On the Extraction of Vocal-related Information to Facilitate the Management of Popular Music Collections

Wei-Ho Tsai
Institute of Information Science, Academia Sinica
Taipei, Taiwan, Republic of China
+886-2-27883799 ext. 2403
wesley@iis.sinica.edu.tw

Hsin-Min Wang
Institute of Information Science, Academia Sinica
Taipei, Taiwan, Republic of China
+886-2-27883799 ext. 1714
whm@iis.sinica.edu.tw

ABSTRACT
With the explosive growth of networked collections of musical material, there is a need to establish a mechanism like a digital library to manage music data. This paper presents a content-based processing paradigm of popular song collections to facilitate the realization of a music digital library. The paradigm is built on the automatic extraction of information of interest from music audio signals. Because the vocal part is often the heart of a popular song, we focus on developing techniques to exploit the solo vocal signals underlying an accompanied performance. This supports the necessary functions of a music digital library, namely, music data organization, music information retrieval/recommendation, and copyright protection.

Categories and Subject Descriptors
H.2.8 [**Database Management**]: Database Applications – *Data mining*; H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing – *Indexing methods*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *Clustering*, *Relevance feedback*, *Retrieval models*; H.3.7 [**Information Storage and Retrieval**]: Digital Libraries –*Systems issues.*

General Terms
Algorithms, Management, Measurement, Documentation, Design, Experimentation, Legal Aspects, Verification.

Keywords
Music Digital Library, Music Information Retrieval, Vocal/Non-vocal Segmentation, Solo Voice Modeling, Query-by-example.

1. INTRODUCTION
With the ever-increasing capabilities of data storage and transmission, and thanks to the recent advances in various digital signal processing technologies, music material has enjoyed an unprecedented growth in terms of production and distribution of late years. Various types of audio formats, coupled with media players, software, and channels have significantly changed the way people listen to music, and thereby made music ubiquitous. However, with the rapid proliferation of musical material comes the ironic dilemma of how can we deal with it: how to locate the desired music from the innumerable options and how to ensure only those that are authorized can access them. This dilemma has motivated recent research [2,3,11,18] into trying to establish a mechanism like a digital library to manage the burgeoning number of music collections, and, if possible, make music as accessible as text that has long been a well-regulated archival resource.

Constructing a music digital library requires the integration of comprehensive technologies and disciplines concerned with library science, computer science, audio engineering, musicology, law, and so on. Although each discipline concentrates on its own set of requirements, it is expected that a music digital library, as far as the functionality is concerned, must be capable of the following operations:

- **Music data organization**, which indexes music data readily available in electronic form for the subsequent preservation, access, research, and other uses.
- **Music information retrieval/recommendation**, which allows users to acquire music documents or relevant objects according to their requests submitted to the system (retrieval) or the suggestions made by the system (recommendation).
- **Copyright protection**, which safeguards the copyrighted music works against unauthorized access, manipulation, or distribution.

Conventionally, music in record shops is cataloged by title, composer, producer, artist, and other objective or subjective classes to allow customers to quickly find the items they want. Such a concept has also existed in most current music encoded in digital formats, in which some descriptive information is delivered together with the actual audio content. Examples of this so-called metadata are the widely used ID3 tags attached to MP3 bitstreams [8] and the forthcoming MPEG-7 standard [12]. From the viewpoint of the traditional bibliographic text processing, it might be rather straightforward for a digital library to manage the music collections through the use of metadata. However, as music presented in audio form encompasses numerous levels of information, ranging from concrete to abstract, the metadata-based approach may not be effective, if the information we want is not indicated in the metadata. Without the relevant descriptive tag attached, it is difficult to browse music data as textual data, because listening to an audio recording takes much more time than reading a text. Moreover, generating metadata for a vast

JCDL'05, June 7–11, 2005, Denver, Colorado, USA.

music database is often associated with tremendous efforts, since it inevitably requires human intervention. On the other hand, metadata can easily to be removed or manipulated, which might be taken advantage of by copyright violators to prevent their piracy from being detected. Viewed in this manner, designing a music digital library cannot rely solely on the common metadata-based mechanism, but requires the use of automatic techniques to analyze the content of music data.

This paper presents a paradigm of content-based processing of music data to facilitate the construction of a music digital library. Our goal is to develop the necessary techniques for automatically extracting information of interest from music audio signals, thereby realizing the above-mentioned basic operations in a music digital library. Relatedly, there has been a copious amount of research working toward a similar goal, such as melody extraction [1,6], instrument recognition [7,10], genre classification [14,26], artist or singer identification [13,15,30], song identification [9,25,28], mood classification [16,31], and lyric transcription [29]. Among these studies, of particular interest and challenge is the problem of probing information residing in vocals, i.e., singing over music. In popular music, the vocal part is often the heart of a song, carrying the melodic hook, artistic style, lyrics, and so on. Extracting vocal-related information is important, but is made notably difficult by the fact that the desired singing signals are inextricably intertwined with the non-stationary signals of background accompaniment. So far, a vast majority of works have almost either bypassed the problem of how to eliminate the background music, or simply dealt with this problem from a fault-tolerance standpoint. Methods that exclusively exploit the underlying solo vocal signals have been rarely explored. Recently, we studied a stochastic modeling technique for representing the acoustic characteristics of a singer's voices in an accompanied recording [24]. This paper extends our stochastic modeling technique to a general framework for extracting vocal-related information from popular music data. As an example in using this framework, we show how a set of undocumented music data can be classified, retrieved, and verified on the basis of the singers' voice characteristics derived from accompanied vocal signals.

The rest of this paper is organized as follows. Section 2 describes an overview of the methods for managing a popular music collection by extracting the desired vocal-related information. Section 3 presents a statistical classifier for distinguishing between vocal segments and accompaniments, which is a first step in extracting the vocal-related information. In Section 4, we introduce a stochastic method for distilling the desired solo voice characteristics from the vocal regions of music recordings. Next, examples of realizing music data organization, retrieval, and copyright protection are discussed in detail in Sections 5, 6, 7 and 8, respectively, by supervised and unsupervised singer classification, singer-based song retrieval, and singer verification. Finally, in Section 9, we present our conclusions.

2. TECHNICAL OVERVIEW

This section provides an overview of the techniques for managing a popular music collection by extracting the desired vocal-related information. Although the vocal-related information discussed in detail here is concerned with the singers' voice characteristics, the techniques should be applicable to a wide variety of other vocal-related information, such as the melody or singing language of a song, without loss of generality.

2.1 Music Data Organization

Automatically classifying music data is the most efficient way to organize a vast undocumented music collection. Like the common classification problem, the methods for classifying music data can be divided into two categories: supervised classification and unsupervised classification. The major difference between these two categories is that the former requires training samples with a ground truth of each predefined class, while the latter works without prior knowledge about what and how many classes are involved in the data. This study presents an example framework for the supervised and unsupervised classification of unlabeled music recordings based on their associated singers (hereafter referred to as supervised and unsupervised singer classification).

In the supervised singer classification, the system takes as input an unknown music recording and produces as output about the most likely singer involved in this recording[1]. This technique can be of great use for those wishing to identify the singer in a live concert recording, which typically lacks metadata when it was made. In addition, singer classification can be used to distinguish among the individual singers of a rock music band, in which the band's songs are usually sung by not only a lead singer, but also a guitarist, a drummer, or other band-members.

As shown in Figure 1, a supervised singer classification system operates in two phases: training and testing. In the training phase, music data from each of the singers concerned is collected beforehand. This data is segmented into vocal and non-vocal regions, where a vocal region consists of concurrent singing and accompaniment, whereas non-vocal regions consist of accompaniment only. Then, the voice characteristics of each singer are statistically modeled from the segmented vocal regions. Since most of the vocal regions in a popular music song contain background music, the interference of this must be eliminated or suppressed to better model a singer's voice characteristics. For this purpose, we have developed a solo voice modeling technique by estimating the characteristics of the background music from the non-vocal segments. As a result of the solo voice modeling, each of the candidate singers is represented by a model. During a test, the classifier decides in favor of a singer for the test recording by choosing which of the models best matches the recording, i.e., the maximum likelihood decision rule. The details of the vocal/non-vocal segmentation, solo voice modeling, and maximum likelihood decision are described in Sections 3, 4, and 5.

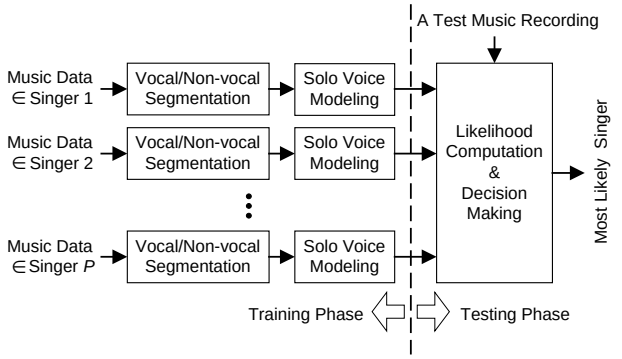


Figure 1. Supervised singer classification for music data organization.

[1] In this work we limit ourselves to single-singer music recordings.

In the unsupervised singer classification, the system partitions a collection of undocumented music recordings into several clusters such that each cluster consists exclusively of recordings from one singer. This technique compensates for the shortcomings of the supervised singer classification, where the inventory of the singers involved in the music collection maybe unknown and hence the voice characteristics of a specific singer cannot be modeled in advance. By using the unsupervised singer classification, the recordings performed by the same singers are grouped together. Therefore, the human efforts required for documentation can be greatly reduced, from having to listen to each recording to only having to listen to a few recordings in each cluster.

Figure 2 shows the framework for unsupervised singer classification. Like the supervised singer classification, each of the music recordings to be clustered is segmented into vocal and non-vocal regions. Solo voice modeling is then applied to the segmented vocal regions. However, in contrast to the supervised system, which creates a solo voice model using all the vocal portions of training data performed by a particular singer, the unsupervised system creates a solo voice model for each of the individual recordings. Here, a solo voice model represents the voice characteristics of a singer involved in a recording. The similarities between music recordings can then be measured by comparing the solo voice models. Finally, the recordings similar enough to each other are grouped into a cluster. The whole process of classification is described in a greater detail in Section 6.

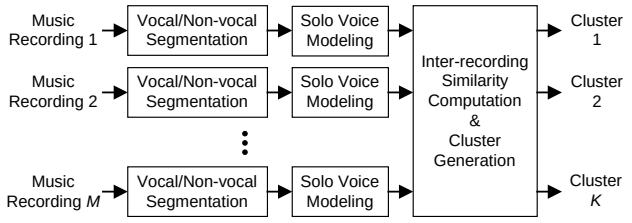


Figure 2. Unsupervised singer classification for music data organization.

2.2 Music Data Retrieval/Recommendation

Music data may be retrieved as straightforwardly as retrieving text documents by using a keyword-based query, if the music data is classified and indexed as a set of documents. However, this is often not the case, because pre-classifying and indexing music data may not be supported in many applications. One application associated with this problem is a distributed peer-to-peer retrieval system [27]. Since the data to be browsed and retrieved is not stored in a centralized unit, pre-classification of music data is considered infeasible. Moreover, the use of a keyword-based query usually requires that users provide an explicit description of the item they are looking for. This usage can be awkward for music data retrieval, since it is sometimes difficult for users to formulate their thoughts on the desired music data as a text query. To avoid these problems, this study investigates a query-by-example framework for retrieving undocumented music data according to an excerpt of audio query submitted by users.

Figure 3 shows a query-by-example system that allows users to locate a specified singer's music recordings from a database via submitting a fragment of music as a query. This system can be of great use to those wishing to listen to a particular artist's songs, but just cannot recall the name of the artist. Such a requirement may also arise when users hear some songs and want to know more about the artists of these songs or want to listen to more songs performed by the same artists. In these cases, a user may query, "Find me all the songs performed by the singer of the attached recording." As with query-by-example, such a technique can also trivially accommodate some features of music recommendation by the system, in which users are encouraged to listen to the songs performed by their favorite singers, or the singers with voices similar to their favorite artists.

To permit that undocumented music data can be retrieved according to the example music query submitted by a user, the system evaluates the similarities between the music query and each of the music recordings to be retrieved. The outcome presented to a user is a ranked list of the relevance of each recording to the query. Here, the greater the similarity between the query and a particular recording, the more relevant the recording is deemed. For the similarity to be connected to the singers' vocal characteristics, each music recording undergoes the above-mentioned vocal/non-vocal segmentation along with the solo voice modeling.

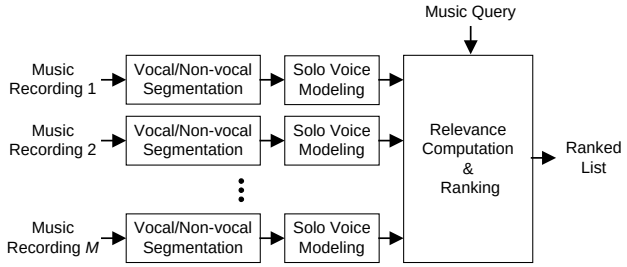


Figure 3. A query-by-example framework for retrieving undocumented music data according to the singer specified by an example music query.

2.3 Copyright Protection

Automatic extraction of vocal-related information can be further used to assist copyright protection of musical works. For example, record companies or copyright holders may want to deploy an automatic system that comprehensively scans any suspect websites and examines if some music files available online violate their copyrights. One promising way of the examination is to recognize if the music files contain the vocals of some singers or artists belonging to their copyrights. Since the examination is performed on the basis of audio content, it is immune from the manipulation of the metadata in the music files.

To make the above idea a reality, we have developed a singer-verification system that determines whether or not an unknown music recording contains a specified singer's voice. Considering an unknown music recording **X**, the singer verification can be formulated as a basic hypothesis test between

H_0: **X** is performed by the specified singer,

and

H_1: **X** is not performed by the specified singer.

The optimum test to decide between these two hypotheses is a likelihood ratio test given by

$$\frac{p(\mathbf{X}\mid H_0)}{p(\mathbf{X}\mid H_1)} \begin{array}{c} H_0 \text{ (Yes)} \\ > \\ \leq \\ H_1 \text{ (No)} \end{array} \delta, \tag{1}$$

where δ is a decision threshold. Figure 4 shows the basic components of a singer-verification system based on the above hypothesis test. Prior to determining if a test music recording is performed by a specified singer, a training process must be carried out to characterize the two hypotheses, H_0 and H_1. To do this, music data, performed and not performed by the hypothesized singer, are collected and used for generating the representative characteristics of H_0 and H_1, respectively. This process is done by the above-mentioned vocal/non-vocal segmentation along with the solo voice modeling.

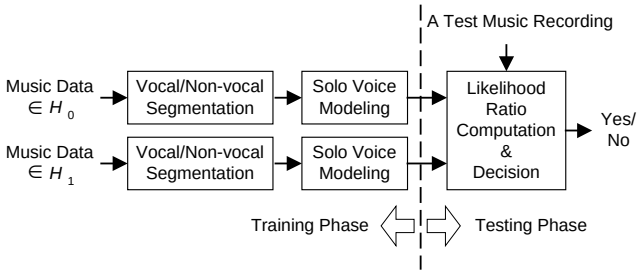


Figure 4. Singer verification for copyright protection.

3. VOCAL/NON-VOCAL SEGMENTATION

As a first step in extracting the vocal-related information, music segments that contain singing voices are located and marked as such. This task can be formulated as a problem of distinguishing between vocal segments and accompaniments. We therefore build a vocal/non-vocal recognizer to solve this problem. The recognizer consists of a front-end signal processor that converts digital waveforms to frame-based cepstral [19] feature vectors, followed by a backend statistical processor that performs modeling and matching. It operates in two phases, training and testing, as shown in Figure 5.

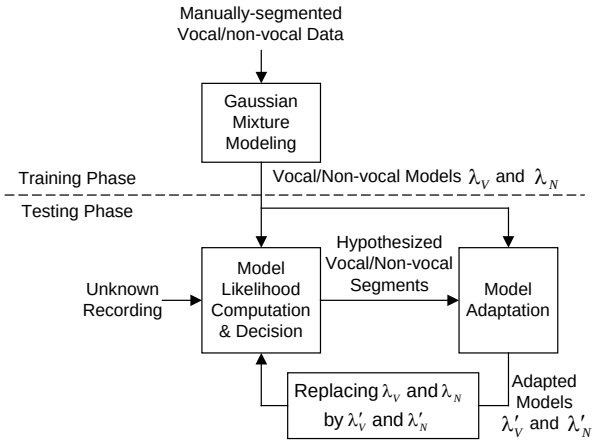


Figure 5. Vocal/non-vocal segmentation.

During training, a music database with manual vocal/non-vocal transcriptions is used to form two separate Gaussian mixture models (GMMs) [20]: a vocal GMM, and a non-vocal GMM. Both GMMs are designed to model the spectral distribution of various broad acoustic classes by a combination of Gaussian components, in which the broad acoustic classes reflect some general vocal tract and instrumental configurations. We denote the vocal GMM as λ_V, and the non-vocal GMM as λ_N. Parameters of the GMMs are initialized via *k*-means clustering and iteratively adjusted via expectation-maximization (EM) [5].

In the testing phase, the recognizer takes as input the T_x-length feature vectors $\mathbf{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_{Tx}\}$ extracted from an unknown recording, and produces as output the frame log-likelihoods $\log p(\boldsymbol{x}_t|\lambda_V)$ and $\log p(\boldsymbol{x}_t|\lambda_N)$, for $1 \leq t \leq T_x$. Since singing tends to be continuous, the recognition is preferably performed in a segment-by-segment manner, rather than a frame-by-frame manner. To reduce the risk of crossing multiple vocal/non-vocal boundaries, a segment is selected and examined in the following way. First, vector clustering is applied to all the frame feature vectors, and each frame is assigned a cluster index associated with its feature vector. Then, each segment is assigned the majority index of its constituent frames, and adjacent segments are merged as a homogeneous segment, if they have the same index. Accordingly, a homogeneous segment can be hypothesized as either vocal or non-vocal using:

$$\frac{1}{W_k}\left(\sum_{i=0}^{W_k-1}\log p(\boldsymbol{x}_{s_k+i}\mid\lambda_V) - \sum_{i=0}^{W_k-1}\log p(\boldsymbol{x}_{s_k+i}\mid\lambda_N)\right) \begin{array}{c} \text{vocal} \\ > \\ \leq \\ \text{non - vocal} \end{array} \eta, \tag{2}$$

where W_k and s_k represent, respectively, the length and starting frame of the *k*-th homogeneous segment, and η is the decision threshold.

As the performance of the above recognizer crucially depends on the reliability of the vocal/non-vocal models, it seems necessary to use training data that exhaustively covers the vocal/non-vocal characteristics of various music styles. However, acquiring such a large amount of training data is usually cost prohibitive, since it requires considerable efforts to manually label the music. To circumvent this problem, we propose tailoring the vocal/non-vocal models for each of the individual test music recordings, instead of designing the models that can cover the universal vocal/non-vocal characteristics. The basic idea is to refine the vocal/non-vocal models by means of the recognition results. It is assumed that the acoustic characteristics of the true vocal/non-vocal segments within each music recording can be inferred largely from the hypothesized vocal/non-vocal segments. Thus, the hypothesized segments can be used to refine the models, so that the recognizer with the refined models then repeats the likelihood computation and decision making, which should improve recognition. There are a number of ways to perform model refinement. This study uses a model adaptation technique based on maximum *a posteriori* estimation [24]. The procedure of recognition and model adaptation is carried out iteratively, until the resulting vocal/non-vocal boundaries are not changed anymore.

4. SOLO VOICE MODELING

Stochastic modeling is a promising way to handle the information to be extracted from the observed data. In modeling the desired vocal-related information, a frequently-encountered problem is

that the observed vocal data is intermixed with the background accompaniments that are irrelevant to the acoustic characteristics to be modeled. The undesired background accompaniment interferes with the stochastic modeling, and hence degrades the performance of an information-extraction system. To tackle this problem, we have developed a stochastic modeling technique for distilling the underlying solo voices in an accompanied recording. This technique is motivated by the observation that in popular music, substantial similarities exist between the non-vocal regions and the accompaniment of the singing regions. It is, therefore, reasonable to assume that the stochastic characteristics of the background music can be approximated by those of the non-vocal regions.

Suppose that a T-length accompanied voice $\mathbf{V} = \{\boldsymbol{v}_1, \boldsymbol{v}_2, ..., \boldsymbol{v}_T\}$ is a mixture of a solo voice $\mathbf{S} = \{\boldsymbol{s}_1, \boldsymbol{s}_2, ..., \boldsymbol{s}_T\}$ and a background music $\mathbf{B} = \{\boldsymbol{b}_1, \boldsymbol{b}_2, ..., \boldsymbol{b}_T\}$. Both **S** and **B** are unobservable, but **B**'s stochastic characteristics can be estimated from the non-vocal segments. Accordingly, it is sufficient to build a stochastic model λ_s for the solo voice **S**, based on the available information from **V** and **B**. Consider an example where our goal is to capture a singer's voice characteristics in a music recording. We further assume that **S** and **B** are, respectively, drawn randomly and independently according to GMMs $\lambda_s = \{w_{s,i}, \boldsymbol{\mu}_{s,i}, \boldsymbol{\Sigma}_{s,i} \mid 1 \le i \le I\}$, and $\lambda_b = \{w_{b,j}, \boldsymbol{\mu}_{b,j}, \boldsymbol{\Sigma}_{b,j} \mid 1 \le j \le J\}$, where $w_{s,i}$ and $w_{b,j}$ are mixture weights, $\boldsymbol{\mu}_{s,i}$ and $\boldsymbol{\mu}_{b,j}$ are mean vectors, and $\boldsymbol{\Sigma}_{s,i}$ and $\boldsymbol{\Sigma}_{b,j}$ are covariance matrices.

If the signal **V** is formed from a generative function $\boldsymbol{v}_t = f(\boldsymbol{s}_t, \boldsymbol{b}_t)$, where $1 \le t \le T$, the probability of **V**, given λ_s and λ_b can be represented by

$$p(\mathbf{V} \mid \lambda_s, \lambda_b) = \prod_{t=1}^{T} \left\{ \sum_{i=1}^{I} \sum_{j=1}^{J} w_{s,i} w_{b,j}\, p(\boldsymbol{v}_t \mid \boldsymbol{\mu}_{s,i}, \boldsymbol{\Sigma}_{s,i}, \boldsymbol{\mu}_{b,j}, \boldsymbol{\Sigma}_{b,j}) \right\}, \tag{3}$$

where

$$p(\boldsymbol{v}_t \mid \boldsymbol{\mu}_{s,i}, \boldsymbol{\Sigma}_{s,i}, \boldsymbol{\mu}_{b,j}, \boldsymbol{\Sigma}_{b,j}) = \iint_{\boldsymbol{v}_t = f(\boldsymbol{s}_t, \boldsymbol{b}_t)} \mathcal{N}(\boldsymbol{s}; \boldsymbol{\mu}_{s,i}, \boldsymbol{\Sigma}_{s,i}) \mathcal{N}(\boldsymbol{b}; \boldsymbol{\mu}_{b,j}, \boldsymbol{\Sigma}_{b,j}) d\boldsymbol{s} d\boldsymbol{b}, \tag{4}$$

and $\mathcal{N}(\cdot)$ denotes a Gaussian density function. In our context, **V**, **S** and **B** are represented in the form of cepstral features, and since **S** and **B** are additive in the time domain or linear-spectrum domain, the accompanied voice can be approximately expressed by $\boldsymbol{v}_t = \log[\exp(\boldsymbol{s}_t) + \exp(\boldsymbol{b}_t)] \approx \max(\boldsymbol{s}_t, \boldsymbol{b}_t)$, $1 \le t \le T$. Thus, Equation (4) is explicitly computed using

$$\begin{aligned} p(\boldsymbol{v}_t \mid \boldsymbol{\mu}_{s,i}, \boldsymbol{\Sigma}_{s,i}, \boldsymbol{\mu}_{b,j}, \boldsymbol{\Sigma}_{b,j}) = {} & \mathcal{N}(\boldsymbol{s}_t; \boldsymbol{\mu}_{s,i}, \boldsymbol{\Sigma}_{s,i}) \int_{-\infty}^{\boldsymbol{v}_t} \mathcal{N}(\boldsymbol{b}; \boldsymbol{\mu}_{b,j}, \boldsymbol{\Sigma}_{b,j}) d\boldsymbol{b} \\ & + \mathcal{N}(\boldsymbol{b}_t; \boldsymbol{\mu}_{b,j}, \boldsymbol{\Sigma}_{b,j}) \int_{-\infty}^{\boldsymbol{v}_t} \mathcal{N}(\boldsymbol{s}; \boldsymbol{\mu}_{s,i}, \boldsymbol{\Sigma}_{s,i}) d\boldsymbol{s}. \end{aligned} \tag{5}$$

To build λ_s, a maximum-likelihood estimation can be made as

$$\lambda_s^* = \arg\max_{\lambda_s} p(\mathbf{V} \mid \lambda_s, \lambda_b). \tag{6}$$

Using the EM algorithm, a new model $\hat{\lambda}_s$ is iteratively estimated by maximizing the auxiliary function

$$Q(\lambda_s, \hat{\lambda}_s) = \sum_{t=1}^{T} \sum_{i=1}^{I} \sum_{j=1}^{J} p(i, j \mid \boldsymbol{v}_t, \lambda_s, \lambda_b) \log p(i, j, \boldsymbol{v}_t \mid \hat{\lambda}_s, \lambda_b), \tag{7}$$

where

$$p(i, j, \boldsymbol{v}_t \mid \hat{\lambda}_s, \lambda_b) = \hat{w}_{s,i} w_{b,j}\, p(\boldsymbol{v}_t \mid \hat{\boldsymbol{\mu}}_{s,i}, \hat{\boldsymbol{\Sigma}}_{s,i}, \boldsymbol{\mu}_{b,j}, \boldsymbol{\Sigma}_{b,j}), \tag{8}$$

and

$$p(i, j \mid \boldsymbol{v}_t, \lambda_s, \lambda_b) = \frac{w_{s,i} w_{b,j}\, p(\boldsymbol{v}_t \mid \boldsymbol{\mu}_{s,i}, \boldsymbol{\Sigma}_{s,i}, \boldsymbol{\mu}_{b,j}, \boldsymbol{\Sigma}_{b,j})}{\sum_{m=1}^{I} \sum_{n=1}^{J} w_{s,m} w_{b,n}\, p(\boldsymbol{v}_t \mid \boldsymbol{\mu}_{s,m}, \boldsymbol{\Sigma}_{s,m}, \boldsymbol{\mu}_{b,n}, \boldsymbol{\Sigma}_{b,n})}. \tag{9}$$

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

$$\hat{w}_{s,i} = \frac{1}{T} \sum_{t=1}^{T} \sum_{j=1}^{J} p(i, j \mid \boldsymbol{v}_t, \lambda_s, \lambda_b), \tag{10}$$

$$\hat{\boldsymbol{\mu}}_{s,i} = \frac{\sum_{t=1}^{T} \sum_{j=1}^{N} p(i, j \mid \boldsymbol{v}_t, \lambda_s, \lambda_b) E\{\boldsymbol{s}_t \mid \boldsymbol{v}_t, \boldsymbol{\mu}_{s,i}, \boldsymbol{\Sigma}_{s,i}, \boldsymbol{\mu}_{b,j}, \boldsymbol{\Sigma}_{b,j}\}}{\sum_{t=1}^{T} \sum_{j=1}^{N} p(i, j \mid \boldsymbol{v}_t, \lambda_s, \lambda_b)}, \tag{11}$$

$$\hat{\boldsymbol{\Sigma}}_{s,i} = \frac{\sum_{t=1}^{T} \sum_{j=1}^{J} p(i, j \mid \boldsymbol{v}_t, \lambda_s, \lambda_b) E\{\boldsymbol{s}_t \boldsymbol{s}_t' \mid \boldsymbol{v}_t, \boldsymbol{\mu}_{s,i}, \boldsymbol{\Sigma}_{s,i}, \boldsymbol{\mu}_{b,j}, \boldsymbol{\Sigma}_{b,j}\}}{\sum_{t=1}^{T} \sum_{j=1}^{J} p(i, j \mid \boldsymbol{v}_t, \lambda_s, \lambda_b)} - \hat{\boldsymbol{\mu}}_{s,i} \hat{\boldsymbol{\mu}}_{s,i}', \tag{12}$$

where prime (′) denotes a vector transpose, and

$$\begin{aligned} & E(\boldsymbol{s}_t \mid \boldsymbol{v}_t, \boldsymbol{\mu}_{s,i}, \boldsymbol{\Sigma}_{s,i}, \boldsymbol{\mu}_{b,j}, \boldsymbol{\Sigma}_{b,j}) \\ & = \frac{\iint_{\boldsymbol{v}_t = f(\boldsymbol{s}_t, \boldsymbol{b}_t)} \boldsymbol{s} \mathcal{N}(\boldsymbol{s}; \boldsymbol{\mu}_{s,i}, \boldsymbol{\Sigma}_{s,i}) \mathcal{N}(\boldsymbol{b}; \boldsymbol{\mu}_{b,j}, \boldsymbol{\Sigma}_{b,j}) d\boldsymbol{s} d\boldsymbol{b}}{p(\boldsymbol{v}_t \mid \boldsymbol{\mu}_{s,i}, \boldsymbol{\Sigma}_{s,i}, \boldsymbol{\mu}_{b,j}, \boldsymbol{\Sigma}_{b,j})}, \end{aligned} \tag{13}$$

$$\begin{aligned} & E(\boldsymbol{s}_t \boldsymbol{s}_t' \mid \boldsymbol{v}_t, \boldsymbol{\mu}_{s,i}, \boldsymbol{\Sigma}_{s,i}, \boldsymbol{\mu}_{b,j}, \boldsymbol{\Sigma}_{b,j}) \\ & = \frac{\iint_{\boldsymbol{v}_t = f(\boldsymbol{s}_t, \boldsymbol{b}_t)} \boldsymbol{s}\boldsymbol{s}' \mathcal{N}(\boldsymbol{s}; \boldsymbol{\mu}_{s,i}, \boldsymbol{\Sigma}_{s,i}) \mathcal{N}(\boldsymbol{b}; \boldsymbol{\mu}_{b,j}, \boldsymbol{\Sigma}_{b,j}) d\boldsymbol{s} d\boldsymbol{b}}{p(\boldsymbol{v}_t \mid \boldsymbol{\mu}_{s,i}, \boldsymbol{\Sigma}_{s,i}, \boldsymbol{\mu}_{b,j}, \boldsymbol{\Sigma}_{b,j})}. \end{aligned} \tag{14}$$

The details of Equations (13) and (14) can be found in [24]. Note that if the number of mixtures in the background music GMM is zero, then the solo voice modeling method degenerates to directly modeling the observed vocal signal, without taking the background music into account.

5. SUPERVISED SINGER CLASSIFICATION

5.1 Methodology

Given an unknown music recording **X**, the objective of the supervised singer classification is to determine which among a group of P singers $\{S_1, S_2, \ldots, S_P\}$ performed **X**. Using the solo voice modeling method described in Sec. 4, the P candidate singers are in turn represented by P singer-specific models $\{\lambda_{s,1}, \lambda_{s,2}, \ldots, \lambda_{s,P}\}$. The most likely singer of **X** can then be determined by finding which of the models best matches **X**. As shown in Figure 6, the procedure begins with the segmentation of **X** into vocal part $\mathbf{X}_V$ and non-vocal part $\mathbf{X}_B$. An approximate background music model λ_B is then created using the non-vocal part $\mathbf{X}_B$. Then, the likelihood that **X** belongs to the i-th singer is computed using the conditional probability of the accompanied vocal signal $\mathbf{X}_V$, given the singer-specific solo voice model, $\lambda_{s,i}$, and the approximate background music model λ_B, i.e., $p(\mathbf{X}_V \mid \lambda_{s,i}, \lambda_B)$. According to the maximum likelihood decision rule, the classifier decides in favor of a singer S^* for **X** satisfying

$$S^* = \arg\max_{1 \le i \le P} p(\mathbf{X}_V \mid \lambda_{s,i}, \lambda_B). \tag{15}$$

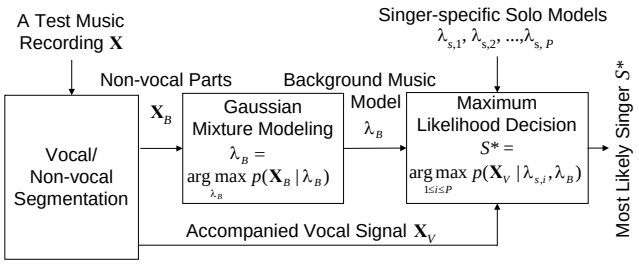


Figure 6. Testing phase of the supervised singer classification.

5.2 Experimental Results

The music data used in this experiment consisted of 416 tracks from Mandarin pop music CDs. The average length of a track was around three minutes. All the tracks were manually labeled with the singer's identity and the vocal/non-vocal boundaries. The database was divided into two subsets, denoted as DB1 and DB2, respectively. DB1 comprised 200 tracks performed by 10 female and 10 male singers, with 10 distinct songs per singer, while DB2 contained the remaining 216 tracks, performed by 13 female and 8 male singers, none of whom appeared in DB1. All music data was 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.

In this experiment, DB2 was used to generate the vocal model λ_V and non-vocal model λ_N. The feature vectors were Mel-scale frequency cepstral coefficients [4], computed using a 32-ms Hamming-windowed frame with 10-ms frame shifts. Performance of the vocal/non-vocal segmentation was evaluated on the basis of frame accuracy, defined by

$$\frac{\#\,\text{correctly-recognized frames}}{\#\,\text{total frames}} \times 100\%.$$

However, 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 computation. The best segmentation accuracy, which we evaluated with DB1, was 82.4%, using a 64-mixture vocal GMM and an 80-mixture non-vocal GMM.

The DB1 subset was further divided into DB1-T for training singer-specific models, and DB1-E for evaluation purposes. DB1-T comprised five tracks per singer, while the DB1-E comprised the remaining five tracks per singer. To test the performance of supervised singer classification with respect to different lengths of music recordings, each of the tracks in DB1-E was divided into several overlapping clips of T feature vectors. A 10-sec clip corresponded 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 experiment was conducted in a clip-by-clip manner, and the classification accuracy was assessed by the percentage of correctly-classified clips over the total number of clips. In the training phase, the number of mixture components used in each of the solo voice models and the background music models was empirically determined to be 48 and 16, respectively. In the testing phase, the online-created background music model was empirically set to have 4 Gaussian components, if the number of the non-vocal frames exceeded 200; otherwise, no background music model was used.

Figure 7 shows the classification results with respect to T = 1000 (10 sec), 3000 (30 sec), 6000 (60 sec), and the entire track, in which the T-length clips that were fully labeled as non-vocal were not used for testing. As expected, the classification accuracy improved as the clip length increased. Furthermore, we can see that the performance of the solo voice modeling method is noticeably better than that of an intuitive method based on GMM without background music modeling. When a test clip contained an entire track, we obtained a classification accuracy of 96%. This indicates that the proposed singer classification method is capable of organizing music data. In addition, an interesting observation can be made when we compare the classification performance using the manual vocal/non-vocal segmentation with automatic segmentation. Intuitively, the classification performance achieved with the manual vocal/non-vocal segmentation should serve as an upper bound for that obtained using automatic segmentation. However, the results contradict that intuition. The major reason for this phenomenon is that automatic segmentation is actually advantageous for pruning some feature vectors that are manually labeled as vocal, but heavily mixed with the loud background music. Therefore, despite some loss of information, pruning such vocal frames can prevent non-singer features interfering with classification.

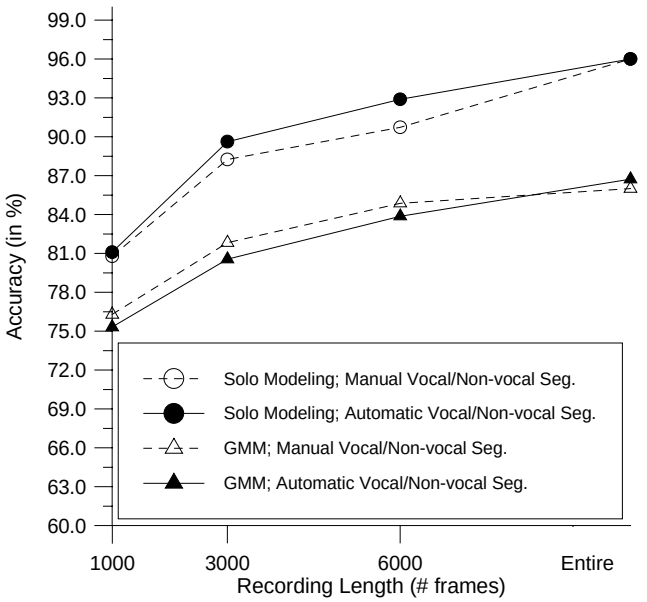


Figure 7. Supervised singer classification results.

6. UNSUPERVISED SINGER CLASSIFICATION

6.1 Methodology

To begin, each of the M music recordings, $\mathbf{X}_1$, $\mathbf{X}_2$, …, $\mathbf{X}_M$, to be classified is segmented into vocal and non-vocal regions. The non-vocal regions of each recording $\mathbf{X}_i$, $1 \le i \le M$, form an approximate background music model $\lambda_{b,i}$. Then, a solo voice model $\lambda_{s,i}$ is created for each recording by using its vocal regions together with the approximate background music model. Accordingly, the M recordings are in turn represented by M sets of solo voice model and background music model.

The similarities between recordings can then be measured by computing the log-likelihood, $L_{i,j} = \log p(\mathbf{X}_{V,i} | \lambda_{s,j}, \lambda_{b,i})$ for $1 \le i, j \le M$, that the vocal regions of the i-th recording $\mathbf{X}_{V,i}$ tests against the models pertaining to the j-th recording. Here, a large log-likelihood $L_{i,j}$ indicates that the underlying singer's voice in the i-th recording is similar to that of the j-th recording. In general, a log-likelihood computed from two same-singer recordings is larger than that computed from two different-singer recordings. However, this cannot always be guaranteed, since perfectly representing a singer's voice characteristics is beyond the capability of the stochastic modeling. To overcome this problem, we propose assigning a log-likelihood vector $\mathbf{L}_i = [L_{i,1}, L_{i,2}, \ldots, L_{i,M}]'$, to each recording, $\mathbf{X}_i$, $1 \le i \le M$, and then computing the similarities between the log-likelihood vectors. This strategy enables that, when measuring the similarity between a pair of recordings, some information from other recordings can be incorporated into the similarity computation for that pair of recordings. Since the few abnormal log-likelihoods within a vector can be diluted by most other normal ones, the similarities between recordings may be determined more reliably.

To quantify the inter-recording similarity, each log-likelihood vector is converted into a characteristic vector $\mathbf{F}_i = [F_{i,1}, F_{i,2}, \ldots, F_{i,M}]'$ by

$$F_{i,j} = \begin{cases} 1.0 & , \ j = i \\ \exp[\alpha(L_{i,j} - L_{i,\varphi})] & , \ j \ne i \end{cases}, \tag{16}$$

where α is a positive constant for scaling, and

$$\varphi = \arg\max_{k \ne i} L_{i,k}. \tag{17}$$

The similarity between two recordings $\mathbf{X}_i$ and $\mathbf{X}_j$ is computed using the cosine measure to the characteristic vectors $\mathbf{F}_i$ and $\mathbf{F}_j$, i.e., $(\mathbf{F}_i \cdot \mathbf{F}_j) / \| \mathbf{F}_i \| \| \mathbf{F}_j \|$. Proceeding in this way, unsupervised singer classification can be formulated as a problem of vector clustering. We can therefore apply the k-means clustering algorithm to solve this problem. Given the number of clusters[2] to be generated, it is hoped that by minimizing the variance of the characteristic vectors in each cluster, the recordings performed by the same singer can be grouped together.

6.2 Experimental Results

The music data used here was an expansion of the data used in Section 5.2. It consisted of 1005 pop music tracks, including 200 tracks of the DB1, 216 tracks of DB2, and 589 newly-added tracks. The 589 tracks, denoted as DB3, were performed by 26 female and 13 male singers, none of whom appeared in DB1 and DB2. Each of the singers in DB3 performed 11 to 18 distinct songs. However, we did not label DB3 with vocal/non-vocal boundaries; hence, the following classification experiment was performed on the basis of automatic vocal/non-vocal segmentation. The configurations of feature vector computation and the models for vocal/non-vocal segmentation were the same as those used in Section 5.2.

Performance of the unsupervised singer classification is evaluated on the basis of average cluster purity [23], defined by

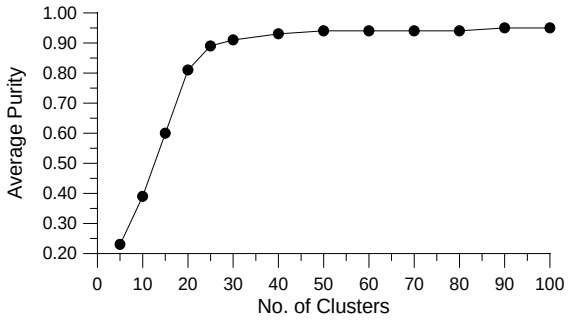
$$\overline{\rho} = \frac{1}{M} \sum_{k=1}^{K} n_k \rho_k, \tag{18}$$

and

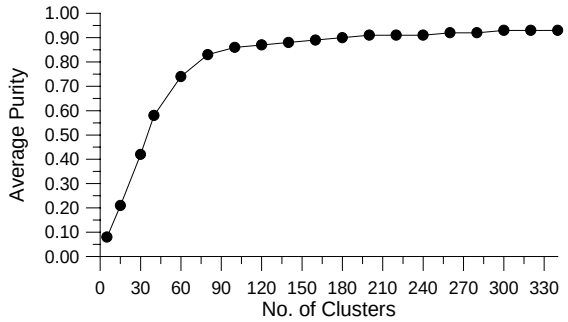
$$\rho_k = \sum_{p=1}^{P} \frac{n_{kp}^2}{n_k^2}, \tag{19}$$

where ρ_k is the purity of the k-th cluster, n_k is the total number of recordings in the k-th cluster, n_{kp} is the number of recordings in the k-th cluster that were performed by the p-th singer, K is the total number of generated clusters, and P is the total number of singers. The purity indicates the level of agreement in a cluster. The larger the value of the purity, the better the classification that is performed. A perfect classification should produce an average purity of one.

We conducted two experiments of unsupervised singer classification on the DB1 (20 singers) and DB1+DB3 (59 singers), respectively. The solo voice model and background music model for each recording were empirically determined to be 24 and 8, respectively. Figures 8a and 8b show the average purity as a function of the number of clusters. We can see that the average purity increases sharply as the number of clusters increases in the beginning, but tends to saturate after too many clusters are created. When the number of clusters is equal to the singer population, the purities of 0.81 and 0.74 are yielded by the DB1 (Figure 8a) and DB1+DB3 (Figure 8b), respectively. We can also see from Figures 8a and 8b that an average purity of more than 0.9 can be achieved when the number of clusters doubles the singer population. In this case, there will be more than 90% recordings tagged with correct singer's identity, if we listen to an arbitrary recording in each cluster and then assign a tag of singer's identity to all the recordings in each cluster.



(a) Evaluation with DB1 (20 singers).



(b) Evaluation with DB1+DB3 (59 singers).

Figure 8. Unsupervised singer classification results.

[2] Interested readers are referred to [24] for the study of determining an appropriate number of clusters.

7. SINGER-BASED RETRIEVAL OF UNDOCUMENTED MUSIC DATA

7.1 Methodology

Similar to the unsupervised singer classification, we generate M sets of solo voice model $\lambda_{s,i}$ and background music model $\lambda_{b,i}$, $1 \le i \le M$, for a collection of M music recordings, $\mathbf{X}_1$, $\mathbf{X}_2$, …, $\mathbf{X}_M$, to be retrieved. Through the computation of inter-recording log-likelihoods, the M recordings can be represented as M characteristic vectors $\mathbf{F}_1$, $\mathbf{F}_2$,…, $\mathbf{F}_M$. This representation resembles the *vector space model*, which is commonly used in the text document retrieval problem. By treating each recording-specific solo voice model as a term defined in a vector space model, an input music query $\mathbf{Y}$ may also be represented as a characteristics vector $\mathbf{Q}$, in a similar way that each $\mathbf{F}_i$ is formed. To do this, we compute the log-likelihoods, $\mathcal{L}(\mathbf{Y},\mathbf{X}_i) = \log p(\mathbf{Y}_V|\lambda_{s,i}, \lambda_{b,y})$, for $1 \le i \le M$, where $\mathbf{Y}_V$ is the vocal portion of $\mathbf{Y}$, and $\lambda_{b,y}$ is the background music model trained using the non-vocal portion of $\mathbf{Y}$. The characteristic vector $\mathbf{Q} = [Q_1, Q_2,\ldots, Q_M]'$ can then be formed by

$$Q_j = \exp[\alpha(\mathcal{L}(\mathbf{Y},\mathbf{X}_i) - \mathcal{L}(\mathbf{Y},\mathbf{X}_\xi))] \ , \ 1 \le j \le M, \qquad (20)$$

where $\xi = \arg\max_i \mathcal{L}(\mathbf{Y},\mathbf{X}_i)$.

Accordingly, the relevance of each recording with respect to a query can be evaluated by using the cosine measure to the characteristic vectors, i.e.,

$$\mathcal{R}(\mathbf{Y},\mathbf{X}_i) = \frac{\mathbf{Q}\cdot\mathbf{F}_i}{\|\mathbf{Q}\|\,\|\mathbf{F}_i\|}, \ 1 \le i \le M. \qquad (21)$$

Let R_i denote the rank of $\mathcal{R}(\mathbf{Y},\mathbf{X}_i)$ among $\mathcal{R}(\mathbf{Y},\mathbf{X}_1)$, $\mathcal{R}(\mathbf{Y},\mathbf{X}_2)$, …, $\mathcal{R}(\mathbf{Y},\mathbf{X}_M)$ in descending order. A music recording $\mathbf{X}_i$ is hypothesized as being performed by the singer of query $\mathbf{Y}$ if $R_i < \Upsilon$, where Υ controls the number of documents that will be presented to the user.

The above music data retrieval method may be improved by applying *relevance feedback* (RF) [22], which refines queries using information from the data considered relevant by users (explicit RF) or by the system itself (blind RF). A typical RF method in text document retrieval is to append relevant words from retrieved documents to the keyword string of a query, and then repeat the retrieval process based on the new query. Its intuitive counterpart in this task may be carried out by concatenating the relevant recordings with the music query, and then re-evaluating the relevance of each recording with respect to the new music query. However, this strategy cannot control the unequal weight of information contributed by different recordings; hence, there may be a severe error whenever the music query is concatenated with a recording that is performed by a different singer. To apply RF more effectively, we propose to refine the characteristic vector of a music query, instead of using direct data concatenation. Specifically, the characteristic vector $\mathbf{Q}$ is refined using

$$\hat{\mathbf{Q}} = \mathbf{Q} + \sum_{i=1}^{M} \beta^{R_i}\,\mathbf{F}_i \ , \qquad (22)$$

where β is a constant, which is assigned to be slightly smaller than one.

7.2 Experimental Results

The validity of the above singer-based music data retrieval method was examined with "DB1+DB3" described in Section 6.2. Our experiments were conducted in a leave-one-out manner, which used one track at a time in DB1+DB3 as a query to retrieve the remaining 788 tracks, and then rotated through all the tracks. The feature vector computation, vocal/non-vocal segmentation, and the solo voice modeling followed the same configurations used in Section 6.2.

The retrieval results were assessed by means of recall rate (RR), precision rate (PR), and mean average precision (mAP), respectively, computed by

$$\text{RR (in \%)} = \frac{\#\text{ relevant recordings presented to a user}}{\#\text{ total relevant recordings}} \times 100\%,$$

$$\text{PR (in \%)} = \frac{\#\text{ relevant recordings presented to a user}}{\#\text{ total recordings presented to a user}} \times 100\%,$$

and

$$\text{mAP} = \frac{1}{n_Q}\sum_{i=1}^{n_Q}\left(\frac{1}{n_{Q,i}}\sum_{k=1}^{n_{Q,i}}\frac{k}{R_{k,i}}\right), \qquad (23)$$

where n_Q is the total number of queries, $n_{Q,i}$ is the number of relevant recordings presented to a user, with respect to the i-th query, and $R_{k,i}$ is the rank of k-th relevant recording in the ranked list.

Figure 9 shows the precision and recall rates with respect to the number of recordings presented to a user. We can see that the method with relevance feedback consistently improves the retrieval performance. The best RR and PR both reached around 70%, as thirteen recordings were presented to a user. With regard to the mean average precision, we obtained an mAP of 100.0%, 95.2%, and 82.6%, when the numbers of recordings presented to a user were 1, 5, and 10, respectively.

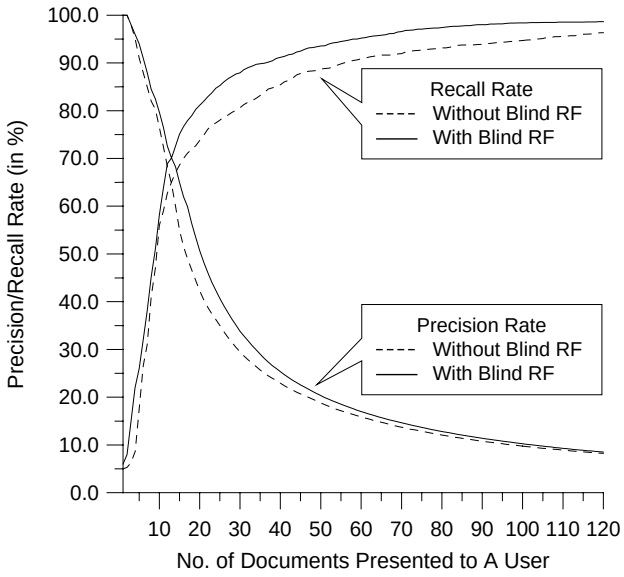


Figure 9. Results of the singer-based music data retrieval.

8. SINGER VERIFICATION FOR COPY-RIGHT PROTECTION

8.1 Methodology

Analogous to supervised singer classification, the operation of a singer-verification system can be divided in two phases: training and testing. In the training phase, music data from the concerned singers must be collected beforehand. This data is segmented into vocal and non-vocal regions, and then used to generate a singer-specific model λ_s^H, based on the solo voice modeling method. In addition, to characterize the irrelevant singers' voices (hypothesis H_1), we generate a universal singer model λ_s^U using all the available music data not performed by the hypothesized singer.

During testing, a background music GMM λ_b is created on-line using the segmented non-vocal regions of a test recording **X**. The system determines whether or not **X** is performed by the hypothesized singer using

$$\log p(\mathbf{X}_V \mid \lambda_s^H, \lambda_b) - \log p(\mathbf{X}_V \mid \lambda_s^U, \lambda_b) \underset{\text{No}}{\overset{\text{Yes}}{\gtrless}} \delta, \tag{24}$$

where $\mathbf{X}_V$ denotes all the segmented vocal regions in **X**, and δ is the decision threshold.

8.2 Experimental Results

The singer-verification experiments were conducted in a leave-one-out manner, which specified one singer at a time in "DB1+DB3" as a hypothesized singer, and then rotated through all the singers. The music data used for training each hypothesized singer model contained five tracks, and the remaining tracks not used for training served as the testing data. All the training material was also used to create the universal singer model λ_s^U, with 48 Gaussian components determined empirically. The number of Gaussian components used in the hypothesized singer model and background music model were the same as those used in the experiment of supervised singer classification.

Performance of the singer verification was evaluated on the basis of miss detection rate (MDR) and false alarm rate (FAR). A miss detection occurs when a music recording performed by the hypothesized singer was determined as "No", while a false alarm occurs when a music recording not performed by the hypothesized singer was determined as "Yes". MDR and FAR are subject to tradeoff, which can be represented by the Detection Error Tradeoff (DET) curve [17]. Figure 10 shows the singer verification results. We can see that the equal error rate (MDR = FAR) is around 14%. Since in practical use, attaining a low MDR is usually more important than attaining a low FAR, the experimental result shows that, by allowing a slightly larger probability of false alarm, say 20%, the proposed system can detect more than 90% suspect recordings. This confirms the feasibility of a copyright-protection technique based on singer verification.

9. CONCLUSIONS

We have presented a paradigm of extracting vocal-related information from music audio signals to facilitate the management of popular music collections. This paradigm begins by segmenting music recordings into vocal and non-vocal portions, followed by a stochastic modeling of solo voice signals underlying the accompanied vocal portions. The stochastic modeling of solo voice signals is done by suppressing the characteristics of the background accompaniments estimated from the non-vocal portions. Following this paradigm, we have developed several useful techniques for the realization of a popular music digital library. Specifically, we have shown that music recordings can be organized automatically according to their associated singers, by using either supervised classification or unsupervised classification. We have also provided a query-by-example framework for music information retrieval, which allows users to locate a specified singer's music documents without explicitly indicating the name of the sought singer. Furthermore, to curb the illegal use of copyrighted music material on networks, we have developed a singer-verification technique that enables copyright holders to rapidly scan suspect websites for pirated material. With regard to practicability, our future work will extend the current techniques to handle a wider variety of music data, including signal degradation or simultaneous vocals. In addition, the implementation details of managing popular music collections by extracting other vocal-related information, such as melody and singing language, will be further investigated.

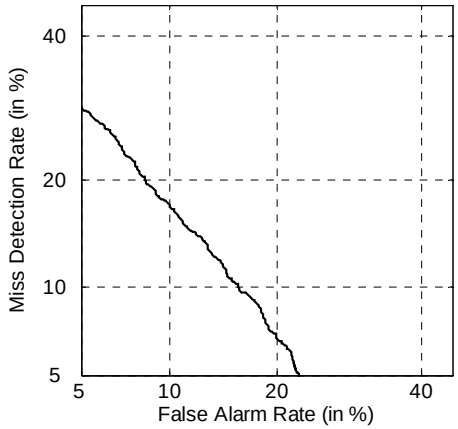


Figure 10. Singer-verification results.

10. ACKNOWLEDGMENTS

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

11. 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), 2001, 1498-1504.

[2] Bainbridge, D., Nevill-Manning, C. G., Witten, I. H., Smith, L. A., McNab, R. J. Towards A digital Library of Popular Music. In *Proceedings of the ACM International Conference on Digital Libraries*, 1999.

[3] Cunningham, S. J., Reeves, N., and Britland M. An Ethnographic Study of Music Information Seeking: Implications for the Design of a Music Digital Library. In *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries*, 2003.

[4] Davis, S. B., and Mermelstein, P. Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences. *IEEE Transactions on Acoustics, speech, and Signal Processing*, 28, 1980, 357-366.

[5] Dempster, A., Laird, N., and Rubin, D. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society*, 39, 1977, 1-38.

[6] Durey, A. S., and Clements, M. A. Features for Melody Spotting Using Hidden Markov Models. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2002.

[7] Eronen, A. Musical Instrument Recognition Using ICA-based Transform of Features and Discriminatively Trained HMMS. In *Proceedings of the International Symposium on Signal Processing and Its Applications*, 2003.

[8] Hacker, S. *MP3: The Definitive Guide*. O'Reilly, 2000.

[9] Haitsma, J., and Kalker, T. A Highly Robust Audio Fingerprinting System, In *Proceedings of the International Conference on Music Information Retrieval*, 2002.

[10] Herrera, P., Amatriain, X., Batlle, E., and Serra. X. Towards Instrument Segmentation for Music Content Description: A Critical Review of Instrument Classification Techniques. In *Proceedings of the International Symposium on Music Information Retrieval*, 2000.

[11] http://music-ir.org/evaluation/wp.html

[12] ISO-IEC/JTC1 SC29 WG11 Moving Pictures Expert Group. Information technology – multimedia content description interface – part 4: Audio. Committee Draft 15938-4, ISO/IEC, 2000.

[13] Kim, Y. E., and Whitman, B. Singer Identification in Popular Music Recordings Using Voice Coding Features. In *Proceedings of the International Conference on Music Information Retrieval*, 2002.

[14] Li, T., Ogihara, M., and Li, Q. A Comparative Study on Content-Based Music Genre Classification. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval*, 2003.

[15] Liu, C. C., and Huang, C. S. A Singer Identification Technique for Content-based Classification of MP3 Music Objects. In *Proceedings of the ACM International Conference on Information and Knowledge Management*, 2002.

[16] Liu, D., Lu, L., and Zhang, H. J. Automatic Mood Detection from Acoustic Music Data. In *Proceedings of the International Conference on Music Information Retrieval*, 2003.

[17] Martin, A., Doddington, G., Kamm, T., Ordowski, M., and Przybocki, M. The DET Curve in Assessment of Detection Task Performance. In *Proceedings of the European Conference on Speech Communication and Technology*, 1997.

[18] McNab, R. J., Smith, L. A., Witten, I. H. Towards the Digital Music Library: Tune Retrieval from Acoustic Input. In *Proceedings of the ACM International Conference on Digital Libraries*, 1996.

[19] Oppenheim, A. V., and Schafer, R. W. Homomorphic Analysis of Speech. *IEEE Transactions on Audio and Electroacoustics*, 16, 1968, 221-226.

[20] Reynolds, D. A., and Rose, R. C. Robust Text-independent Speaker Identification Using Gaussian Mixture Speaker Models. *IEEE Transactions on Speech and Audio Processing*, 3 (1), 1995, 72-83.

[21] Reynolds, D. A., Quatieri, T. F., and Dunn, R. B. Speaker Verification Using Adapted Gaussian Mixture Models. *Digital Signal Processing*, 10, 2000, 19-41.

[22] Salton, G. *Automatic Text Processing*. Addison-Wesley, Reading, MA, 1989.

[23] Solomonoff, A., Mielke, A., Schmidt, M., and Gish, H. Clustering Speakers by Their Voices. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1998.

[24] Tsai, W. H., Wang, H. M., and Rodgers D. Blind Clustering of Popular Music Recordings Based on Singer Voice Characteristics. *Computer Music Journal*, 28 (3), 2004, 68-78.

[25] Tsai, W. H., and Wang, H. M. A Query-by-example Framework to Retrieve Music Documents by Singer. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, 2004.

[26] Tzanetakis, G., and Cook, P. Musical Genre Classification of Audio Signals. *IEEE Transactions on Speech and Audio Processing*, 10 (5), 2002, 293-302.

[27] Tzanetakis, G., Gao, J., and Steenkiste, P. A Scalable Peer-to-Peer System for Music Content and Information Retrieval. In *Proceedings of the International Conference on Music Information Retrieval*, 2003.

[28] Venkatachalam, V., Cazzanti, L., Dhillon, N., and Wells, M. Automatic Identification of Sound Recordings. *IEEE Signal Processing Magazine*, March 2004, 92-99.

[29] Wang, C. K., Lyu, R. Y., and Chiang, Y. C. An Automatic Singing Transcription System with Multilingual Singing Lyric Recognizer and Robust Melody Tracker. In *Proceedings of the European Conference on Speech Communication and Technology*, 2003.

[30] Whitman, B., Flake, G., and Lawrence, S. Artist Detection in Music with Minnowmatch. In *Proceedings of the IEEE Workshop on Neural Networks for Signal Processing*, 2001.

[31] Yang, D., and Lee, W. Disambiguating Music Emotion Using Software Agents. In *Proceedings of the International Conference on Music Information Retrieval*, 2004.