A Survey of Audio-Based Music Classification and Annotation

Zhouyu Fu, Guojun Lu, Kai Ming Ting, and Dengsheng Zhang

***Abstract*—Music information retrieval (MIR) is an emerging research area that receives growing attention from both the research community and music industry. It addresses the problem of querying and retrieving certain types of music from large music data set. Classification is a fundamental problem in MIR. Many tasks in MIR can be naturally cast in a classification setting, such as genre classification, mood classification, artist recognition, instrument recognition, etc. Music annotation, a new research area in MIR that has attracted much attention in recent years, is also a classification problem in the general sense. Due to the importance of music classification in MIR research, rapid development of new methods, and lack of review papers on recent progress of the field, we provide a comprehensive review on audio-based classification in this paper and systematically summarize the state-of-the-art techniques for music classification. Specifically, we have stressed the difference in the features and the types of classifiers used for different classification tasks. This survey emphasizes on recent development of the techniques and discusses several open issues for future research.**

***Index Terms*—Acoustic signal processing, classification algorithms, feature extraction, music information retrieval.**

I. Introduction

With the development of information and multimedia technologies, digital music has become widely available from different media, including radio broadcasting, digital storage such as compact discs (CDs), the Internet, etc. The vast amount of music accessible to the general public calls for developing tools to effectively and efficiently retrieve and manage the music of interest to the end users. Music information retrieval (MIR) is an emerging research area in multimedia to cope with such necessity. A key problem in MIR is classification, which assigns labels to each song based on genre, mood, artists, etc. Music classification is an interesting topic with many potential applications. It provides important functionalities for music retrieval. This is because most end users may only be interested in certain types of music. Thus, a classification system would enable them to search for the music they are interested in. On the other hand, different music types have different properties. We can manage them more effectively and efficiently once they are categorized into different groups.

Manuscript received February 05, 2010; revised June 10, 2010 and September 15, 2010; accepted November 19, 2010. Date of publication December 10, 2010; date of current version March 18, 2011. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Gerald Schuller.

The authors are with the Faculty of Information Technology, Gippsland School of Information Technology, Monash University, Gippsland Campus, Churchill, Victoria 3842, Australia (e-mail: zhouyu.fu@monash.edu; guojun.lu@monash.edu; kaiming.ting@monash.edu; dengsheng.zhang@monash.edu).

Digital Object Identifier 10.1109/TMM.2010.2098858

Music classification has received much attention from MIR researchers in recent years. In the MIR community, an annual event Music Information Retrieval Evaluation eXchange[1] (MIREX) is held for competitions on important tasks in MIR since 2004. Most of the high-level tasks in MIREX competitions are relevant to music classification. Those tasks directly related to music classification are listed in the following.

- Genre Classification [1]–[19]
- Mood Classification [12], [20]–[30]
- Artist Identification [31]–[39]
- Instrument Recognition [40]–[51]
- Music Annotation [52]–[66]

In this paper, we provide an overview of features and techniques used for the above music classification tasks. There have been a few survey articles in the relevant research field [67]–[69] in previous years. Scaringella *et al.* [67] reviewed the techniques of audio feature extraction and classification for the task of genre classification only. The review paper of Weihs *et al.* [68] focused on music classification in general but did not pay much attention to the subtle differences between different tasks, as different types of features may vary in performance for different tasks. Moreover, the field of music classification research is developing rapidly in the past few years, with new features and types of classifiers being developed and used. More importantly, the task of music annotation has recently gained much popularity in the MIR community since the work of Turnbull *et al.* [54] in 2007. The purpose of music annotation is to annotate each piece of song with a set of semantically meaningful text annotations called tags. A tag can be any relevant musical term that describes the genre, mood, instrumentation, and style of the song. Hence, music annotation can be treated as a classification problem in the general sense, where tags are class labels that cover different semantic categories. In contrast, standard classification tasks are confined to a specific semantic category like genre, mood, or instrument.

It is worth mentioning that music annotation, as an emerging task in music classification, was not covered in previous review papers. The most recent review paper of Casey *et al.* [69] focused on the general MIR research and did not concentrate on music classification in particular. Nor did it cover recent development of techniques in music classification and annotation. Thus, a comprehensive survey is needed to cover the recent progress in the field of music classification and annotation. In addition, we should pay attention to the differences between the classification tasks listed above and identify the features and classification techniques suited for individual tasks. Note that we focus exclusively on audio-based music classification and

[1]http://www.music-ir.com/mirex

annotation in this survey. Research and studies on classification based on symbolic representations of music are not covered here.

The remainder of this review paper is organized as follows. In Section II, we discuss audio feature extraction for music classification in general. In Section III, we discuss classification techniques and issues specific to music classification. We cover this section with a review of individual classification tasks, which emphasizes the task-specific characteristics and highlights the important features, classifiers, and related issues for each task. To the best of our knowledge, this task-oriented discussion, despite its importance, has not been found in previous review papers. Potential research issues for future study are discussed in Section IV, which is another novelty of our review paper. Conclusions are given in Section V.

II. Audio Features for Music Classification

The key components of a classification system are feature extraction and classifier learning [70]. Feature extraction addresses the problem of how to represent the examples to be classified in terms of feature vectors or pairwise similarities. The purpose of classifier learning is to find a mapping from the feature space to the output labels so as to minimize the prediction error. We focus on music classification based on audio signals unless otherwise stated.

A. Overview of Audio Features

Many audio features have been proposed in the literature for music classification. Different taxonomies exist for the categorization of audio features. Weihs *et al.* [68] have categorized the audio features into four subcategories, namely short-term features, long-term features, semantic features, and compositional features. Scaringella [67] followed a more standard taxonomy by dividing audio features used for genre classification into three groups based on timbre, rhythm, and pitch information, respectively. Each taxonomy attempts to capture audio features from certain perspective. Instead of a single-level taxonomy, here we unify the two taxonomies and present a hierarchical taxonomy in Fig. 1 that characterizes audio features from different perspectives and levels.

From the perspective of music understanding, we can divide audio features into two levels, low-level and mid-level features, as illustrated in the bottom two rows of Fig. 1 along with top-level labels. Low-level features can be further divided into two classes of timbre and temporal features. Timbre features capture the tonal quality of sound that is related to different instrumentation, whereas temporal features capture the variation and evolution of timbre over time. Low-level features are obtained directly from various signal processing techniques like Fourier transform, spectral/cepstral analysis, autoregressive modeling, etc. Each class of low-level features also consists of many different features, as shown in Fig. 1 by the abbreviations below the name of the class in the box. The meanings of these abbreviations will become clearer shortly in the discussion below. Low-level features have been used predominantly in music classification, due to the simple procedures to obtain them and their good performance. However, they are not closely related to the intrinsic properties of music

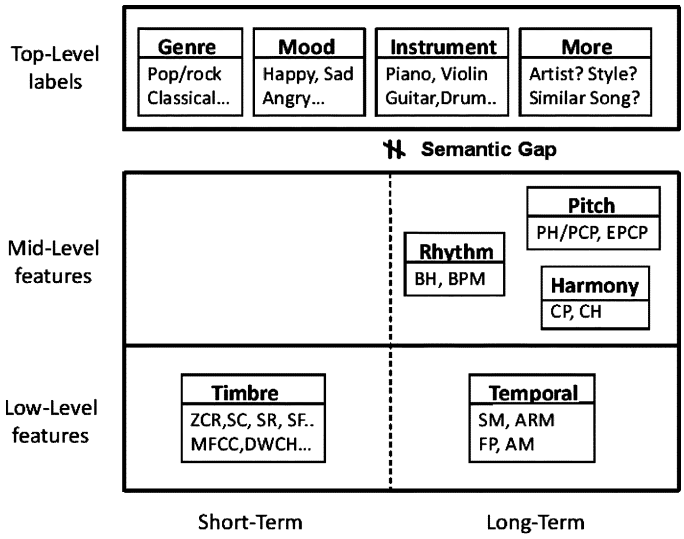

Fig. 1. Characterization of audio features.

as perceived by human listeners. Mid-level features provide a closer relationship and include mainly three classes of features, namely rhythm, pitch, and harmony. These features are usually extracted on top of low-level ones. At the top level, semantic labels provide information on how humans understand and interpret music, like genre, mood, style, etc. This is an abstract level as the labels cannot be readily obtained from lower level features as indicated by the semantic gap between mid-level features and labels. The purpose of content-based music classification is to bridge the semantic gap by inferring the labels from low-/mid-level features.

From a different perspective, audio features can also be categorized into short-term features and long-term features, as illustrated by columns of Fig. 1. Short-term features like timbre features usually capture the characteristics of the audio signal in frames with 10–100 ms duration, whereas long-term features like temporal and rhythm features capture the long-term effect and interaction of the signal and are normally extracted from local windows with longer durations. Hence, the main difference here is the length of local windows used for feature extraction. In the following, we focus on the semantic-based taxonomy and discuss in more detail the common low-level and mid-level features.

B. Low-Level Features

Low-level features are the important building block for audio classification systems. They are easy to extract and have demonstrated good performance in virtually all music classification tasks. Table I summarizes the common low-level features proposed in the literature and used in various music classification tasks.

The majority of the features listed in Table I are timbre features. As a basic element of music, timbre is a term describing the quality of a sound [81]. Different timbres are produced by different types of sound sources, like different voices and musical instruments. Timbre in music is similar to color in images. Despite the large number of timbre features proposed for music classification, the extraction of timbre features is closely

TABLE I
SUMMARY OF COMMON LOW-LEVEL FEATURES USED IN MUSIC CLASSIFICATION

Class	Feature Type	Used in
Timbre	Zero Crossing Rate (ZCR)	[1], [2], [9], [10]
	Spectral Centroid (SC)	[1], [2], [9], [10], [25]
	Spectral Rolloff (SR)	[1], [2], [9], [10], [25]
	Spectral Flux (SF)	[1], [2], [25]
	Spectral Bandwidth (SB)	[1], [9], [10], [25]
	Spectral Flatness Measure (SFM)	[71], [72], [73]
	Spectral Crest Factor (SCF)	[71], [72], [73]
	Amplitude Spectrum Envelop (ASE)	[74], [18]
	Octave based Spectral Contrast (OSC)	[75], [25], [27], [18]
	Daubechies Wavelet Coef Histogram (DWCH)	[2], [12], [27], [63]
	Mel-frequency Cepstrum Coefficient (MFCC)	[1], [35], [9], [39]
	Fourier Cepstrum Coefficient	[76], [9]
	Linear Predictive Cepstrum Coefficient (LPCC)	[34], [39]
	Stereo Panning Spectrum Features (SPSF)	[77], [78]
Temporal	Statistical Moments (SM)	[1], [2], [35], [9]
	Amplitude Modulation (AM)	[79], [80], [10], [18]
	Auto-Regressive Modeling (ARM)	[4], [15]

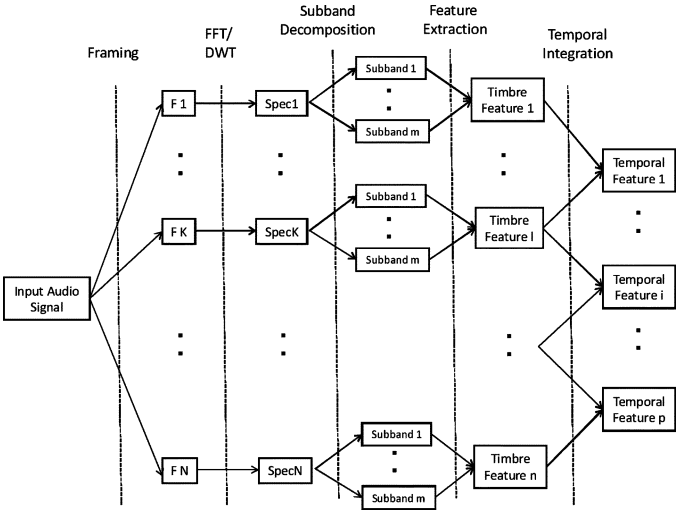

Fig. 2. Illustration of steps for low-level feature extraction.

related to spectral analysis of the audio signal and follows some standard steps, as illustrated in Fig. 2. A song is usually quite long in duration and may last for minutes. Hence, the input audio signal for each song may contain millions of samples given high sampling frequency of over 10 kHz. Instead of doing song-level signal analysis directly, a song is usually split into many local frames in the first step to facilitate subsequent frame-level timbre feature extraction. This has two main advantages. Firstly, it leads to a more efficient feature extraction process, since we only need to apply spectral analysis to short-time signals. Secondly, it is more effective to model the timbre features in frames with 10–100 ms duration. Due to the non-stationary nature of music, the audio signal of a song is normally a non-stationary time series whose statistical properties depend on time. By applying frame-level feature analysis, we can assume that the audio signal within each frame is stationary. Thus, the underlying timbre information can be more reliably captured.

After framing, spectral analysis techniques such as fast Fourier transform (FFT) and discrete wavelet transform (DWT) are then applied to the windowed signal in each local frame. From the output magnitude spectra, we can define some summary features such as spectral centroid (SC), spectral rolloff (SR), spectral flux (SF), and spectral bandwidth (SB) capturing simple statistics of the spectra. Hereafter, we term the collection of these features as short time Fourier transform (STFT) features. To extract more powerful features such as MFCC, OSC, DWCH, and MPEG-7 audio descriptors like SFM, SCF, and ASE, a subband analysis is performed by decomposing the power spectrum into subbands and applying feature extraction in each subband. The details of subband configuration and feature extraction vary for different types of features. Normally, the subbands are arranged with logarithmically spaced spectral ranges (OSC, ASE, DWCH) following the constant Q-transform [82]. For MFCC, the subbands are linearly spaced in lower frequencies (< 1 KHz) and logarithmically spaced in higher frequencies. The reason for using log-scale at the higher frequency range is to mirror the human auditory system, whereby finer spectral resolution is achieved at lower frequencies. The arrangement was empirically found to be a reasonable trade off, given the success of MFCC feature for music classification. Notice for each of these features, we can construct the corresponding derivative feature by taking the first-order or second-order differences in feature values between the current frame and previous frames. In this way, the feature set can be expanded solely based on the use of low-level features [10].

A number of issues on the use of timbre features deserve our special attention. Firstly, the local windows used in timbre feature extraction are usually taken from fixed-size intervals regardless of the underlying musical events. It would be desirable to split the signal into unequal intervals corresponding to separate notes where the boundaries are aligned with the onset and offset of the notes [8]. Secondly, the bank of band-pass filters is usually organized to cover octave-based logarithmic spectral range. Different implementations of filter bank adopt slightly different spectral ranges and pass bands. These would influence the classification performance in a data-dependent way. Thirdly, psycho-acoustic transformations are usually applied to normalize the spectrum and filter output values [3]. They have been shown to contribute positively to genre and mood classification [3], [83]. Fourth, phase information is usually discarded from the Fourier spectrum, but it may convey important information on the music being analyzed. The utility of phase information for music classification remains an open problem. More-

over, modern music industry has made extensive use of multiple sound channels during the music recording and production stages, most notably stereo sound. Most current music classification systems, however, have discarded the stereo information and consider the mixed sound signal by summing up the two channels. There has been limited work on utilizing stereo information for automatic music classification until recently [77], [78]. A stereo panning spectral feature (SPSF) was proposed in [77] which captures the contrast in the spectral distributions between left and right channels over different frequency ranges. The resulting SPSF, when combined with the standard MFCC feature, was shown to achieve superior classification performance as compared to using MFCC alone.

Temporal features form another important class of low-level features that capture the temporal evolution of the signal. It is worth noticing that temporal features are usually constructed on top of timbre features or the spectrogram obtained during timbre feature extraction. As illustrated in Fig. 2, timbre features extracted from several frames are further integrated to create the temporal feature. In this way, we can create a richer set of features for classification with different combination of timbre and temporal features. The simplest type of temporal features is statistical moments such as mean, variance, covariance, and kurtosis of timbre features collected from a larger local window called texture window [1]. Two variations have demonstrated good performance for music classification when combined with timbre features, namely $MuVar$ [1] and $MuCov$ [35]. The former takes the mean and variance of timbre feature vectors within the texture window and concatenate them into a summary vector, whereas the latter takes the mean and upper triangular part of the covariance matrix for concatenation. The same operation can be repeated for a even larger window at a higher scale, yielding $MuVar^2$ and $MuCov^2$. MuVar^2 combined with MFCC is the default audio feature extraction tool used in the music analysis software Marsyas [84].[2] A texture window for the computation of statistical moments over timbre features usually spans for a duration of a few seconds and contain hundreds to thousands of frames. Hence, the set of timbre feature vectors within the texture window can be treated as multivariate time series data. Signal processing techniques can then be applied to the time series data to extract temporal features. One big advantage for doing so is that temporal information has been preserved by treating the local feature set as a multivariate time series. Nevertheless, this is at the cost of increased complexity in modeling and computation. One can apply spectral transformation like STFT to the time series of local features and derive novel features, including fluctuation pattern (FP) [79], rhythmic pattern [14], rhythmic coefficient [85], and more sophisticated features derived from modulation spectrum analysis [18]. Specifically, Morchen [10] has studied the temporal integration problem in detail and listed a comprehensive set of operations that can be applied to the set of timbre features to produce new features at a coarser scale revealing the long-term trend of music. Many features proposed therein were generated with the similar rationale by analyzing the modulation of the amplitude spectrum. Hence, we call the whole family of these methods amplitude modulation (AM) in the following text. Multivariate autoregressive models have also been used to model the temporal evolution of local feature series [15]. Technically, any of the signal processing techniques used in timbre feature extraction can also be adopted to obtain temporal features, although only a few of them have been used in practice. The difference is that timbre feature extraction is performed on the input audio signal in local windows, whereas temporal feature extraction is performed on the series of extracted timbre features within larger texture windows.

Besides standard temporal feature extraction techniques mentioned above, probabilistic modeling of temporal features has also been applied to music classification like hidden Markov models (HMM) [6], [86]–[88]. The HMM is a statistical model for time series data with hidden states. For an audio signal, each frame is associated with a single state in HMM. The state of each frame is unobservable but it depends on the states of previous frames. The output of each frame, i.e., the feature vector extracted from the local frame, is generated from a probability model conditioned on the state. The purpose of HMM training is to estimate the parameters of the output probability model given each state and the transition model between neighboring states that maximizes the likelihood function. The trained model can then be used to estimate the likelihood that a new signal is generated from the model. To apply HMMs for music classification, one can use the inferred states as mid-level features for classification [86]–[88]. Alternatively, a template HMM can be built for each label class [6], [86]. A new song is assigned to the class whose template best matches the song with highest likelihood value.

C. Mid-Level Features

Despite the good performance of low-level features for music classification, they do not capture the intrinsic properties of music that humans perceive and appreciate. Low-level features are the dominant features used for various music classification tasks. Despite that, mid-level features have found their application for certain problems that require the use of higher level semantics like query by example [89], [90] and cover song detection[3][91]–[93]. The commonly used mid-level features in music analysis include

- Rhythm—recurring pattern of tension and release in music;
- Pitch—perceived fundamental frequency of the sound;
- Harmony—combination of notes simultaneously, to produce chords, and successively, to produce chord progressions [81].

Although the above features are easily identified and appreciated by music listeners, it is not straightforward to define them explicitly and extract them reliably from raw audio signals for the purpose of music analysis.

Rhythm is the most widely used mid-level feature in audio-based music classification. It describes how certain patterns occur and recur in the music and is related to the "danceability" of the music. Beat and tempo (beat-per-minute,

[2]http://marsyas.info

[3]The purpose of cover song detection is to identify the same song in the data set remixed with different style or played by a different artist

BPM) are two important cues that describe rhythmic content of the music which have been utilized in music classification. Tzanetakis proposed the use of beat histogram (BH) [1] for genre classification. The auto-correlation of the time-domain envelop signal is calculated. The peaks of the auto-correlation function are then identified which correspond to potential regularity of the music under analysis. The beat histogram models the distributions of the regularities exhibited in the envelop signal, where rhythmic features can be obtained such as magnitudes and locations of dominant peaks and BPM. Rhythm features have demonstrated good empirical performance for mood classification [21], [22], [27], [83]. This may be explained from the fact that the mood of a song is highly correlated with rhythm. For example, like sad songs usually have a slow rhythm, whereas angry songs usually have a fast rhythm.

Pitch is another important component of music. It is determined by what the ear judges to be the most fundamental frequency of the sound [81]. In the strict sense, however, a pitch is not equal to the fundamental frequency for two reasons. Firstly, the perception of pitch is completely subjective whereas frequency measurement is objective. Other factors like differences in timbre, loudness, and musical context also affect pitch [81]. Secondly, a musical note played on most instruments consists of a series of harmonic-related frequency, including the fundamental frequency and partials at integer multiples, and is normally perceived as one sound with a single pitch. Hence, pitch information extraction in real audio signals is more than locating the fundamental frequency. For frame level pitch analysis, various multi-pitch estimation algorithms have been developed [94], [95] to identify the top candidate pitches for each frame. Song level pitch feature representation like the pitch histogram (PH) can also be derived and applied to classification [1], [96]. Pitch histogram has been used in music genre and mood classification [1], [20] in early years of MIR research along with low-level timbre features like MFCC and other spectral features. It models the distribution of candidate pitches extracted from all frames. Each histogram bin captures the occurrence frequency of the corresponding pitch. Another important concept about pitch is the pitch class or chroma, which defines an equivalent class of pitches. The pitch class of a pitch depends on the relative position of the pitch within an octave by mapping all pitches from different octaves into a single octave in the diatonic scale. For example, low C, middle C, and high C all belong to the same pitch class. Pitch class features like pitch class profile (PCP) [97] and harmonic pitch class profile (HPCP) [98] have been developed and widely used in various tasks like melody analysis and transcription [93], [99], [100]. The chroma feature is a simplified version of HPCP [101] and can be obtained directly by transforming the spectrum values without any attempt on pitch detection. It has been previously used in music classification in conjunction with MFCC and the combined feature was shown to outperform the MFCC feature alone [102].

Harmony involves the use of chords. A chord is the fundamental constituent of harmony which involves the simultaneous combination of two or more notes [81]. Harmony is achieved by chord progression, a series of chords played successively. In contrast to melody, which captures the horizontal information of music, harmony explores the vertical dimension. Chord information like chord sequences (CS) can be extracted from the music audio data making use of various chord detection and recognition algorithms [97], [103], [104]. All these methods begin with pitch detection using either standard or enhanced pitch features [103] to identify the fundamental frequency and its partials. Then each pitch histogram feature is compared with the chord template to identify the existence of possible chords. Chord features complement pitch features in melody-based similarity matching and cover song detection [101], [105]. Despite its widespread use in cover song detection, it is not popular with standard music classification tasks. In mood classification, it has been shown that the incorporation of chord features can improve classification performance when used in combination with timbre and rhythm features [73]. A complete set of mid-level features like rhythm, pitch, and chords can be obtained from an automatic music transcription system. Music transcription refers to the analysis of an acoustic music signal so as to record information like the pitch, onset time, and duration of each sound. Intuitively, it can be regarded as the inverse process of producing scores from music recordings. Details of music transcription is beyond the scope of this paper. Interested readers are referred to the book [106] for more details. Lidy *et al.* [14] have also showed that the use of additional features obtained from an automatic transcription system can improve the performance of genre classification when combined with timbre features.

To summarize, the choice of audio features is much dependent on the problems we deal with. Timbre features are suitable for genre and instrument classification but not appropriate for comparing the melody similarity of two songs. For mood classification, a large amount of works used rhythm features [21], [22], [25], [27]. While pitch and harmonic features are not quite popular with standard classification systems based on genre, artist, mood, etc., they are the most important feature types for song similarity retrieval and cover song detection at melodic level [92], [93], [98], [105], where timbre features fail to achieve good results. This is corroborated by a recent comparative study on music similarity [107], which showed that timbre features best explain for the instrumentation of the music. Different melodies played by the same instrument would produce more similar timbre features than those corresponding to the same melody with different instrumentation. In general, there is no single set of task-independent features that can consistently outperform the others. In Section III, we will discuss in more detail the features and classifiers best suited for each music classification task.

D. Song-Level Feature Representation

Depending on the problem, music classification can be performed at either segment or song level. The former includes instrument recognition, where instrumentation may vary from segment to segment, and hence, a prediction is made for each segment. The later embodies the majority of classification tasks, like genre, mood, singer identification, and annotation, where a decision is returned for each song.

As mentioned earlier, current music analysis systems perform feature extraction at segment level. The segment for feature extraction can be a small frame with 10–100 ms duration as for the case of timbre features, or a larger texture window with a couple of seconds in length that comprises many frames for temporal features. In either case, a song usually contains many segments, and a feature vector can be obtained for each segment. A question arises naturally then for song-level classification: how do we obtain a feature representation at song level from features gathered from segments?

For different types of classifiers, feature representation for a song can take one out of three possible forms. It can be a single feature vector for each song, a similarity measure returned for each pair of songs, or a feature set of local feature vectors. For the single vector representation, a straightforward approach is to take the average [1] or median values [3] for each feature attribute over all segments so as to construct a song-level summary feature vector. The resulting feature vector can then be directly exploited by a standard classifier for classification. This was mainly applied to temporal features, for example MuVar and MuCov together with MFCC features [1], [35]. A single feature vector can also be computed for a song by using a global codebook model [108]–[110]. The global codebook is constructed by clustering the local feature vectors and taking the cluster centroids as the codebook vectors. Each local feature vector in the song is then mapped to the nearest codebook vector. After that, a histogram is computed counting the occurrence frequencies of the codebook vectors. The histogram captures the distributions of codebook vectors in each song and can be used directly as the feature vector for the song. Extension to the global codebook model has also been proposed in [110] by using the hierarchical Dirichlet process (HDP) to model the distribution of local feature vectors during codebook construction.

Alternatively, one can define the similarity between any two songs and use the pairwise similarity values for classification. Song similarity is usually evaluated by comparing the distributions of short-term timbre features in each song [111]–[113]. The main reason for focusing on timbre features is the large number of features available for accurate statistical modeling, as compared to relatively fewer long-term features in one song. A probability model is first estimated for each song to model the distribution of these features. The similarity between any two songs can then be established by comparing the estimated probability models based on some divergence criterion such as the Kullback-Leibler (KL) divergence.

Different probability models have been used for distribution modeling in the literature. These include standard parametric models like single Gaussian with full covariance matrix [113], Gaussian mixture models (GMM) [112], and non-parametric models like K-means cluster model [111]. A comparative study conducted by Aucouturier and Pachet [86] have shown that the performance of a simple Gaussian model for genre classification is highly competitive with those of GMM and K-means clustering. This is mostly due to the difficulty in evaluating the similarity between two GMM models. Unlike the single Gaussian model, no closed-form derivation of KL divergence is available for GMM comparisons. A Monte Carlo sampling approach is usually taken for the computation of KL divergence, which is both computationally expensive and numerically unstable. Comparing two cluster models is also difficult, and the earth mover distance is adopted for this purpose in [111].

We can also keep the feature set of local feature vectors for each song and use them directly for classification. Two classifiers that can be used this way are the GMM classifier [70], and the convolutional neural network (CNN) model [114], [115]. Details will be presented in Section III.

III. Music Classification—Classifier Learning and Tasks

In this section, we first review the classification techniques for music classification and annotation. Issues for individual tasks are then discussed. We will focus on the task-specific issues and try to discover certain types of features and classifiers suited for the particular classification task.

A. Classifier Learning

1) Classifiers for Music Classification: We first study the classification techniques for the standard classification problem. This is the setting for the majority of music classification tasks. In standard classification, we are presented with a training data set where each example comes with a label. The purpose is to design a classification rule that can best predict the labels for unseen data.

Classifier design is a standard topic in pattern classification [70]. The common choices of classifiers are K-nearest neighbor (K-NN) [116], support vector machine (SVM) [117], and GMM classifier [70]. Various other classifiers have also been used for different music classification tasks, including logistic regression [10], [39], artificial neural networks (ANN) [4], [6], [50], [118], decision trees [5], [48], [85], linear discriminant analysis (LDA) [2], [18], [42], nearest centroid (NC) [18], [28], and sparse representation-based classifier (SRC) [16], [19].

K-NN [116] and SVM [117] are the two most popular classifiers used for both general classification problems and in music classification as well. K-NN uses training data directly for the classification of testing data. The label of the testing instance is predicted by majority voting on the labels of the K nearest instances in the training set. SVM is the state-of-the-art binary classifier based on the large margin principle. Given labeled instances from two classes, SVM finds the optimal separating hyperplane which maximizes the distance between support vectors and the hyperplane. The support vectors are those instances closest to the hyperplane whose labels are most likely to be confused. Therefore, the SVM has good classification performance since it focuses on the difficult instances. Both K-NN and SVM are applicable to single feature vector representations and pairwise similarity values as well. In the latter case, a kernel matrix is built from pairwise similarity values that can be used directly by the SVM [36].

The use of GMM as a classifier should not be confused with its use for modeling the timbre features as discussed in Section II-D. In the latter case, GMMs are used for song-level similarity computation. This is different from classifier learning. For the GMM classifier, we fit the Gaussian mixture model over the distributions of song-level features in each class. With

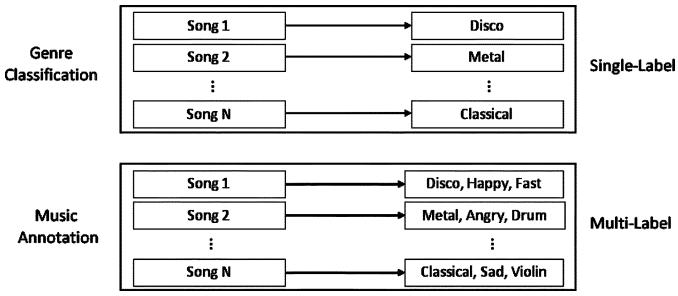

Fig. 3. Genre classification versus music annotation as single-label versus multi-label learning problems.

the class conditional probability distribution, a testing example can be labeled according to the following Bayes rule

$$f(x) = \arg\max_j P(y = j|x)$$
$$P(y = j|x) = \frac{P(x|y = j)P(y = j)}{\sum_j P(x|y = j)P(y = j)}. \quad (1)$$

The decision is based on the maximizer of the posterior probability $P(y|x)$(y specifies the labels, x specifies the data). $P(x|y)$ specifies the conditional probability of example x for class label y estimated from the training data using GMM, and $P(y)$ is the prior probability specifying the proportion of label y in the training data. Specifically, GMM classifier can be used for feature set input, too. By assuming that timbre features in each class are independent and identically distributed, we can apply the product rule to estimate the class conditional probability for feature sets.

Another classifier that can directly handle feature set classification is convolutional neural network (CNN) [114], which is a generalization of the standard neural network model by taking convolutions over the segments of the input signal. Hence, such model can be used for audio classification based on sequence of timbre features like raw MFCC features. This is demonstrated in [115] with applications on general audio classification using a convolutional deep belief network (CDBN), an extension of CNN with multiple layers of network.

2) Classifiers for Music Annotation: Music annotation is a general classification task. In standard classification, each instance has a single label only. For instance, in music genre classification, each song is assigned to a single genre class, as illustrated on the top panel of Fig. 3. This is not necessarily the case for some real-world classification problems, where an instance may be related to multiple categories. Music annotation is one such example. In music annotation, a song is described by many semantically meaningful tags, which can be anything from genre, to mood, to the style of music. The purpose of music annotation is to label new songs with all relevant tags. This can be naturally cast as a multi-label learning (MLL) problem [119].

MLL is a classification paradigm that deals with assignment with multiple labels. It was initially developed for text mining [120] and image classification [119] and was recently applied to music annotation [55]. There are two approaches to solving MLL problems. The first approach converts the original problem to multiple binary classification problems, and is classifier independent. A typical example is binary relevance (BR) [119]. A BR framework learns a separate binary classifier for each label class by treating instances from the label class as positive instances and the remaining instances as negative ones. The learned binary classifiers are then used for prediction. The second approach is algorithm dependent which adapts existing classification techniques to the setting of MLL, such as MLL versions of K-NN [29] and SVM [63], [120].

3) Feature Learning: Another important issue we address here is feature learning. While this may seem like a problem with features, it is actually closely related to classifier learning. This is because the purpose of feature learning is to automatically select and extract features for improving the classification performance [5] over common audio features obtained following the standard pipelines as described in the previous section.

There is a subtle difference between automatic feature selection and extraction. In the former case, features are directly selected from a large number of candidate input features based on some feature selection rules [5], [11]. For feature extraction, features are obtained from transformations of the input features based on some feature mapping or projection rule [115], [121]. Feature selection/extraction can be done in either supervised or unsupervised fashion. In the supervised setting, labeled data are used to assist the selection or extraction of useful features that best discriminate between different labels [121]. One possible approach for feature selection is to learn a front-end classifier like logistic regressor, which can be trained efficiently, and rank the attributes based on the classifier weights [11]. The lowest ranked feature attributes are then discarded in training the final classifier. Alternatively, we can perform linear feature extraction by learning a transformation matrix to project higher dimensional feature vectors to a lower dimensional subspace that preserves most of the discriminant information. This is achieved by various metric learning algorithms found to be useful for feature learning in music classification [121]. An important metric learning method useful for genre classification is linear discriminant analysis (LDA) [18], which finds the optimal transformation by maximizing between-class scatter while minimizing intra-class scatter. Unsupervised feature extraction methods process input features based on modeling the underlying structure of the audio signal without making use of the label information [19], [115], [122]. A standard method for unsupervised feature extraction is principal component analysis (PCA), which projects the input features to a lower dimensional space that maximally preserves the covariance. PCA is normally used as the postprocessing step for decorrelation in the extraction of standard timbre features like OSC. Non-negative matrix factorization (NMF) [122] provides an alternative approach to unsupervised feature extraction, which aims at obtaining a factorization of the matrix of feature vectors into the product of two low rank matrices with non-negative entries. NMF can recover lower dimensional features with non-negative feature values. This is quite useful for music feature representation and is found to deliver good empirical performance for genre classification [122]. An extension of NMF to tensors, called non-negative

TABLE II
PERFORMANCE COMPARISON OF GENRE CLASSIFICATION ALGORITHMS ON THE STANDARD GTZAN DATA SET

Reference	Features	Classifier	Accuracy (%)
[1]^a	{STFT+MFCC}×MuVar+beat+pitch	K-NN	60
[1]^b	{STFT+MFCC}×MuVar+beat+pitch	GMM	61
*[79]	{MFCC}×FP	SVM	77.7 ± 2.8
*[113]^a	{MFCC}×GMM	K-NN	70.6 ± 3.0
*[113]^b	{MFCC}×GMM	SVM	70.4 ± 3.1
[12]^a	{STFT+MFCC}×MuVar+beat+pitch	SVM	72 ± 5.1
[12]^b	{STFT+MFCC}×MuVar	SVM	71.8 ± 4.8
[12]^c	DWCH+STFT+MFCC×MuVar	SVM	78.5 ± 4.1
*[35]	MFCC×MuCov	SVM	78.6 ± 2.4
[84]	STFT+MFCC×MuVar^2	SVM	79.8
[9]	STFT+FFT+MFCC+LPC	AdaBoost.DT	82.4
[16]	CR×NTF	SVM	78.2 ± 3.8
[18]	{MFCC+ASE+OSC}×FP×LDA	NC	90.6 ± 3.1
[19]	CR×NTF	SRC	92.4 ± 2.0
[123]	{MFCC+ASE+OSC}×{MuCov,FP}+beat+chord	SVM (MKL)	90.4
[123]	{MFCC+ASE+OSC}×{MuCov,FP}+beat+chord	SVM (SG)	90.9

tensor factorization (NTF), is also used in music genre classification for input tensor features and has demonstrated the best performance when combined with specific features and classifiers [19]. We will mention this approach in more details in Section III-A4.

4) Feature Combination and Classifier Fusion: If multiple features are available, we can combine them in some way for music classification. Feature combination from different sources is an effective way to enhance the performance of music classification systems [12], [13], [39], [123]. A straightforward way to feature combination is to concatenate all features into a single feature vector, as shown in [1] and [12] for combining timbre with beat and pitch features. Feature combination can also be integrated with classifier learning. Multiple kernel learning (MKL) [124] is one such framework developed particularly for SVM classifiers. The purpose of MKL is to learn an optimal linear combination of features for SVM classification. MKL has recently been applied to music classification and found to outperform any of the single feature types [56], [123].

As an alternative to feature combination, we can also perform decision-level fusion to combine multiple decisions from different classifiers. There are many ways to perform decision-level fusion, including majority voting, sum rule which takes the average of decision values returned by individual classifiers, etc. A more general framework is established by the technique of stacked generalization (SG) [125], which provides a cascaded framework for classification by stacking classifiers on top of classifiers. In the SG framework, classifiers at the first level are trained on individual features, and classifiers at the second level are trained by using the decision values returned by level-1 classifiers as new features. Hence, SG obtains the fusion rule through supervised learning. The choice of classifiers used for SG is quite flexible. Normally SVMs are used within SG for optimized performance as in [123] and [126]. Different combination strategies have been studied in [123], showing that SG and MKL achieve the best performances for multi-feature music genre classification, outperforming other existing methods by a significant margin. Another important class of feature combination methods is based on ensemble methods for classification. One such example is AdaBoost with decision trees (AdaBoost.DT) [9], [27], which combines decision tree classifiers with the boosting framework [127]. Each decision tree classifier is trained on a single type of feature.

B. Classification Tasks

Now we review the usefulness of features and classifiers, and discuss current issues and challenges for each individual task listed in Section I.

1) Genre Classification: Genre classification is the most widely studied area in MIR since the seminal work of Tzanetakis and Cook [1]. They also provided a publicly available benchmark data set with 1000 songs in raw audio format evenly distributed in ten genres. Other publicly available data sets for genre classification include the ISMIR 2004 data set[4] and the Dortmund data set [128]. The availability of public data sets has made it possible to compare different approaches for the performance of genre classification on an equal basis. Much work has been done in this area [2]–[9], [11], [12], [14]–[19]. Here, we have listed a few representative methods in Table II and reported the features and classifiers used, and the accuracy rates achieved by each method. The plus sign in the table denotes multiple features used and the multiplication sign denote sequence of operations for feature extraction. For example, "$\{\mathrm{STFT} + \mathrm{MFCC}\}\mathrm{xMuVar} + \mathrm{beat} + \mathrm{pitch}$" in the feature column of the first row means that STFT and MFCC features are extracted at frame level and subsequently processed by MuVar to obtain the temporal feature vector. This vector is then concatenated by the beat and pitch features obtained at song level to produce the final feature vector for classification. Some results in the table are based on our own implementations, and we highlight them by prepending an asterisk in the references shown in the table. Otherwise, the results are based on those reported in the original paper.

It is worth mentioning that Table II by no means covers the full spectrum of available methods for genre classification due to the vast amount of work done in this area. Moreover, the results should be treated with caution due to variations in implementation details even for the same feature. For instance, the top 13 coefficients were used for MFCC feature extracted in [1] and [12], whereas 20 feature coefficients were adopted by [35].

[4]http://ismir2004.ismir.net/genre_contest/index.htm

Despite this, the results reported in the table do reveal some interesting and important information about the usefulness of certain feature types and classifiers for the classification task. MFCC is inarguably the single most important feature type that was widely used for genre classification. It alone can produce quite good classification result [35] compared to many other methods that use more sophisticated collections of features. Other low-level features are also used in combination with MFCC in many systems to further improve the performance, such as STFT [1] [a,b], [12][a,b,c], [9], [84], OSC [18], [123], ASE [18], [123], and DWCH [12]. The importance of mid-level features such as beat and pitch is difficult to judge. They were found to perform poorly for genre classification when used alone, but can be used to complement spectral features to further improve the classification performance [1], [129]. The only exception is the chord feature, which performs quite well for genre classification [57], [123] and can further boost the performance when combining with other features [123]. From Table II, we also note that the integration of timbre features with temporal features are important for genre classification. Performances of MFCC features with MuCov/FP are better than global GMM modeling of MFCC features taken from local frames [113], which does not consider any temporal information. Performances of different temporal features can also vary, even used with the same timbre feature. For example, $\mathrm{MFCC} + \mathrm{MuCov}$ [35] and $\mathrm{MFCC} + \mathrm{FP}$ [79] perform much better than $\mathrm{MFCC} + \mathrm{MuVar}$ [1]. This is understandable, since MuCov takes the mean and covariance over frame-level MFCC features and FP performs further spectral analysis by treating the frame-level MFCC features as new time series data. Both of them are more sophisticated than MuVar, which only uses the mean and variance of feature values across different frames. Taking MuVar operation twice, i.e., MuVar^2[84], to MFCC features can improve over $\mathrm{MFCC} + \mathrm{MuVar}$ combination.

The best performance for genre classification on the GTZAN benchmark dataset is achieved by Panagakis *et al.* [19] using cortical representation (CR) features inspired by the human auditory system [130]. Raw CR features are formulated as tensors in very high dimensions and subsequently processed by NTF to produce the lower-dimensional feature vector for classification [16]. In [19], the SRC was employed for the classification of processed CR features yielding top performance for genre classification with average accuracy rate of 92.4% from ten-fold cross validation. The performance boost is largely due to the classifier used. When the SVM classifier was applied to the same features (e.g., [16], [19]), it produced lower accuracy rates. SRC is a novel classification technique by exploiting the sparsity in feature representation. It was initially employed for face recognition in [131], where more details on SRC can be found. The use of SRC in [19] is the first time that it was applied to a music classification task. Whether SRC is a better classification tool than SVM for music genre classification with other audio features still requires further examination. Despite this, SVM is still among the top performing classifiers and has been used predominantly in music classification and consistently outperformed other standard classifiers like K-NN and GMM.

Another classification tool that has been recently explored for music genre classification is CDBN [115], which can be applied to the feature set of MFCC features directly. The results in [115] showed that CDBN outperforms standard SVM when using the same MFCC feature. This is especially true for level 2 CDBN with two hidden layers and the cases of fewer labeled instances, which justifies the representation power of CDBN and the use of deep learning for audio classification, since SVM is basically a shallow single layer network. Nevertheless, considering that the data set used in [115] is a non-standard one and a 3-s segment is chosen from each song for testing the classification performance, further tests on benchmark data sets using the standard evaluation procedure are needed to justify the utility of CDBN.

Despite the use of CR feature only for the top performing method in [19], better classification results are usually achieved by combining multiple feature types obtained from different channels. This has been demonstrated by various methods on feature combination [9], [18], [123]. Moreover, since different classifiers were utilized by these methods, we can see that the improvement in performance is likely to be contributed by the features used. This further justifies the usefulness of feature combination regardless of the classifiers being used. Different SVM-based combination schemes were compared in [123] with the same feature set consisting of both low-and mid-level features, which shows that SG [125] and MKL [124] are the two most effective combination methods for SVM classifiers in genre classification.

2) Mood Classification: Mood classification is another important application area in MIR. The purpose is to classify songs into different emotional categories like happy, sad, and angry. Despite its importance, it is quite difficult to evaluate the performance of mood classification algorithms. This is due to the following two main obstacles. The first is the lack of publicly available benchmark data sets. Different researchers have used their own data sets, which makes it very difficult to compare different methods on an equal basis. Much effort has been made to maintain a standardized data set for MIREX mood classification contest in recent years. The second and more serious main obstacle for mood classification is the difficulty in obtaining the ground truth. Unlike genres and artists, mood definitions are quite vague and much dependent on individual taste and choices. Different ground truth models have been developed by psychologists and used by MIR researchers, including Hevner's model [132], Thayer's model [133], and the TWC model [134]. These are, however, from subjective judgment and lack general support. "Democracy" is one way to obtain objective ground truth information from majority voting of individual opinions by exploring the mood metadata and collecting opinion polls from users, critics, and fans via collaborative filtering [135]. However, as noted in [136], performance evaluation is still influenced by various effects in the data creation and evaluation process.

Despite the problem with the objectiveness of mood classification, a lot of research has been done in mood classification. The features and classification methods used for mood classification have a lot in common with genre classification, with much more emphasis on low-level spectral features [12], [20], [23], [25]. There are some noticeable differences in the use of specific feature types though. First, rhythmic features have been used a lot in many mood classification systems [21], [22], [27], [83]. Whether rhythmic information alone can yield good mood

classification performance is still under debate; it does play a much more important role in mood classification as shown by [21] and [83] than genre and other classification tasks. Articulation-based features, which measure the smoothness or continuity in the transition between multiple sounds in the song and were never used for other classification tasks, have also been used in some mood classification algorithms [28], [83] along with other features. This is based on the consideration that the transition between notes in a song with placid emotions (happy, sad) would be more smooth than the transition in a song describing strong emotions (angry) [83]. Moreover, happy or sad songs are usually perceived with slow tempos, whereas angry songs usually have faster tempos [83]. These findings may support the use of rhythm and articulation features for mood classification. For low-level spectral features, Jiang *et al.* [75] proposed the octave-based spectral contrast (OSC) feature for mood classification, which captures the relative contrast information in each spectral subband. It has shown in many mood classification systems [25], [27], [75] that OSC is more preferable to MFCC for music classification with better empirical performance.

It is also worth mentioning that mood classification is more natural as an MLL problem. This is because a song can exhibit different moods at different time. In [2], Li *et al.* first pointed out the importance of MLL for mood classification and proposed a simple MLL algorithm for doing so by converting multi-label classification into multiple binary classification problems. More sophisticated MLL algorithm is proposed in [29], which adapts the standard K-NN classifier to the multi-label setting. with the multi-label classification of music emotions.

3) Artist Identification: We address the problem of artist identification in this subsection. This involves several related subtasks—artist identification, singer recognition, and composer recognition. Since different artists, singers, and composers have their distinctive styles of performing, singing, and composition, it is possible to distinguish songs performed/sung/written by different artists/singers/composers based on the musical styles reflected in song audio, which is the purpose for general audio-based artist identification. Mandel and Ellis proposed an SVM-based method for artist identification by using song-level features aggregated from the low-order statistics of MFCC coefficients [35]. They also released the USPOP2002 data set and made the MFCC features publicly available, which includes a total of 706 albums and 8764 songs from 400 artists in popular music. Their method, with little adaptation, can also be applied to composer identification and performed well in composer identification competitions in MIREX. Hence, the method is more for style identification than singer identification.

Following an alternative line of research, many methods for singer recognition have also been proposed in the past few years [31], [34], [37]–[39]. The main difference that distinguishes singer recognition from artist style identification is in feature extraction. Most singer recognition systems apply a preprocessing segmentation step to segment the audio signal into vocal and instrument part, and then perform classification based on audio features extracted from the vocal part of the signal [33], [37], [38]. This separate approach has been demonstrated to be effective for singer recognition. Alternatively, features extracted from both vocal and instrument segments can be combined for improved recognition performance, as different singers have distinctive music style that may influence the instrument accompaniment [39].

Vocal/nonvocal segmentation is itself a classification problem that can be solved by modeling the distributions of audio features extracted from the training segments for each class [32], [34]. HMM models are often used for the labeling of sequences of segments. Audio features used for segmentation and subsequent classification include standard MFCC features [31], [33], [37], LPCC features [34], [39] and harmonic features [34], and novel cepstrum-based features that model the vibrato of singers [38].

The performance of artist identification is also heavily influenced by the album effect [35], [38], [39]. Songs from the same album are more similar to each other compared to songs from a different album even for the same singer/artist, which leads to an overestimate of classification accuracy if the testing data set contains songs from the same album as those in the training data set. To avoid album effect and ensure fair performance evaluation, the training and testing data sets should not share songs from the same album.

4) Instrument Recognition: Instrument recognition is another interesting classification problem in MIR. The purpose is to identify the types of instruments playing at different intervals in the raw audio. Unlike classification of genres, moods, and artists, instrument recognition is rather a sequence labeling problem, where classification is done at segment level to assign a single instrument label (for solo music) or multiple instrument labels (for polyphonic music) to each segment of the song. Early research on instrument recognition is focused on recognition of instrument from solo music with a single instrument playing at the same time [40]–[44], which is somehow not practical for real music performances. Current research efforts have shifted from solo to polyphonic music that deals with multiple instruments simultaneously [45]–[51].

One big challenge is the huge number of combinations of instrumentation encountered in polyphonic music. Essid *et al.* [45] proposed an automatic method to discover such combinations of instruments in Jazz music by using hierarchical clustering, with each cluster center defining an instrument class. A standard classification algorithm is then taken by making use of timbre and temporal features to classify each audio segment to one of the clusters. Alternatively, one can simply identify the presence/absence of a single instrument in the signal of instrument mixture [48]. Thus, the problem of combinatorial explosion in determining all instruments in the mixture is effectively sidestepped. From an alternative viewpoint, this is equivalent to posing instrument recognition as a multi-label classification problem, where multiple instrument labels can be assigned to each segment. It would be interesting to address instrument recognition from the perspective of multi-label learning so that the methods for music annotation as discussed in Section IV can be readily used. Whether multi-label learning can be used to improve the performance of instrument recognition is still an open question though. Another problem that complicates instrument recognition from raw music audio is source interference, where

TABLE III
PERFORMANCE COMPARISON OF MUSIC ANNOTATION ALGORITHMS ON CAL500 DATA SET

Method	Features	Classifier	Precision-At-10	AUC
MixHier [54]	dMFCC	GMM	0.265 ± 0.007	0.710 ± 0.004
Autotag [58]	dMFCC	AdaBoost	0.281	0.678
CBA [65]	dMFCC	CBA	0.286	0.765
AudioSVM [126][a]	dMFCC$\times$MuVar^2	SVM	N/A	0.78
AffinitySVM [126][b]	dMFCC$\times$MuVar^2	SVM+SG	N/A	0.85

the louder instrument masks other instruments at overlapping partials when they are played simultaneously. This was also addressed in the literature [46], [51]. Whether source separation methods [137] can aid and improve the performance of instrument recommendation is still an open question.

5) Music Annotation: Music annotation is a paradigm in MIR pioneered by the seminal work of Slaney *et al.* [52] It provides an indirect approach to overcome the semantic gap for content-based music retrieval. Each song is labeled with semantically meaningful text annotations called tags. Tags can be any text that describes the music given by the listeners, like genres, moods, artists, styles, etc. The purpose of music annotation is to find a mapping from the audio content to the tags of text annotations. This not only enables us to label new songs with appropriate tags, but more importantly, helps to convert the music retrieval problem to text retrieval by substituting songs with text annotations. This enables us to search for relevant music based on text queries. In this way, the two seemingly different tasks can be unified under the same framework, namely music annotation and query by text, the latter being the inverse problem of the former.

To achieve music annotation, the relationships between the audio content and text annotations need be modeled first. Turnbull *et al.* [54], [55] used a hierarchical Gaussian mixture model to describe the audio features extracted from music segments for each individual tag. Both retrieval and annotation can be achieved via finding the songs/tags that maximize the conditional probability model. They collected a 500-song data set (CAL500) with human annotated ground truth to test their algorithm and made the data set publicly available as a testbed. Table III lists the performance of some annotation algorithms on the CAL500 data set along with the features and classification techniques used by these methods. Standard evaluation metrics for multi-label learning have been adopted to measure the performance of music annotation, including average Area Under ROC (AUC) values over different tags as well as average precision for each song by keeping ten labels (precision-at-10) in label ranking. Feature "dMFCC" in the table denotes delta-MFCC feature comprising standard MFCC feature along with first-and second-order differences. Since music annotation is an emerging research area in MIR, not many methods have been developed and most current methods make predominant use of delta-MFCC features. The way MFCC features are organized, however, is an important issue here. Frame-level MFCC features are used in [55] and [65] and modeled using the GMM model and global codebook model, respectively, whereas Ness *et al.* used summary MFCC features [126] by taking the mean and variance of the frame-level MFCC features (MuVar) and achieved better annotation results in terms of higher AUC value. Nevertheless, the SVM classifier is used in [126] as compared to the simpler classifiers used in [55] and [65]. This further complicates the issue as it is hard to judge whether the improvement is contributed by the classifier or feature representation. There is still much room for improvement in the features used for annotation, with open questions like whether a more sophisticated feature integration method like MuCov or FP can further improve the performance. We conjecture that the results from the large amount of work on audio features for genre classification may have some implications on how to obtain better features for music annotation.

The classification problem in music annotation is somehow different from that in standard classification tasks, like genre and mood. The multi-label nature of the annotation problem makes generative approaches more pertinent. Generative approaches model the distributions of each class directly and produce a ranking for the each tag, which can be used to determine the relevance between each song and tag. A natural choice here is the GMM model that models the distributions of frame-level MFCCs for each label class via a mixture of Gaussian [55]. Different probability models can also be used to model the two-way relations between music and tag labels, like the simpler yet more effective codeword Bernoulli average (CBA) model based on the global codebook representation of the MFCC features for each tag [65]. Alternatively, one can also consider the distribution of audio features for both words and anti-words, the complement of annotation words [57]. This leads to a discriminant supervised learning framework for text-based query using both words and anti-words.

One important issue with multi-label learning is the correlations between labels. For music annotation where labels are taken from a large set of tags describing the genre, mood, style, and instrument of the songs, we would expect that these tags are highly correlated to one another. For instance, a song annotated with genre of "rock" may also likely to have instrument labels of "guitar" and "drum". On the other hand, it is very unlikely for a song to have the labels "disco" and "violin" at the same time. There are two ways to take label correlations into account. Firstly, we can adopt SG [125] by training a second-level classifier on top of the decision values returned by the initial classifier trained on individual labels. It has been shown in [126] and [138] that SG can further improve the annotation performance over the initial classifiers by a significant margin. Alternatively, music annotation can also naturally be cast directly as a ranking problem. Discriminative ranking models can be trained by directly considering the rank orders for different labels [59], [63], [139]. The standard SVM can be adapted to incorporate rank information and solve ranking problems [120], [139]. The formulation of rank SVM [120] is much more complicated compared

to the standard version and may not be suited for large-scale annotation problems. It would be interesting to develop efficient methods for ranking directly.

IV. Research Issues

In this section, based on our overview of the techniques and tasks relevant to music classification, we identify and discuss four open issues in music classification research that require further investigation in the future. These issues are outlined in Sections IV-A–IV-D.

A. Large-Scale Content-Based Music Classification With Few Labeled Data

Current music classification systems are mostly tested on small or median size data sets with thousands of songs. Research has been focused on improving the classification performance rather than efficiency. Modern music industry is growing rapidly. To keep track with the ever-increasing number of new songs made available everyday by both professionals and anonymous authors, we need faster music analysis and classification systems capable of handling large-scale data sets with millions of songs. This poses two main challenges to the current techniques for music classification. Scalability is the first important issue regarding both processing time and storage. Feature extraction for music audio analysis is a quite time-consuming process. The extraction of standard audio features, like MFCC, pitch, and rhythmic features, requires applying spectral transformation like FFT to local frames as the preliminary step. This is quite costly even for songs with moderate lengths. In order to accelerate large-scale music classification tasks, faster preprocessing procedures are needed to streamline feature extraction. Storage is also an issue for scalability. With the growing size of music data, a compact song-level representation is needed to reduce the space complexity for audio processing and analysis and prevents database overflow. Besides scalability issues, the second main challenge for large-scale music classification systems is the difficulty in gaining ground truth information. It is well-known that the performance of a classification system heavily depends on the amount of data used for training. However, before we can make use of the data to improve classification performance, we need ground truth labels for the songs in the training data set. This may be a problem for tasks like mood classification and instrument recognition, which require extensive effort for the acquisition of labels. Hence, it would be most desirable to relieve from the dependence on ground truth labels and be able to train the music classification system for large-scale data sets from few labeled data.

Few research efforts have been made to address large-scale music classification. From our point of view, two novel machine learning paradigms, namely online learning [140] and semi-supervised learning (SSL) [141], provide techniques suitable for tackling the two main challenges mentioned above. Online learning [140], in contrast to traditional batch learning methods for classification, provides a general framework for incrementally training the classifier as training instances are presented in the data stream. Each training instance is processed only once and then discarded by the online learning system. While online learning requires labeled data to proceed, SSL provides a general framework for training a classification system from scarcely labeled data [141]. The idea is to use unlabeled data to provide additional information on the distributions of data in each class and guide the learning process. In this way, the necessity for labeled data as in standard supervised learning can be relaxed. The technique of SSL was employed for music genre classification [17] and instrument recognition [142], showing that SSL can achieve comparable performance with classifier learned in the supervised manner while using much fewer examples in training. By combining SSL with online learning, it is possible to train the music classification systems more efficiently and effectively without heavy dependence on labeled data. The combination is a hard machine learning problem which has not yet been investigated, but it provides a possible direction for future research and one solution to overcome the shortcomings of current music classification systems.

B. Music Mining From Multiple Sources

With the ever-growing music industry and expansion of fan community on the Internet, it is quite convenient to gain information about songs through multiple sources nowadays. Information gained from alternative sources can be utilized to facilitate and complement audio-based music classification. Web mining is an important channel to gain additional information of music. We can search music by text using song titles, artists, and styles on text-based search engines. Another important type of information gained from web mining is social tags, which are text descriptions and annotations of online music data by musical fans. Social tags are widely available on music recommendation websites like *last.fm*[5] and *MOG*.[6] They provide rich information on the style and characteristics of the music work that can be utilized for music classification and retrieval [53]. This is especially important for music annotation, as social tags directly reveal the label information for the music. However, we should emphasize that social tags do not necessarily equate with ground truth, as they are provided by volunteers and fans reflecting their personal opinions and judgments, and thus are inevitably biased and noisy. Combining social tags with audio features can further improve the performance of content-based systems and is currently a hot research topic in MIR [60]–[63]. Besides web mining and social tags, collaborative filtering is another source for additional information on music. It discovers useful information from user playlists by studying the correlations between songs in different playlists. Collaborative filtering can be used in combination with social tags and audio features for music annotation, as demonstrated in [64]. Another important application of social tags and collaborative filtering is to help validate the ground truth such as the case for mood classification [30], [135], where the definition of ground truth is quite vague. Overall, music analysis from multiple sources enables us to better understand the music and motivates novel paradigms for music annotation and retrieval.

A main assumption held by current systems on music annotation based on social tags and collaborative filtering is that these

[5]http://www.last.fm

[6]http://mog.com

additional sources of information can be conveniently obtained. However, this is not necessarily the case for testing songs in the data set because we need song-level metadata like titles and artists to retrieve social tags and other sources of information from text retrieval. Hence, an important question for future research is how to get social tag information for testing songs. A possible way to do this is through recursive classifier learning, by iteratively performing the two steps of classifier learning and web mining using predicted labels from the learned classifier.

C. Learning Music Similarity Retrieval

An important problem in MIR not directly related to music classification is similarity retrieval. Given a query song, the purpose of similarity retrieval is to retrieve similar songs in the data set. Depending on the definition of similarity, this can result in very different songs being returned by the retrieval system. While early retrieval systems rely predominantly on timbre similarity [112], [113], [118], some applications require different forms of similarities being used. For instance, cover song detection [92], [93] and query by example/humming [89], [90] more or less require similarity comparison to be evaluated at melodic and harmonic level. On the other hand, timbre-based retrieval systems were not used for such applications.

The fact that we need different similarities for different tasks motivates a new scenario of similarity retrieval based on learned similarity. Standard similarity retrieval is an unsupervised learning problem which does not involve any classifier learning. By making use of exemplar pairs of similar songs and dissimilar songs, we can pose similarity retrieval as a binary classification problem. The purpose is to learn a classifier that can predict whether a pair of songs are similar. This idea has been employed lately in [143] for cover song detection and is found to achieve superior performance than traditional systems with predefined similarity measure based on the affinity of melody. It is possible to employ a similar idea for content-based music search engines, while the data needed for similarity learning can be initially acquired from text retrieval results, with songs co-occurring in the same query being deemed as similar pairs. Another application for supervised similarity retrieval is in learning user preference. This is especially useful for improving over initial query results. There are two schemes particularly suited for this task—relevance feedback and active learning. Relevance feedback is a user feedback mechanism widely used in image retrieval [144]. In the relevance feedback framework, the retrieval process is repeated for several iterations. After each iteration, query results from the previous iteration are reviewed by the user. The feedbacks from the user are then used to refine the similarity measure, which is used to create new query results for the next iteration. Active learning is another effective technique to model user preference [36], [144], which applies supervised learning to relevance feedback. In active learning, after each iteration, pairs of similar/dissimilar songs are added to the initial training data for similarity learning according to user feedback. New similarity metric is learned by updating the classifier with augmented training set. Then example pairs that are most likely to be confused by the current classifier are picked and presented to the user for future rounds of feedback and classifier learning.

D. Perceptual Features for Music Classification

The issues we discussed above are more related to the learning and classification techniques for music classification. Although classification is a very important aspect of music classification research, audio features also play an important role in content-based music classification systems. The dominant features used in current classification systems are low-level features. By the terminology of [69], low-level features have high specificity, which describe the characteristics of the audio signal at very high resolution and can immediately reflect the subtle changes in the signal. Hence, low-level features should be best suited for high-specificity tasks to identify the exact content of audio recordings, like fingerprinting for near-duplicate song detection and copyright monitoring. In contrast, music classification systems are low-specificity systems focusing on the broader nature of music like genre and mood that are not directly conveyed by the low-level signal. Despite good empirical performance delivered by the low-level features in current music classification systems, mid-level features describe the music content at a more abstract level and should be better suited for low-specificity systems of music classification.

There are two general approaches to mid-level feature extraction, respectively, through models of music and models of auditory perception and cognition. Most research efforts have been made in the first approach via explicit extraction of features such as rhythm, pitch, and harmony from audio signals based on the general knowledge and understanding of music [1], [104], [106], [129], [145]. The mid-level features obtained through music modeling are likely to improve the performance of the classification system when combined with low-level features, yet fail to deliver satisfactory performance when used alone [1], [12], [129]. This is most likely due to the difficulty in obtaining such features reliably as we need to resort to heuristics for feature extraction on top of spectral analysis. Alternatively, recent research on perceptual features based on human auditory and neural models have seen success in both general audio classification [115], [146] and music genre classification [16], [19]. These include cortical representations inspired by auditory model [16], [19], sparse coding model consistent with the mechanism of information storage in human brain [146], and convolutional network model that mimics the process of information processing by the neural system [114], [115]. Note that as a classification model, the convolutional neural network contains many hidden layers with many nodes that act like processing units from which the perceptual features are processed. Hence, the training of a convolutional neural network is an integrated process of learning perceptual features and classification rule. In contrast to mid-level musical features, perceptual features are more relevant to how human perceives and processes music in human auditory and neural systems and thus may better characterize music at a higher level of abstraction important for low-specificity classification tasks. There is great potential for the use of perceptual features for music classification, which has not yet been fully investigated and would be one of the interesting directions for future research.

V. Conclusions

In this survey, we have reviewed the recent development in music classification and annotation. The survey has provided an up-to-date discussion of audio features and classification techniques used in the literature. In addition, we have also reviewed individual tasks for music classification and annotation and identified both task-specific issues and general open problems that require further investigation in the future.

To identify the limit of a music classification system, some work has been done on comparing the performance of automatic genre classification with human performance [147]–[149]. In [147], it was found that humans have a strong ability to identify genre classes and correct decisions can be made within a short time span of 10–100 ms. Gaps between human performance and that of the genre classification system were observed in [148] and [149]. From those evidences on genre classification, it can be seen that there is still much room for further improvement over current automatic music classification systems.

References

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

[2] T. Li, M. Ogihara, and Q. Li, "A comparative study of content-based music genre classification," in *Proc. SIGIR*, 2003.

[3] T. Lidy and A. Rauber, "Evaluation of feature extractors and psychoacoustic transformations for music genre classification," in *Proc. Int. Conf. Music Information Retrieval*, 2005.

[4] A. Meng and J. Shawe-Taylor, "An investigation of feature models for music genre classification using the support vector classifier," in *Proc. Int. Conf. Music Information Retrieval*, 2005.

[5] I. Mierswa and K. Morik, "Automatic feature extraction for classifying audio data," *Mach. Learn.*, vol. 58, pp. 127–149, 2005.

[6] N. Scaringella and G. Zoia, "On the modelling of time information for automatic genre recognition systems in audio signals," in *Proc. Int. Conf. Music Information Retrieval*, 2005.

[7] D. Turnbull and C. Elkan, "Fast recognition of musical genres using RBF networks," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 4, pp. 580–584, 2005.

[8] K. West and S. Cox, "Finding an optimal segmentation for audio genre classification," in *Proc. Int. Conf. Music Information Retrieval*, 2005.

[9] J. Bergstra, N. Casagrande, D. Erhan, D. Eck, and B. Kegl, "Aggregate features and ada boost for music classification," *Mach. Learn.*, vol. 65, no. 2–3, pp. 473–484, 2006.

[10] F. Morchen, A. Ultsch, M. Thies, and I. Lohken, "Modeling timbre distance with temporal statistics from polyphonic music," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 1, pp. 81–90, 2006.

[11] F. Mochen, I. Mierswa, and A. Ultsch, "Understandable models of music collections based on exhaustive feature generation with temporal statistics," in *Proc. ACM SIGKDD*, 2006.

[12] T. Li and M. Ogihara, "Toward intelligent music information retrieval," *IEEE Trans. Multimedia*, vol. 8, no. 3, pp. 564–574, 2006.

[13] J. Shen, J. Shepherd, and A. Ngu, "Towards effective content-based music retrieval with multiple acoustic feature combination," *IEEE Trans. Multimedia*, vol. 8, no. 6, pp. 1179–1189, 2006.

[14] T. Lidy, A. Rauber, A. Pertusa, and J. Inesta, "Improving genre classification by combination of audio and symbolic descriptors using a transcription system," in *Proc. Int. Conf. Music Information Retrieval*, 2007.

[15] A. Meng, P. Ahrendt, and J. Larsen, "Temporal feature integration for music genre classification," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 5, pp. 1654–1664, 2007.

[16] I. Panagakis, E. Benetos, and C. Kotropoulos, "Music genre classification: A multilinear approach," in *Proc. Int. Conf. Music Information Retrieval*, 2008.

[17] Y. Song and C. Zhang, "Content based information fusion for semi-supervised music genre classification," *IEEE Trans. Multimedia*, vol. 10, no. 1, pp. 145–152, 2008.

[18] C.-H. Lin, J.-L. Shih, K.-M. Yu, and H.-S. Lin, "Automatic music genre classification based on modulation spectral analysis of spectral and cepstral features," *IEEE Trans. Multimedia*, vol. 11, no. 4, pp. 670–682, 2009.

[19] I. Panagakis, C. Kotropoulos, and G. R. Arce, "Music genre classification using locality preserving non-negative tensor factorization and sparse representations," in *Proc. Int. Conf. Music Information Retrieval*, 2009.

[20] T. Li and M. Ogihara, "Detecting emotion in music," in *Proc. Int. Conf. Music Information Retrieval*, 2003.

[21] Y. Feng, Y. Zhuang, and Y. Pan, "Music retrieval by detecting mood via computational media aesthetics," in *Proc. Int. Conf. Web Intelligence*, 2003.

[22] D. Yang and W. Lee, "Disambiguating music emotion using software agents," in *Proc. Int. Conf. Music Information Retrieval*, 2003.

[23] M. D. Korhonen and M. J. D. A. Clausi, "Modeling emotional content of music using system identification," *IEEE Trans. Syst., Man, Cybern.*, vol. 36, no. 3, pp. 588–599, 2006.

[24] Y.-H. Yang, C.-C. Liu, and H. H. Chen, "Music emotion classification: A fuzzy approach," in *Proc. ACM Multimedia*, 2006.

[25] L. Lu, D. Liu, and H.-J. Zhang, "Automatic mood detection and tracking of music audio signals," *IEEE Trans. Speech Audio Process.*, vol. 14, no. 1, pp. 5–18, 2006.

[26] W.-L. Cheung and G. Lu, "Music emotion annotation by machine learning," in *Proc. Int. Workshop Multimedia Signal Processing*, 2008.

[27] Y.-H. Yang, Y.-C. Lin, Y.-F. Su, and H. H. Chen, "A regression approach to music emotion recognition," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 16, no. 2, pp. 448–457, 2008.

[28] L. Mion and G. D. Poli, "Score-independent audio features for description of music expression," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 16, no. 2, pp. 458–466, 2008.

[29] K. Trohidis, G. Tsoumakas, G. Kalliris, and I. Vlahavas, "Multi-label classification of music into emotions," in *Proc. Int. Conf. Music Information Retrieval*, 2008.

[30] C. Laurier, M. Sordo, J. Serra, and P. Herrera, "Music mood representations from social tags," in *Proc. Int. Conf. Music Information Retrieval*, 2009.

[31] B. Whitman, G. Flake, and S. Lawrence, "Artist detection in music with minnowmatch," in *Proc. IEEE Workshop Neural Networks for Signal Processing*, 2001.

[32] A. L. Berenzweig and D. P. W. Ellis, "Locating singing voice segments within music signals," in *Proc. IEEE Workshop Applications of Signal Processing to Audio and Acoustics*, 2001.

[33] A. L. Berenzweig, D. P. W. Ellis, and S. Lawrence, "Using voice segments to improve artist classification of music," in *Proc. Int. Conf. Virtual, Synthetic and Entertainment Audio*, 2002.

[34] Y. E. Kim and B. Whitman, "Singer identification in popular music recordings using voice coding features," in *Proc. Int. Conf. Music Information Retrieval*, 2002.

[35] M. Mandel and D. Ellis, "Song-level features and SVMs for music classification," in *Proc. Int. Conf. Music Information Retrieval*, 2005.

[36] M. Mandel, G. Poliner, and D. Ellis, "Support vector machine active learning for music retrieval," *Multimedia Syst.*, vol. 12, no. 1, pp. 3–13, 2006.

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

[38] T.-L. Nwe and H. Li, "Exploring vibrato-motivated acoustic features for singer identification," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 2, pp. 519–530, 2007.

[39] J. Shen, J. Shepherd, B. Cui, and K.-L. Tan, "A novel framework for efficient automated singer identification in large music databases," *ACM Trans. Inf. Syst.*, vol. 27, no. 3, pp. 1–31, 2009.

[40] J. Marques and P. J. Moreno, *A Study of Musical Instrument Classification Using Gaussian Mixture Models and Support Vector Machines*. Cambridge, MA: Cambridge Research Lab., 1999, Tech. Rep.

[41] J. C. Brown, "Computer identification of musical instruments using pattern recognition with cepstral coefficients as features," *J. Acoust. Soc. Amer.*, vol. 105, pp. 1933–1941, 1999.

[42] G. Agostini, M. Longari, and E. Pollastri, "Musical instrument timbres classification with spectral features," *EURASIP J. Appl. Signal Process.*, vol. 2003, no. 1, pp. 5–14, 2003.

[43] S. Essid, G. Richard, and B. David, "Musical instrument recognition based on class pairwise feature selection," in *Proc. Int. Conf. Music Information Retrieval*, 2004.

[44] S. Essid, G. Richard, and B. David, "Musical instrument recognition by pairwise classification strategies," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 4, pp. 1401–1412, 2006.
[45] S. Essid, G. Richard, and B. David, "Instrument recognition in polyphonic music based on automatic taxonomies," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 1, pp. 68–80, 2006.
[46] T. Kitahara, M. Goto, K. Komatani, T. Ogata, and H. Okuno, "Instrument identification in polyphonic music: Feature weighting to minimize influence of sound overlaps," *EURASIP J. Appl. Signal Process.*, vol. 2007, no. 1, pp. 155–155, 2007.
[47] P. Leveau, D. Sodoyer, and L. Daudet, "Automatic instrument recognition in a polyphonic mixture using sparse representation," in *Proc. Int. Conf. Music Information Retrieval*, 2007.
[48] D. Little and B. Pardo, "Learning musical instruments from mixtures of audio with weak labels," in *Proc. Int. Conf. Music Information Retrieval*, 2008.
[49] F. Fuhrmann, M. Haro, and P. Herrera, "Scalability, generality and temporal aspects in automatic recognition of predominant musical instruments in polyphonic music," in *Proc. Int. Conf. Music Information Retrieval*, 2009.
[50] P. Hamel, S. Wood, and D. Eck, "Automatic identification of instrument classes in polyphonic and poly-instrument audio," in *Proc. Int. Conf. Music Information Retrieval*, 2009.
[51] T. Heittola, A. Klapuri, and T. Virtanen, "Musical instrument recognition in polyphonic audio using source-filter model for sound separation," in *Proc. Int. Conf. Music Information Retrieval*, 2009.
[52] M. Slaney, "Semantic-audio retrieval," in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing*, 2007.
[53] M. Levy and M. Sandler, "A semantic space for music derived from social tags," in *Proc. Int. Conf. Music Information Retrieval*, 2007.
[54] D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet, "Towards musical query-by-semantic description using the cal500 data set," in *Proc. ACM Information Retrieval*, 2007.
[55] D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet, "Semantic annotation and retrieval of music and sound effects," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 16, no. 2, pp. 467–476, 2008.
[56] L. Barrington, M. Yazdani, D. Turnbull, and G. Lanckriet, "Combining feature Kernels for semantic music retrieval," in *Proc. Int. Conf. Music Information Retrieval*, 2008.
[57] Z.-S. Chen, J.-M. Zen, and J.-S. Jang, "Music annotation and retrieval system using anti-models," in *Proc. Audio Eng. Soc.*, 2008.
[58] T. Bertin-Mahieux, D. Eck, F. Maillet, and P. Lamere, "Autotagger: A model for predicting social tags from acoustic features on large music databases," *J. New Music Res.*, vol. 37, no. 2, pp. 115–135, 2008.
[59] G. Chechik, E. Le, M. Rehn, S. Bengio, and D. Lyon, "Large-scale content-based audio retrieval from text queries," in *Proc. ACM Multimedia Information Retrieval*, 2008.
[60] M. Levy and M. Sandler, "Music information retrieval using social tags and audio," *IEEE Trans. Multimedia*, vol. 11, no. 3, pp. 383–395, 2009.
[61] P. Knees, T. Pohle, M. Schedl, D. Schnitzer, K. Seyerlehner, and G. Widmer, "Augmenting text-based music retrieval with audio similarity," in *Proc. Int. Conf. Music Information Retrieval*, 2009.
[62] D. Turnbull, L. Barrington, M. Yazdani, and G. Lanckriet, "Combining audio content and social context for semantic music discovery," in *Proc. ACM Information Retrieval*, 2009.
[63] F. Wang, X. Wang, B. Shao, T. Li, and M. Ogihara, "Tag integrated multi-label music style classification with hypergraph," in *Proc. Int. Conf. Music Information Retrieval*, 2009.
[64] B. Tomasik, J. H. Kim, M. Ladlow, M. Augat, D. Tingle, R. Wicentowski, and D. Turnbull, "Using regression to combine data sources for semantic music discovery," in *Proc. Int. Conf. Music Information Retrieval*, 2009.
[65] M. Hoffman, D. Blei, and P. Cook, "Easy as CBA: A simple probabilistic model for tagging music," in *Proc. Int. Conf. Music Information Retrieval*, 2009.
[66] J. H. Kim, B. Tomasik, and D. Turnbull, "Using artist similarity to propagate semantic information," in *Proc. Int. Conf. Music Information Retrieval*, 2009.
[67] N. Scaringella, G. Zoia, and D. Mlynek, "Automatic genre classification of music content—A survey," *IEEE Signal Process. Mag.*, vol. 23, no. 2, pp. 133–141, 2006.
[68] C. Weihs, U. Ligges, F. Morchen, and D. Mullensiefen, "Classification in music research," *Adv. Data Anal. Classificat.*, vol. 1, no. 3, pp. 255–291, 2007.
[69] M. Casey, R. Veltkamp, M. Goto, M. Leman, C. Rhodes, and M. Slaney, "Content-based music information retrieval: Current directions and future challenges," *Proc. IEEE*, vol. 96, no. 4, pp. 668–696, 2008.
[70] R. O. Duda and P. E. Hart, *Pattern Classification*, 2nd ed. New York: Wiley, 2000.
[71] E. Allamanche, J. Herre, O. Hellmuth, B. Froba, T. Kastner, and M. Cremer, "Content-based identification of audio material using MPEG-7 low level description," in *Proc. Int. Conf. Music Information Retrieval*, 2001.
[72] E. Benetos, M. Kotti, and C. Kotropoulos, "Musical instrument classification using non-negative matrix factorization algorithms and subset feature selection," in *Proc. Int. Conf. Acoustics, Speech, Signal Processing*, 2006.
[73] H.-T. Cheng, Y.-H. Yang, Y.-C. Lin, I.-B. Liao, and H. H. Chen, "Automatic chord recognition for music classification and retrieval," in *Proc. Int. Conf. Multimedia Expo*, 2008.
[74] H. G. Kim, N. Moreau, and T. Sikora, "Audio classification based on MPEG-7 spectral basis representation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 5, pp. 716–725, 2004.
[75] D.-N. Jiang, L. Lu, H.-J. Zhang, and J.-H. Tao, "Music type classification by spectral contrast feature," in *Proc. Int. Conf. Multimedia Expo*, 2002.
[76] C. C. Lin, S. H. Chen, T. K. Truong, and Y. Chang, "Audio classification and categorization based on wavelets and support vector machine," *IEEE Trans. Speech Audio Process.*, vol. 13, no. 5, pp. 644–651, 2005.
[77] G. Tzanetakis, R. Jones, and K. McNally, "Stereo panning features for classifying recording production style," in *Proc. Int. Conf. Music Information Retrieval*, 2008.
[78] G. Tzanetakis, L. Martins, K. McNally, and R. Jones, "Stereo panning information for music information retrieval tasks," *J. Audio Eng. Soc.*, vol. 58, no. 5, pp. 409–417, 2010.
[79] E. Pampalk, A. Rauber, and D. Merkl, "Content-based organization and visualization of music archives," in *Proc. ACM Multimedia*, 2002.
[80] E. Pampalk, A. Flexer, and G. Widmer, "Improvements of audio-based music similarity and genre classification," in *Proc. Int. Conf. Music Information Retrieval*, 2005.
[81] L. Macy, Grove Music Online. [Online]. Available: http://www.oxfordmusiconline.com/public/book/omo_gmo.
[82] J. C. Brown, "Calculation of a constant Q spectral transform," *J. Acoust. Soc. Amer.*, vol. 89, no. 1, pp. 425–434, 1991.
[83] B.-Y. Chua, "Automatic extraction of perceptual features and categorization of music emotional expression from polyphonic music audio signals," Ph.D. dissertation, Gippsland School of Info. Tech., Monash Univ., Churchill, Australia, 2007.
[84] G. Tzanetakis, "Marsyas-0.2: A case study in implementing music information retrieval systems," in *Proc. Intelligent Mobile Information Systems*.
[85] K. West, "Novel techniques for audio music classification and search," Ph.D. dissertation, Univ. East Anglia, Norwich, U.K., 2008.
[86] J. Aucouturier and F. Pachet, "Improving timbre similarity: How high is the sky?," *J. Negative Results Speech Audio Sci*, vol. 1, no. 1, pp. 1–13, 2004.
[87] J. Reed and C.-H. Lee, "A study on music genre classification based on universal acoustic models," in *Proc. Int. Conf. Music Information Retrieval*, 2006.
[88] J. Reed and C.-H. Lee, "On the importance of modeling temporal information in music tag annotation," in *Proc. Int. Conf. Acoustics, Speech and Signal Processing*, 2009.
[89] J.-S. Jang and H.-R. Lee, "A general framework of progressive filtering and its application to query by singing/humming," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 16, no. 2, pp. 350–358, 2008.
[90] E. Unal, E. Chew, P. Georgiou, and S. Narayanan, "Challenging uncertainty in query by humming systems: A fingerprinting approach," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 16, no. 2, pp. 359–371, 2008.
[91] E. Gomez and P. Herrera, "The song remains the same: Identifying versions of the same piece using tonal descriptors," in *Proc. Int. Conf. Music Information Retrieval*, 2006.
[92] W. H. Tsai, H. M. Yu, and H. M. Wang, "A query-by-example technique for retrieving cover versions of popular songs with similar melodies," in *Proc. Int. Conf. Music Information Retrieval*, 2005.
[93] J. Serra, E. Gomez, P. Herrera, and X. Serra, "Chroma binary similarity and local alignment applied to cover song identification," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 16, no. 6, pp. 1138–1151, 2008.
[94] T. Tolonen and M. Karjalainen, "A computationally efficient multipitch analysis model," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 6, pp. 708–716, 2000.

[95] A. Klapuri, "Multiple fundamental frequency estimation based on harmonicity and spectral smoothness," *IEEE Trans. Speech Audio Process.*, 2000.

[96] G. Tzanetakis, A. Ermolinskyi, and P. Cook, "Pitch histograms in audio and symbolic music information retrieval," in *Proc. Int. Conf. Music Information Retrieval*, 2002.

[97] T. Fujishima, "Realtime chord recognition of musical sound: A system using common lisp music," in *Proc. Int. Computer Music Conf.*, 1999, pp. 464–467.

[98] E. Gomez, "Tonal description of music audio signals," Ph.D. dissertation, Dept. Technol., Universitat Pompeu Fabra, Barcelona, Spain, 2006.

[99] M. Marolt, "A mid-level melody-based representation for calculating audio similarity," in *Proc. Int. Conf. Music Information Retrieval*, 2006.

[100] G. Poliner, D. Ellis, A. Ehmann, E. Gomez, S. Streich, and B. S. Ong, "Melody transcription from music audio: Approaches and evaluation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, no. 4, pp. 1247–1256, 2007.

[101] D. Ellis and G. Poliner, "Identifying cover songs with chroma features and dynamic programming beat tracking," in *Proc. Int. Conf. Acoustics, Speech and Signal Process.*, 2007.

[102] D. Ellis, "Classifying music audio with timbral and chroma features," in *Proc. Int. Conf. Music Information Retrieval*, 2007.

[103] E. Gomez and P. Herrera, "Estimating the tonality of polyphonic audio files: Cognitive versus machine learning modelling strategies," in *Proc. Int. Conf. Music Information Retrieval*, 2004.

[104] K. Lee, "Automatic chord recognition using enhanced pitch class profile," in *Proc. Int. Computer Music Conf.*, 2006.

[105] J. P. Bello, "Audio-based cover song retrieval using approximate chord sequences: Testing shifts, gaps, swaps and beats," in *Proc. Int. Conf. Music Information Retrieval*, 2007.

[106] A. Klapuri and M. Davy, *Signal Processing Methods for Music Transcription*. New York: Springer, 2006.

[107] J. Jensen, M. Christensen, D. Ellis, and S. Jensen, "Quantitative analysis of a common audio similarity measure," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 17, no. 4, pp. 693–702, 2009.

[108] J. T. Foote, "Content-based retrieval of music and audio," in *Proc. SPIE Multimedia Storage and Archiving Systems II*, 1997, pp. 138–147.

[109] K. Seyerlehner, G. Widmer, and P. Knees, "Frame-level audio similarity—A codebook approach," in *Proc. Int. Conf. Digital Audio Effects*, 2008.

[110] M. Hoffman, D. Blei, and P. Cook, "Content-based musical similarity computation using the hierarchical dirichlet process," in *Proc. Int. Conf. Music Information Retrieval*, 2008.

[111] B. Logan and A. Salomon, "A music similarity function based on signal analysis," in *Proc. Int. Conf. Multimedia Expo*, 2001.

[112] J. Aucouturier and F. Pachet, "Music similarity measures: What's the use?," in *Proc. Int. Conf. Music Information Retrieval*, 2002.

[113] E. Pampalk, S. Dixon, and G. Widmer, "On the evaluation of perceptual similarity measures for music," in *Proc. Int. Conf. Music Information Retrieval*, 2003.

[114] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[115] H. Lee, Y. Largman, P. Pham, and A. Y. Ng, "Unsupervised feature learning for audio classification using convolutional deep belief networks," in *Proc. Advances in Neural Information Processing Systems*, 2009.

[116] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inf. Theory*, vol. 13, no. 1, pp. 21–27, 1967.

[117] B. E. Boser, I. Guyon, and V. Vapnik, "A training algorithm for optimal margin classifiers," in *Proc. ACM Conf. Computational Learning Theory*, 1992, pp. 144–152.

[118] A. Berenzweig, B. Logan, D. Ellis, and B. Whitman, "A large-scale evaluation of acoustic and subjective music similarity measures," in *Proc. Int. Conf. Music Information Retrieval*, 2003.

[119] M. Boutell, X. Shen, J. Luo, and C. Brown, "Learning multi-label semantic scene classification," *Pattern Recognit.*, vol. 37, no. 9, pp. 1757–1771, 2004.

[120] A. Elisseeff and J. Weston, "A Kernel method for multi-labelled classification," in *Proc. Neural Information Processing Systems*, 2000.

[121] M. Slaney, K. Weinberger, and W. White, "Learning a metric for music similarity," in *Proc. Int. Conf. Music Information Retrieval*, 2008.

[122] A. Holzapfel and Y. Stylianou, "Music genre classification using nonnegative matrix factorization-based features," *IEEE Trans. Audio, Speech, Lang. Processing*, vol. 16, no. 2, pp. 424–434, 2008.

[123] Z. Fu, G. Lu, K. Ting, and D. Zhang, "On feature combination for music classification," in *Proc. Int. Workshop Statistical Pattern Recognition*, 2010.

[124] G. Lanckriet, N. Cristianini, P. Bartlett, L. E. Ghaoui, and M. I. Jordan, "Learning the Kernel matrix with semidefinite programming," *J. Mach. Learn. Res.*, vol. 5, pp. 27–72, 2004.

[125] D. Wolpert, "Stacked generalization," *Neural Netw.*, vol. 5, no. 2, pp. 241–259, 1992.

[126] S. R. Ness, A. Theocharis, G. Tzanetakis, and L. G. Martins, "Improving automatic music tag annotation using stacked generalization of probabilistic SVM outputs," in *Proc. ACM Multimedia*, 2009.

[127] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, 1997.

[128] H. Homburg, I. Mierswa, B. Moller, K. Morik, and M. Wurst, "A benchmark dataset for audio classification and clustering," in *Proc. Int. Conf. Music Info. Retrieval*, 2005.

[129] E. Tsunoo, G. Tzanetakis, N. Ono, and S. Sagayama, "Audio genre classification using percussive pattern clustering combined with timbral features," in *Proc. Int. Conf. Music Information Retrieval*, 2009.

[130] K. Wang and S. A. Shamma, "Spectral shape analysis in the central auditory system," *IEEE Trans. Speech Audio Process.*, vol. 3, no. 5, pp. 382–396, 1995.

[131] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.

[132] K. Hevner, "Experimental studies of the elements of expression in music," *Amer. J. Psychol.*, vol. 48, no. 2, pp. 246–268, 1936.

[133] R. Thayer, *The Biopsychology of Mood and Arousal*. Oxford, U.K.: Oxford Univ. Press, 1989.

[134] A. Tellegen, D. Watson, and L. A. Clark, "On the dimensional and hierarchical structure of affect," *Psychol. Sci.*, vol. 10, no. 4, pp. 297–303, 1999.

[135] X. Hu and J. S. Downie, "Exploring mood metadata: Relationships with genre, artist and usage metadata," in *Proc. Int. Conf. Music Information Retrieval*, 2007.

[136] X. Hu, J. S. Downie, C. Laurier, M. Bay, and A. F. Ehmann, "The 2007 mirex audio mood classification task: Lessons learned," in *Proc. Int. Conf. Music Information Retrieval*, 2008.

[137] Y. Cho and L. K. Saul, "Learning dictionaries of stable autoregressive models for audio scene analysis," in *Proc. Int. Conf. Machine Learning*, 2009.

[138] F. Pachet and P. Roy, "Improving multilabel analysis of music titles: A large-scale validation of the correction approach," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 17, no. 2, pp. 335–343, 2009.

[139] D. Grangier and S. Bengio, "A discriminative Kernel-based approach to rank images from text queries," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 8, pp. 1371–1384, 2007.

[140] S. Shalev-Shwartz, "Online learning: Theory, algorithms, and applications," Ph.D. dissertation, Hebrew Univ., Jerusalem, Israel, 2007.

[141] J. Zhu, Semi-Supervised Learning Literature Survey. [Online]. Available: http://pages.cs.wisc.edu/~jerryzhu/research/ssl/semireview.html.

[142] Y. Moh and J. M. Buhmann, "Manifold regularization for semi-super sequential learning," in *Proc. Int. Conf. Acoustics, Speech and Signal Processing*, 2009.

[143] S. Ravuri and D. Ellis, "Cover song detection: From high scores to general classification," in *Proc. Int. Conf. Acoustics, Speech and Signal Processing*, 2010.

[144] T. S. Huang, C. K. Dagli, S. Rajaram, E. Y. Chang, M. I. Mandel, G. E. Poliner, and D. P. W. Ellis, "Active learning for interactive multimedia retrieval," *Proc. IEEE*, vol. 96, no. 4, pp. 648–667, 2008.

[145] J. T. Foote, M. Cooper, and U. Nam, "Audio retrieval by rhythmic similarity," in *Proc. Int. Conf. Music Information Retrieval*, 2002.

[146] R. Grosse, R. Raina, H. Kwong, and A. Y. Ng, "Shift-invariant sparse coding for audio classification," in *Proc. Uncertainty in Artificial Intelligence*, 2007.

[147] R. Gjerdingen and D. Perrott, "Scanning the dial: The rapid recognition of music genres," *J. New Music Res.*, vol. 37, no. 2, pp. 93–100, 2008.

[148] S. Lippens, J. P. Martens, M. Leman, B. Baets, H. Meyer, and G. Tzanetakis, "A comparison of human and automatic musical genre classification," in *Proc. Int. Conf. Acoustics, Speech and Signal Processing*, 2004.

[149] G. Wiggins and T. Crawford, "How many beans make five? The consensus problem in music-genre classification and a new evaluation method for single-genre categorisation systems," in *Proc. Int. Conf. Music Information Retrieval*, 2007.

Zhouyu Fu received the B.Eng. degree from Zhejiang University, Hangzhou, China, in 2001, the M.Eng. degree from the Institute of Automation, the Chinese Academy of Sciences, Beijing, China, in 2004, and the Ph.D. degree in information engineering from the Australian National University, Acton, in 2009.

He is currently a postdoctoral research fellow in Gippsland School of Information Technology at Monash University, Churchill, Australia. His research interests are in the areas of pattern recognition, machine learning, computer vision, and multimedia information retrieval.

Guojun Lu received the B.Eng. degree from Nanjing Institute of Technology (now South East University), Nanjing, China, in 1984 and the Ph.D. degree from Loughborough University, Loughborough, U.K., in 1990

He is currently a Professor of the Faculty of Information Technology, Monash University, Churchill, Australia. He has held positions at Loughborough University, National University of Singapore, and Deakin University. His main research interests are in multimedia communications and multimedia information indexing and retrieval. He has published over 150 refereed journal and conference papers and two books in these areas. He has over 15 years' teaching experience in three different universities, and has successfully supervised many research students.

Kai-Ming Ting received the Ph.D. degree from the University of Sydney, Sydney, Australia.

He has worked at the University of Waikato, New Zealand and Deakin University, Australia . He has been with Monash University, Churchill, Australia, since 2001 and is now an Associate Professor in the Gippsland School of Information Technology. His current research interests are in the areas of anomaly detection, ensemble approaches, data stream, swarm intelligence, data mining, and machine learning in general.

Dr. Ting is a current member of the Editorial Board of the *Data Mining and Knowledge Discovery Journal*. He has served as one of the three co-chairs for the Twelfth Pacific-Asia Conference on Knowledge Discovery and Data Mining, and a member of the program committee for a number of international conferences including the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining and the International Conference on Machine Learning. He has received research fundings from the Australian Research Council, the U.S. Air Force of Scientific Research (AFOSR/AOARD), and the Australian Institute of Sports

Dengsheng Zhang received the Ph.D. degree in computing from Monash University, Churchill, Australia, in 2002.

He joined Monash University in 2002. He is currently a senior lecturer in Faculty of Information Technology at Monash University. He has over 15 years of research experience in multimedia information processing and retrieval areas and has published over 60 refereed international journal and conference papers in his career. His main research interests include pattern recognition, multimedia information processing, and retrieval.

Dr. Zhang has won the 2009 Faculty of Information Technology Early Career Researcher Award for his outstanding research achievement.