Weber, D. M. "Language identification incorporating lexical information", *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, Phoenix, USA, 1999.

[5] Eronen, A. "Musical instrument recognition using ICA-based transform of features and discriminatively trained HMMs," *Proceedings of the International Symposium on Signal Processing and Its Applications*, Paris, France, 2003.

[6] Harbeck, S., and Ohler, U. "Multigrams for language identification", *Proceedings of the European Conference on Speech Communication and Technology*, Budapest, Hungary, 1999.

[7] Hazen, T. J. and Zue, V. W. "Segment-based automatic language identification", *Journal of the Acoustical Society of America*, 101(4): 2323-2331, 1997.

[8] House, A. S. and Neuburg, E. P. "Toward automatic identification of the language of an utterance. I. Preliminary methodological considerations", *Journal of the Acoustical Society of America*, 62(3): 708-713, 1977.

[9] Jelinek, F. "Self-organized language modeling for speech recognition", *Readings in Speech Recognition*, Palo Alto, CA: Morgan Kaufmann, 1990, chap 8, pp. 450-506.

[10] Kim, Y. E. and Whitman, B. "Singer identification in popular music recordings using voice coding features", *Proceedings of the International Conference on Music Information Retrieval*, Paris, France, 2002.

[11] Medina, R. A., Smith, L. A., and Wagner, D. R. "Content-based indexing of musical scores", *Proceedings of the Joint Conference on Digital Libraries*, Texas, USA, 2003.

[12] Muthusamy, Y. K., Barnard, E., and Cole, R. A. "Reviewing automatic language identification", *IEEE Signal Processing Magazine*, 4: 33-41, 1994.

[13] Nadas, A., Nahamoo, D., and Picheny, M. A., "Speech recognition using noise-adaptive prototypes", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(10): 1495-1503, 1989.

[14] Nakagawa, S., Ueda, Y., and Seino. "Speaker-independent, text-independent language identification by HMM", *Proceedings of the International Conference on Spoken Language Processing*, Alberta, Canada, 1992.

[15] Navratil, J. and Zuhlke, W. "An efficient phonotactic-acoustic system for language identification", *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, Seattle, USA, 1998.

[16] Navratil, J. "Spoken language recognition – A step toward multilinguality in speech processing", *IEEE Transactions on Speech and Audio Processing*, 9(6): 678-685, 2001.

[17] Rose, R. C., Hofstetter, E. M., and Reynolds, D. A. "Integrated models of signal and background with application to speaker identification in noise", *IEEE Transactions on Speech and Audio Processing*, 2(2): 245-257, 1994.

[18] Tsai, W. H., Wang, H. M., Rodgers, D., Cheng, S. S., and Yu, H. M. "Blind clustering of popular music recordings based on singer voice characteristics", *Proceedings of the International Conference on Music Information Retrieval*, Baltimore, USA, 2003.

[19] Tzanetakis, G. and Cook, P. "Musical genre classification of audio signals", *IEEE Transactions on Speech and Audio Processing*, 10(5): 293-302, 2002.

[20] Wang, C. K., Lyu, R. Y., and Chiang, Y. C. "An automatic singing transcription system with multilingual singing lyric recognizer and robust melody tracker", *Proceedings of the European Conference on Speech Communication and Technology*, Geneva, Switzerland, 2003.

[21] Zissman, M. A. "Comparison of four approaches to automatic language identification of telephone speech", *IEEE Transactions on Speech and Audio Processing*, 4(1): 31-44, 1995.