Music Information Retrieval:
Recent Developments and Applications

Markus Schedl
Johannes Kepler University Linz, Austria
markus.schedl@jku.at

Emilia Gómez
Universitat Pompeu Fabra, Barcelona, Spain
emilia.gomez@upf.edu

Julián Urbano
Universitat Pompeu Fabra, Barcelona, Spain
julian.urbano@upf.edu
# Contents

1 Introduction to Music Information Retrieval 128  
1.1 Motivation ........................................ 128  
1.2 History and evolution .......................... 129  
1.3 Music modalities and representations ............ 130  
1.4 Applications ...................................... 132  
1.5 Research topics and tasks ...................... 141  
1.6 Scope and related surveys ..................... 141  
1.7 Organization of this survey .................. 143  

2 Music Content Description and Indexing 145  
2.1 Music feature extraction ....................... 146  
2.2 Music similarity ................................. 166  
2.3 Music classification and auto-tagging .......... 170  
2.4 Discussion and challenges ................... 172  

3 Context-based Music Description and Indexing 174  
3.1 Contextual data sources ......................... 175  
3.2 Extracting information on music entities .......... 176  
3.3 Music similarity based on the Vector Space Model ...... 181  
3.4 Music similarity based on Co-occurrence Analysis ...... 185  
3.5 Discussion and challenges ................... 190
4 User Properties and User Context 193
  4.1 User studies ........................ 194
  4.2 Computational user modeling  .... 196
  4.3 User-adapted music similarity  ... 198
  4.4 Semantic labeling via games with a purpose .... 200
  4.5 Music discovery systems based on user preferences . . . . 202
  4.6 Discussion and challenges ................ 204

5 Evaluation in Music Information Retrieval 208
  5.1 Why evaluation in Music Information Retrieval is hard ... 209
  5.2 Evaluation initiatives ........................ 213
  5.3 Research on Music Information Retrieval evaluation  .... 220
  5.4 Discussion and challenges .................... 222

6 Conclusions and Open Challenges 226

Acknowledgements 231

References 232
Abstract

We provide a survey of the field of Music Information Retrieval (MIR), in particular paying attention to latest developments, such as semantic auto-tagging and user-centric retrieval and recommendation approaches. We first elaborate on well-established and proven methods for feature extraction and music indexing, from both the audio signal and contextual data sources about music items, such as web pages or collaborative tags. These in turn enable a wide variety of music retrieval tasks, such as semantic music search or music identification (“query by example”). Subsequently, we review current work on user analysis and modeling in the context of music recommendation and retrieval, addressing the recent trend towards user-centric and adaptive approaches and systems. A discussion follows about the important aspect of how various MIR approaches to different problems are evaluated and compared. Eventually, a discussion about the major open challenges concludes the survey.

DOI: 10.1561/1500000042.
1

Introduction to Music Information Retrieval

1.1 Motivation

Music is a pervasive topic in our society as almost everyone enjoys listening to it and many also create. Broadly speaking, the research field of Music Information Retrieval (MIR) is foremost concerned with the extraction and inference of meaningful features from music (from the audio signal, symbolic representation or external sources such as web pages), indexing of music using these features, and the development of different search and retrieval schemes (for instance, content-based search, music recommendation systems, or user interfaces for browsing large music collections), as defined by Downie [52]. As a consequence, MIR aims at making the world’s vast store of music available to individuals [52]. To this end, different representations of music-related subjects (e.g., songwriters, composers, performers, consumer) and items (music pieces, albums, video clips, etc.) are considered.

Given the relevance of music in our society, it comes as a surprise that the research field of MIR is a relatively young one, having its origin less than two decades ago. However, since then MIR has experienced a constant upward trend as a research field. Some of the most important reasons for its success are (i) the development of audio compression
techniques in the late 1990s, (ii) increasing computing power of personal computers, which in turn enabled users and applications to extract music features in a reasonable time, (iii) the widespread availability of mobile music players, and more recently (iv) the emergence of music streaming services such as Spotify\(^1\), Grooveshark\(^2\), Rdio\(^3\) or Deezer\(^4\), to name a few, which promise unlimited music consumption every time and everywhere.

### 1.2 History and evolution

Whereas early MIR research focused on working with symbolic representations of music pieces (i.e. a structured, digital representation of musical scores such as MIDI), increased computing power enabled the application of the full armory of signal processing techniques directly to the music audio signal during the early 2000s. It allowed the processing not only of music scores (mainly available for Western Classical music) but all kinds of recorded music, by deriving different music qualities (e.g. rhythm, timbre, melody or harmony) from the audio signal itself, which is still a frequently pursued endeavor in today’s MIR research as stated by Casey et al. \[28\].

In addition, many important attributes of music (e.g. genre) are related not only to music content, but also to contextual/cultural aspects that can be modeled from user-generated information available for instance on the Internet. To this end, since the mid-2000s different data sources have been analyzed and exploited: web pages, microblogging messages from Twitter\(^5\), images of album covers, collaboratively generated tags and data from games with a purpose.

Recently and in line with other related disciplines, MIR is seeing a shift — away from system-centric towards user-centric designs, both in models and evaluation procedures as mentioned by different authors such as Casey et al. \[28\] and Schedl et al. \[241\]. In the case of

---

\(^1\)http://www.spotify.com

\(^2\)http://grooveshark.com/

\(^3\)http://www.rdio.com/

\(^4\)http://www.deezer.com

\(^5\)http://www.twitter.com
user-centric models, aspects such as serendipity (measuring how positively surprising a recommendation is), novelty, hotness, or location-and time-awareness have begun to be incorporated into models of users’ individual music taste as well as into actual music retrieval and recommendation systems (for instance, in the work by Zhang et al. [307]).

As for evaluation, user-centric strategies aim at taking into account different factors in the perception of music qualities, in particular of music similarity. This is particularly important as the notions of music similarity and of music genre (the latter often being used as a proxy for the former) are ill-defined. In fact several authors such as Lippens et al. [157] or Seyerlehner [252] have shown that human agreement on which music pieces belong to a particular genre ranges only between 75% and 80%. Likewise, the agreement among humans on the similarity between two music pieces is also bounded at about 80% as stated in the literature [282, 230, 287, 112].

1.3 Music modalities and representations

Music is a highly multimodal human artifact. It can come as audio, symbolic representation (score), text (lyrics), image (photograph of a musician or album cover), gesture (performer) or even only a mental model of a particular tune. Usually, however, it is a mixture of these representations that form an individual’s model of a music entity. In addition, as pointed out by Schedl et al. [230], human perception of music, and of music similarity in particular, is influenced by a wide variety of factors as diverse as lyrics, beat, perception of the performer by the user’s friends, or current mental state of the user. Computational MIR approaches typically use features and create models to describe music by one or more of the following categories of music perception: music content, music context, user properties, and user context, as shown in Figure 1.1 and specified below.

From a general point of view, music content refers to aspects that are encoded in the audio signal, while music context comprises factors that cannot be extracted directly from the audio but are nevertheless related to the music item, artist, or performer. To give some exam-
1.3. Music modalities and representations

![Diagram of music modalities and representations]

**Figure 1.1:** Categorization of perceptual music descriptors proposed in [230]

- Melodic structure, rhythmic structure, melody, and timbre features belong to the former category, whereas information about an artist’s cultural or political background, semantic labels, and album cover artwork belong to the latter. When focusing on the user, **user context** aspects represent dynamic and frequently changing factors, such as the user’s current social context, activity, or emotion. In contrast, **user properties** refer to constant or only slowly changing characteristics of the user, such as her music taste or music education, but also the user’s (or her friends’) opinion towards a performer. The aspects belonging to user properties and user context can also be related to long-term and short-time interests or preferences. While user properties are tied to general, long-term goals, user context much stronger influences short-time listening needs.

Please note that there are interconnections between some features from different categories. For instance, aspects reflected in collaborative tags (e.g. musical genre) can be modeled by music content (e.g.
instrumentation) while some others (e.g. geographical location, influences) are linked to music context. Another example is semantic labels, which can be used to describe both the mood of a music piece and the emotion of a user as reviewed by Yang and Chen [305].

Ideally, music retrieval and recommendation approaches should incorporate aspects of several categories to overcome the “semantic gap”, that is, the mismatch between machine-extractable music features and semantic descriptors that are meaningful to human music perception.

1.4 Applications

MIR as a research field is driven by a set of core applications that we present here from a user point of view.

1.4.1 Music retrieval

Music retrieval applications are intended to help users find music in large collections by a particular similarity criterion. Casey et al. [28] and Grosche et al. [89] propose a way to classify retrieval scenarios according to specificity (high specificity to identify a given audio signal and low to get statistically similar or categorically similar music pieces) and granularity or temporal scope (large granularity to retrieve complete music pieces and small granularity to locate specific time locations or fragments). Some of the most popular music retrieval tasks are summarized in the following, including pointers to respective scientific and industrial work.

*Audio identification* or *fingerprinting* is a retrieval scenario requiring high specificity and low granularity. The goal here is to retrieve or identify the same fragment of a given music recording with some robustness requirements (e.g. recording noise, coding). Well-known approaches such as the one proposed by Wang [297] have been integrated into commercially available systems, such as *Shazam* [6] (described in [297]), *Vericast* [7] or *Gracenote MusicID* [8]. Audio fingerprinting technolo-

---

6http://www.shazam.com
7http://www.bmat.com/products/vericast/
8http://www.gracenote.com/music/recognition/
1.4. Applications

Technologies are useful, for instance, to identify and distribute music royalties among music authors.

**Audio alignment**, matching or synchronization is a similar scenario of music retrieval where, in addition to identifying a given audio fragment, the aim is to locally link time positions from two music signals. Moreover, depending on the robustness of the audio features, one could also align different performances of the same piece. For instance, *MATCH* by Dixon and Widmer [48] and the system by Müller et al. [180] are able to align different versions of Classical music pieces by applying variants of the *Dynamic Time Warping* algorithm on sequences of features extracted from audio signals.

**Cover song identification** is a retrieval scenario that goes beyond the previous one (lower specificity level), as the goal here is to retrieve different versions of the same song, which may vary in many aspects such as instrumentation, key, harmony or structure. Systems for version identification, as reviewed by Serrà et al. [248], are mostly based on describing the melody or harmony of music signals and aligning these descriptors by local or global alignment methods. Web sites such as *The Covers Project*[^9] are specialized in cover songs as a way to study musical influences and quotations.

In **Query by humming** and **query by tapping**, the goal is to retrieve music from a given melodic or rhythmic input (in audio or symbolic format) which is described in terms of features and is compared to the documents in a music collection. One of the first proposed systems is *MUSART* by Birmingham et al. [43]. Music collections for this task were traditionally built with music scores, user hummed or tapped queries –more recently with audio signals as in the system by Salamon et al. [218]. Commercial systems are also exploiting the idea of retrieving music by singing, humming or typing. One example is *SoundHound*[^10], that matches users’ hummed queries against a proprietary database of hummed songs.

The previously mentioned applications are based on the comparison of a target music signal against a database (also referred as *query by ex-[^9]http://www.coversproject.com/
ample), but users may want to find music fulfilling certain requirements (e.g. “give me songs with a tempo of 100 bpm or in C major”) as stated by Isaacson [110]. In fact, humans mostly use tags or semantic descriptors (e.g. “happy” or “rock”) to refer to music. Semantic/tag-based or category-based retrieval systems such as the ones proposed by Knees et al. [125] or Turnbull et al. [278] rely on methods for the estimation of semantic labels from music. This retrieval scenario is characterized by a low specificity and long-term granularity. An example of such semantic search engines is SearchSounds by Celma et al. [31, 266], which exploits user-generated content from music blogs to find music via arbitrary text queries such as “funky guitar riffs”, expanding results with audio-based features. A screenshot of the user interface for the sample query “metal” can be seen in Figure 1.2. Another example is Gedoodle by Knees et al. [125], which is based on audio features and corresponding similarities enriched with editorial metadata (artist, album, and track names from ID3 tags) to gather related web pages. Both complementary pieces of information are then fused to map semantic user queries to actual music pieces. Figure 1.3 shows the results for the query “traditional irish”.

Figure 1.2: SearchSounds user interface for the query “metal”.

1.4. Applications

1.4.2 Music recommendation

Music recommendation systems typically propose a list of music pieces based on modeling the user’s musical preferences. Ricci et al. [212] and Celma [30] state the main requirements of a recommender system in general and for music in particular: accuracy (recommendations should match one’s musical preferences), diversity (as opposed to similarity, as users tend to be more satisfied with recommendations when they show a certain level of diversity), transparency (users trust systems when they understand why it recommends a music piece) and serendipity (a measure of “how surprising a recommendation is”). Well-known commercial
systems are Last.fm\textsuperscript{11}, based on collaborative filtering, and Pandora\textsuperscript{12}, based on expert annotation of music pieces.

Recent methods proposed in the literature focus on user-aware, personalized, and multimodal recommendation. For example, Baltrunas et al. [7] propose their InCarMusic system for music recommendation in a car; Zhang et al. [307] present their Auralist music recommender with a special focus on serendipity; Schedl et al. [231, 238] investigate position- and location-aware music recommendation techniques based on microblogs; Forsblum et al. [70] propose a location-based recommender for serendipitous discovery of events at a music festival; Wang et al. [298] present a probabilistic model to integrate music content and user context features to satisfy user’s short-term listening needs; Teng et al. [276] relate sensor features gathered from mobile devices with music listening events to improve mobile music recommendation.

1.4.3 Music playlist generation

Automatic music playlist generation, which is sometimes informally called “Automatic DJing”, can be regarded as highly related to music recommendation. Its aim is to create an ordered list of results, such as music tracks or artists, to provide meaningful playlists enjoyable by the listener. This is also the main difference to general music recommendation, where the order in which the user listens to the recommended songs is assumed not to matter. Another difference between music recommendation and playlist generation is that the former typically aims at proposing new songs not known by the user, while the latter aims at reorganizing already known material.

A study conducted by Pohle et al. [206], in which humans evaluated the quality of automatically generated playlists, showed that similarity between consecutive tracks is an important requirement for a good playlist. Too much similarity between consecutive tracks, however, makes listeners feel bored by the playlist.

Schedl et al. [231] hence identify important requirements other than similarity: familiarity/popularity (all-time popularity of an artist or

\textsuperscript{11}http://www.lastfm.com
\textsuperscript{12}http://www.pandora.com
1.4. Applications

track), hotness/trendiness (amount of attention/buzz an artist currently receives), recentness (the amount of time passed since a track was released), and novelty (whether a track or artist is known by the user). These factors and some others contribute to a serendipitous listening experience, which means that the user is positively surprised because he encountered an unexpected, but interesting artist or song. More details as well as models for such serendipitous music retrieval systems can be found in [231] and in the work by Zhang et al. [307].

To give an example of an existing application that employs a content-based automatic playlist generation approach, Figure 1.4 depicts a screenshot of the Intelligent iPod\(^\text{13}\) [246]. Audio features and corresponding similarities are directly extracted from the music collection residing on the mobile device. Based on these similarities, a playlist is created and visualized by means of a color stripe, where different colors correspond to different music styles, cf. (2) in Figure 1.4. The user can interact with the player with the scroll wheel to easily access the various music regions, cf. (4) in Figure 1.4.

Automatic playlist generation is also exploited in commercial products. To give an example, YAMAHA BODiBEAT\(^\text{14}\) uses a set of body sensors to track one’s workout and generate a playlist to match one’s running pace.

1.4.4 Music browsing interfaces

Intelligent user interfaces that support the user in experiencing serendipitous listening encounters are becoming more and more important, in particular to deal with the abundance of music available to consumers today, for instance via music streaming services. These interfaces should hence support browsing through music collections in an intuitive way as well as retrieving specific items. In the following, we give a few examples of proposed interfaces of this kind.

The first one is the nepTune\(^\text{15}\) interface proposed by Knees et al. [128], where music content features are extracted from a given mu-

\(^{13}\)http://www.cp.jku.at/projects/intelligent-ipod
\(^{14}\)http://www.yamaha.com
\(^{15}\)http://www.cp.jku.at/projects/neptune
sic collection and then clustered. The resulting clusters are visualized by creating a virtual landscape of the music collection. The user can then navigate through this artificial landscape in a manner similar to a flight simulator game. Figure 1.5 shows screenshots of the nepTune interface. In both versions, the visualization is based on the metaphor of “Islands of Music” [193], according to which densely populated clusters of songs are visualized as mountains, whereas sparsely populated regions are visualized as beaches and oceans.

A similar three-dimensional browsing interface for music collections is presented by Lübbers and Jarke [161]. Unlike nepTune, which employs the “Islands of Music” metaphor, their system uses an inverse height map, by means of which clusters of music items are visualized as valleys separated by mountains corresponding to sparse regions. In addition, Lübbers and Jarke’s interface supports user adaptation by providing means of deforming the landscape.

Figure 1.4: Intelligent iPod mobile browsing interface.
1.4. Applications

Figure 1.5: nepTune music browsing interface.

Musicream\footnote{http://staff.aist.go.jp/m.goto/Musicream} by Goto and Goto \cite{80} is another example of a user interface that fosters unexpected, serendipitous encounters with music, this time with the metaphor of a water tap. Figure 1.6 depicts a screenshot of the application. The interface includes a set of colored taps (in the top right of the figure), each corresponding to a different style of music. When the user decides to open the virtual handle, the respective tap creates a flow of songs. The user can then grab and play songs, or stick them together to create playlists (depicted on the left side of the figure). When creating playlists in this way, similar songs can be easily connected, whereas repellent forces are present between dissimilar songs, making it much harder to connect them.

Songrium\footnote{http://songrium.jp} is a collection of web applications designed to enrich the music listening experience. It has been developed and is maintained by the National Institute of Advanced Industrial Science and Technology (AIST) in Japan. As illustrated by Hamasaki and Goto \cite{90}, Songrium offers various ways to browse music, for instance, via vi-
sualizing songs in a graph using audio-based similarity for placement (“Music Star Map”), via visualizing a song and its derivative works in a solar system-like structure (“Planet View”), or via exploring music by following directed edges between songs, which can be annotated by users (“Arrow View”).

1.4.5 Beyond retrieval

MIR techniques are also exploited in other contexts, beyond the standard retrieval scenarios. One example is the computational music theory field, for which music content description techniques offer the possibility to perform comparative studies using large datasets and to formalize expert knowledge. In addition, music creation applications benefit from music retrieval techniques, for instance via “audio mosaicing”, where a target music track is analyzed, its audio descriptors extracted for small fragments, and these fragments substituted with
similar but novel fragments from a large music dataset. These applications are further reviewed in a recent "Roadmap for Music Information ReSearch" build by a community of researchers in the context of the MIReS project\(^\text{18}\) [250].

1.5 Research topics and tasks

We have seen that research on MIR comprises a rich and diverse set of areas whose scope goes well beyond mere retrieval of documents, as pointed out by several authors such as Downie et al. [55, 20], Lee et al. [147, 148] and Bainbridge et al. [6]. MIR researchers have then been focusing on a set of concrete research tasks, which are the basis for final applications. Although most of the tasks will be reviewed within this manuscript, we already provide at this point an overview of some of the most important ones (including references) in Table 1.1.

A first group of topics are related to the extraction of meaningful features from music content and context. These features are then used to compute similarity between two musical pieces or to classify music pieces according to different criteria (e.g. mood, instrument, or genre). Features, similarity algorithms and classification methods are then tailored to different applications as described below.

1.6 Scope and related surveys

The field of MIR has undergone considerable changes during recent years. Dating back to 2006, Orio [186] presented one of the earliest survey articles on MIR, targeted at a general Information Retrieval audience who is already familiar with textual information. Orio does a great job in introducing music terminology and categories of music features that are important for retrieval. He further identifies different users of an MIR system and discusses their individual needs and requirements towards such systems. The challenges of extracting timbre, rhythm, and melody from audio and MIDI representations of music are discussed. To showcase a music search scenario, Orio discusses different

\(^{18}\)http://mires.eecs.qmul.ac.uk/
ways of music retrieval via melody. He further addresses the topics of automatic playlist generation, of visualizing and browsing music collections, and of audio-based classification. Eventually, Orio concludes by reporting on early benchmarking activities to evaluate MIR tasks. Although Orio’s work gives a thorough introduction to MIR, many new research directions have emerged within the field since then. For instance, research on web-, social media-, and tag-based MIR could not be included in his survey. Also benchmarking activities in MIR were still in their fledgling stages at that time. Besides contextual MIR and evaluation, considerable progress has been made in the tasks listed in Table 1.1. Some of them even emerged only after the publication of [186]; for instance, auto-tagging or context-aware music retrieval.

Other related surveys include [28], where Casey et al. give an overview of the field of MIR from a signal processing perspective. They hence strongly focus on audio analysis and music content-based similarity and retrieval. In a more recent book chapter [227], Schedl gives an overview of music information extraction from the Web, covering the automatic extraction of song lyrics, members and instrumentation of bands, country of origin, and images of album cover artwork. In addition, different contextual approaches to estimate similarity between artists and between songs are reviewed. Knees and Schedl [127], give a survey of music similarity and recommendation methods that exploit contextual data sources. Celma’s book [30] comprehensively addressed the problem of music recommendation from different perspectives, paying particular attention to the often neglected “long tail” of little-known music and how it can be made available to the interested music aficionado.

In contrast to these reviews, in this survey we (i) also discuss the very current topics of user-centric and contextual MIR, (ii) set the discussed techniques in a greater context, (iii) show applications and combinations of techniques, not only addressing single aspects of MIR such as music similarity, and (iv) take into account more recent work.

Given the focus of the survey at hand on recent developments in MIR, we decided to omit most work on symbolic (MIDI) music representations. Such work is already covered in detail in Orio’s article
1.7. Organization of this survey

[186]. Furthermore, such work has been seeing a decreasing number of publications during the past few years. Another limitation of the scope is the focus on Western music, which is due to the fact that MIR research on music of other cultural areas is very sparse, as evidenced by Serra [249].

As MIR is a highly multidisciplinary research field, the annual “International Society for Music Information Retrieval” conference\(^{19}\) (ISMIR) brings together researchers of fields as diverse as Electrical Engineering, Library Science, Psychology, Computer Science, Sociology, Mathematics, Music Theory, and Law. The series of ISMIR conferences are a good starting point to dig deeper into the topics covered in this survey. To explore particular topics or papers presented at ISMIR, the reader can use the ISMIR Cloud Browser\(^{20}\) [88].

### 1.7 Organization of this survey

This survey is organized as follows. In Section 2 we give an overview of music content-based approaches to infer music descriptors. We discuss different categories of feature extractors (from low-level to semantically meaningful, high-level) and show how they can be used to infer music similarity and to classify music. In Section 3 we first discuss data sources belonging to the music context, such as web pages, microblogs, or music playlists. We then cover the tasks of extracting information about music entities from web sources and of music similarity computation for retrieval from contextual sources. Section 4 covers a very current topic in MIR research, i.e. the role of the user, which has been neglected for a long time in the community. We review ideas on how to model the user, highlight the crucial role the user has when elaborating MIR systems, and point to some of the few works that take the user context and the user properties into account. In Section 5 we give a comprehensive overview on evaluation initiatives in MIR and discuss their challenges. Section 6 summarizes this survey and highlights some of the grand challenges MIR is facing.

\(^{19}\)http://www.ismir.net

\(^{20}\)http://dc.ofai.at/browser/all
Table 1.1: Typical MIR subfields and tasks.

<table>
<thead>
<tr>
<th>Task</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FEATURE EXTRACTION</strong></td>
<td></td>
</tr>
<tr>
<td>Timbre description</td>
<td>Peeters et al. [200], Herrera et al. [99]</td>
</tr>
<tr>
<td>Music transcription and melody extraction</td>
<td>Klapuri &amp; Davy [122], Salamon &amp; Gómez [219], Hewlett &amp; Selridge-Field [103]</td>
</tr>
<tr>
<td>Onset detection, beat tracking, and tempo estimation</td>
<td>Belio et al. [10], Gómez [83], McKinney &amp; Breebaart [171]</td>
</tr>
<tr>
<td>Tonality estimation: chroma, chord, and key</td>
<td>Wakefield [296], Chew [34], Gómez [79], Papadopoulos &amp; Peeters [197], Oudre et al. [188], Temperley [274]</td>
</tr>
<tr>
<td>Structural analysis, segmentation and summarization</td>
<td>Cooper &amp; Foote [37], Peeters et al. [202], Chai [32]</td>
</tr>
<tr>
<td><strong>SIMILARITY</strong></td>
<td></td>
</tr>
<tr>
<td>Similarity measurement</td>
<td>Bogdanov et al. [18], Slaney et al. [28], Schedl et al. [236, 228]</td>
</tr>
<tr>
<td>Cover song identification</td>
<td>Serra et al. [118], Bertin-Mahieux &amp; Ellis [14]</td>
</tr>
<tr>
<td>Query by humming</td>
<td>Kosugi et al. [132], Salamon et al. [218], Dannenberg et al. [43]</td>
</tr>
<tr>
<td><strong>CLASSIFICATION</strong></td>
<td></td>
</tr>
<tr>
<td>Emotion and mood recognition</td>
<td>Yang &amp; Chen [304, 305], Laurier et al. [139]</td>
</tr>
<tr>
<td>Genre classification</td>
<td>Tzanetakis &amp; Cook [281], Knees et al. [124]</td>
</tr>
<tr>
<td>Instrument classification</td>
<td>Herrera et al. [102]</td>
</tr>
<tr>
<td>Composer, artist and singer identification</td>
<td>Kim et al. [118]</td>
</tr>
<tr>
<td>Auto-tagging</td>
<td>Sorio [264], Coviello et al. [39], Miotto &amp; Orio [173]</td>
</tr>
<tr>
<td><strong>APPLICATIONS</strong></td>
<td></td>
</tr>
<tr>
<td>Audio fingerprinting</td>
<td>Wang [297], Cano et al. [24]</td>
</tr>
<tr>
<td>Content-based querying and retrieval</td>
<td>Slaney et al. [28]</td>
</tr>
<tr>
<td>Music recommendation</td>
<td>Celma [30], Zhang et al. [307], Kaminskas et al. [114]</td>
</tr>
<tr>
<td>Playlist generation</td>
<td>Pohle et al. [268], Reynolds et al. [211], Pampalk et al. [196], Aucouturier &amp; Pachet [2]</td>
</tr>
<tr>
<td>Audio-to-score alignment and music synchronization</td>
<td>Dixon &amp; Warner [48], Scott &amp; Nurse [267], Leitich et al. [150], Lamere et al. [136], Pampalk &amp; Goto [195]</td>
</tr>
<tr>
<td>Song/artist popularity estimation</td>
<td>Schedl et al. [237], Pachet &amp; Roy [190], Koenigstein &amp; Shavitt [130]</td>
</tr>
<tr>
<td>Music visualization</td>
<td>Müller &amp; Jiang [199], Mardirossian &amp; Chew [166], Cooper et al. [38], Foote [68], Gómez &amp; Bonada [75]</td>
</tr>
<tr>
<td>Browsing user interfaces</td>
<td>Stober &amp; Nürnberger [250], Lamere et al. [136], Pampalk &amp; Goto [195]</td>
</tr>
<tr>
<td>Interfaces for music interaction</td>
<td>Steward &amp; Sandier [268]</td>
</tr>
<tr>
<td>Personalized, context-aware and adaptive systems</td>
<td>Schedl &amp; Schnitzer [238], Stober [269], Kaminskas et al. [114], Baltrunas et al. [7]</td>
</tr>
</tbody>
</table>
A content descriptor is defined in the MPEG-7 standard as a distinctive characteristic of the data which signifies something to somebody [220]. The term music content is considered in the literature as the implicit information that is related to a piece of music and that is represented in the piece itself (see Figure 2.1). Music content description technologies then try to automatically extract meaningful characteristics, called descriptors or features, from music material.

Music content descriptors can be classified according to three main criteria, as proposed by Gouyon et al. [85] and Leman et al. [152] among
others: (1) abstraction level: from low-level signal descriptors to high-level semantic descriptors; (2) temporal scope: descriptors can refer to a certain time location (instantaneous or frame-based), to a segment or to a complete music piece (global); and (3) musical facets: melody, rhythm, harmony/tonality, timbre/instrumentation, dynamics, structure or spatial location.

We present here the main techniques for music content description, focusing on the analysis of music audio signals. This description is crucial for MIR because, unlike the words, sentences, and paragraphs of text documents, music does not have an explicit, easily-recovered structure. The extracted descriptors are then exploited to index large music collections and provide retrieval capabilities according to different contexts and user needs.

2.1 Music feature extraction

2.1.1 Time and frequency domain representation

Techniques for the automatic description of music recordings are based on the computation of time and frequency representations of audio signals. We summarize here the main concepts and procedures to obtain such representations.

The frequency of a simple sinusoid is defined as the number of times that a cycle is repeated per second, and it is usually measured in cycles per second, or Hertz (Hz). As an example, a sinusoidal wave with a frequency $f = 440 \text{ Hz}$ performs 440 cycles per second. The inverse of the frequency $f$ is called the period $T$ ($f = \frac{1}{T}$), which is measured in seconds and indicates the temporal duration of one oscillation of the sinusoidal signal.

In time domain, analog signals $x(t)$ are sampled each $T_s$ seconds to obtain digital signal representations $x[n]$, where $n = i \cdot T_s$, $i = 0, 1, 2, ...$ and $f_s = \frac{1}{T_s}$ is the sampling rate in samples per second (Hz). According to the Nyquist-Shannon sampling theorem, a given audio signal should be at least sampled to the double of its maximum frequency to avoid the so-called aliasing, i.e. the introduction of artifacts during the sampling process. Time-domain representations, illustrated in Figure 2.2,
2.1. **Music feature extraction**

are suitable to extract descriptors related to the temporal evolution of the waveform $x[n]$, such as the location of major changes in signal properties.

The frequency spectrum of a time-domain signal is a representation of that signal in the frequency domain. It can be generated via the Fourier Transform (FT) of the signal, and the resulting values are usually presented as amplitude and phase, both plotted versus frequency, as illustrated in Figure 2.3. For sampled signals $x[n]$ we use the Discrete version of the Fourier Transform (DFT). Spectrum analysis is usually carried out in short segments of the sound signal (called *frames*), in order to capture the variations in frequency content along time (Short-Time Fourier Transform - STFT). This is mathematically expressed by multiplying the discrete signal $x[n]$ by a window function $w[n]$, which typically has a bell-shaped form and is zero-valued outside of the considered interval. STFT is displayed as a spectrogram, as illustrated in Figure 2.4.
Figure 2.3: Frequency-domain representation (long-term spectrum average normalized by its maximum value) of a flute sound (top), an oboe sound (middle) and a trumpet sound (bottom) playing the same note, C4. We observe that the harmonics are located in the same frequency positions for all these sounds, \(i \cdot f_0\), where \(i = 1, 2, 3, ...\) but there are differences on the spectral shape. The flute timbre is soft, characterized by energy decreasing harmonics compared to the fundamental frequency. The oboe and trumpet sounds have more energy in high-frequency harmonics (the frequency component in \(2 \cdot f_0\) is the one with highest energy), generating a brighter timbre.

The main parameters that influence the analysis are the frame size \(N\), the overlap between consecutive frames and the shape of the window function \(w[n]\). The frame size \(N\) (in samples) determines the frequency resolution \(\Delta f = \frac{f_s}{N}\) Hz, i.e. the distance between consecutive bins in the frequency domain. The compromise between having a good temporal resolution (using short frames) or a good frequency resolution (using long frames) is an important factor that should be adapted to the
2.1. Music feature extraction

Figure 2.4: Spectrogram (x-axis: time; y-axis: frequency) of a sound made of two sinusoids (one with a fixed frequency and another one with a decreasing frequency) and analyzed with a window of around 6 ms, providing good temporal resolution (top) and 50 ms providing good frequency resolution (bottom). We observe that good temporal resolution allows to analyze temporal transitions and good frequency resolution allows to distinguish close frequencies.

temporal and frequency characteristics of the signal under analysis. An example of the compromise between time and frequency resolution is illustrated in Figure 2.4.

Sound spectrum, as illustrated in Figure 2.3, is one of the main factors determining the timbre or the quality of a sound or note, as it describes the relative amplitude of the different frequencies of complex sounds.

2.1.2 Low-level descriptors and timbre

Low-level descriptors are computed from the audio signal in a direct or derived way, e.g. from its frequency representation. They have little meaning to users but they are easily exploited by computer systems.
They are usually related to loudness and timbre, considered as the color or quality of sound as described by Wessel [301]. Timbre has been found to be related to three main properties of music signals: temporal evolution of energy (as illustrated in Figure 2.2), spectral envelope shape (relative strength of the different frequency components, illustrated in Figure 2.3), and time variation of the spectrum. Low-level descriptors are then devoted to represent these characteristics.

Low-level descriptors are the basis for high-level analyses, so they should provide a proper representation of the sound under study. They should also be deterministic, computable for any signal (including silence or noise) and robust (e.g. to different coding formats, this can be application dependent). Although there is no standard way to compute low-level descriptors, they have a great influence on the behavior of the final application. A widely cited description of the procedure for low-level description extraction is presented by Peeters in [201] and [200], and illustrated in Figure 2.5. Instantaneous (frame-based) descriptors are obtained in both time and frequency domains, and then segment or global descriptors are computed after temporal modeling.

Well-known instantaneous temporal descriptors are the short-time Zero Crossing Rate (measuring the number of times the signal crosses the zero axis per second and related to noisiness and high frequency content) and energy (represented by the root mean square RMS value.
2.1. Music feature extraction

Common global temporal descriptors are log attack time (duration of the note onset) and temporal centroid (measuring the temporal location of the signal energy and useful to distinguish sustained vs. non-sustained sounds).

Mel-Frequency Cepstrum Coefficients (MFCCs) have been widely used to represent in a compact way (with a finite number of coefficients) a signal spectrum. They were proposed in the context of speech recognition (see Rabiner and Schafer [208]) and applied to music by Logan et al. [158]. They are computed as illustrated in Figure 2.6. The magnitude spectrum is filtered with a set of triangular filters with bandwidths following a Mel-frequency scale (emulating the behavior of the human hearing system). For each of the filters, the log of the energy is computed and a Discrete Cosine Transform (DCT) is applied to obtain the final set of coefficients (13 is a typical number used in the literature).

Other descriptors are spectral moments (spectral centroid, spread, skewness, and kurtosis), spectral slope, spectral roll-off (upper frequency spanning 95% of the spectral energy), spectral flatness, and spectral flux (correlation between consecutive magnitude spectra). Figure 2.7 shows an example of low-level instantaneous descriptors computed over an

Figure 2.6: Block diagram for the computation of MFCCs
Figure 2.7: Example of some low-level descriptors computed from a sound from the song "You’ve got a friend" by James Taylor (voice, guitar and percussion). Audio signal (top); Spectrogram and spectral centroid (second panel); Spectral Flux (third panel); RMS (bottom).

excerpt of the song "You’ve got a friend" by James Taylor (voice, guitar and percussion), computed using the *libxtract* Vamp plugin\(^1\) in *Sonic Visualizer*\(^2\).

Perceptual models can be further applied to represent perceptually-based low-level descriptors such as *loudness* or *sharpness*, and temporal

\(^1\)http://libxtract.sourceforge.net
\(^2\)http://www.sonicvisualizer.org
evolution of instantaneous descriptors can be studied by means of simple statistics (e.g., mean, standard deviations, or derivatives).

Low-level descriptors are often the basis for representing timbre in higher-level descriptors such as instrument, rhythm or genre. In addition, they have been directly used for audio fingerprinting as compact content-based signatures summarizing audio recordings.

### 2.1.3 Pitch content descriptors

Musical sounds are complex waveforms consisting of several components. Periodic signals (with period $T_0$ seconds) in time domain are harmonic in frequency-domain, so that their frequency components $f_i = i \cdot f_0$ are multiples of the so-called fundamental frequency $f_0 = \frac{1}{T_0}$. The harmonic series is related to the main musical intervals and establishes the acoustic foundations of the theory of musical consonance and scales as explained by Sethares in [251].

The perceptual counterpart of fundamental frequency is pitch, which is a subjective quality often described as highness or lowness. According to Hartman [96], sound has certain pitch if it can be reliably matched by adjusting the frequency of a sine wave of arbitrary amplitude. Although the pitch of complex tones is usually related to the pitch of the fundamental frequency, it can be influenced by other factors such as timbre. Some studies have shown that one can perceive the pitch of a complex tone even though the frequency component corresponding to the pitch may not be present (missing fundamental) and that non-periodic sounds (e.g., bell sounds) can also be perceived as having a certain pitch. We refer to the work of Schmuckler [244] and de Cheveigné [44] for a comprehensive review on the issue of pitch perception. Although not being the same, the terms pitch and fundamental frequency are often used as synonyms in the literature.

In music, the pitch scale is logarithmic (i.e. adding a certain musical interval corresponds to multiplying $f_0$ by a given factor) and intervals are measured in cents (1 semitone = 100 cents). Twelve-tone equal temperament divides the octave (i.e. multiply $f_0$ by a factor of 2), into 12 semitones of 100 cents each. In Western music, the set of pitches that are a whole number of octaves apart share the same pitch class
or *chroma*. For example, the pitch class A consists of the A’s in all octaves.

Pitch content descriptors are among the core of melody, harmony, and tonality description, having as their main goal to estimate periodicity in music signals from its time-domain or frequency-domain representation. A large number of approaches for $f_0$ estimation from monophonic signals (a single note present at a time) has been proposed in the literature, and adapted to different musical instruments, as reviewed by Gómez et al. [78]. Well-known approaches measure periodicity by maximizing autocorrelation (or minimizing distance) in time or frequency domain, such as the well-known YIN algorithm by de Cheveigné and Kawahara [45], which is based on time-domain distance computation. Alternative methods compare the magnitude spectrum with an ideal harmonic series (e.g. two-way mismatch by Maher and Beaugrand [162]), apply auditory modeling (e.g. as proposed by Klapuri [120]) or are based on the cepstrum (i.e. Inverse Fourier transform of the logarithm of the magnitude spectrum), as in Noll [183]).

Despite of all this research effort, up to our knowledge there is no standard method capable of working well for any sound in all conditions. The main difficulties of the task rely on the presence of quasi-periodicities, the fact that multiple periodicities are associated to a given $f_0$, and the existence of temporal variations, ambiguous events and noise.

The problem of mapping a sound signal from time-frequency domain to a “time-$f_0$” domain has turned out to be especially hard in the case of polyphonic signals where several sound sources are active at the same time. Multi-pitch (multiple $f_0$) estimation can be considered as one of the main challenges in the field, as we need to deal with masking, overlapping tones, mixture of harmonic and non-harmonic sources, and the fact that the number of sources might be unknown. Approaches thus focus on three simplified tasks: (1) the extraction of the $f_0$ envelope corresponding to the predominant instrument in complex polyphonies (e.g. the singing voice in popular music), a task commonly denoted as melody extraction [216]; (2) the estimation of multiple $f_0$ on simple polyphonies (few overlapping notes): (3) the computation of chroma
features, where multiple $f_0$ values are jointly analyzed and mapped to a single octave [296].

**Predominant melody extraction**

Predominant $f_0$ algorithms are an extension of methods working in monophonic music signals, but based on the assumption that there is a predominant sound source (e.g., singing voice or soloist instrument) in the spectrum. The main goal is then to identify a predominant harmonic structure in the spectral domain. There are two main approaches to melody extraction: *salience-based* algorithms, based on estimating the salience of each possible $f_0$ value (within the melody range) over time from the signal spectrum, and methods based on *source separation*, which first try to isolate the predominant source from the background and then apply monophonic $f_0$ estimation. For a detailed review on the state-of-the-art, applications, and challenges of melody extraction we refer to the work by Salamon et al. [216].

A state-of-the-art salience-based method by Salamon and Gómez [215] is shown in Figure 2.8. First, the audio signal is converted to the frequency domain incorporating some equal loudness filter and frequency/amplitude correction, and the spectral peaks are detected. Those spectral peaks are used to build the “salience function”, a time-$f_0$ representation of the signal. By analyzing the peaks of this salience function, a set of $f_0$ contours are built, being time continuous sequences of $f_0$ candidates grouped using auditory streaming cues. By studying contour characteristics, the system distinguishes between melodic and non-melodic contours to obtain the final melody $f_0$ sequence. An example of the output of this melody extraction approach, extracted with the MELODIA tool\(^3\), is illustrated in Figure 2.9.

Current methods work well (around 75% of overall accuracy according to Salamon et al. [216]) for music with a predominant instrument (mostly evaluated in singing voice), but there are still limitations in voicing detection (estimating whether or not a predominant instrument is present) and in the presence of strong accompaniment.

\(^3\)http://mtg.upf.edu/technologies/melodia
Multi-pitch estimation

Multi-pitch (multi-\(f_0\)) estimation methods try to estimate all the pitches within a mixture. As for melody extraction, current algorithms are based either on source separation or saliency analysis.

Methods based on source separation may follow an iterative process, where the predominant \(f_0\) is estimated, a predominant spectrum is built from this \(f_0\) information, and is subtracted from the original spectrum. A well-known algorithm of this kind is the one proposed by
2.1. *Music feature extraction*

![Image of music feature extraction](image)

**Figure 2.9:** Example of the output of the melody extraction algorithm proposed by Salamon and Gómez [215]. Waveform (top pane); spectrogram and extracted melody $f_0$ sequence in red color (second pane); salience function (third pane); $f_0$ contours (bottom pane). This figure was generated by the MELODIA Vamp plugin.
Klapuri [123] and illustrated in Figure 2.10. It consists of three main blocks that are shared by alternative proposals in the literature: auditory modeling, bandwise processing, and periodicity estimation. First, the signal is input to a model of the peripheral auditory system consisting of a bank of 72 filters with center frequencies on the critical-band scale (approximation of logarithm bandwidths of the filters in human hearing) covering the range from 60 Hz to 5.2 KHz. The output of the filterbank is compressed, half-wave rectified, and low-pass filtered to further model the mechanisms of the inner ear. This auditory modeling step is followed by the computation of the magnitude spectra per channel. Within-band magnitude spectra are summed to obtain a summary magnitude spectrum, where the predominant $f_0$ is estimated. Then, harmonics corresponding to the $f_0$ candidate are located and a harmonic model is applied to build the predominant magnitude spectrum, which is subtracted from the original spectrum.

Another set of approaches are based on a joint $f_0$ estimation, with the goal of finding an optimal set of $N$ $f_0$ candidates for $N$ harmonic series that best approximate the frequency spectrum. Multi-band or multi-resolution approaches for frequency analysis are often considered in this context (e.g. by Dressler [58]), and the joint estimation is usually performed by partially assigning spectral peaks to harmonic positions as proposed by Klapuri in [121].
2.1. Music feature extraction

State-of-the-art algorithms are evaluated on simple polyphonies. For instance, there was a maximum of five simultaneous notes at the 2013 Music Information Retrieval Evaluation eXchange\(^4\) (MIREX), a community-based international evaluation campaign that takes place in the context of the International Conferences on Music Information Retrieval (ISMIR). Current approaches (Yeh et al. [306] and Dressler [59]) yield an accuracy around 65\%, showing the difficulty of the task.

**Chroma feature extraction**

Chroma features, as illustrated in Figure 2.11, represent the intensity of each of the 12 pitch classes of an equal-tempered chromatic scale, and are computed from the frequency spectrum.

Chroma features can be extracted from monophonic and polyphonic music signals. As with pitch estimation methods, chroma feature extractors should be robust to noise (non-pitched sounds) and independent of timbre (spectral envelope), dynamics, and tuning. Several approaches exist for chroma feature extraction (we refer to the work by

---

\(^4\)http://music-ir.org/mirex/
Gómez [74] for a review), following the steps illustrated in Figure 2.12.

The signal is first analyzed in order to obtain its frequency domain representation, using a high frequency resolution. The main frequency components (e.g., spectral peaks) are then mapped to pitch class values according to an estimated tuning frequency. For most approaches, a frequency value partially contributes to a set of ’sub-harmonic’ fundamental frequency (and associated pitch class) candidates. The chroma vector is computed with a given interval resolution (number of bins per octave) and is finally post-processed to obtain the final chroma representation. Timbre invariance is achieved by different transformations such as spectral whitening [74] or cepstrum liftering (discarding low cepstrum coefficients) as proposed by Müller and Ewert [177]. Some approaches for chroma estimation are implemented into downloadable tools, e.g., the HPCP Vamp plugin\(^5\) implemented the approach in [74] and the Chroma Matlab toolbox\(^6\) implementing the features from [177].

2.1.4 Melody, harmony, and tonality

The pitch content descriptors previously described are the basis for higher-level music analysis, which are useful not only for users with knowledge in music theory, but also for the general public (major and minor mode, for instance, has been found to correlate with emotion).

\(^5\)http://mtg.upf.edu/technologies/hpcp

\(^6\)http://www.mpi-inf.mpg.de/resources/MIR/chromatoolbox/
2.1. Music feature extraction

Pitches are combined sequentially to form melodies and simultaneously to form chords. These two concepts converge into describing tonality, understood as the architectural organization of pitch material in a given musical piece.

The majority of empirical research on tonality modeling has been devoted to Western music, where we define key as a system of relationships between a series of pitches having a tonic as its most important element, followed by the dominant (5th degree of the scale) and sub-dominant (4th degree of the scale). In Western music, there are two basic modes, major and minor, each of them having different position of intervals within their respective scales. When each tonic manages both a major and a minor mode, there exist a total of 24 keys, considering an equal-tempered scale (12 equally distributed semitones within an octave).

There are different studies related to the computational modelling of tonality from score information, as reviewed by Chew [34]. A well-known method to estimate the key from score representations is the one proposed by Krumhansl et al. [134], based on measuring the correlation of pitch duration information (histogram of relative durations of each of the 12 pitch-classes of the scale) with a set of key profiles. These major/minor key profiles, shown in Figure 2.13, represent the stability of the 12 pitch classes relative to a given key. They were based on data from experiments by Krumhansl and Kessler in which subjects were asked to rate how "well" each pitch class "fit with" a prior context establishing a key, such a cadence or scale. As an alternative to human ratings, some approaches are based on learning these profiles from music theory books, as proposed by Temperley [274] or MIDI files, as proposed by Chai [33]. Current methods provide a very good accuracy (92 % in Classical music according to MIREX) in estimating the key from MIDI files, such as the method proposed by Temperley [275].

Some of these methods have been adapted to audio signals by exploiting pitch content descriptors, mainly chroma features, as proposed by Gómez [74], Chuan and Chew [35], and Papadopoulos and Peeters [198]. Accuracies of state-of-the-art methods fall below those obtained by their MIDI-based counterparts (around 80%). This is due to the dif-
difficulty of extracting pitch content information from polyphonic music audio signals, which is implicitly given in MIDI files (see Section 2.1.3).

Giving just a key value is poor in terms of description, as a musical piece rarely maintains the same tonal center allover its duration. According to Leman [151], tonal context is built up at different time scales, at least one time frame for local events (pitches and chords) and another one for global events (key). Template-based approaches have also been applied to short segments to estimate chords instead of key, e.g., by Oudre et al. [188] as illustrated in Figure 2.14. Probabilistic models (Hidden Markov Models) have also been adapted to this task, e.g., by Papadopoulos and Peeters [197]. Recently, multi-scale approaches, such as the one by Sapp [223], have been adapted to deal with music signals as illustrated in Figure 2.15 [167].

Current methods for tonality representation have been adapted to different repertoire, mostly parameters such as the interval resolution (e.g. to cope with different tuning systems as those found in non-Western music) or the used profiles. Some examples in different repertoire are Makkam music [109] or Indian music [217].
2.1. Music feature extraction

Figure 2.14: System for real-time tonality description and visualization from audio signals, presented in [75]. Top: chroma features; bottom: estimated chord (or key) mapped to the harmonic network representation.

Figure 2.15: Multi-resolution tonality description (keyscape) as presented by Martorell and Gómez [167].

2.1.5 Novelty detection and segmentation

Novelty relates to the detection of changes in the audio signal and is commonly used to segment music signals into relevant portions such as notes or sections with different instrumentation. Two main tasks in the MIR literature are related to novelty detection: onset detection and audio segmentation.

The goal of onset detection algorithms is to locate the start time (onset) of new events (transients or notes) in the signal. Onset is defined
as a single instant chosen to mark the start of the (attack) transient. The task and techniques are similar to those found for other modalities, e.g. the location of shot boundaries in video [154]. Onset detection is an important step for higher-level music description, e.g. music transcription, melody, or rhythm characterization.

Bello et al. [10] provide a good overview of the challenges and approaches for onset detection. According to the authors, the main difficulties for this task are the presence of slow transients, ambiguous events (e.g., vibrato, tremolo, glissandi) and polyphonies (onsets from different sources). Onsets are usually characterized by a fast amplitude increase, so methods for onset detection are based on detecting fast changes in time-domain energy (e.g. by means of log energy derivative) or the presence of high frequency components (e.g. using low-level features such as spectral flux). This procedure is illustrated in Figure 2.16. For polyphonic music signals, this approach is often extended to multiple frequency bands as proposed by Klapuri [119]. Detecting notes is slightly different than detecting onsets, as consecutive notes can be only perceived by a pitch glide, so that approaches for onset detection would fail. Note segmentation approaches then combine the location of energy and $f_0$ variations in the signal, which is especially challenging for instruments with soft changes such as the singing voice [76].

Segmentation of an audio stream into homogeneous sections is needed in different contexts such as speech vs. music segmentation, singing voice location, or instrument segmentation. Low-level features related to timbre, score-representations, pitch or chroma have been used in the literature for audio segmentation following two main approaches: model-free methods based on signal features and algorithms that rely on probabilistic models. Model-free approaches follow the same prin-
2.1. Music feature extraction

principle as the onset detection algorithms previously introduced, and use the amount of change of a feature vector as a boundary detector: when this amount is higher than a given threshold, a boundary change decision is taken. Threshold adjustment requires a certain amount of trial-and-error, or fine-tuned adjustments regarding different segmentation classes. Furthermore a smoothing window is usually applied. Model-based segmentation requires previous training based on low-level descriptors and annotated data. Hidden Markov Models, Gaussian Mixture Models, Auto-Regressive models, and Support Vector Machines are some of the techniques exploited in this context. We refer to Ong [185] for a review of approaches.

2.1.6 Rhythm

Rhythm is related to the architectural organization of musical events along time (temporal hierarchy) and incorporates regularity (or organization) and differentiation as stated by Desain and Windsor [47]. The main rhythm descriptors to be extracted from music signals are related to four different components: timing (when events occur), tempo (how often events occur), meter (what structure best describes the event occurrences) and grouping (how events are structured in motives or phrases).

Methods for computational rhythm description are based on measuring periodicity of events, represented by onsets (see Section 2.1.5) or low-level features, mainly energy (on a single or multiple frequency bands) and spectral descriptors. This is illustrated in Figure 2.17, computed using the algorithm proposed by Stark et al. [267] and available online. Methods for periodicity detection are then analogous to algorithms used for pitch estimation, presented in Section 2.1.3, but based on low-level descriptors.

Most of the existing literature focuses on estimating tempo and beat position and inferring high-level rhythmic descriptors related to meter, syncopation (displacement of the rhythmic accents), or rhythmic pattern. The overall block diagram is shown in Figure 2.18. We refer to Gouyon [84] for a review on rhythm description systems.

\[\text{http://www.vamp-plugins.org/plugin-doc/qm-vamp-plugins.html}\]
Holzapfel et al. [104] perform a comparative evaluation of beat tracking algorithms, finding that the main limitations of existing systems are to deal with non-percussive material (e.g., vocal music) with soft onsets, and to handle short-time deviations, varying tempo, and integrating knowledge on tempo perception (double or half errors) [171].

2.2 Music similarity

Similarity is a very active topic of research in MIR as it is in the core of many applications, such as music retrieval and music recommendation systems. In music content description, we consider similarity in two different time scales: locally, when we try to locate similar excerpts from the same musical piece (self-similarity analysis) or between different pieces, and globally if we intend to compute a global distance between two musical pieces. The distinction between local and global similarity/retrieval is also found in other modalities (e.g., passage retrieval
2.2. Music similarity

The main research problem in music similarity to define a suitable distance or similarity measure. We have to select the musical facets and descriptors involved, the abstraction level (too concrete would discard variations and too abstract would yield false positives), and the desired granularity level or temporal scope. Moreover, similarity depends on the application (as seen in Section 1) and might be a subjective quality that requires human modeling (e.g. Vignoli and Pauws [292]).

2.2.1 Self-similarity analysis and music structure

Structure is related to similarity, proximity, and continuity; so research on structural analysis of music signals is mainly linked to two research goals: detecting signal changes (as presented in Section 2.1.5) and detecting repetitions, exact or with variations, within the same musical piece. This task is also denoted as self-similarity analysis. One practical goal, for instance, is to detect the chorus of a song. Self-similarity analysis is based on the computation of a self-similarity matrix, as proposed by Foote [68]. Such a matrix is built by pairwise comparison of feature vectors from two different frames of a music recording. An example of a self-similarity matrix is shown in Figure 2.19. Repetitions are detected by locating diagonals over this matrix, and some musical
restrictions might be applied for final segment selection and labeling.

An important application of self-similarity analysis is music summarization, as songs may be represented by their most frequently repeated segments [37, 33].

2.2.2 Global similarity

The concept of similarity is a key aspect of indexing, retrieval, recommendation, and classification. Global similarity computation is usually based either on content descriptors or on context information (see Section 3).

Traditional approaches for content-based music similarity were mostly based on low-level timbre descriptors, as proposed by Aucouturier and Pachet [3, 189] and Pampalk [194]. Foote [69] proposed the exploitation of rhythmic features (melodic and tonal information was later incorporated), mainly in the context of cover version identification (see Serrà et al. [248] for an extensive review of methods).
2.2. Music similarity

A recent example of a state-of-the-art approach is the Block-level Framework (BLF) proposed by Seyerlehner et al. [254]. This framework describes a music piece by first modeling it as overlapping blocks of the magnitude spectrum of its audio signal. To account for the musical nature of the audio under consideration, the magnitude spectrum with linear frequency resolution is mapped onto the logarithmic Cent scale. Based on these Cent spectrum representations, BLF defines several features that are computed on blocks of frames (Figure 2.20): spectral pattern (SP) characterizes the frequency content, Delta Spectral Pattern (DSP) emphasizes note onsets, Variance Delta Spectral Pattern (VDSP) aims at capturing variations of onsets over time, Logarithmic Fluctuation Pattern (LFP) describes the periodicity of beats, Correlation Pattern (CP) models the correlation between different frequency bands, and Spectral Contrast Pattern (SCP) uses the difference between spectral peaks and valleys to identify tonal and percussive components. Figure 2.20 illustrates the different features for a piano piece by Shostakovich. The y-axis represents the frequency bands and the x-axis the sorted temporal components of the blocks.
Recent work on global similarity complements low-level descriptors with semantic descriptors obtained through automatic classification (see Section 2.3), as proposed by Bogdanov et al. [19, 17] for music similarity and recommendation. Global similarity can also be based on local similarity. To this end, algorithms for sequence alignment have been used, for instance, to obtain a global similarity value in the context of cover version identification by Serrà [248] and Müller et al. [180].

Music similarity is still an ill-defined concept, often indirectly evaluated in the context of artist classification, cover version identification, by means of co-occurrence analysis of songs in personal collections and playlists [12, 13] or by surveys [292]. Section 4 reviews some strategies to adapt similarity measures to different user contexts, and Section 5 provides further details on the quantitative and qualitative evaluation of similarity measures.

2.3 Music classification and auto-tagging

Until now we have reviewed methods to extract descriptors related to melody, rhythm, timbre, or harmony from music signals. These descriptors can be used to infer higher-level semantic categories via classification methods. Such high-level aspects are typically closer to the way humans would describe music, for instance, by a genre or instrument.

In general, we can distinguish between approaches that classify a given music piece into one out of a set of categories (music classification) and approaches that assign a number of semantic labels (or “tags”) to a piece (music auto-tagging). Auto-tagging frequently uses tags from a folksonomy, e.g. from Last.fm users, and can be thought of as a multi-label classification problem.

Research efforts on music classification have been devoted to classify music in terms of instrument (Herrera et al. [102]), genre (Tzanetakis and Cook [281], Scaringella et al. [225]), mood (Laurier et al. [139]) or culture (Gómez et al. [77]), among others. Results for this task vary depending on different factors such as the number of classes, the objectivity of class instances (mood, for instance, is a quite subjective concept), the representativeness of the collection used for training, and
2.3. *Music classification and auto-tagging*

The process of music auto-tagging is illustrated in Figure 2.21 proposed by Sordo [264]. Given a tagged music collection (training set), features are extracted from the audio, possibly followed by a dimensionality reduction or feature selection step, to increase computational performance. Subsequently, tag models are learned by classifiers, based on the relationship between feature vectors and tags. After this training phase on labeled data, the classifiers can be used to predict tags for previously unseen music items. Features frequently used in auto-taggers include rhythm and timbre descriptors (Mandel et al. [165]), but also high-level features may be considered (Sordo [264]).

Some recent approaches to music auto-tagging are summarized as follows. Sordo [264] presents a method called weighted vote $k$-Nearest Neighbor (kNN) classifier. Given a song $s$ to be tagged and a training set of labeled songs, the proposed approach identifies the $k$ closest neighbors $N$ of $s$ according to their feature vector representation. Hereafter, the frequencies of each tag assigned to $N$ are summed up and the most frequent tags of $N$ (in relation to the value of $k$) are predicted for $s$. 

---

**Figure 2.21**: Schematic illustration of a music auto-tagger, according to Sordo [264].
Similar to Sordo, Kim et al. [116] employ a kNN classifier to autotag artists. They investigate different artist similarity measures, in particular, similarities derived from artist co-occurrences in Last.fm playlists, from Last.fm tags, from web pages about the artists, and from music content features.

Mandel et al. [165] propose an approach that learns tag language models on the level of song segments, using conditional Restricted Boltzmann Machines [262]. Three sets of vocabularies are considered: user annotations gathered via Amazon’s Mechanical Turk, tags acquired from the tagging game MajorMiner [164], and tags extracted from Last.fm. The authors further suggest to take into account not only song segments, but include into their model also annotations on the track level and the user level.

Seyerlehner et al. [253] propose an auto-tagger that combines various audio features modeled within their block-level framework [254], as previously described. A Random Forest classifier is then used to learn associations between songs and tags.

A very recent trend is to employ two-stage algorithms. Such algorithms in a first step derive higher-level information from music content features, for instance, weights of descriptive terms. These new representations, sometimes combined with the original audio features, are subsequently used by a classifier to learn semantic labels (Coviello et al. [39]; Miotto et al. [172]).

2.4 Discussion and challenges

We have reviewed the main methods for extracting meaningful descriptions for music signals related to different musical facets such as timbre, melody, harmony, and rhythm, and we have seen that these descriptors can be exploited in the context of similarity and classification, among others. The underlying technologies work to a certain extent (state-of-the-art algorithms for feature extraction have an accuracy around 80%, depending on the task), but show a “glass-ceiling” effect. This can be explained by several factors, such as the subjectivity of some labeling tasks and the existence of a conceptual (semantic) gap between content
feature extractors and expert analyses. Furthermore, current technologies should be adapted to the repertoire under study (e.g., focus on mainstream popular music; limitations, for instance, for Classical music or for repertoires outside of the so-called Western tradition).

Recent strategies to overcome these limitations are the development of repertoire-specific methods, the integration of feature extractions and expert annotations (computer-assisted description), the development of personalized and adaptive descriptors, and the integration of multiple modalities (score, audio, and video) for automatic music description.
As we have seen in the previous chapter, there exists a lot of work aiming at uncovering from the audio signal meaningful music qualities that can be used for music similarity and retrieval tasks. However, as long ago as 2004, Aucouturier and Pachet [3] speculated that there is an upper limit of performance levels achievable with music content-based approaches. Motivated by the fact that there are seemingly aspects that are not encoded in the audio signal or that cannot be extracted from it, but which are nevertheless important to the human perception of music (e.g., meaning of lyrics or cultural background of songwriter), MIR researchers started to look into data sources that relate to the music context of a piece or an artist. Most of the corresponding approaches rely on Text-IR techniques, which are adapted to suit music indexing and retrieval. However, there is a major difference to Text-IR: in music retrieval, it is not only the information need that needs to be satisfied by returning relevant documents, but there is also the entertainment need of the listener and her frequent desire to retrieve serendipitous music items. Serendipity in this context refers to the discovery of an interesting and unexpected music item.

In this section, we first briefly present data sources that are fre-
3.1. **Contextual data sources**

Since the early 2000s, web pages have been used as an extensive data source (Cohen and Fan [36]; Whitman and Lawrence [302]; Baumann and Hummel [9]; Knees et al. [124]). Only slightly later, music-related information extracted from peer-to-peer networks started to be used for music similarity estimation by Whitman and Lawrence [302], Ellis et al. [64], Berenzweig et al. [13], and Logan et al. [159]. Another contextual data source is music playlists shared on dedicated web platforms such as Art of the Mix\(^1\). Playlists have been exploited, among others, by Pachet et al. [191], Cano and Koppenberger [27], and Bacigalupo et al. [4]. A lot of MIR research benefits from collaboratively generated tags. Such tags are either gathered via games with a purpose (Law et al. [140]; Mandel and Ellis [164]; Turnbull et al. [277]; Law and von Ahn [141]) or from Last.fm (Levy and Sandler [153]; Geleijnse et al. [71]). Probably the most recent data source for music retrieval and recommendation tasks is microblogs, exploited by Zangerle et al. [309] and Schedl et al. [228, 232]. In addition to the aforementioned sources that are already quite well researched, Celma [31] exploit RSS feeds of music blogs, while Hu et al. [107] mine product reviews.

The main challenge with all contextual data sources is to reliably identify resources that refer to a music item or an artist. In the case of web pages, this is typically achieved by issuing music-related queries to search engines and analyzing the fetched web pages, as done by Whitman and Lawrence [302] as well as Knees et al. [124]. In the case

\(^1\)http://www.artofthemix.org
of microblogs, researchers typically rely on filtering posts by hashtags (Zangerle et al. [309]; Schell [228]).

Contextual data sources can be used to mine pieces of information relevant to music entities. Respective work is summarized in Section 3.2. The large body of work involving music context in similarity and retrieval tasks can be broadly categorized into approaches that represent music entities as high-dimensional feature vectors according to the Vector Space Model (VSM) [221, 5] and into approaches that employ co-occurrence analysis. The former category is addressed in Section 3.3, the latter in Section 3.4.

### 3.2 Extracting information on music entities

The automated extraction of music-related pieces of information from unstructured or semi-structured data sources, sometimes called Music Information Extraction (MIE), is a small subfield of MIR. Nevertheless it is highly related to context-based music description and indexing. An overview of work addressing some categories of music-related information and of common methods is thus given in the following.

#### 3.2.1 Band members and their roles

In order to predict members of a band and their roles, i.e. instruments they play, Schedl et al. [242] propose an approach that first crawls web pages about the band under consideration. From the set of crawled web pages, n-grams are extracted and several filtering steps (e.g., with respect to word capitalization and common speech terms) are performed in order to construct a set of potential band members. A rule-based approach is then applied to each candidate member and its surrounding text. The frequency of patterns such as "[member] plays the [instrument]" is used to compute a confidence score and eventually predict the (member, instrument) pairs with highest confidence. This approach yielded a precision of 61% at 26% recall on a collection of 500 band members.

Extending work by Krenmayer [133], Knees and Schedl [126] propose two approaches to band member detection from web pages. They
use a Part-of-Speech (PoS) tagger [22], a gazetteer annotator to identify keywords related to genres, instruments, and roles, among others, and finally they perform a transducing step on named entities, annotations, and lexical metadata. This final step yields a set of rules similar to the approach by Schedl et al. [242]. The authors further investigate a Machine Learning approach, employing a Support Vector Machine (SVM) [290], to predict for each token in the corpus of music-related web pages whether it is a band member or not. To this end, the authors construct feature vectors including orthographic properties, PoS information, and gazetteer-based entity information. On a collection of 51 Metal bands, the rule-based approach yielded precision values of about 80% at 60% recall, whereas the SVM-based approach performed inferior, given its 78% precision at 50% recall.

3.2.2 Artist’s or band’s country of origin

Identifying an artist’s or a band’s country of origin provides valuable clues of their background and musical context. For instance, an artist’s geographic and cultural context, political background, or song lyrics are likely strongly related to his or her origin. Work on this task has been performed by Govaerts and Duval [86] and by Schedl et al. [240]. While the former mines these pieces of information from specific web sites, the latter distills the country of origin from web pages identified by a search engine.

Govaerts and Duval search for occurrences of country names in biographies from Wikipedia\textsuperscript{2} and Last.fm, as well as in properties such as “origin”, nationality”, “birth place”, and “residence” from Freebase\textsuperscript{3}. The authors then apply simple heuristics to predict the most probable country of origin for the artist or band under consideration. An example of such a heuristic is predicting the country that most frequently occurs in an artist’s biography. Another one favors early occurrences of country names in the text. When using Freebase as data source, the authors again predict the country that most frequently occurs in the related properties of the artist or band. Combining the results of the different

\textsuperscript{2}http://www.wikipedia.org
\textsuperscript{3}http://www.freebase.com
data sources and heuristics, Govaerts and Duval [86] report a precision of 77% at 59% recall.

Schedl et al. [240] propose three approaches to country of origin detection. The first one is a heuristic which compares the page count estimates returned by Google for queries of the form "artist/band" "country" and simply predicts the country with highest page count value for a given artist or band. The second approach takes into account the actual content of the web pages. To this end, up to 100 top-ranked web pages for each artist are downloaded and \( tf \cdot idf \) weights are computed. The country of origin for a given artist or band is eventually predicted as the country with highest \( tf \cdot idf \) score using as query the artist name. The third approach relies as proxy on text distance between country and key terms such as “born” or “founded”. For an artist or band \( a \) under consideration, this approach predicts as country of origin \( c \) the country whose name occurs closest to any of the key terms in any web page retrieved for \( c \). It was shown that the approach based on \( tf \cdot idf \) weighting reaches a precision level of 71% at 100% recall and hence outperforms the other two methods.

3.2.3 Album cover artwork

Automatically determining the image of an album cover, given only album and performer name is dealt with by Schedl et al. [233, 243]. This is to the best of our knowledge the only work on this task. The authors first use search engine results to crawl web pages of artists and albums under consideration. Subsequently, both the text and the HTML tags of the crawled web pages are indexed at the word level. The distances at the level of words and at the level of characters between artist/album names and \(<img>\) tags is computed thereafter. Using Formula 3.1, where \( p(\cdot) \) refers to the offset of artist \( a \), album \( b \), and image tag \( img \) in the web page, and \( \tau \) is a threshold variable, a set of candidate cover artworks is constructed by fetching the corresponding images.

\[
|p(a) - p(img)| + |p(b) - p(img)| \leq \tau
\]  

(3.1)

Since this set still contains a lot of irrelevant images, content-based filtering is performed. First, non-square images are discarded, using
3.2. Extracting information on music entities

simple filtering by width/height-ratio. Also images showing scanned compact discs are identified and removed from the set. To this end, a circle detection technique is employed. From the remaining set, those images with minimal distance according to Formula 3.1 are output as album covers. On a test collection of 255 albums, this approach yielded correct prediction rates of 83%.

3.2.4 Artist popularity and cultural listening patterns

The popularity of a performer or a music piece can be considered highly relevant. In particular the music business shows great interest in good estimates for the popularity of music releases and promising artists. There are hence several companies that focus on this task, for instance, Musicmetric\textsuperscript{4} and Media Measurement\textsuperscript{5}. Although predicting whether a song will become a hit or not would be highly desirable, approaches to “hit song science” have produced rather disappointing results so far, as shown by Pachet and Roy [190]. In the following, we hence focus on work that describes music popularity rather than predicting it.

To this end, different data sources have been investigated: search engine page counts, microblogging activity, query logs and shared folders of peer-to-peer networks, and play counts of Last.fm users.

Koenigstein and Shavitt [130] analyzed search queries issued in a peer-to-peer network. The authors inferred user locations from IP addresses and were thus able to compare charts created from the query terms with official music charts, such as the “Billboard Hot 100”\textsuperscript{6} in the USA. They found that many artists that enter the “Billboard Hot 100” are already frequently sought for one to two weeks earlier.

Schedl et al. [237] show that the popularity approximations of music artists correlate only weakly between different data sources. A remarkable exception was a higher correlation found between shared folders in peer-to-peer networks and page count estimates, probably explained by the fact that these two data sources accumulate data rather than reflect current trends, like charts based on record sales or postings of

\footnotesize{\textsuperscript{4}http://www.musicmetric.com} \textsuperscript{5}http://www.mediameasurement.com \textsuperscript{6}http://www.billboard.com/charts/hot-100
Figure 3.1: Popularity of songs by Madonna on Twitter. The steady large green bar near the base is her consistently popular song “Like A Prayer”, while the light blue and the orange bars are “Girls Gone Wild” and “Masterpiece”, respectively, by her March 2012 release “MDNA”.

Twitter users. The authors hence conclude that music popularity is multifaceted and that different data sources reflect different aspects of popularity.

More recently, Twitter has become a frequently researched source for estimating the popularity of all kinds of subjects and objects. Hauger and Schedl [97] look into tweets including hash tags that typically indicate music listening, for instance, #nowplaying or #itunes. They employ a cascade of pattern matching approaches to map such tweets to artists and songs. Accumulating the listening events per artist over time, in bins of one week, reveals detailed listening statistics and time-dependent popularity estimates (cf. Figure 3.1). From the figure, an interesting observation can be made. While the share of all-time hits such as “Like a Prayer” or “La isla bonita” remains quite constant over time, songs with spiraling listening activities clearly indicate new record releases. For instance, “Girl Gone Wild” from the album
“MDNA” started its rise in the end of February 2012. However, the album was released not earlier than on March 23. We hence see a pre-release phenomenon similar to the one found by Koenigstein and Shavitt in peer-to-peer data [130].

Leveraging microblogs with attached information about the user’s location further allows to perform an in-depth analysis of cultural listening patterns. While MIR research has focused on Western music since its emergence, recent work by Serra [249] highlights the importance of culture-specific studies of music perception, consumption, and creation. When it comes to music consumption, a starting point to conduct such studies might be data sources such as *MusicMicro* by Schedl [229] or the *Million Musical Tweets Dataset* (MMTD) by Hauger et al. [98], which offer information on listening activities inferred from microblogs, together with temporal and spatial data. One has to keep in mind, though, that such sources are highly biased towards users of *Twitter*, who do not necessarily constitute a representative sample of the overall population.

### 3.3 Music similarity based on the Vector Space Model

The classical Text-IR strategy of document modeling via first constructing a bag-of-words representation of the documents under consideration and subsequently computing a term weight vector for each document was adopted already quite early in MIR research based on music context sources. In the following, we review methods that use as data source either music-related web pages, microblogs, or collaborative tags.

#### 3.3.1 Music-related web pages

Among the earliest works is Whitman and Lawrence’s [302], in which they analyzed a number of term sets to construct corresponding indexes from music-related web pages. These web pages were fetched from the results of queries "artist" music review and "artist" genre style to the *Google* search engine. Adding keywords like “music” or “review” is required to focus the search towards music-related web
pages and disambiguate bands such as “Tool” or “Kiss”. Applying a Part-of-Speech (PoS) tagger by Brill [22] on the corpus of web pages under consideration, Whitman and Lawrence create different dictionaries comprising either noun phrases, adjectives, artist names, unigrams, or bigrams, which they subsequently use to index the web pages. The authors then estimate the similarity between pairs of artists via a distance function computed on the $tf \cdot idf$ vectors of the respective artists. It was shown that indexing n-grams and noun phrases coupled with term weighting outperforms simply indexing artist names and adjectives for the task of artist similarity estimation.

Whitman and Lawrence’s approach [302] was later refined by Bausmann and Hummel in [9]. After having downloaded artist-related web pages in the same way as Whitman and Lawrence did, they employed some filtering methods, such as discarding web pages with a large size and text blocks that do not comprise at least a single sentence. Bausmann and Hummel’s approach further performs keyword spotting in the URL, the title, and the first text block of each web page. The presence of keywords used in the original query to Google increases a page score, which is eventually used to filter web pages that score too low. Another refinement was the use of a logarithmic idf formulation, instead of the simple variant $w_{t,a} = tf_{t,a} / df_t$ employed by Whitman and Lawrence. $tf_{t,a}$ denotes the number of occurrences of term $t$ in all web pages of artist $a$; $df_t$ is the number of web pages in which term $t$ occurs at least once, considering the entire corpus.

Knees et al. [124] further refined earlier approaches by considering all unigrams in the corpus of fetched web pages to construct the index and by using the $tf \cdot idf$ formulation shown in Equation 3.2, where $N$ is the number of pages in the corpus, $tf_{t,a}$ and $df_t$ defined as above.

$$w_{t,a} = (1 + \log tf_{t,a}) \cdot \log \frac{N}{df_t} \quad (3.2)$$

Calculating artist similarities based on the cosine measure between two artists’ $tf \cdot idf$ vectors, the authors achieved up to 77% accuracy in a genre prediction task. In this task, each artist in a collection of 224 artists, equally distributed over 14 genres, was used as query for which the closest artists according to the similarity measure were retrieved. A
retrieved artist was considered relevant if her genre equaled the genre of the query. An extensive comparative study conducted by Schedl et al. [236] assesses the influence on retrieval performance of different aspects in modeling artist similarity based on web pages. On two standardized artist collections, Schedl et al. analyze factors such as different \( tf \) and \( idf \) variants, similarity measures, and methods to aggregate the web pages of each artist. The authors conclude that (i) logarithmic formulations of both \( tf \) and \( idf \) weights perform best, (ii) cosine similarity or Jaccard overlap should be used as similarity measure, and (iii) all web pages of each artist should be concatenated into one big document that represents the artist. However, they also notice that a small change of a single factor can sometimes have a strong impact on performance. Although Schedl et al. investigate several thousands of combinations of the aforementioned aspects and perform experiments on two music collections, the results might only hold for popular artists as both collections omit artists from the “long tail”. Their choice of using genre as proxy for similarity can also be questioned, but is quite common in MIR experiments.

### 3.3.2 Microblogs

A similar study, but this time using as data source microblogs is presented by Schedl [228]. The author queried Twitter over a period of three months for microblogs related to music artists. Tweets including the artist name (and optionally the term “music”) have been gathered, irrespective of the user. Similar to the studies on web pages presented above, each microblog is treated as a document. Different aggregation strategies to construct a single representation for each artist are investigated. In addition, various dictionaries to index the resulting corpus are considered. Evaluation is conducted again using genre as relevance criterion, similar to Schedl’s earlier investigations on web pages [236]. The average over the mean average precision (MAP) values resulting from using as query each artist in the collection is used as performance measure. It is shown that 64% MAP can be reached on the collection of 224 artists proposed by Knees et al. [124], already introduced above. Results of more than 23,000 single experiments yielded the following
findings: (i) query scheme "artist" without any additional keywords performs best (otherwise the set of tweets is too restricted), (ii) most robust MAP scores are achieved using a domain-specific index term set, (iii) normalizing documents does not improve results (because of the small variance in length of tweets), and (iv) for the same reason, Inner product as similarity measure does not perform significantly worse than cosine. As with the study presented on web pages [236] in the last section, these findings may not be generalizable to larger music collections including lesser known artists. However, the data sparsity for such “long tail” artists is not specific to microblogs, but a general problem in context-based MIR, as already pointed out by Celma [30] and Lamere [135] (cf. Section 3.5).

### 3.3.3 Collaborative tags

During the past few years, users of social music platforms and players of tagging games (cf. Section 4.4) have created a considerable amount of music annotations in the form of collaborative tags. These tags hence represent a valuable source for music similarity and retrieval tasks. Tag-based music retrieval approaches further offer some advantages over approaches based on web pages or microblogs: (i) the dictionary used for indexing is much smaller, typically less noisy, and includes semantically meaningful descriptors that form a folksonomy7 and (ii) tags are not only available on the artist level, but also on the level of albums and tracks. On the down side, however, considerable tagging coverage requires a large and active user community. Moreover, tag-based approaches typically suffer from a “popularity bias”, i.e. tags are available in abundance for popular artists or songs, whereas the “long tail” of largely unknown music suffers from marginal coverage (Celma [30]; Lamere [135]). This is true to a smaller extent also for microblogs. The “community bias” is a further frequently reported problem. It refers to the fact that users of a particular music platform that allows tagging, for instance Last.fm, seldom correspond to the average music listener. As these biases yield distortions in similarity estimates, they are detri-

---

7A folksonomy is a user-generated categorization scheme to annotate items. Unlike a taxonomy, a folksonomy is organized in a flat, non-hierarchical manner.
3.4. **Music similarity based on Co-occurrence Analysis**

Using collaborative tags extracted from music platforms, Levy and Sandler [153] aim at describing music pieces in a semantic space. To this end, they gather tags from *Last.fm* and *MusicStrands*[^8], a former web service for sharing playlists. The tags found for each track are tokenized and three strategies to construct the term vector space via $tf \cdot idf$ vectors are assessed: (i) weighting the $tf_{t,p}$ value of tag $t$ and music piece $p$ using as weight the number of users who assigned $t$ to $p$, (ii) restricting the space to tags occurring in a dictionary of adjectives, and (iii) use standard $tf \cdot idf$ weighting on all tags. Similarities between $tf \cdot idf$ vectors are computed as cosine similarity. To evaluate their approach, Levy and Sandler construct a retrieval task in which each track serves as seed once. MAP is computed as performance measure, using matching genre labels as proxy for relevance, as described above. The authors find that using a dictionary of adjectives for indexing worsens retrieval performance. In contrast, incorporating user-based tag weighting improves MAP. They, however, raise questions about whether this improvement in MAP is truly important to listeners (cf. Section 5). Finally, the authors also investigate dimensionality reduction of the term weight vectors via Latent Semantic Analysis (LSA) [46], which is shown to slightly improve performance.

Geleijnse et al. [71] exploit *Last.fm* tags to generate a “tag ground truth” for artists. Redundant and noisy tags on the artist level are first discarded, using the tags assigned to the tracks by the artist under consideration. Artist similarities are then calculated as the number of overlapping tags in corresponding artists’ tag profiles. Evaluation against the “similar artists” function provided by *Last.fm* shows a significantly higher number of overlapping tags between artists *Last.fm* judges as similar than between randomly selected pairs of artists.

### 3.4 Music similarity based on Co-occurrence Analysis

Compared to the approaches relying on the Vector Space Model, which have been elaborated on in the previous section, approaches based on

[^8]: [http://music.strands.com](http://music.strands.com)
co-occurrence analysis derive similarity information from counts of how frequently two music entities occur together in documents of a music-related corpus. The underlying assumption is that two music items or artists are similar if they frequently co-occur. Approaches reflecting this idea have been proposed for different data sources, the most prominent of which are music playlists, peer-to-peer networks, web pages, and recently microblogs.

3.4.1 Music playlists

The earliest approaches based on the idea of using co-occurrences to estimate contextual music similarity exploited music playlists of various kinds. Pachet et al. [191] consider playlists of a French radio station as well as playlists given by compilation compact discs. They compute relative frequencies of two artists’ or songs’ co-occurrences in the set of playlists under consideration. These relative frequencies can be thought of as an approximation of the probability that a given artist \( a_i \) occurs in a randomly selected playlist which is known to contain artist \( a_j \). After correcting for the asymmetry of this function, the resulting values can be used as similarity measure. The corresponding similarity function is shown in Equation 3.3, in which \( f(a_i) \) denotes the total number of playlists containing artist \( a_i \) and \( f(a_i, a_j) \) represents the number of playlists in which both artists \( a_i \) and \( a_j \) co-occur.

\[
sim(a_i, a_j) = \frac{1}{2} \left[ \frac{f(a_i, a_j)}{f(a_i)} + \frac{f(a_j, a_i)}{f(a_j)} \right]
\]  

(3.3)

A shortcoming of this simple approach is that Equation 3.3 is not capable of capturing indirect links, i.e. inferring similarity between artists \( a_i \) and \( a_k \) from the fact that artists \( a_i \) and \( a_j \) as well as artists \( a_j \) and \( a_k \) frequently co-occur. This is why Pachet et al. further propose the use of Pearson’s correlation coefficient between co-occurrence vectors of artists \( a_i \) and \( a_j \) to estimate similarity. Assuming that the set of music playlists under consideration contains \( N \) unique artists, the \( N \)-dimensional co-occurrence vector of \( a_i \) contains in each dimension \( u \) the frequency of co-occurrences of artists \( a_i \) with \( a_u \).

Assessing both methods on a small set of 100 artists, Pachet at
al. found, however, that the direct co-occurrence approach outperforms the correlation-based co-occurrences.

A few years after Pachet et al.’s initial work, Baccigalupo et al. [4] benefited from the trend of sharing user-generated content. They gathered over one million user-generated music playlists shared on MusicStrands, identified and subsequently filtered the most popular 4,000 artists. To estimate similarity between artists $a_i$ and $a_j$, the authors also rely on co-occurrence counts. In addition, Baccigalupo et al. consider important the distance at which $a_i$ and $a_j$ co-occur within each playlist. The overall dissimilarity between $a_i$ and $a_j$ is hence computed according to Equation 3.4, where $f_h(a_i, a_j)$ denotes the number of playlists in which $a_i$ and $a_j$ co-occur at a distance of $h$, i.e. having exactly $h$ other artists in between them. The authors empirically determined weights for different values of $h$: $\beta_0 = 1.00$, $\beta_1 = 0.80$, and $\beta_2 = 0.64$. To account for the popularity bias, $\text{dis}(a_i, a_j)$ is eventually normalized with the distance to the most popular artist. Baccigalupo et al. do not evaluate their approach for the task of similarity measurement, but propose it to model multi-genre affinities for artists.

$$\text{dis}(a_i, a_j) = \sum_{h=0}^{2} \beta_h \cdot [f_h(a_i, a_j) + f_h(a_j, a_i)]$$

### 3.4.2 Peer-to-peer networks

Information about artist or song co-occurrences in music collections shared via peer-to-peer networks is another valuable source to infer music similarity. Already in the early years of MIR research, Whitman and Lawrence [302] targeted this particular source in that they acquired 1.6 million user-song relations from shared folders in the OpenNap\(^9\) network. From these relations, the authors propose to estimate artist similarity via Equation 3.5, where $f(a_i)$ is the number of users who share artist $a_i$ and $f(a_i, a_j)$ is the number of users who share both artists $a_i$ and $a_j$. The final term in the expression mitigates the popularity bias by dividing the difference in popularity between $a_i$ and $a_j$.

\(^9\)http://opennap.sourceforge.net
by the maximum popularity of any artist in the collection.

$$sim(a_i, a_j) = \frac{f(a_i, a_j)}{f(a_j)} \cdot \left(1 - \frac{|f(a_i) - f(a_j)|}{\max_k f(a_k)}\right)$$

(3.5)

More recently, Shavitt and Weinsberg [255] proposed an approach to music recommendation, which makes use of metadata about audio files shared in peer-to-peer networks. The authors first collected information on shared folders for 1.2 million Gnutella [213] users. In total, more than half a million individual songs were identified. The user-song relations are then used to construct a 2-mode-graph modeling both. A user sharing a song is simply represented by an edge between the respective song and user nodes. Shavitt and Weinsberg found that a majority of users tend to share similar songs, but only a few unique ones. Clustering the user-artist matrix corresponding to the 2-mode-graph (by simple $k$-means clustering), the authors construct an artist recommender that suggests artists listened to by the users in the same cluster as the target user. They further propose an approach to song recommendation that alleviates the problem of popularity bias. To this end, distances between songs $s_i$ and $s_j$ are computed according to Equation 3.6, where $f(s_i, s_j)$ denotes the number of users who share both songs $s_i$ and $s_j$ and $c(s_i)$ refers to the total number of occurrences of $s_i$ in the entire corpus. The denominator corrects the frequency of co-occurrences in the numerator by increasing distance if both songs are very popular, and are hence likely to co-occur in many playlists, regardless of their actual similarity.

$$\text{dis}(s_i, s_j) = -\log_2 \left( \frac{f(s_i, s_j)}{\sqrt{c(s_i) \cdot c(s_j)}} \right)$$

(3.6)

Although evaluation experiments showed that average precision and recall values are both around 12%, Shavitt and Weinsberg claim these to be quite good results, given the real-world dataset, in particular the large number of songs and the high inconsistencies in metadata.

3.4.3 Web pages

There are also a few works on music-related co-occurrence analysis drawing from web pages. Among the earliest, Zadel and Fujinaga [308]
3.4. Music similarity based on Co-occurrence Analysis

use an Amazon\textsuperscript{10} web service to identify possibly related artists in a given artist collection. In order to quantify two artists’ degree of relatedness $\text{sim}(a_i, a_j)$, Zadel and Fujinaga subsequently query the Google search engine and record the page count estimates $pc(a_i, a_j)$ and $pc(a_i)$, respectively, for the query "artist $a_i$" "artist $a_j$" and "artist $a_i$", for all combinations of artists $a_i$ and $a_j$. The normalized co-occurrence frequencies are then used to compute a similarity score between $a_i$ and $a_j$, as shown in Equation 3.7.

$$
\text{sim}(a_i, a_j) = \frac{pc(a_i, a_j)}{\min(pc(a_i), pc(a_j))} \\
(3.7)
$$

Schedl et al. [234] propose a similar approach, however, without the initial acquisition step of possibly related artists via a web service. Instead, they directly query Google for each pair of artists in the collection. The authors also investigate different query schemes, such as "artist $a_i$" "artist $a_i$" music review, to disambiguate artist names that equal common speech, for instance “Pink”, “Kiss”, or “Tool”. Also a slightly different similarity measure is employed by Schedl et al., namely the measure given in Equation 3.3, where $f(a_i, a_j)$ denotes the page count estimate for queries of the form "artist $a_i$" "artist $a_j$" [music-related keywords] and $f(a_i)$ denotes this estimate for queries "artist $a_i$" [music-related keywords]. Evaluation on a test collection of 224 artists, uniquely distributed over 14 genres, yielded an overall precision@$k$ of 85\%, when the relevance of a retrieved artist is defined as being assigned to the same genre as the query artist.

Despite the seemingly good performance of this approach, a big shortcoming is that the number of queries that need to be issued to the search engine grows quadratically with the number of artists in the collection, which renders this approach infeasible for real-world music collections. Mitigating this problem, Cohen and Fan [36] as well as Schedl [226] propose to download a number of top-ranked web pages retrieved by Google as result to the query "artist $a_i$" [music-related keywords], instead of recording pairwise page count estimates. The

\textsuperscript{10}http://www.amazon.com
fetched web pages are then indexed using as index terms the list of all artists in the music collection under consideration. This allows to define a co-occurrence score as the relative frequency of artist $a_i$’s occurrence on web pages downloaded for artist $a_j$. Similarities are then estimated again according to Equation 3.3. More important, this approach decreases the number of required queries to a function linear in the size of the artist collection, without decreasing performance.

### 3.4.4 Microblogs

Quite recently, co-occurrence approaches to music similarity from microblogs have been proposed by Zangerle et al. [309] and by Schedl et al. [232]. So far, all of them use Twitter for data acquisition. To this end, it is first necessary to identify music-related messages in a stream of tweets. Both Zangerle and Schedl achieve this by filtering the stream according to hashtags, such as #nowplaying or #music. Mapping the remaining content of the tweet to known artist and song names (given in a database), it is possible to identify individual listening events of users. Aggregating these events per user obviously yields a set of songs the respective user indicated to have listened to, which represents a simple user model. Computing the absolute number of user models in which songs $s_i$ and $s_j$ co-occur, Zangerle et al. [309] define a similarity measure, which they subsequently use to build a simple music recommender. In contrast, Schedl et al. [232] conduct a comprehensive evaluation of different normalization strategies for the raw co-occurrence counts, however, only on the level of artists instead of songs. They found the similarity measure given in Equation 3.8 to perform best when using the “similar artist” relation from Last.fm as ground truth.

$$sim(a_i, a_j) = \frac{f(a_i, a_j)}{\sqrt{f(a_i) \cdot f(a_j)}}$$ (3.8)

### 3.5 Discussion and challenges

Although features extracted from contextual data sources are used successfully stand-alone or to complement content-based approaches for
3.5. Discussion and challenges

music similarity, retrieval, and information extraction tasks, several challenges are faced when exploiting them:

- **Availability of data:** Although we require only a piece of metadata (e.g. band name), coverage in web- and social media-based data sources is typically sparse, particularly for lesser known music.

- **Level of detail:** As a consequence of sparse coverage, usually information can only be found in sufficient amounts on the level of artists and performers, but not of songs. To give an example, Lamere [135] has shown that the average number of tags assigned to each song on Last.fm equals only 0.25.

- **Noisy data:** Web pages and microblogs that are irrelevant for the requested music item or artist, as well as typos in collaboratively generated tags, are examples of noise in contextual data sources.

- **Community bias:** Users of social music platforms are not representative of the entire population of music listeners. For instance, the genre Viking Metal has the same importance as the genre Country among users of Last.fm, based on an analysis by Lamere [135]. As a consequence, the amount of information available can be very unbalanced between different styles of music and only reflects the interest of the community that uses the platform under consideration.

- **Hacking and vandalism:** Users of social music platforms who deliberately inject erroneous information into the system are another problem. For example, as pointed out by Lamere [135], Paris Hilton was for a long time the top recommended artist for the genre “brutal death metal” on Last.fm, which can only be interpreted as a joke.

- **Cold start problem:** Newly released music pieces or albums do not have any coverage on the web or in social media (except for pre-release information). In contrast to music content-based methods which can immediately be employed as soon as the audio
is available, music context approaches require some time until information becomes available.

- **Popularity bias:** Artists or songs that are very popular may unjustifiably influence music similarity and retrieval approaches. To give an example, in music similarity computation, the popularity bias may result in artists such as Madonna being estimated as similar to almost all other artists. Such undesirable effects typically lead to high “hubness” in music recommendation systems as shown by Schnitzer et al. [245], meaning that extraordinarily popular artists are recommended very frequently, disfavoring lesser known artists, and in turn hindering serendipitous music encounters.

Despite these challenges, music retrieval and recommendation based on contextual data have been proved very successful, as underlined for instance by Slaney [259].
The user plays a key role for all MIR applications. Concepts and tasks such as similarity, semantic labels, and structuring music collections are strongly dependent on users’ cultural background, interests, musical knowledge, and usage intention, among other factors. User properties relate directly to the notion of a personalized system incorporating static or only slowly changing aspects, while user context relates to the notion of a context-aware system that continuously adapts to dynamic changes in the user’s environment or her intrinsically affective and cognitive states. It is known in MIR and related fields that several concepts used to develop and evaluate systems are subjective, thus varying between individuals (e.g. relevance or similarity). However, not until recently are these user- and culture-specific aspects being integrated when elaborating music retrieval and recommendation approaches.

In this section, we review the main efforts within the MIR community to model and analyze user behavior and to incorporate this knowledge into MIR systems. To this end, we start in Section 4.1 with a summary on empirical user studies performed in MIR, and some inferred design recommendations. Subsequently, we present in Section 4.2 the main categories of approaches to model users in MIR and to in-
corporate these models into retrieval and recommendation systems. As the notion of musical similarity is of particular importance for MIR, but depends on individual perceptual aspects of the listener, we review methods on adaptive music similarity measures in Section 4.3. Several “games with a purpose” for semantic labeling of music are presented in Section 4.4. Given their direct user involvement, such games are a valuable source for information that can be incorporated into user-centric MIR applications. At the same time, they represent MIR applications themselves. In Section 4.5, we eventually present two applications that exploit users’ listening preferences, either by questionnaires or postings about music listening, in order to build music discovery systems.

4.1 User studies

As pointed out by Weigl and Guastavino [300], the MIR field has been more focused on developing systems and algorithms than on understanding user needs and behavior. In their review of the literature on empirical user studies, they found out that research focuses on different aspects: general user requirements, user requirements in specific contexts, preference and perception modeling (e.g., factors for disliking songs or effects of musical expertise and culture), analysis of textual queries, employment of user studies to generate ground truth data for evaluation (see Section 5), organization of music collections, strategies when seeking new music and information behavior in passive or serendipitous encounters with new music. Weigl and Guastavino conclude that there is not one standard methodology for these experiments and there is a bias towards qualitative studies and male subjects from similar backgrounds. The authors make a few recommendations for MIR system design, summarized as follows:

- **Undirected browsing:** emphasis should be placed on serendipitous discovery processes by means of browsing applications, where the user should be provided with some “entry points” to the catalogue. Audio preview (by intelligent music summaries) and visual representations of the music (e.g., album covers or symbolic representations) are identified as useful features for a system.
4.1. User studies

- *Goal-directed search and organization*: allow for different search strategies to retrieve music in specific contexts, as individuals prefer different approaches to search for new music according to their background, research experience, and application (e.g., search by similarity, textual queries, or music features). In addition, people organize music on the basis of the situation in which they intend to listen to it, so the incorporation of the user context can be valuable for MIR systems.

- *Social- and metadata-based recommendations*: while editorial metadata is widely used to organize music collections, MIR systems should allow “fuzzy” search on it and the possibility for users to define their own metadata. In addition, social aspects in music discovery and recommendation are a key component to integrate in MIR systems.

- *User devices and interfaces*: user interfaces should be simple, easy to use, attractive, playful, and should include visual representations of music items. Interfaces may also be adapted to different audiences (e.g., children, young users or elderly people). Portable devices seem to be a good option for MIR systems because they can be used ubiquitously, and online support should be available, including supporting descriptors of the search criteria.

In a recent study, Lee and Cunningham [146] analyze previous user studies related to MIR, which they categorize as “studies of users” and “studies involving users”. In particular, they further categorize as: empirical studies on the needs and behaviors of humans, experiments involving users on a particular task, analysis of user-generated data, and surveys and reviews of the above. Their results corroborate the widespread appeal of music as a subject for research, as indicated by the diversity of areas and venues these studies originated from, as well as their citation patterns. They also argue that MIR is a fast-changing field not only for researchers, but also for end users. For instance, Lee and Waterman [144] observed clear changes in the popularity of music platforms, illustrating that what users need and what they expect from music services is most likely changing rapidly as well.
Lee and Cunningham also observed that many user studies were based on small user samples, and likely biased too because of the sampling methods used. To this threat to validity they also add the possibility of a novelty bias by which users tend to prefer new systems or interfaces just because they are new. This effect could also be amplified in many cases where there is a clear selection bias and the users of the study tend to be recruited from the same institution as the researchers. Finally, they observe a clear disconnect between how MIR tasks are designed to evaluate systems, and how end users are supposed to use those systems; they conclude that suggestions made in the user studies can be difficult and costly to implement, especially in the long run.

4.2 Computational user modeling

In what follows, we give a brief overview of strategies to incorporate user-centric information into music retrieval and recommendation systems. Such strategies can be divided into (i) personalization based on static (explicit or implicit) ratings, (ii) dynamically adapting the retrieval process to immediate user feedback, and (iii) considering comprehensive models of the user and her context.

4.2.1 Personalization based on ratings

Current personalized music access systems typically model the user in a rather simplistic way. It is common in collaborative filtering approaches, such as the ones by Sarwar et al. [224] and Linden et al. [156], to build user profiles only from information about a user $u$ expressing an interest in item $i$. This expression of interest can either be given by explicit feedback or derived from implicit feedback. An example for the former are “like” or “dislike” buttons provided to the user. The latter can be represented by skipping a song in a playlist.

As a very simple form, interest can be inferred from clicking events on a particular item, from purchasing transactions, or from listening events to music pieces. These interest-relationships between user $u$ and item $i$ are then stored in a binary matrix $R$, where element $r_{u,i}$ de-
notes the presence or absence of a relationship between $u$ and $i$. A slightly more elaborate representation is the one typically employed by the Recommender Systems community, which consists in using explicit ratings instead of binary values to represent $R$. To this end, Likert-type scales that allow users to assign “stars” to an item are very frequent, a typical choice being to offer the user a range from one to five stars. For instance, Koren et al. [131] followed this approach to recommend novel items via matrix factorization techniques.

### 4.2.2 Dynamic user feedback

An enhancement of these static rating-based systems are systems that directly incorporate explicit user feedback. Nürnberger and Detyniecki [184] propose a variant of the Self-Organizing Map (cf. the nep-Tune interface in Section 1.4.4) which adapts to user feedback. While the user visually reorganizes music items on the map, the clustering of the SOM changes accordingly. Knees and Widmer [129] incorporated relevance feedback [214] into a text-based, semantic music search engine to adapt the retrieval process. Pohle et al. [205] present an adaptive music retrieval system, based on users weighting concepts. To this end, a clustering of collaborative tags extracted from Last.fm is performed, from which a small number of musical concepts are derived via Non-Negative Matrix Factorization (NMF) [142]. A user interface then allows for adjusting the importance or weights of the individual concepts, based on which artists that best match the resulting distribution of the concepts are recommended to the user.\footnote{Due to its integration into Last.fm and the resulting legal issues, we cannot give a screenshot of the system here. The interested reader may, however, contact the first author for more details.} Zhang et al. [310] propose a very similar kind of personalization strategy via user-adjusted weights.

### 4.2.3 Context-awareness

Approaches for context-aware music retrieval and recommendation differ significantly in terms of how the user context is defined, gathered, and incorporated. The majority of them rely solely on one or a few
aspects. For instance, Cebrian et al. [29] used temporal features, and Lee and Lee [143] used listening history and weather conditions. On the other hand, comprehensive user models are rare in MIR. One of the few exceptions is Cunningham et al.’s study [42] that investigates if and how various factors relate to music taste (e.g., human movement, emotional status, and external factors such as temperature and lightning conditions). Based on the findings, the authors present a fuzzy logic model to create playlists.

Some works target mobile music consumption, typically matching music with the current pace of the user while doing sports (Moens et al. [175]; Biehl et al. [16]; Elliott and Tomlinson [62]; Dornbush et al. [49]; Cunningham et al. [42]). To this end, either the user’s location or heartbeat is used to infer jogging or walking pace. Kaminskas and Ricci [113] aim at matching tags that describe a particular place of interest, such as a monument, with tags describing music. Employing text-based similarity measures between the two sets of tags, they build a system for location-aware music recommendation. Baltrunas et al. [7] suggest a context-aware music recommender for car driving. To this end, they take into account eight different contextual factors, including driving style, mood, road type, weather, and traffic conditions. Their model adapts according to explicit human feedback. A more detailed survey on personalized and context-aware music retrieval is given by Schedl et al. [230].

4.3 User-adapted music similarity

There have been some efforts to adapt music similarity measures according to the user. Schedl et al. [241] summarize three different strategies. The first one (direct manipulation) consists in letting users control the weight of the different musical descriptors (e.g., tempo, timbre, or genre) for the final similarity measure. This approach requires much user effort for a high number of descriptors, and is limited by the fact that the user should make her or his preference explicit. The second strategy is based on gathering user feedback on the similarity of pairs of songs, which is further exploited to adjust the similarity model. The
third strategy is based on collection clustering, that is, the user is asked
to group songs in a 2-D plot (e.g. built by means of Self-Organizing
Maps), and each movement of a song causes a weight change in the
underlying similarity measure.

One can also consider the problem of adapting a music similarity
measure as a metric learning problem subject to so-called relative dis-
tance constraints, so that the task of learning a suitable adaptation
of a similarity measure can be formulated as a constraint optimization
problem. A comprehensive work on the adaptation of the different steps
of a MIR system is provided by Stober [269]; feature extraction, defi-
nition of idiosyncratic genres adapted to the user’s personal listening
habits, visualization and music similarity.

Assuming perception of music and hence quality judgment of mu-
sic recommendations are influenced by the position (GPS coordinates)
and location (semantically meaningful indication of spatial position) of
the user, Schedl and Schnitzer [239, 238] propose methods to integrate
this kind of information into a hybrid similarity measure. This hybrid
similarity measure encodes aspects of music content, music context,
and user context (cf. Section 1.3). The former two are addressed by
linearly combining state-of-the-art similarity functions based on music
content (audio signal) and music context (web pages). The user con-
text is then integrated by weighting the aggregate similarity measure
according to the spatial distance of all other users to the seed user
requesting music recommendations. To this end, Schedl and Schnitzer
exploit the MusicMicro dataset of geo-annotated music listening events
derived from microblogs [229]. They first compute for each user \(u\) the
geo-spatial centroid of her listening activity \(\mu(u)\), based on all of her
listening-related tweets. To recommend music to \(u\), the geodesic dis-
tance \(g(u, v)\) between \(\mu(u)\) and \(\mu(v)\) is computed for all potential target
users \(v\). The authors incorporate \(g(u, v)\) into a standard collaborative
filtering approach, giving higher weight to nearby users than to users
far away. They experiment with linear and with exponential weighting
of the geodesic distance. Conducting cross-fold validation experiments
on the MusicMicro collection, it is shown that such a location-specific
adaptation of music similarities by giving higher weights to geograph-
ically close users can outperform both standard collaborative filtering and content-based approaches.

4.4 Semantic labeling via games with a purpose

During the past few years, the success of platforms fostering collaborative tagging of all kinds of multimedia material has led to an abundance of more or less meaningful descriptors of various music entities (e.g., performers, composers, albums, or songs) As such tags establish a relationship between music entities and users, they can be regarded as contributing to a user profile. The platform most frequently exploited in the context of MIR is certainly Last.fm. An overview of methods using the Last.fm folksonomy was already given in Section 3.3.3.

However, one shortcoming when working with Last.fm tags is that many of them are irrelevant to create a descriptive, semantic profile; for instance, opinion tags such as “love”, “favorite”, or “great live band” do not contribute a lot to a semantic artist profile, compared to more objective labels such as instruments or epochs. Less noisy and more meaningful tags should result from users playing games with a purpose (GWAP). The idea of these games is to solve problems that a computer cannot solve, i.e. problems that require human intelligence. They obviously have to be entertaining enough to attract and keep users playing. Such games have been used first in 2004 to label images, via the ESP game [293].

In the field of MIR, Law et al. proposed the TagATune game in 2007 [140]. In TagATune, two players are paired and played the same sound or song. Their only means of communication is via text messages. The players are not explicitly told to provide descriptors, but to guess what their partners are thinking. In contrast to the ESP game, TagATune was found to yield much more subjective, ambiguous, and imaginative labels, which is likely the result of a higher variance in human perception of music than of images. To remedy this problem, Law et al. refined their game and based it on a method they call “input agreement” [141]. In the new version of TagATune, a screenshot of which is depicted in Figure 4.1, two players are again paired, but
4.4. Semantic labeling via games with a purpose

are then either played two different songs or same songs. They have to find out as quickly as possible whether their input songs match or not. Law and van Ahn show that this setting is better suited to obtain objective and stable semantic descriptors. Unlike in the first version, participants frequently used negated key words, such as “no guitar”. Law and van Ahn further claim that games based on input agreement are more popular and yield a higher number of tags. TagATune also offers a bonus round, in which users are presented three songs, one seed and two target songs. Users have to choose which of the targets is more similar to the seed. This yields a dataset of relative similarity judgments. From such a dataset, similarity measures claimed to reflect human perception of music better than measures based on audio content analysis can be learned, as shown by Wolff and Weyde [303] as well as Stober [269], also see Section 4.3.

Another GWAP for music annotation is the ListenGame, presented by Turnbull et al. [277]. Players are paired and played the same song. They subsequently have to choose from a list of words the one that best and the one that worst describes the song. Users get immediate feedback about which tags other players have chosen. To the collected data, Turnbull et al. apply Mixture Hierarchies Expectation Maximization (MH-EM) [291] to learn semantic associations between words and
songs. These associations are weighted and can therefore be used to construct tag weight vectors for songs and in turn to define a similarity measure for retrieval.

Mandel and Ellis present in [164] another GWAP called *MajorMiner*. It differs from the other games presented so far in that it uses a more fine-grained scoring scheme. Players receive more points for new tags to stimulate the creation of a larger semantic corpus. More precisely, a player who first uses a tag $t$ to describe a particular song scores two points if $t$ is later confirmed (used again) by another player. The third and subsequent players that use the same tag $t$ do not receive any points.

Kim et al. [117] designed a GWAP called *MoodSwings*, where users provide mood descriptors for songs. However, unlike in the other games, these tags are not associated to the whole song but to specific points in time. Two players listen to the same music clip simultaneously, and move their mouse around a game board representing the valence-arousal space. The mood of each player is sampled every second, and the mood of the other player is displayed every few seconds. The more the two players agree with each other, the more points they score.

4.5 Music discovery systems based on user preferences

One way to obtain information about users is by assessing their music listening preferences. Hanani et al. [92] identified two main strategies: inferring information from user behavior or respective data on a large scale or by means of surveys and questionnaires to explicitly gather qualitative statements and ratings.

In the following, we present two systems that gather musical preferences and integrate them into music access systems: *Music Avatar* and *Music Tweet Map*. While the former gathers user preferences from questionnaires, the latter infers such information from microblogs identified as referring to listening events.
4.5. Music discovery systems based on user preferences

4.5.1 Musical preferences and their visualization

Bogdanov et al. [17] present the Music Avatar project as an example of musical preference modeling and visualization, where musical preference is modeled by analyzing a set of preferred music tracks provided by the user in questionnaires. Different low-level and semantic features are computed by means of automatic classification, following the methods introduced in Section 2. Next, the system summarizes these track-level descriptors to obtain a user profile. Finally, this collection-wise description is mapped onto the visual domain by creating a humanoid cartoony character that represents the user’s musical preferences, as illustrated in Figure 4.2. This user modeling strategy has been further exploited by the authors in the context of music recommendation [17].

In addition to static musical preferences, interesting information is provided by listening patterns. Herrera et al. [101] propose an approach to analyze and predict temporal patterns in listening behaviors with the help of circular statistics. They show that for certain users, artists and genres, temporal patterns of listening behavior can be exploited by MIR systems to predict music listening preference.

---

\(^2\) http://mtg.upf.edu/project/musicalavatar
4.5.2 Visual analysis of geo-located music listening events

Interesting insights can also be gained by contrasting listening patterns among users and locations. Hauger and Schedl [97] study location-specific listening events and similarity relations by filtering the Twitter stream for music-related messages that include hash tags such as #nowplaying and subsequently indexing the resulting tweets using lists of artist and song names. Hauger et al. [98] construct a dataset of music listening activities of microbloggers. Making use of position information frequently revealed by Twitter users, the resulting location-annotated listening events can be used to investigate music preferences around the world, and to construct user-specific, location-aware music recommendation models. The former is made possible by user interfaces such as Music Tweet Map\(^3\); the latter is dealt with in the next section.

The Music Tweet Map offers a wide range of functions, for instance, exploring music listening preferences according to time and location, analyzing the popularity of artists and songs over time, exploring artists similar to a seed artist, clustering artists according to latent topics, and metadata-based search, as a matter of fact. To give some illustrations of these capabilities, Figure 4.3 shows listening activities in the Netherlands. The number of tweets in each region is illustrated by the size of the respective circle. Different colors refer to different topics, typically related to genre. Figure 4.4 shows how to access songs at a specific location, here the Technical University of Delft. The illustration further reveals statistics per tweet and per user and respective topic distributions. Artists similar to a given seed, in this case Eminem, can be explored as shown in Figure 4.5. Different shades of red indicate the similarity level to the seed, whereas listening events to the seed itself are depicted in black. For an illustration of the artist popularity charts, see Figure 3.1.

4.6 Discussion and challenges

As investigated in this section, research on user-aware music retrieval is still in its infancy. Although some promising first steps into the right

\(^3\)http://www.cp.jku.at/projects/MusicTweetMap
4.6. Discussion and challenges

Figure 4.3: Exploring listening patterns by microbloggers in the Netherlands, using Music Tweet Map.

direction have been made, almost all work models the user in a quite simplistic way, for instance, via musical genre preference or time and location of music listening. In some more recent works, specific music consumption scenarios are addressed, for instance listening while driving by Baltrunas et al. [7] or while doing sports by Moens et al. [175].
Even though these approaches already enable personalized music recommendation systems, falling short of regarding the user and her context in a comprehensive way, user satisfaction of resulting systems tends to be low. Kaminskas et al. [114] show this by contrasting personalized with context-aware algorithms. Again, a personalized system is one that models the user in a static way, for instance, via general listening preferences or musical education; whereas a context-aware system is one that dynamically adapts its user model according to changes in the user’s intrinsic or extrinsic characteristics, such as affective state or environmental surrounding, respectively. Another important aspect to increase user satisfaction, and shortcoming of most existing approaches, is to explain results of a music retrieval or music recommendation system to the end users, so they can understand why a particular item has been recommended.

Many questions related to user-centric music retrieval and recommendation still require extensive research. Among these, some of the most important ones are: how to model the user in a comprehensive way; which aspects of the user properties and the user context are the most important ones for which music retrieval task; how user properties and context influence music perception and preference; whether to
4.6. Discussion and challenges

Figure 4.5: Exploring artists similar to Eminem, listened to in the USA, using Music Tweet Map.

take culture-specific aspects into account and, if so, how; how the user’s musical preference and current affective state influence each other; and provided we gain deep insights into the above issues, how to eventually build user-centric systems for different usages.
Evaluation of MIR systems is typically based on test collections or datasets [222], following the Cranfield paradigm traditionally employed in Text IR [93]. Nonetheless, there are some clear differences in how both fields have evolved during the past decade [289, 55].

The Text IR field has a long tradition of conferences mainly devoted to the evaluation of retrieval systems for the variety of tasks found in the field. Examples are the Text REtrieval Conference (TREC) [295], the National Institute of Informatics-Testbeds and Community for Information access Research (NTCIR) [115], the Conference and Labs of the Evaluation Forum (CLEF) [21] or the INitiative for the Evaluation of XML retrieval (INEX) [87]. Every year, a programme committee selects a set of tasks for which to evaluate new systems, based on the general interests of the research community, the state of the art in each case, and the availability of resources. Each task is then organized by a group of experts, who design the evaluation experiments, select the evaluation measures to score systems, find or create a suitable test collection, and plan and schedule all phases of the experiments. Research teams interested in participating in a specific task can use the published data to run their systems and submit the output back to the
task organizers. Using various resources, the organizers then evaluate all submitted systems and publish the results of the experiment. During the actual conference, organizers discuss the results and participants show their approach to solve the problem, thus fostering cross-team collaboration and refinement of retrieval techniques. In addition, these experiments often serve as testbeds to try or validate new evaluation methods that would otherwise remain very difficult to study because they usually require large amounts of resources.

Similar evaluation conferences have appeared in other fields related to Multimedia. TRECVID [260] began in 2001 as part of the TREC series, and has continued as a stand-alone conference dedicated to video retrieval since 2003. ImageCLEF [176] started in 2003 as part of the CLEF series, dedicated to evaluating systems for image retrieval. MediaEval [137] started in 2010 as a continuation of the VideoCLEF task hosted in 2008 and 2009, though it has focused on a variety of multimedia tasks related not only to video, but also to audio, image, etc. Unfortunately, no such evaluation conference exists in MIR, even though MIREX has been established as the de facto evaluation forum alongside the annual ISMIR conference.

This section elaborates on the complexity of evaluating MIR systems and describes the evaluation initiatives that have appeared with the years. Specific research on evaluation in MIR is outlined later, which describes the current status of the matter and the challenges found as of today.

5.1 Why evaluation in Music Information Retrieval is hard

5.1.1 Complexity of musical information

As early pointed out by Downie [53], evaluation in Music IR differs in several ways from evaluation in Text IR. The most important difference is related to the availability of data. For instance, textual documents and images are readily available on the Internet, but this is not the case for music. Obtaining music files is expensive due to rights-holders and copyright laws, so the creation of publicly accessible collections has been practically impossible, let alone their creation at a large-scale. Re-
searchers can not generally resort to user-generated music documents either because their creation is not as ubiquitous as text, video or images. Every regular user can write a blog post or upload pictures or videos taken with a camera or cell phone, but recording a music piece requires a certain degree of musical knowledge and equipment. The result has been that research teams acquired their private collections of audio files with which they evaluated their systems, posing obvious problems not only in terms of reproducibility of research, but also in terms of its validity because these collections are usually poorly described [289, 203].

Even if data were readily available, another difference is that multimedia information is inherently more complex than text [53]. Musical information is multifaceted, comprising pitch, rhythm, harmony, timbre, lyrics, performance, etc. There are also different ways of representing music, such as scores or MIDI files and analog or digital audio formats. A music piece can be transposed in pitch, played with different instruments and different ornaments, or have its lyrics altered and still be perceived as the same piece [247]. In addition, text is explicitly structured (i.e. letters, words, sentences, etc.), and while similar structure is found in music (i.e. notes, bars, etc.), such structure is not explicit at all in audio signals. A similar distinction can be found for instance in video retrieval, where there is no visual equivalent to words and much of the research is likewise devoted to the development of descriptors that might play that role [260]

Finally, the storage and processing requirements for an MIR system are typically orders of magnitude larger. For instance, the size of an average Web document is in the kilobyte range, while a digital audio file is several dozen megabytes long. Even a lossy compression format like MP3 requires several megabytes to store a single music track, and the mere use of a lossy encoding can have negative effects on certain types of MIR algorithms that employ low-level features [256, 91, 111, 283]. All these characteristics of musical information are at the root of the complexity not only of developing MIR techniques, but also of the definition and elaboration of resources for their evaluation.
5.1. Why evaluation in Music Information Retrieval is hard

5.1.2 Tasks and evaluation datasets

Because of the copyright restrictions on musical data, public collections very rarely contain the raw audio signal of music pieces. There are some exceptions, such as the GTZAN collection [281] (1,000 audio clips for genre classification), the RWC databases [81, 82] (465 general purpose clips), the Music Audio Benchmark Data Set [106] (1,886 songs for classification and clustering) or the ENST-Drums database [72] (456 audio-visual sequences featuring professional drummers). For the most part though, the only viable alternative is to distribute datasets as various sets of features computed by third parties, such as in the Latin Music Database [257] or the recent Million Song Dataset [15, 210]. This approach is sometimes adequate for certain tasks where systems do not typically analyze audio at a low level, such as music recommendation. Nonetheless, it clearly hinders research in the sense that we are limited to whatever features are published and however they are computed; it is just impossible to try that new feature that worked out well in our private datasets. In some other tasks such as beat tracking it is just impossible to work even from low-level features; algorithms need to have the actual audio signal to produce their output. Another consequence of the problems to publicly distribute musical data is that collections tend to be very small, usually containing just a few dozen songs and rarely having over a thousand of them. In addition, and also to overcome legal issues, these musical documents are often just short clips extracted from the full songs, not the full songs themselves. Even different clips from the same song are often considered as different songs altogether, creating the illusion of large datasets.

MIR is highly multimodal [178], as seen in Sections 2 and 3. As a consequence, it is often hard to come up with suitable datasets for a given task, and researchers usually make do with alternative forms of data, assuming it is still valid. For example, synthesized MIDI files have been used for multiple $f_0$ estimation from audio signals, which is of course problematic to the point of being unrealistic [182]. Another example can be found in melody extraction, as we need the original multi-track recording of a song to produce annotations, and these are very rarely available.
Also, many MIR tasks require a certain level of music expertise from data annotators, which poses an additional problem when creating datasets. For example, annotating the chords found in a music piece can be a very complex task, especially in certain music genres like Jazz. A non-expert might be able to annotate simple chords that sound similar to the true chords (e.g. C instead of D9), or somewhat complex ones that could be mistaken with the original chords (e.g. inversions); but identifying the true chords requires a certain level of expertise. Even music experts might sometimes not agree, since analyzing music involves a subjective component. This does not imply that this task is not useful or relevant; while musicologists for instance may require the complex chords, there are simpler use cases where the simplified chords are sufficient and even preferred, such as for novice guitar players who want chords to be identified on the fly to play on top of some song. For some other tasks, making annotations for a single audio clip just a few seconds long can take several hours, and in some cases it is not even clear how annotations should be made [219, 95]. For example, it is quite clear what a melody extraction algorithm should do: identify the main voice or instrument in a music piece and obtain its melody pitch contour [207]. However, this may become confusing for example when we find instruments playing melodies alternating with vocals. There are other points that can be debated when evaluating these systems, such as determining a meaningful frame size to annotate pitch, an acceptable threshold to consider a pitch estimate correct, or the degree to which pitch should be discretized.

Research on MIR comprises a rich and diverse set of areas whose scope go well beyond mere retrieval of documents [55, 20, 147, 6, 148]. Three main types of tasks can be identified when considering system-oriented evaluation of MIR techniques: retrieval, where systems return a list of documents in response to some query (e.g. music recommendation or query by humming); annotation, where systems provide annotations for different segments of a music piece (e.g. melody extraction or chord estimation); and classification, where systems provide annotations for the full songs rather than for different segments (e.g. mood or genre classification). The immediate result from this diversity is that
all tasks have certain particularities for evaluation, especially in terms of data types, effectiveness measures and user models. We can also distinguish between low-level tasks such as Beat Tracking that serve to evaluate algorithms integrated for other high-level tasks such as Genre Classification, similar to Boundary Detection or other component tasks in TRECVID. As shown in Table 5.1, these low-level tasks indeed correspond to a large fraction of research on MIR.

5.2 Evaluation initiatives

The ISMIR series of conferences started in 2000 as the premier forum for research on MIR, and early in its second edition the community was well aware of the need of having a periodic evaluation forum similar to those in Text IR. Reflecting upon the tradition of formal evaluations in Text IR, the “ISMIR 2001 resolution on the need to create standardized MIR test collections, tasks, and evaluation metrics for MIR research and development” was signed by the attendees as proof of the concern regarding the lack of formal evaluations in Music IR and the willingness to carry out the work and research necessary to initiate such an endeavor [51]. A series of workshops and panels were then organized in conjunction with the JCDL 2002, ISMIR 2002, SIGIR 2003 and ISMIR 2003 conferences to further discuss the establishment of a periodic evaluation forum for MIR [50]. Two clear topics emerged: the application of a TREC-like system-oriented evaluation framework for MIR [294], and the need to deeply consider its strengths and weaknesses when specifically applied to the music domain [209]. Several evaluation initiatives for MIR have emerged since, which we describe below.

5.2.1 ADC: 2004

The first attempt to organize an international evaluation exercise for MIR was the Audio Description Contest\(^1\) (ADC), in conjunction with the 5th ISMIR conference in Barcelona, 2004 [26]. ADC was organized and hosted by the Music Technology Group at Universitat Pompeu Fabra, who initially proposed 10 different tasks to the MIR community:

\(^1\)http://ismir2004.ismir.net/ISMIR_Contest.html
Melody Extraction, Artist Identification, Rhythm Classification, Music Genre Classification, Tempo Induction, Audio Fingerprinting, Musical Instrument Classification, Key and Chord Extraction, Music Structure Analysis and Chorus Detection. After public discussions within the community, the first five tasks were finally selected to run as part of ADC. A total of 20 participants from 12 different research teams took part in one or more of these five tasks.

The definition of evaluation measures and selection of statistical methods to compare systems was agreed upon after discussions held by the task participants themselves. In terms of data and annotations, copyright-free material was distributed to participants when available, but only low-level features were distributed for the most part [25]. This served two purposes: first, it allowed participants to train their systems for the task; second, it allowed both participants and organizers to make sure all formats were correct and that system outputs were the same when systems were run by participants and by organizers. This was critical, because it was the organizers who ran the systems with the final test data, not the participants. This was necessary to avoid legal liabilities.

A public panel was held during the ISMIR 2004 conference to unveil the results obtained in ADC and to foster discussion among the community to establish a periodic evaluation exercise like ADC. There was general agreement on the benefit of doing so, but it was also clear that such an endeavor should be based on the availability of public data so that researchers could test their systems before submission and improve them between editions.

5.2.2 MIREX: 2005–today

After the success of the Audio Description Contest in 2004, the Music Information Retrieval Evaluation eXchange$^2$ (MIREX) was established and first run in 2005 in conjunction with the 6th annual ISMIR conference, held in London [57]. MIREX is annually organized since then by the International Music Information Retrieval Systems Evaluation

---

$^2$http://www.music-ir.org/mirex/wiki/
5.2. Evaluation initiatives

Table 5.1: Number of runs (system-dataset pairs) per task in all MIREX editions so far. These figures are not official; they have been manually gathered from the MIREX website.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Audio Artist Identification</td>
<td>7</td>
<td>7</td>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Audio Drum Detection</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Audio Genre Classification</td>
<td>15</td>
<td>7</td>
<td>26</td>
<td>65</td>
<td>48</td>
<td>31</td>
<td>31</td>
<td>26</td>
<td></td>
</tr>
<tr>
<td>Audio Key Finding</td>
<td>7</td>
<td></td>
<td>5</td>
<td>8</td>
<td>6</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Audio Melody Extraction</td>
<td>10</td>
<td>10</td>
<td>21</td>
<td>72</td>
<td>30</td>
<td>60</td>
<td>30</td>
<td>24</td>
<td></td>
</tr>
<tr>
<td>Audio Onset Detection</td>
<td>9</td>
<td>13</td>
<td>17</td>
<td>12</td>
<td>18</td>
<td>8</td>
<td>10</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>Audio Tempo Extraction</td>
<td>13</td>
<td>7</td>
<td></td>
<td>7</td>
<td>6</td>
<td>4</td>
<td>11</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Symbolic Genre Classification</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Symbolic Melodic Similarity</td>
<td>7</td>
<td>18</td>
<td>8</td>
<td>13</td>
<td>11</td>
<td>6</td>
<td>5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Symbolic Key Finding</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Audio Beat Tracking</td>
<td>5</td>
<td></td>
<td>33</td>
<td>26</td>
<td>24</td>
<td>20</td>
<td>60</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Audio Cover Song Identification</td>
<td>8</td>
<td>8</td>
<td>6</td>
<td>6</td>
<td>4</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Audio Music Similarity</td>
<td>6</td>
<td>12</td>
<td>15</td>
<td>8</td>
<td>18</td>
<td>10</td>
<td>8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Query-by-Singing/Humming</td>
<td>23</td>
<td>20</td>
<td>16</td>
<td>9</td>
<td>20</td>
<td>12</td>
<td>24</td>
<td>28</td>
<td></td>
</tr>
<tr>
<td>Score Following</td>
<td>2</td>
<td>4</td>
<td>5</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Audio Classical Composer Identif</td>
<td>7</td>
<td>11</td>
<td>30</td>
<td>27</td>
<td>16</td>
<td>15</td>
<td>14</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Audio Music Mood Classification</td>
<td>9</td>
<td>13</td>
<td>33</td>
<td>36</td>
<td>17</td>
<td>20</td>
<td>23</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Multiple F0 Estimation &amp; Tracking</td>
<td>27</td>
<td>28</td>
<td>39</td>
<td>23</td>
<td>16</td>
<td>16</td>
<td>6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Audio Chord Detection</td>
<td>15</td>
<td>18</td>
<td>15</td>
<td>18</td>
<td>22</td>
<td>32</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Audio Tag Classification</td>
<td>11</td>
<td>34</td>
<td>26</td>
<td>30</td>
<td>18</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Query-by-Tapping</td>
<td>5</td>
<td>9</td>
<td>6</td>
<td>3</td>
<td>6</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Audio Structure Segmentation</td>
<td>5</td>
<td>12</td>
<td>12</td>
<td>27</td>
<td>35</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Discovery of Repeated Themes &amp; Sections</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>16</td>
</tr>
</tbody>
</table>

Laboratory (IMIRSEL), based at the University of Illinois at Urbana-Champaign [56].

The choice of tasks, evaluation measures and data was again based on open proposals and discussions through electronic mailing lists and a wiki website. IMIRSEL provided the necessary communication mechanisms for that, as well as the computational infrastructure and the M2K execution platform to automate the evaluation process [56]. For its first edition in 2005, MIREX hosted the same tasks in ADC plus five additional tasks, mainly related to symbolic data processing as opposed to just audio (see Table 5.1). The number of participants increased to 82 individuals from 41 different research teams, who submitted a total of 86 different systems to evaluate.

The principal characteristic of MIREX is that it is based on an algorithm-to-data paradigm, where participants submit the code or binaries for their systems and IMIRSEL then runs them with the pertinent datasets, which are hidden from participants to avoid legal issues and also on the grounds of preventing overfitting. Releasing datasets after they are used would of course help IMIRSEL in running MIREX
and researchers in analyzing and improving their systems, but it would require the creation of new datasets the following year, meaning that new annotations would have to be acquired and that cross-year comparisons would be more difficult.

MIREX runs annually, and a brief overview of results is usually given during the last day of the ISMIR conference, along with a poster session where participants can share their approaches to solve each task. Over 2,000 different runs have been evaluated in MIREX since 2005 for 23 different tasks, making it the premier evaluation forum in MIR research. As a rule of thumb, MIREX runs a task if appropriate data is available (usually from previous years) and at least two teams are willing to participate. As seen in Table 5.1, MIREX has clearly focused on audio-based tasks.

5.2.3 MusiClef: 2011-2013

Despite its success among the community, MIREX is limited in the sense that all datasets are hidden to participants even after all results are published. While this allows IMIRSEL to avoid overfitting and cheating when using the same datasets in subsequent years, it also prevents participants from fully exploiting the experimental results to further improve their systems. To partially overcome this situation, the MusiClef campaign was initiated in 2011 as part of the annual CLEF conference [187]. Two tasks were proposed for the first edition, clearly based on real-world scenarios of application. The first task paired with LaCosa, an Italian TV broadcasting provider, and aimed at music categorization for TV show soundtrack selection. The second task paired with the Fonoteca at the University of Alicante, aiming at automatically identifying Classical music in a loosely labeled corpus of digitized analog vinyls.

Standard training data was made available to participants, and multi-modal data (e.g. user tags, comments, and reviews) was also included for participants to exploit. The audio content-based features were computed with the MIRToolbox [138], but MusiClef organizers also allowed, even encouraged, participants to submit their code to remotely compute custom features from the dataset, thus allowing them
to apply a much wider range of techniques. Overall, the availability of data and openness of feature extractors represented a step towards reproducibility of experiments. Another differentiating characteristic is the development of several baseline implementations.

MusiClef moved to the MediaEval conference in 2012 [155]. For this edition, it built upon the 2011 dataset [235], with a task on multi-modal music tagging based on music content and user-generated data. A soundtrack selection task for commercials was run at MediaEval 2013, in which systems had to analyze music usage in TV commercials and determine music that fits a given commercial video\(^3\). It was again a multi-modal task, with metadata regarding TV commercial videos from Youtube, web pages, social tags, image features, and music audio features. Unlike in previous years, ground truth data was acquired via crowdsourcing platforms. This was suitable because the task was not to predict the soundtrack actually accompanying the real video, but the music which people think is best suited to describe or underline the advertised product or brand. This might be quite different from what the respective companies’ PR departments think. Unfortunately, MusiClef did not have much support from the MIR community and stopped in 2013, probably because the tasks were still too challenging and high level for current MIR technology and did not seem appealing to researchers.

5.2.4 MSD Challenge: 2012

The Million Song Dataset (MSD) [15] represented a significant breakthrough in terms of data availability and size (it contains features and metadata for a million contemporary popular music tracks). It contains metadata and audio features for a million contemporary popular music tracks, encouraging research that scales to commercial sizes. Audio features were computed with The Echo Nest API\(^4\), and the data is linked with 7digital\(^5\) to provide 30 seconds samples of songs, with

---

\(^3\)http://multimediaeval.org/mediaeval2013/soundtrack2013/
\(^4\)http://developer.echonest.com
\(^5\)http://www.7digital.com
MusicBrainz\textsuperscript{6} and Play.me\textsuperscript{7} to gather additional metadata, or even the lyrics through MusiXmatch\textsuperscript{8}.

Following the creation of the dataset, the MSD Challenge\textsuperscript{9} [170] was organized in 2012, reflecting upon the success of the previous Netflix challenge [11] on movie recommendation and the 2011 KDD Cup on music recommendation [60]. The task in this case consisted in predicting the listening history of users for which half was exposed. The challenge was open in the sense that any source of information, of any kind, was permitted and encouraged. Like in MusiClef, training data was available, and the annotations used in the final test dataset were also made public when the challenge was over. Reference baseline implementations were also available.

The MSD Challenge had an enormous success in terms of participation, with 150 teams submitting almost 1,000 different runs. The reason for such high level of participation is probably that the task was amenable to researchers outside the MIR field, especially those focused on Machine Learning and Learning to Rank. Because music tracks in the MSD were already described as feature vectors, participants did not necessarily need to have knowledge on music or signal processing. A second set of user listening history was intentionally left unused for a second round of the MSD Challenge, initially planned for 2013. However, for various logistics issues it was postponed. No more user data is available though, so no more editions are planned afterwards. The MSD Challenge is thus a one or two times initiative, at least in its current form.

5.2.5 MediaEval: 2012-today

As mentioned above, the MusiClef campaign was collocated with the MediaEval series of conferences in 2012 and 2013, but other music-related tasks have emerged there as well. The Emotion in Music task appeared in 2013 to continue the Affection tasks held in previous

\textsuperscript{6}http://www.musicbrainz.org
\textsuperscript{7}http://www.playme.com/
\textsuperscript{8}http://www.musixmatch.com
\textsuperscript{9}http://labrosa.ee.columbia.edu/millionsong/challenge
5.2. Evaluation initiatives

years [263]. It contained two tasks: in the first task participants had to automatically determine emotional dimensions of a song continuously in time, such as arousal and valence; in the second task they had to provide similar descriptors but statically, ignoring time. A dataset with 1,000 creative commons songs was distributed among participants, and crowdsourcing methods were again employed to evaluate systems. These tasks are again scheduled for 2014, with a brand new dataset.

Two other tasks are planned for MediaEval 2014 as well. The C@merata task is a question answering task focused on Classical music scores. Systems receive a series of questions in English referring to different features of a music score (e.g. “perfect cadence” or “harmonic fourth”, and they have to return passages from the score that contain the features in the question. The Crowdsourcing task is aimed at classification of multimedia comments from SoundCloud\textsuperscript{10} by incorporating human computation into systems. In particular, systems had to sort timed-comments made by users who were listening to particular songs, focusing on whether comments are local (i.e. pertaining or not to some specific moment of the song) and technical.

5.2.6 Networked platforms

Two alternatives have been explored in response to the restrictions for distributing MIR datasets: publishing features about the data, or having the algorithms go to the data instead of the data go to the algorithms. Several lines of work to improve these two scenarios and exploring the feasibility of mixing them up have appeared recently [192, 169, 210]. For example, MIREX-DIY is a Web-based platform to allow researchers upload their systems on demand, have them executed remotely with the pertinent datasets, and then download the results of the evaluation experiment [61]. In addition, this would provide archival evaluation data, similar to that found in Text IR forums like TREC and platforms like evaluatIR [1].

\textsuperscript{10}https://soundcloud.com/
5.3 Research on Music Information Retrieval evaluation

Carrying out an evaluation experiment in Information Retrieval is certainly not straightforward; several aspects of the experimental designs have to be considered in terms of validity, reliability and efficiency [273, 289]. Consequently, there has been a wealth of research investigating how to improve evaluation frameworks, that is, evaluating different ways to evaluate systems. With the years, this research has unveiled various caveats of IR evaluation frameworks and their underlying assumptions, studying alternatives to mitigate those problems [222, 93]. However, there has been a lack of such research in MIR, which is particularly striking given that the MIR field basically adopted the body of knowledge on evaluation in Text IR as of the early 2000s. Since then, the state of the art on evaluation has moved forward, but virtually no research has been conducted to revise its suitability for MIR. Compared to Text IR, research about evaluation receives about half as much attention (e.g. 11% of papers in SIGIR vs. 6% in ISMIR), although it seems clear that the little research being conducted does have an impact on the community [289].

Although much of the research related to evaluation in MIR has been devoted to the development of datasets and the establishment of periodic evaluation exercises, some work has addressed other specific problems with the evaluation frameworks in use. As mentioned before, making annotations for some MIR tasks can be very time consuming and generally requires some level of musical expertise. Nonetheless, the inherently entertaining nature of music makes it possible to resort to non-experts for some types of tasks. As mentioned in Section 4, several games with a purpose have been developed to gather music annotations, such as TagATune [140], MajorMiner [164], MoodSwings [117], and ListenGame [277]. The suitability of paid crowdsourcing platforms such as Amazon’s Mechanical Turk has been studied by Urbano et al. [287] and Lee [145] to gather music similarity judgments as opposed to music experts [112]. Mandel et al. [163] also explored crowdsourcing alternatives to gather semantic tags, and Lee and Hu [149] did so to gather music mood descriptors. Similarly, Sordo et al. [265] compared experts and user communities to create music genre taxonomies.
Typke et al. [279] studied the suitability of alternative forms of ground truth for similarity tasks, based on relevance scales with a variable number of levels; they also designed an evaluation measure specifically conceived for this case [280]. Urbano et al. [285, 287] showed various inconsistencies with that kind of similarity judgments and proposed low-cost alternatives based on preference judgments that resulted in more robust annotations. Other measures have been specifically defined for certain tasks. For instance, Moelants and McKinney [174] focused on tempo extraction, Poliner et al. [207] devised measures for melody extraction, and recent work by Harte [94], Mauch [168] and Pauwels and Peeters [199] proposed and revised measures for chord detection. Other work studied the reliability of annotations for highly subjective tasks such as artist similarity and mood classification [63, 258].

Because of the limitations when creating datasets, a concern among researchers is the reliability of results based on rather small datasets. For example, Salamon and Urbano [219] showed that traditional datasets for melody extraction are clearly too small, while the ones that are reliable are too focused to generalize results. Similarly, Urbano [282] showed that collections in music similarity are generally larger than needed, which is particularly interesting given that new judgments are collected every year for this task. Flexer [65] discussed the appropriate use of statistical methods to improve the reliability of results. Urbano et al. [286, 284] visited current statistical practice to improve statistical power, reduce costs, and correctly interpret results.

In order to support the creation of large datasets, low-cost evaluation methodologies have been explored for some tasks. For instance, Urbano and Schell [288, 282] proposed probabilistic evaluation in music similarity to reduce annotation cost to less than 5%, and Holzapfel et al. [105] proposed selective sampling to differentiate easy and challenging music pieces for beat tracking without annotations. Finally, some work has pointed out more fundamental questions regarding the validity of evaluation experiments for some particular tasks, such as music similarity [284, 108] or genre classification [272, 271]. In particular, Schell et al. [230] and Lee and Cunningham [146] discuss the need to incorporate better user models in evaluation of MIR systems.
5.4 Discussion and challenges

Evaluation of MIR systems has always been identified as one of the major challenges in the field. Several efforts have set out to develop and provide the necessary infrastructure, technology and methodologies to carry out these evaluations. There is no doubt that the MIR community has enormously benefited from these initiatives for fostering these experiments and establishing specific evaluation frameworks [40, 41]. However, it is also becoming clear that it has reached a point where these evaluation frameworks and general practice do not allow researchers to improve as much and as well as they would want [289]. The main reasons are: 1) the impossibility of conducting error analysis due to the unavailability of public datasets and the closed nature of MIREX (e.g. if a system performs particularly badly for some song, there is no way of knowing why or what song that is to begin with), 2) the lack of a larger discussion and agreement in terms of task and measure definitions (in fact, most tasks are initiated because some dataset becomes available after a PhD student donates it, so the task is ultimately defined by an individual researcher or team), 3) the fact that basically the same datasets are being used year after year, so tasks do not evolve and their research problems are found to be less challenging with time, and 4) the lack of evaluation data to conduct the necessary research to improve evaluation frameworks. The root problems here are the distribution of copyrighted material and the cost of building new datasets.

To visualize the partial impact of this close-datasets policy, we plotted in Figure 5.1 the maximum and median performance scores of algorithms submitted to MIREX for a selection of tasks. All algorithms within the same task were evaluated with the same dataset over the years and using the same measures, so scores are comparable. As can be seen, most tasks have rapidly reached a steady point where algorithms have not improved any further. On the one hand, this evidences that researchers are not able to analyze their algorithms in detail after

\[\text{We selected tasks with a sufficiently large history and level of participation, ignoring very recent datasets with too few algorithms to appreciate trends.}\]
being evaluated, and they end up submitting basically the same algorithms or small variations tested with their private datasets; in some cases, the best algorithms are not even submitted again because they would obtain the same result. On the other hand, it also evidences the “glass ceiling” effect mentioned in Section 2 whereby current audio descriptors are effective up to a point. In some cases, like Audio Melody Extraction, we can observe how algorithms have even performed worse with the years. The reason for this may be that new datasets were introduced in 2008 and 2009, so researchers adapted their algorithms. However, these datasets have later been shown to be unreliable [219], so we see better results with some of them but worse results with others.

Solving these issues has been identified as one of the grand challenges in MIR research [100, 250], and some special sessions during the ISMIR 2012 and 2013 conferences were specifically planned to address them [204, 8]. Several lines of work have been identified to improve the current situation [289]. For instance, the creation of standardized music
Evaluation in Music Information Retrieval

corpora that can be distributed throughout researchers and used across tasks, seeking multimodal data when possible. In other areas like video retrieval, this has been possible thanks to data donations and creative-commons video material [260]. For instance, some datasets employed in TRECVID originated from the Internet Archive, television shows from the Netherlands Institute for Sound and Vision, indoor surveillance camera videos from UK airports, or the Heterogeneous Audio Visual Internet Corpus. The main problem in MIR is that creating music is more complex, and while vast amounts of unlicensed video or images are being recorded everywhere, most of the music is copyrighted. Consequently, collections are usually small and contain just metadata and feature vectors, but the MIR community must pursue the collaboration with music providers to gain access to the raw data or, at least, the possibility to remotely compute custom features. In all cases, it is very important that these corpora are controlled and that all research is conducted on the exact same original data.

Annotations and ground truth data should also be public for researchers to further improve their systems out of forums like MIREX, push the state of the art and pursue new challenges. This in turn could be problematic if no new data is generated from time to time and researchers stick to the same datasets over the years. To avoid this, low-cost evaluation methodologies and annotation procedures should be adopted to renew and improve the datasets used. These new annotations can be released every so often for researchers to further train their systems, while a separate dataset is kept private and reused for several years to measure progress. Finally, the inclusion of strong baselines to compare systems should be further promoted and demanded in all MIR research. Ideally, these would be the best systems found in the annual evaluation forums, but this requires the establishment of common evaluation datasets.

The MIR community also needs to explore alternative evaluation models beyond the algorithm-to-data paradigm currently followed in MIREX, which is extremely time consuming for IMIRSEL, becomes prohibitive when funding runs out, does not allow us to evaluate interactive systems and hence tend to ignore final users. Again, this would
require common evaluation datasets if it were finally the participants who run their own systems. But most importantly, it is paramount that all evaluation data generated every year, in its raw and unedited form, be published afterwards; this is an invaluable resource for conducting meta-evaluation research to improve MIR evaluation frameworks and practices [311]. Very recent examples of meta-evaluation studies, possible only with the release of evaluation data, were conducted by Smith and Chew [261] for the MIREX Music Segmentation task, by Flexer et al. [67, 66] for Audio Music Similarity, and by Burgoyne et al. [23] for Audio Chord Detection.
Conclusions and Open Challenges

Music Information Retrieval is a young but established multidisciplinary field of research. As stated by Herrera et al. [100], “even though the origin of MIR can be tracked back to the 1960’s, the first International Conference on Music Information Retrieval, started in 2000 as a symposium, has exerted on the sense of belongingness to a research community”.

Although the field is constantly evolving, there exists already a set of mature techniques that have become standard in certain applications. In this survey, we provided an introduction to MIR and detailed some applications and tasks (Section 1). We reviewed the main approaches for automatic indexing of music material based on its content (Section 2) and context (Section 3). We have also seen that retrieval success is highly dependent on user factors (Section 4). For this reason, defining proper evaluation strategies, highly involving end users, at the different steps of the process and tailored to the MIR task under investigation is important to measure the success of the employed techniques. Current efforts in evaluation of MIR algorithms were presented in Section 5.

Some of the grand challenges still to be solved in MIR have already
been pointed out by leading MIR researchers, among others, Downie et al. [54], Goto [79] and Serra et al. [250].

Downie et al. [54] mentioned five early challenges in MIR: further study and understanding of the users in MIR; to dig deeper into the music itself to develop better high level descriptors; expand the musical horizon beyond modern, Western music; rebalance the amount of research devoted to different types of musical information; and the development of full-featured, multifaceted, robust and scalable MIR system (this they mention as “The Grand Challenge” in MIR).

Goto [79] later identified five grand challenges: delivering the best music for each person by context-aware generation or retrieval of appropriate music; predicting music trends; enriching human-music relationships by reconsidering the concept of originality; providing new ways of musical expression and representation to enhance human abilities of enjoying music; and solving the global problems our worldwide society faces (e.g. decreasing energy consumption in the music production and distribution processes).

Serra et al. [250] propose to consider a broader area of Music Information Research, defined as “a research field which focuses on the processing of digital data related to music, including gathering and organization of machine-readable musical data, development of data representations, and methodologies to process and understand that data”. They argue that researchers should focus on four main perspectives detailed in Figure 6.1: technological perspective, user perspective, social and cultural perspective, and exploitation perspective.

Adding to the previously presented aspects, we believe that the following challenges still need to be faced:

**Data availability**: we need to identify all relevant sources of data describing music, guarantee their quality, clarify the related legal and ethical concerns, make these data available for the community and create open repositories to keep them controlled and foster reproducible research.

**Collaborative creation of resources**: the lack of appropriate resources for MIR research is partially caused by both legal and practical
issues; to circumvent this situation we should develop low-cost, large-scale and collaborative methods to build this infrastructure and, most importantly, seek the direct involvement of end users who willingly produce and share data.

**Research that scales:** a recurrent criticism of MIR research is that the techniques developed are not practical because they hardly scale to commercial sizes. With the increasing availability of large-scale resources and computational power, we should be able to adapt MIR methods to scale to millions of music items.

**Glass ceiling effect:** we need to address the current limitation of algorithms for music description, by developing more musically meaningful descriptions, adapting descriptors to different repertoires, and considering specific user needs.

**Adaptation and generality:** we have to increase the flexibility and generality of current techniques and representations, and at the
same time work towards methods adapted to the application needs.

**Semantic gap**: there is the need of addressing the conceptual and semantic gap between computational models of music description and similarity and user or expert musical analyses, as well as the need of exploiting the opportunities for computer-assisted or adapted paradigms.

**Evaluation**: challenges in this area are related to the integration of methodologies from other disciplines, take into account the validity of algorithms in the development of real-world applications and define meaningful evaluation tasks and methodologies valid and reliable in the short and long term; particular attention should be paid to involving the end user during evaluation.

**User studies**: we need to research ways to model the user and his cultural and environmental context, investigating his individual information or entertainment needs to create accurate user models.

**Multimodality**: we need to understand how the diverse multimodal features that are already available can be integrated to build personalized music retrieval systems, and to present this multimodal information to the user in a most beneficial way.

**Music similarity measurement**: we need to understand how the low-level and mid-level content and context features relate to the human perception of music, and how to use this knowledge to construct multifaceted similarity measures that reflect human perception of similarity. To answer these questions, we first need to investigate whether there are relations that are generally valid, independent of individual and culture, or if perception of music is too individual to derive such patterns.

**Multidisciplinarity**: although MIR is already a multidisciplinary field, we still need to systematize cross-disciplinary transfer of knowledge and methodologies, as well as to extend and strengthen existing links with other disciplines.

To round off this survey, we would like to present some visionary challenges we contemplate in the field of MIR:

**Anticipatory music recommendation systems**: we foresee systems that anticipate the human’s music preference, anytime and any-
where, taking into account external (e.g. environmental) and internal (e.g. mood or health state) factors, in order to automatically deliver the appropriate music for each human in the world.

**Music through the senses:** music is foremost an auditory phenomenon, but it is also visual (e.g. dance, gestures of the performer, facial expression) and tactile (e.g. instrument touch); MIR systems should account for such different sensory modalities in order to provide a truly engaging experience, both in music creation and listening.

**Musical companions:** music has a great potential in different aspects of human life such as cognitive development, education, therapy and well-being; we foresee MIR systems as personalized musical companions along our lives, systems connecting music to personal memories or experiences, and systems helping to regulate emotions, for instance in stressful or even depressing human situations. MIR will hence help improving mankind’s overall well-being.
Acknowledgements

This work is supported by the Austrian Science Fund (FWF): P22856 and P25655, as well as by the European Union Seventh Framework Programme FP7 / 2007-2013 through the PHENICX project under grant agreement no. 601166, the A4U postdoctoral grants programme, and the Junta of Andalusia through the COFLA2 project (P12-TIC-1362). The authors would further like to thank Masataka Goto, Mohamed Sordo, and Edith Law for granting permission to include their visual material in this survey, and Juan J. Bosch for his comments on the manuscript. Furthermore, the authors would like to express their gratitude to the anonymous reviewers for their highly valuable suggestions for improving the manuscript.


References


References


References


References


References


References


References


References


References


References


References


References


References


References


References


