High-Level Musical Content-Based Music Information Retrieval: A State-of-the-Art Review

Cinar Gedizlioglu (Corresponding author)
Computer Engineering, Izmir University of Economics, Balcova, Izmir / Turkey
E-mail: cg2269@nyu.edu

Kutluhan Erol
Computer Engineering, Izmir University of Economics, Balcova, Izmir / Turkey
E-mail: kutluhan.erol@gmail.com

Abstract

Music Information Retrieval (MIR) is an interdisciplinary field that involves automating music processing for the purpose of accessing and managing large music collections. Motivated by the rapid growth of digital music content, it encompasses a broad set of strategies from diverse disciplines. In this paper, we provide a task-centric perspective of this field with particular emphasis on high-level content analysis. We provide a general context to explain the contributions from various disciplines for those wishing to learn about this field. We also discuss challenges and future directions to stimulate further research.

Keywords: music analysis, information retrieval, machine learning, audio analysis, representation

DOI: 10.7176/JSTR/6-10-08

1. Introduction

Over the last fifteen years, musical access has transformed dramatically. With the rapid rise of digital media, traditional ways of accessing music, such as records or radio broadcasts have given way to more personalized ways of listening to, creating or learning about music. Music downloads have long surpassed CD sales, music recommender systems such as Spotify and Pandora are enjoying great success. Computer tools to engage with music have risen in popularity, and music creation is much easier with tools such as autotune, sampling, and digital audio workstations (software for editing and processing digital audio) (Colby, 2004). According to IFPI (*IFPI Digital Music Report 2015*, 2015), in 2014, for the first time, revenue from digital channels reached the level of physical format sales, both at 46%, and music subscription services enjoyed a rise of 39% in revenue. As a result of this transformation, the availability of music has increased dramatically and personal music collections grew rapidly: In 2011, average iTunes account contained about three thousand songs (Kahney, 2011); a survey from 2017 indicates (Chu, 2017) two thirds of respondents had over 5000, and ten percent, over a hundred thousand.

This massive growth of the music industry created several challenges. The volume of digital media continues to grow exponentially, leading to an increasing personal need for efficient search tools for music collections. Existing tools suffer from labelling challenges for new data, especially related to more elusive features, such as mood, tempo, and timbre. Additionally, new areas have emerged, such as affordable personal karaoke machines, user-friendly mobile music production applications (Weinberg et al., 2009)[5], and interactive music systems (Collins, 2008; Granger et al., 2018). All these challenges require automation in certain principles of music, such as melody or harmony. The solutions consist of disparate strategies, which collectively form the interdisciplinary field of MIR. With the rise of the digital media, MIR gained popularity in academic and industrial research laboratories. To accomplish the task (and many subtasks) of automating music processing, the field employs many widely different, highly specialized strategies. Thus, it can be difficult, initially, to grasp the overall picture, see how these approaches fit together, and how they can be combined for specific purposes.

We can group these strategies into two broad categories: low level versus high level content analysis.

Furthermore, tiers of abstraction within both categories exist. Low-level content analysis involves processing raw audio input utilizing signal processing techniques. For example, spectral flux is a measurement of change in the power spectrum in the signal, calculated by finding the Euclidean distance between the power spectra of two different frames of a single audio signal (Burger et al., 2013). Mel-frequency cepstral coefficients are computed by taking the Fourier transform of the frequency spectrum after mapping it to the mel-scale, used as features for model training for many tasks (Logan, 2000). Aside from these simple audio measurements, some further analysis using these measurements can also be categorized as low-level content. Examples include onset detection (crucial measurement as a first step for beat tracking purposes) or timbre detection. High-level content analysis involves musical concepts such as melody, harmony or tempo to describe the contents of the music. High-level features are abstractions of musical concepts inferred through lower-level feature measurements. For a list of common high-level features, see (Casey et al., 2008). From an analytical standpoint, even though these features are intuitive for human perception, research to automate this involves sophisticated algorithms and models, which proved to be challenging. This paper focuses on high-level musical content description, with detailed explanations for the most common tasks within.

There are many issues to consider before embarking on a MIR task. A major issue is to determine the input media for the rest of the process, since the collective methods used are adapted to the media, and these adaptations differ from task to task, both in terms of implementation and efficiency. Sections 1 and 2 describe different input formats that could be considered for MIR tasks. Section 3 mentions many subtasks of high-level content identification and covers the advances in each subtask in detail. Section 4 discusses current challenges, and Section 5 concludes the paper with future directions for the field.

2. Raw Audio Input

Raw audio refers to digital audio file formats such as WAV, AIFF, OGG, or MP3. Using raw audio as input leads to signal processing methods for retrieval of low-level features. Precise measuring of a single aspect of an audio is extremely difficult because a particular aspect cannot be perfectly singled out from a signal. Therefore, analysis of low-level features is often imperfect, and noisy. While these features can sometimes have research benefits on their own, such as genre classification of pieces (Kim & Nam, 2019), such subtasks are often unintuitive, since in isolation they have little if any musical significance to humans. Therefore, the primary purpose of low-level features is to infer high-level features, which in turn might be used to infer even more abstract (higher-level) features. For example, a folding process of the frequency spectrum (a low-level feature, computed using the Fourier transform) of a given musical piece yields a chromagram (or pitch class profiles, a level of abstraction) (Fujishima, 1999), which is the occurrence frequency of the 12 pitch classes in that particular piece. The chromagram is then used to infer key or chord information (a further level of abstraction) about the piece.

Imprecision of low-level features are further exacerbated by each level of additional abstraction. Therefore, it is crucial to employ algorithms that are as precise as possible for each step. To reduce the imprecision inherent to low-level features, symbolic representations as input media can be preferred. Most of the low-level features are already encoded in symbolic notation, therefore symbolic representations offer researchers convenience. In contrast, symbolic representations do not capture actual audio, but only represent it in a certain notation, and are therefore unable to reflect nuances. Thus, certain high-level features can only be extracted if the input is in a raw audio form, the most obvious examples of which are timbre, and lyrics.

3. Symbolic Representations

In order to represent music in a way that can be clearly understood and read by the computer, numerous digital representations called Music Representation Languages (MRLs) have been developed or proposed for use in MIR systems. These MRLs have varying approaches on representing a piece of music: Some emphasize the musical aspects of a piece (e.g. score notation, instrument information, composer, movement no), such as MuseData (Hewlett, 1997) or DARMS (Pool, 1996), while others, such as MIDI, have more computational concerns.

The most popular of these MRLs is MIDI, not only because it is computationally efficient, but also because it is the best documented MRL, and is suitable for almost every operating system. Surprisingly, it is also the most limited in terms of the number of aspects of music that it can represent (Fujishima, 1999), since it is designed to be read by hardware, and therefore consists of hardware protocols and instructions, rather than a more “musical” representation. Deeper exploration of different MRLs, their thorough analyses and reviews can be found in (McLane, 1996; Repetto & Selfridge-Field, 1997; Selfridge-Field, 1994).

Since symbolic representations already contain many low-level features explicitly, inferring high-level features is less complex, bypassing signal processing altogether. It is also more precise for the same reason. This leads to the elimination of errors which stem from any signal processing issue, such as production artefacts or algorithmic errors. Computationally, processing symbolic representations is less time-consuming than processing raw audio. Collectively, these reasons render MRLs more desirable over raw audio for extracting high-level musical content. Since MIDI is well documented and compatible with most systems, as stated above, it is usually the representation of choice for MIR processes.

There are also important trade-offs in choosing a symbolic representation as a medium. Automated conversion of raw audio to symbolic representation is an infeasible operation, in terms of the resulting product. And since music is initially recorded in a raw audio form, such conversion should be made by hand, which is a tedious process. Therefore, the amount of content in raw audio form is vastly higher than that in symbolic formats. This also means that if a specific set of pieces are to be studied, the availability of these pieces in symbolic representations becomes a concern. Additionally, symbolic formats do not represent the performance, but the piece. This means that any intentional nuance of the performers would be excluded. If these nuances are of any importance to a study, then symbolic representations are ineffective.

In the earlier years of MIR, the majority of researchers in the field of MIR sought to research using symbolic representations due to their convenience and precision. This led to a decision by MIREX to limit the number of papers using symbolic notation as input. Raw audio-based research therefore rose in numbers. However, symbolic representations remain as a popular option for research, and heavy considerations should be made about which input form to use before embarking on a specific MIR task.

4. High-Level Musical Features

An intuitive starting point for many music information retrieval tasks is to analyse high-level content, such as harmony, melody or tempo. There are many use cases for high-level musical content identification. Examples include finding work containing a melodic fragment, finding music that "sounds like" a given recording, mapping a performance onto another independent of tempo and rhythmic patterns, or finding music that matches a user's personal profile. Tagging or labelling a musical piece (tags and labels are later used to be able to efficiently find a specific set of pieces) usually relies on its high-level musical content. One exception is tagging metadata, such as artist, album or release date, which is often already available independently. For example, automatically tagging a song as "happy" might be more likely if the song is in a major key, or music can be recommended to users depending on the overall tempo of their listening history. Throughout the evolution of MIR, the task of extracting high-level musical features proved to be a great challenge, and this subtask has been a subject of intense research. The music information retrieval evaluation exchange (known as MIREX) is a valuable medium keeping pace with latest developments in many applications within MIR, including high-level musical feature extraction (Downie et al., 2010, 2008).

Following subsections covers different high-level features, their tasks, subtasks and the methods employed for each one. The tasks mentioned here are:

- Melody tracking: automatic identification and analysis of a melody line within a piece or an excerpt,
- Beat tracking: automatic estimation of temporal parameters of a piece such as beat, tempo or rhythm,
- Estimation of key and chord progressions,
- Music structure: automatic identification of music segments such as beats, measures, themes, phrases or movements.

The advances in these fields, comparisons between certain methods and evaluations will also be mentioned.

4.1 Melody Tracking

One of the most commercially attractive applications for melody tracking is retrieval. This includes helping users access music by automatically analysing a given melody line. Recommender systems apply this method to recommend pieces with similar melody lines. Some applications enable users to find songs by humming melody lines. Copyright issues are analysed by investigating melody, the most defining, and the most predominant characteristic of any given piece of music. Similarly, melody tracking is also a primary tool in identifying cover songs (Juheon Lee, Sungkyun Chang, Donmoon Lee, 2015). Production tools are also enhanced by this feature, allowing musicians/producers to isolate melodies for later use, or remove a melody line from a given musical excerpt. The ability to automate this process is also a highly beneficial application for karaoke bars.

Melody tracking tasks can be categorized into three main subtasks: melody extraction from polyphonic audio, Query-by-Humming/Singing (QbHS), and Symbolic Melody Similarity (SMS). These subtasks deal with similar issues, but employ completely different methods, and face different technical problems.

The purpose of audio melody extraction is to extract and identify a melodic contour from polyphonic audio. While “melody” is in itself a difficult term to define, researchers accept more simplified definitions tailored to music processing. The most common definition of a melody line is the voice or instrument that is the most predominant, and its pitch values at different frames. Therefore, one assumption made by researchers is that melody is monophonic.

Even with this assumption, melody extraction from audio proves challenging, mainly due to three factors. First, a polyphonic audio (where more than one voice may be present at a given time interval) makes it difficult to attribute specific frequency bands to specific instruments. This issue is further complicated by post-production techniques, which can alter audio in such ways which can blur note onsets and offsets within the audio. Second, in a piece where there might be more than one melody line (or one melody line with accompaniment), the algorithm should be able to correctly identify the predominant voice. Finally, the time intervals where the melody is not present should also be identified (also known as the “voicing detection” problem). For detailed explanations of recent audio melody extraction algorithms and their performances, see (Bosch et al., 2016; Kumar et al., 2020).

Query-by-Humming/Singing involves users finding songs by humming sections of their melodies, and the system retrieves songs within its database with the closest correspondence. The queries (hummings), as well as the templates from the database which are in MIDI format, are converted to vectors which contain MIDI pitch values. In the case of templates, this conversion process is a straightforward isolation of a vector already existing in a MIDI file. In the case of queries, the process involves melody extraction techniques since the hummings are initially in a raw audio format. Some form of prior cleaning is usually necessary to decrease the influence of octave and precision errors caused by out-of-tune humming.

The problem in general with a typical QbHS system arises when queries are in a different key than the database templates. Some form of matching algorithm is necessary to eliminate the influence of these differences. The simplest solution applied by researchers is to subtract the mean pitch from the query sequence (Jeon & Ma, 2011). This method is problematic when the sung query is only part of a larger template. To eliminate this factor, a Dynamic Time-Warping (DTW) algorithm is applied to the query vector in (Stasiak, 2014), after a set of pre-processing steps. DTW involves non-linearly rescaling the time domain of two sequences, such that they have the same number of frames which accurately map on to their corresponding partners. An illustration of the DTW algorithm can be seen in Figure 1. It tries to minimize the cost of the mappings where the cost is the sum of differences between each frame. However, as of MIREX 2016, the cumulative accuracy is no higher than 80 percent for QbHS tasks.

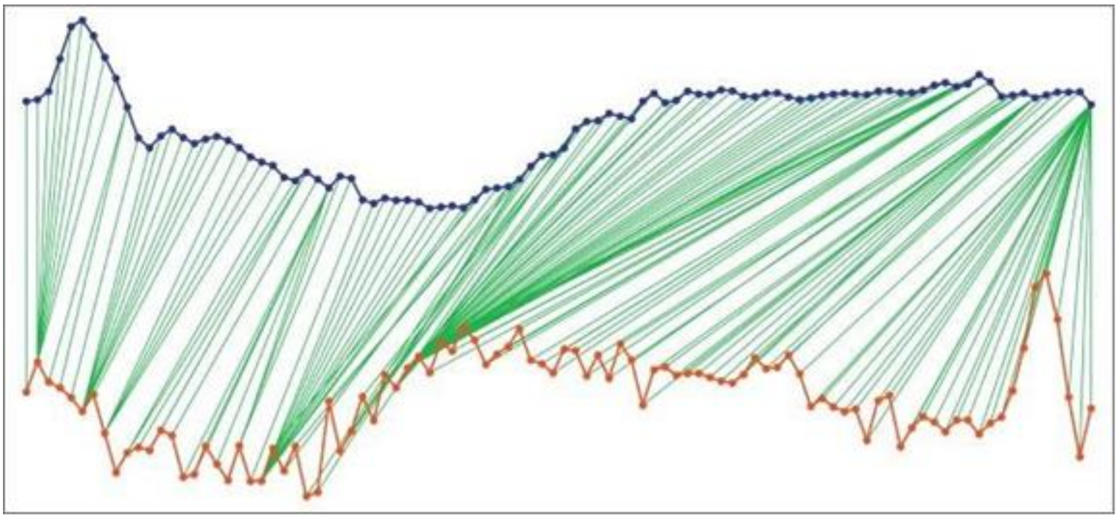

Figure 1. The rescaling of the time domain of two vectors using DTW.

Symbolic representations can also be useful in melody tracking. One official task listed in MIREX is called "Symbolic Melodic Similarity" (SMS). The aim of an SMS task is to be able to list and rank melodically similar excerpts, for a given query. Extracting melody from a symbolized piece of music can be done via several methods, depending on the melody representation. For the SMS task, two melodic representations are suitable: melodies as 1-D strings of characters or geometric curves.

In 1-D strings of characters representation, each character can represent one note or a consecutive note sequence. Similarities between melodies can therefore be found by applying well-known string-matching algorithms, such as finding edit distance, finding substring occurrences or finding longest common subsequence.

Geometric curve representation of melodic lines was the best-performing algorithm for SMS in MIREX 2014 (and later MIREX 2015), proposed by Urbano (2014). According to this method shown in Figure 2, pitch occurrences are represented as points on a 2-D plane, and the melodic line is the interpolating curve, as fitted by second degree splines. Finding melodic similarity is simplified to the comparison of different 2-D curves. However, this best-performing algorithm offers accuracy levels of no more than 75%.

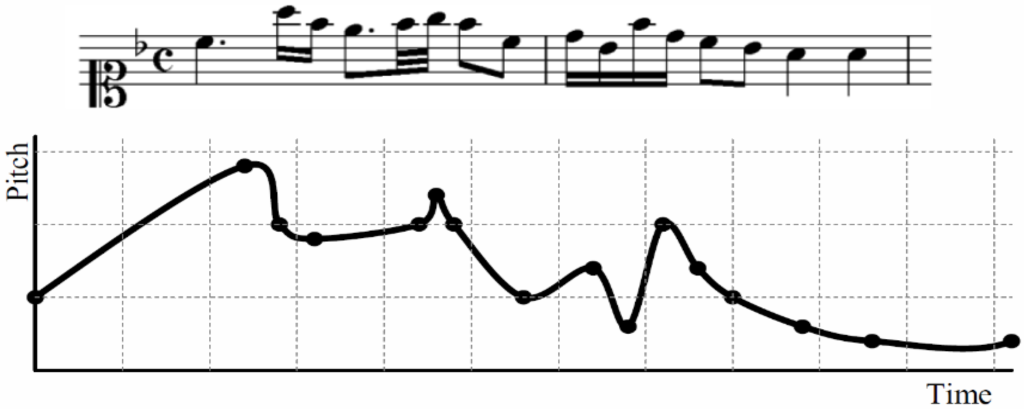

Figure 2. Melody representation as a geometric curve, proposed by Urbano (2014).

4.2 Beat Tracking

Beat tracking is mainly concerned with the automatic estimation of temporal parameters of music, such as beat, measure, rhythm, tempo and meter. These estimated parameters can later be used for retrieval, classification, recommendation or higher-level content identification. Most research in this area is concerned with audio signals. Some temporal parameters which would be difficult to precisely identify are already encoded in symbolic representations, albeit omitting the nuances. An example would be the period of a beat (or a quarter note, the two can be used interchangeably), which is the time difference between the onsets of two successive beats. This information is, by default, included in any MIDI file, therefore is trivial to obtain. Identification of higher-level content, however (such as measure information or rhythmic pattern) requires sophisticated algorithm design using beat information. While these algorithms are not usually concerned with symbolic representations, they can be accordingly modified.

Once beats are estimated, they are treated as the temporal unit for high-level computation, allowing the estimation of implicit musical temporal units, such as measures or rhythmic patterns. Detecting musical structure in a piece (such as identifying chorus sections (Bartsch & Wakefield, 2002; de Berardinis et al., 2020)) is therefore facilitated by the beat information. This feature is especially useful for editing recorded audio for production studios (Fazekas & Sandler, 2007). Measure estimation can also provide a smoother segmentation option for later processing. Furthermore, the temporal axis of musical excerpts can be normalized using beat information, facilitating the identification of cover pieces (Tralie, 2017a, 2017b). Examples of industrial-scale applications of beat tracking, especially important in the field of entertainment, include audio beat synchronization with lighting effects.

As already mentioned above, precision in identifying the beat structure is a problem, mainly stemming from the nature of audio signals and the extraneous information carried, as discussed in Section 2. Algorithms exist to minimize the influence of this information (Cannam et al., 2015), but desirable accuracy levels remain elusive.

One of the biggest challenges for this field is the existence of temporally complex musical pieces. A beat structure that is not explicit in its presentation makes beat estimation highly problematic. Rhythmic complexity and tempo variations are the biggest contributors to ambiguous beat structures. In the presence of an ambiguous beat structure, multiple possible solutions exist, therefore, probabilistic models are employed to select the best option out of many (Boulanger-Lewandowski et al., 2013).

4.3 Chord & Key Detection

Extracting key information from musical excerpts has been a research of interest for over three decades. Krumhansl & Kessler provided key profiles for major and minor keys using an experiment where participants (with at least 5 years of formal musical training) rated how well certain probe tones fit an element (e.g. how well C# fits a IV-V-I cadence in A major) (Krumhansl & Kessler, 1982). These key profiles form the basis of most of the succeeding key-finding algorithms. These are in the form of 12-bin vectors (the correlation value of every chromatic tone) for every key (24 in total). Key-profiles for C major and C minor keys can be seen in Figure 3. According to the Krumhansl-Schmuckler key-finding algorithm, these key profiles are matched with other 12-bin vectors taken from excerpts and the key that yields the highest correlation is identified as the key for the excerpt (Krumhansl, 1990). This method can be found in many MIR-based toolboxes, and is the method of choice for much key extraction research.

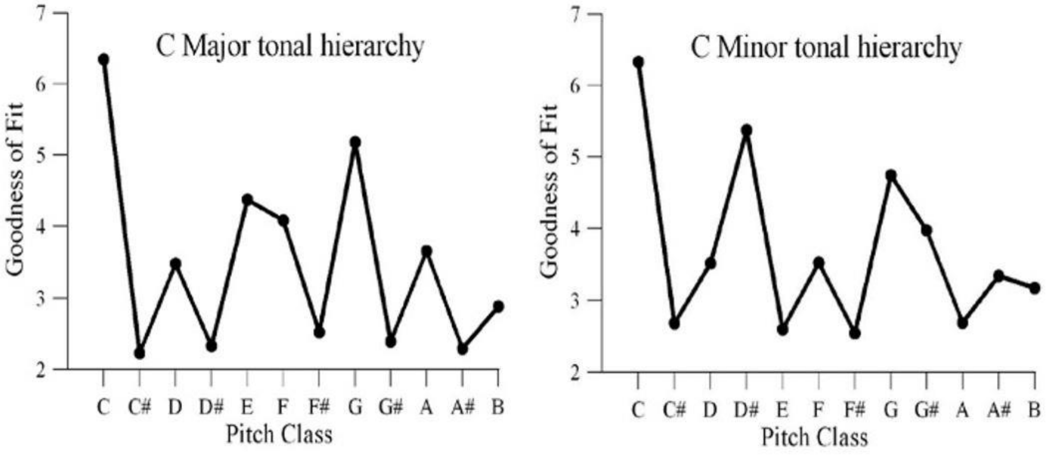

Figure 3. Key profiles for C major and C minor keys.

The much-used Krumhansl-Schmuckler algorithm contained several problems, primarily identified by David Temperley who proposed certain improvements (Temperley, 2001). The Krumhansl-Schmuckler algorithm measures how much a pitch-class is present, factoring in the number of pitch-class occurrences and their durations in a segment. This leads to over-weighting repeated notes. Consider a segment which consists of a C major triad, followed by a series of repeated E's. The system would favour E minor in this case, rather than the correct choice, C major. Temperley suggests a modification where the 12-bin vectors would simply contain 1 or 0 for each pitch-class, without factoring in any form of weight (aside from the key-profile elements), and the algorithm would proceed with the same correlation process. This is shown in Temperley's studies to be substantially more effective.

Another problem is indicated to be the Krumhansl-Schmuckler algorithm's inability to identify modulations, since it is applied to an excerpt as a whole, therefore yielding a single key. Segmentation is proposed by Temperley as a solution. Each segment would carry a different value for key information. Segments are chosen to be the smallest level of metrical units longer than 1 second. To avoid frequent modulations in order to adhere to musical modulation traditions, Temperley imposed penalties if a key for any segment differed from that of the previous segment. This proposed method, while reducing the over-frequency of modulations, carries with it the risk of snowballing, where the occurrence of an incorrectly assigned key would affect the following segments' assignments because of the imposed penalty for modulations.

Expanding upon this study, Temperley proposed treating the key-profiles as probabilities, indicating the probability of each pitch-class occurring in a segment of the given key (Temperley, 2007). Choosing a key for a segment is then a matter of choosing the most probable, given the pitch-class set that is observed.

An interesting study by Chew (Chew, 2007) uses a structure called the "Spiral Array" as key profiles. The Spiral Array is a model where pitch classes are represented by 3D spatial coordinates along a spiral. Chords are defined as triangles in the spiral, and keys are structures consisting of three such triangles (using the tonic, the dominant and the subdominant chords). This mathematical model is a very efficient attempt to model human key-finding, since pitch-class relations are based on fifth intervals (i.e. an increment in the spiral corresponds to an upper perfect fifth interval). Thus, closer pitch classes along the spiral have closer relations in terms of the chords and keys to which they belong. Chew's modeling of key-profiles yields generally accurate assessments of keys and key modulations.

Many more algorithms attempt to map 12-bin key profiles to multiple-dimension representations to improve the modelling of human hearing. The first well-known multiple-dimension space related to pitch-class interactions is "The Harmonic Network" or Tonnetz (shown in Figure 4) (Cohn, 1998). This concept is the basis for the key-finding algorithms that use multiple-dimension representations, and the Spiral Array mentioned above is a mere wrapping up of the Tonnetz to form a 3D spiral. More such representations exist, such as the "Tonal Centroid Space" (Harte et al., 2007) or the "Tonal Interval Space" (Bernardes et al., 2016).

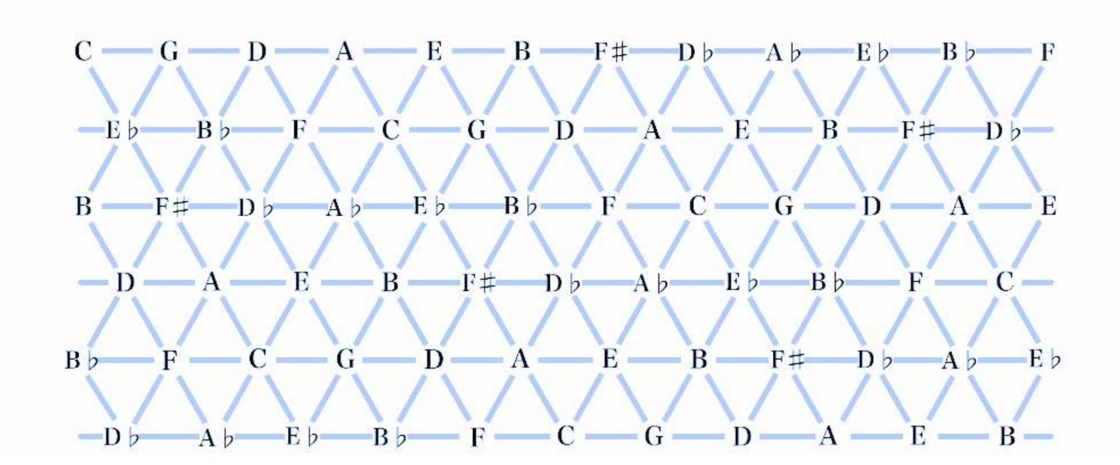
C G D A E B F♯ D♭ A♭ E♭ B♭ F
E♭ B♭ F C G D A E B F♯ D♭
B F♯ D♭ A♭ E♭ B♭ F C G D A E
D A E B F♯ D♭ A♭ E♭ B♭ F C
B♭ F C G D A E B F♯ D♭ A♭ E♭
D♭ A♭ E♭ B♭ F C G D A E B

Figure 4. The Harmonic Network, or Tonnetz.

Chord recognition is a similar task, in which the contents of every segment of a piece is examined, and a chord is inferred. Similar probabilistic models are used (Jiang et al., 2018), but chord recognition has more depth, due to a wider range of variations in terms of chord type (e.g. diminished, augmented, suspended) or inversion. For maximum accuracy, these features should also be determined. This results in extra constraints in the probabilistic models. To achieve such accuracy, different representations for chords are suggested, which are in turn used as features for probabilistic models. At MIREX 2019, one submission for the annual chord estimation task employs convolutional neural networks for 11 different types of chords (S. Lee et al., 2019). The accuracy, however, fails to exceed the baseline chord estimation algorithms, which are annually being resubmitted to MIREX since 2016.

4.4 Music Structure

Segmentation in MIR is an essential task, providing both high-level content and a means to access other high-level content. A segment is usually defined as a region with some internal similarity or consistency. In many low-level content tasks dealing with raw audio, segments are created using windowing techniques for signal processing. These short segments usually last for up to 100 milliseconds. In other tasks, such as tempo tracking, a larger segment size is needed to capture meaningful rhythmic patterns.

In higher-level tasks, a more accurate modelling of the piece is often necessary, achieved by musical segmentation. Depending on the task itself, segment sizes can vary, relative to the beat structure of the piece; they can range from fractions of a beat to entire movements of a complete piece of music. Automatic extraction of different sections of a piece (such as a phrase, a theme or a movement) is especially beneficial in allowing recording studios to edit these sections separately (Fazekas & Sandler, 2007). Alternatively, these sections can be used by live performers to create a mash-up from sections of existing songs.

Beat-based segmentation is essential for beat-tracking tasks such as tempo, meter of rhythmic pattern extraction. It also facilitates the normalization of the time domain, therefore time-invariant analysis can be made. Cover song identification exemplifies a task which greatly benefits from time-invariant analysis (Ellis & Poliner, 2007).

4.5 Mood Detection

Mood detection can be considered an even further level of abstraction among the MIR subtasks. It involves using high-level features such as timbre, tempo, and key as features for further learning algorithms. This is a subtask designed specifically for automatic metadata tagging of a piece, and its target platform consists of the recommendation systems.

Two critical steps in a mood detection task are the feature selection and the feature training steps. Careful considerations should be made before deciding the useful features for determining the mood of a given dataset. According to prior research, key, intensity, timbre and rhythm are the most beneficial features overall (Tzanetakis & Cook, 2002), but this may vary across datasets, especially if songs are from different genres.

Developments in the machine learning field have been especially beneficial for mood detection. With the resurgence of deep learning methods, mood detection gained popularity and research in this area began to output more accurate results. The feature selection step is also automated, where low-level features are directly and automatically chosen by a feature selection algorithm. MIREX 2018 included an automatic mood classification task and submitted papers mention the use of recurrent neural networks as the primary choice of learning algorithm (Song et al., 2018).

5. Challenges in MIR Progress

Over the last two decades, the many technological advances have included digital signal processing and machine learning fields. These advances have allowed for new prospects in MIR and its subfields, but progress has not met expectations. Figure 5 shows the performances of all algorithms submitted to MIREX for several tasks (Schedl et al., 2014). As seen from the figure, the performances show a “glass ceiling” effect, and for some, it is possible that performance will decrease as time passes. The figure only includes research submitted up to 2013, but the glass-ceiling effect continues to present. This section tries to shed more light on these challenges with some points of discussion.

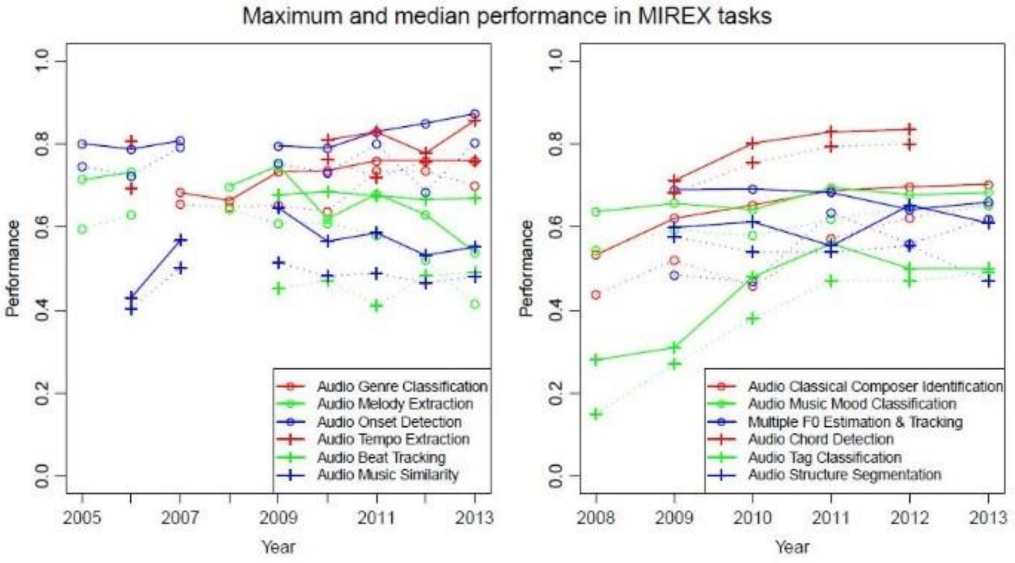

Figure 5. Maximum and median performance for several MIREX tasks [46].

5.1 Dataset Issues

One of the biggest issues in MIR arise from the datasets used for research. Datasets not only establish the ground truth for evaluating MIR algorithms: they guide the research efforts to devise new algorithms. They are also used for training new machine learning models for sophisticated estimation algorithms. As such, both the evaluations and the algorithms are highly sensitive to errors and precision levels in the datasets.

Some datasets contain pieces which suffer from production incapabilities (e.g. slightly faster tempo at one point, detuned audio). These shortcomings are therefore reflected in the estimation algorithms, where precision errors appear, and such pieces should be edited accordingly. Additionally, some pieces are completely inappropriate for the task at hand. An example is given in (Humphrey & Bello, 2015) where a chord estimation algorithm attempts to estimate the chords of pieces with no basic chord structure (some examples given by Humphrey and Bello are "Revolution 9" and "Love You To" by The Beatles, and "Brass Monkey" by the Beastie Boys). Datasets should be constructed based on specific tasks to avoid analyzing non-analyzable pieces.

Since music is subjective by nature, in some cases, an excerpt can be interpreted in multiple ways. The algorithms are designed to represent only one of these ways, and if this representation does not match the ground truth, it is labeled as incorrect. Therefore, when creating ground truth, it would be beneficial to take all of the reasonable solutions into consideration, perhaps by assigning likelihood scores to each label.

Because all of these datasets were constructed using a wide range of sources, and often very different methods, most have different formats, which makes them difficult to use without further processing by any given MIR algorithm. Insufficient documentation for these datasets further increases the difficulty of adapting to each different format. These datasets should be unified under a single format, with sufficient documentation. Such an effort would require a joint task force to drive consensus and receive broad acceptance.

Additionally, the sparseness of datasets is a great challenge in itself. Since it is cumbersome to collaborate with musicians to gather datasets, researchers often use the same existing datasets for many consecutive tasks. Alternatively, tasks are initiated purely because a PhD student donates a dataset to researchers, meaning that tasks are defined by individual researchers, and there are no standards for task definitions or evaluations. An additional issue is that MIREX as a platform has a closed nature (algorithm failures are impossible to track), making it impossible for researchers to learn from failed approaches. This often leads to researchers replicating efforts redundantly for similar tasks. Therefore, collaborating with music providers to frequently obtain new datasets, and building systems to standardize these datasets is becoming a necessity for efficient research.

5.2 User Behavior & Interaction

Currently, the interaction between users and the field of MIR is mostly limited to recommendation systems and tools for music access. The field is mostly focused on developing systems and algorithms, as opposed to analyzing user needs and behavior. For more consistent and more effective user feedback, it is important to broaden the ways in which the user can directly interact with the system, for example, by including different visualization tools, social engagement systems for music, or gaming based on musical concepts. These tools not only provide better user feedback, but also additional contextual information based on group influences in addition to individual preferences.

Since most researchers within the MIR community focus on Western music, different cultures have largely been neglected. High-level content is often designed using Western tonal system such as rhythm, melody, key, scale or tuning. User studies are conducted where subjects respond to stimuli (usually consisting of Western music excerpts) in the Western context. As a result, datasets for MIR research often consist of pieces composed using the Western tonal system. The symbolic representations for music and MIR algorithms in general are catered to the same tonal system. This situation not only impacts researchers and algorithms, but also machine learning outcomes, as they reflect bias in the data. A more generalized approach is necessary for universal applications of MIR tasks, including algorithms, as well as data formats and dataset collections.

5.3 Empirical Studies

A large portion of the research done in MIR aims to propose better tools for users to access music. Empirical research thus seems like a very beneficial method to employ, but these are paid little attention, as research in MIR is mostly systems-focused. Their low impact in MIR research stems from reasons discussed below.

Music is a subjective domain, and it is usually very difficult for non-musicians to explain musical concepts or how they are affected by these. This semantic gap between researchers and users (even musically trained ones) creates problems. Systems may not function as intended, since users may interpret aspects of the system differently. Additionally, difficulties may be encountered during user studies, where subjects may provide inaccurate responses simply because he/she misinterpreted the problem at hand.

Another problem is the scale and the subject demographic for empirical studies. The majority of recent empirical studies have been small-scale, conducted mainly with students, co-workers, or generally people who have some interest in music. This makes them largely non-generalizable and thus unreliable. This is to a large degree unavoidable, since in a fast-changing field such as MIR, large-scale studies are time-consuming, and the study might possibly become irrelevant on completion. Secondly, obtaining mailing lists for surveys are often not possible due to privacy concerns. Researchers therefore resort to easily obtainable samples, given by students and co-workers. Studies are available which provide insightful overviews of the lack of impact for user studies in general (J. H. Lee et al., 2016; J. H. Lee & Cunningham, 2012).

User-centric MIR research is still in its infancy, and many related questions still need extensive research. Modelling each user, therefore creating a personalized system is necessary. Examples can be given where the system is built around specific music listening scenarios, such as driving (Baltrunas et al., 2011) or working out (Moens et al., 2010). However, there are many unexplored but important questions related to user modelling, such as which content or context is more relevant for user modelling, how they influence the user or one another, and if these answers are clear, how to actually build a user-centric system.

6. Conclusion

Despite the challenges that are present within the field of MIR today, overcoming these is possible through careful consideration and steps, especially with the use of powerful machinery and sophisticated algorithms. This paper provided a broad representation of the field, outlining task hierarchy within, and detailing each sub-field. High-level content was given the highest priority, and explained in detail, with historical research background for each content area. Last, the most important challenges that encapsulate the field were revealed, along with discussion points to overcome these challenges. We hope that our efforts in introducing the field and its challenges will attract greater research interest, and increase overall effectiveness of research, overcoming the glass-ceiling effect.

References

Baltrunas, L., Kaminskas, M., Ludwig, B., Moling, O., Ricci, F., Aydin, A., Lüke, K.-H., & Schwaiger, R. (2011). Incarmusic: Context-aware music recommendations in a car. *International Conference on Electronic Commerce and Web Technologies*, 14–15. https://doi.org/10.1007/978-3-642-23014-1

Bartsch, M. A., & Wakefield, G. H. (2002). *To catch a chorus: using chroma-based representations for audio thumbnailing*. *October*, 15–18. https://doi.org/10.1109/aspaa.2001.969531

Bernardes, G., Cocharro, D., Caetano, M., Guedes, C., & Davies, M. E. P. (2016). A multi-level tonal interval space for modelling pitch relatedness and musical consonance. *Journal of New Music Research*, *45*(4), 281–294. https://doi.org/10.1080/09298215.2016.1182192

Bosch, J. J., Bittner, R. M., Salamon, J., & Gomez, E. (2016). A comparison of melody extraction methods based on source-filter modelling. *ISMIR*, *1*.

Boulanger-Lewandowski, N., Bengio, Y., & Vincent, P. (2013). Audio Chord Recognition with Recurrent Neural Networks. *ISMIR*, 335–340. http://www-etud.iro. umontreal. ca/~ boulanni /ISMIR2013.pdf

Burger, B., Ahokas, R., Keipi, A., & Toivianen, P. (2013). Relationships Between Spectral Flux , Perceived Rhythmic Strength , and the Propensity To Move. *Proceedings of the Sound and Music Computing Conference 2013, SMC 2013*, 179–184. http://www.logos-verlag.de/cgi-bin/buch/isbn/3472

Cannam, C., Benetos, E., Mauch, M., Davies, M. E. P., Dixon, S., Landone, C., Noland, K., & Stowell, D. (2015). MIREX 2015: Vamp plugins from the Centre for Digital Music. *Music Information Retrieval Evaluation EXchange*, 0–3. http://www.music-ir.org/ mirex/ abstracts/ 2013/CF2.pdf

Casey, M. A., Veltkamp, R., Goto, M., Leman, M., Rhodes, C., & Slaney, M. (2008). Content-Based Music Information Retrieval: Current Directions and Future Challenges. *Proceedings of the IEEE*, *96*(4), 668–696. https://doi.org/10.1109/JPROC.2008.916370

Chew, E. (2007). *The Spiral Array: An Algorithm for Determining Key Boundaries*. 18–31. https:// doi.org/10.1007/3-540-45722-4_4

Chu, R. (2017). *Survey: How big is your music collection?* https://www.nativsound. com/en/ blog/survey-music-collection-size

Cohn, R. (1998). Introduction to Neo-Riemannian Theory: A Survey and a Historical Perspective. *Journal of Music Theory*, *42*, 167–180. https://doi.org/10.2307/843871

Colby, L. N. (2004). *Digital Audio Workstation* (1st ed.). McGraw-Hill Education.

Collins, K. (2008). *From Pac-Man to Pop Music: Interactive Audio in Games and New Media* (1st ed.). Routledge.

de Berardinis, J., Vamvakaris, M., Cangelosi, A., & Coutinho, E. (2020). Unveiling the Hierarchical Structure of Music by Multi-Resolution Community Detection. *Transactions of the International Society for Music Information Retrieval*, *3*(1), 82–97. https://doi.org /10. 5334/tismir.41

Downie, S. J., Ehmann, A. F., & Bay, M. (2010). The Music Information Retrieval Evaluation eXchange: Some Observations and Insights. *Studies in Computational Intelligence*, *274*, 93–115. https://doi.org/10.1007/978-3-642-11674-2_5

Downie, S. J., Ehmann, A. F., & Lee, J. H. (2008). The Music Information Retrieval Evaluation eXchange (MIREX): Community-led formal evaluations. *Digital Humanities*, 239–240. http://www.ekl.oulu.fi/dh2008/Digital Humanities 2008 Book of Abstracts.pdf

Ellis, D. P. W., & Poliner, G. E. (2007). Identifying "cover songs" with chroma features and dynamic programming beat tracking. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, *4*. https://doi.org/10.1109/ICASSP.2007.367348

Fazekas, G., & Sandler, M. (2007). Intelligent Editing of Studio Recordings with the help of Automatic Music Structure Extraction. *AES 122nd Convention*, 1–14.

Fujishima, T. (1999). Realtime chord recognition of musical sound: A system using common lisp music. *International Computer Music Conference (ICMC)*, 464–467.

Granger, J., Aviles, M., Kirby, J., Griffin, A., Yoon, J., Lara-Garduno, R., & Hammond, T. (2018). Lumanote: A real-time interactive music composition assistant. *CEUR Workshop Proceedings*, *2068*.

Harte, C., Sandler, M., & Gasser, M. (2007). *Detecting harmonic change in musical audio*. 21. https://doi.org/10.1145/1178723.1178727

Hewlett, W. B. (1997). MuseData: Multipurpose Representation. In *Beyond MIDI: The Handbook of Musical Codes* (pp. 402–447). MIT Press.

Humphrey, E. J., & Bello, J. P. (2015). Four Timely Insights on Automatic Chord Estimation. *ISMIR*, 673–679.

IFPI Digital Music Report 2015. (2015). http://www.ifpi.org/downloads/Digital-Music-Report-2015.pdf

Jeon, W., & Ma, C. (2011). Efficient search of music pitch contours using wavelet transforms and segmented dynamic time warping. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, *2*, 2304–2307. https://doi.org /10.1109/ ICASSP. 2011.5946943

Jiang, J., Chen, K., Li, W., & Xia, G. (2018). *MIREX 2018 Submission : A Structural Chord Representation for Automatic Large-Vocabulary Chord Transcription*.

Juheon Lee, Sungkyun Chang, Donmoon Lee, K. L. (2015). Covernet: Cover Song Identification Using Cross-Similarity Matrix With Convolutional Neural Network. *32nd International Conference on Machine Learning, ICML 2015*, *1*, 448–456.

Kahney, L. (2011). *Average iTunes Library = 3K Songs And Is Heavily Mislabeled [And Other Interesting Stats]*. https://www.cultofmac.com/103614/103614/

Kim, T., & Nam, J. (2019). *MIREX 2019: Temporal Feedback Convolutional Recurrent Neural Networks for Music Genre Classification*.

Krumhansl, C. L. (1990). *Cognitive Foundations of Musical Pitch*. Oxford University Press.

Krumhansl, C. L., & Kessler, E. J. (1982). Tracing the dynamic changes in perceived tonal organisation in a spatial representation of musical keys Key-Finding with Interval Profiles. *Psychological Review*, *89*(2), 334–368.

Kumar, R., Biswas, A., & Roy, P. (2020). Melody Extraction from Music: A Comprehensive Study. In *Applications of Machine Learning* (pp. 141–155). https://doi.org/10.1007/978-981-15-3357-0_10

Lee, J. H., Cho, H., & Kim, Y.-S. (2016). Users' music information needs and behaviors: Design implications for music information retrieval systems. *Journal of the Association for Information Science and Technology*, *67*(June), 1301–1330. https://doi.org/10.1002/asi.23471

Lee, J. H., & Cunningham, S. J. (2012). The Impact (or Non-Impact) of User Studies in Music Information Retrieval. *13th International Society for Music Information Retrieval Conference (ISMIR'12), Proc.*, *Ismir*, 391–396.

Lee, S., Jang, J. R., Pool, M., Pool, M., & Pool, M. (2019). *Mirex 2019 Submission : Chord Estimation*. 2–3.

Logan, B. (2000). Mel Frequency Cepstral Coefficients for Music Modeling. *International Symposium on Music Information Retrieval*. https://doi.org/10.1.1.11.9216

McLane, A. (1996). Music as Information. *Annual Review of Information Science and Technology (ARIST)*, *31*, 225.

Moens, B., Noorden, L. Van, & Leman, M. (2010). D-jogger: Syncing music with walking. *Proc. SMC.*, 451–456. https://biblio.ugent.be/publication/1070528/file/1070538.pdf

Pool, O. E. (1996). The Apollo project: Software for musical analysis using DARMS. *Computing in Musicology*, *10*, 123–130. https://doi.org/10.1021/jm900849h

Repetto, D. I., & Selfridge-Field, E. (1997). *Beyond MIDI: The Handbook of Musical Codes* (1st ed.). MIT Press.

Schedl, M., Gomez, E., & Urbano, J. (2014). Music Information Retrieval: Recent Developments and Applications. In *Foundations and Trends® in Information Retrieval* (Vol. 8, Issues 2–3, pp. 127–261). https://doi.org/10.1561/1500000045

Selfridge-Field, E. (1994). Optical recognition of musical notation: A survey of current work. *Computing in Musicology*, *9*.

Song, G., Ding, S., & Wang, Z. (2018). *Audio Classification Tasks Using Recurrent Neural Network*. http://arxiv.org/abs/1606.00298

Stasiak, B. (2014). Follow that tune-adaptive approach to DTW-based Query-by-Humming system. *Archives of Acoustics*, *39*(4), 467–476. https://doi.org/10.2478/aoa-2014-0050

Temperley, D. (2001). *The cognition of basic musical structures*. https://doi.org /10. 1525 /mp. 2005.23.2.189

Temperley, D. (2007). *A Bayesian Approach to Key-Finding*. 195–206. https://doi.org/10.1007/3-540-45722-4_18

Tralie, C. J. (2017a). Early MFCC and HPCP fusion for robust cover song identification. *Proceedings of the 18th International Society for Music Information Retrieval Conference, ISMIR 2017*, 294–301.

Tralie, C. J. (2017b). *Mirex 2017: Cover Song Identification Using Similarity Fusion of HPCPs, MFCCs, and MFCC SSMS*. http://labrosa.

Tzanetakis, G., & Cook, P. (2002). Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, *10*(5), 293–302. https://doi.org/10.1109/TSA.2002.800560

Urbano, J. (2014). MelodyShape at MIREX 2014 Symbolic Melodic Similarity. *Technical Report, Music Information Retrieval Evaluation EXchange*, 0–3.

Weinberg, G., Beck, A., & Godfrey, M. (2009). ZooZBeat: a Gesture-based Mobile Music Studio. *Proceedings of the International Conference on New Interfaces for Musical Expression*, 312–315. http://users.notam02.no/arkiv/proceedings/NIME2009/nime2009/pdf/author/nm090164.pdf