Chapter 5

Music information retrieval

5.1 Introduction

Volume 2 of the subject guide for 2910227, *Interactive multimedia* introduced the very basics of metadata- and content-based multimedia information retrieval. This chapter expands on that introduction in the specific context of music, giving an overview of the field of music information retrieval, some currently existing systems (whether research prototypes or commercially-deployed) and how they work, and some examples of problems yet unsolved.

Figure 5.1 enumerates a number of tasks commonly attempted in the field of music information retrieval, arranged by 'specificity', which can be thought of as how discriminating a particular task is, or how clear the demarcation between relevant and non-relevant (or 'right' and 'wrong') retrieval results is. As will become clear through the course of this chapter, these and other tasks in music information retrieval have applications in domains as varied as digital libraries, consumer digital devices, content delivery and musical performance.

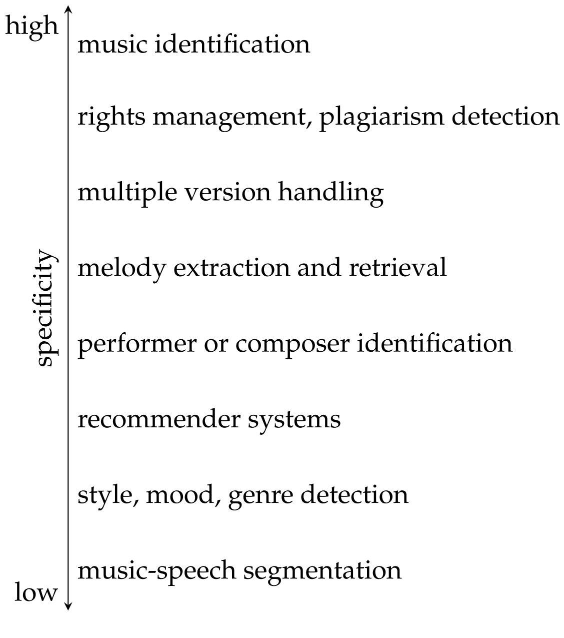
high
music identification
rights management, plagiarism detection
multiple version handling
melody extraction and retrieval
specificity
performer or composer identification
recommender systems
style, mood, genre detection
music-speech segmentation
low

Figure 5.1: An enumeration of some tasks in the general field of music information retrieval, arranged on a scale of 'specificity' after Byrd (2008) and Casey et al. (2008). The specificity of a retrieval task relates to how much acoustic and musical material a retrieved result must share with a query to be considered relevant, and how many documents in total could be considered to be relevant retrieval results.

This chapter includes a number of references to the published scientific literature. Where possible, those references have been made to papers or other documents which were freely accessible online and findable by basic web searches at the time this chapter was written. In addition to this, the International Symposium of Music Information Retrieval publishes the complete cumulative proceedings of the 10 (so far) annual events held, available at `http://www.ismir.net/proceedings/`, where much more information on the topics introduced here can be found.

Additional reading

Müller, M., *Information Retrieval for Music and Motion*. (Berlin: Springer-Verlag, 2007) [ISBN 9783540740476 (hbk)].

Witten, I.H., A. Moffat and T.C. Bell *Managing Gigabytes: Compressing and Indexing Documents and Images*. (San Francisco: Morgan Kaufmann Publishing, 1999) [ISBN 1558605703 (hbk)].

Van Rijsbergen, C.J. *Information Retrieval*. Available at `http://www.dcs.gla.ac.uk/Keith/Preface.html`.

5.2 Genre classification

A huge variety of different kinds of sounds are considered to be music: the very definition of music is fraught with difficulty. Faced with such diversity, between musics created at different times, in different geographical locations, in response to different social contexts, it is natural to attempt to classify music into different varieties or **genres**. For example, one can classify by the geographical area of origin; or by cultural standards (such as the distinction between art music, popular music and traditional music). Each of these broad categories still contains a diverse set of sounds, and can be divided into many smaller sub-categories.

The Western music industry produces genre labels (and indeed genre **taxonomies**: particular divisions of the category of music) for its own purposes. For example, physical shops must organise their music collections somehow, and so Western music retailers divide up the space available to them for categories such as 'Classical', 'Jazz', 'Rock' and 'Soundtracks'; in some cases dividing the space within those categories alphabetically by composer or artist; and in some cases having subgenres. This physical division of the space guides the shopper towards the items they are most likely to want to purchase (Pachet & Cazaly 2000). The same steering also occurs in online music retail; Internet music sellers often allow the user the option to browse the available collection by genre, sometimes to quite a depth of categorisation.

A commonly-attempted task in music information retrieval is to automatically associate a genre label with a given track, working solely or primarily from its audio content. One approach to this is to treat it as a classification problem, and to model genres as clusters in some feature space; by learning the parameters of those clusters from labelled training data, one can then assign a cluster (and hence a genre label) to an unknown track.

For example, a track's acoustic content might be modelled by an overall average spectral feature such as the **cepstrum**, which is a d-dimensional quantity. If for a given genre label we have a set of N tracks for which we have extracted the cepstrum, and we make the assumption that the clusters are represented by single

Gaussians, we can represent the cluster's density function as:

$$p^{(i)}(\mathbf{x}|\mathbf{m},\mathbf{S}) \propto \frac{1}{\sqrt{|\mathbf{S}|}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mathbf{m})\mathbf{S}^{-1}(\mathbf{x}-\mathbf{m})^T\right)$$

by estimating $\mathbf{m}^{(i)}$ and $\mathbf{S}^{(i)}$ from the training data as:

$$\begin{aligned}\hat{\mathbf{m}}^{(i)} &= \frac{1}{N}\sum_N \mathbf{x}^{(i)} \\ \hat{\mathbf{S}}^{(i)} &= \frac{1}{N-1}\sum_N (\mathbf{x}^{(i)} - \hat{\mathbf{m}}^{(i)})^2.\end{aligned}$$

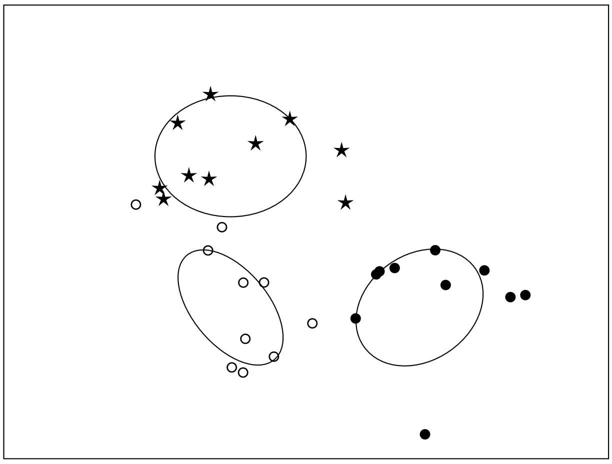

Figure 5.2: Schematic illustration of clustering in a feature space: the individual points represent the position of a piece of music in that feature space, in this context attempting to capture the notion of 'genre'. Because the clusters (whose central support is represented by the ellipses) do not overlap very much, this clustering can be used as a classifier.

Figure 5.2 shows this estimation and clustering schematically: the ellipses represent the concentration of the clusters corresponding to each of the three genre labels present in the model. In this schematic case, the clustering can be used as a classifier; however, in the application of clustering to the genre classification task, the picture is more often like Figure 5.3, where there is significant overlap between the clusters in the feature space, and hence significant mislabelling if the clusters are used to classify unknown data.

While improvements can be made to the situation in Figure 5.3 by the use of more sophisticated features and clustering models (Pampalk et al. 2005), it should be said that because of the different possible meanings and uses of 'genre' that there will always be ambiguity present.

Firstly, the genre of a track is not necessarily related to how something sounds: in many cases, the genre of a track was chosen by a recording label essentially to attempt to maximise the exposure of that track in shops. If a shop had a particular genre near the front, or if the newspapers had been talking about a specific musical genre in the recent past, that genre label would be chosen for more tracks.

In addition, genre being a personal and social construct, the identification of a track's genre differs when performed by different people: not because people do not

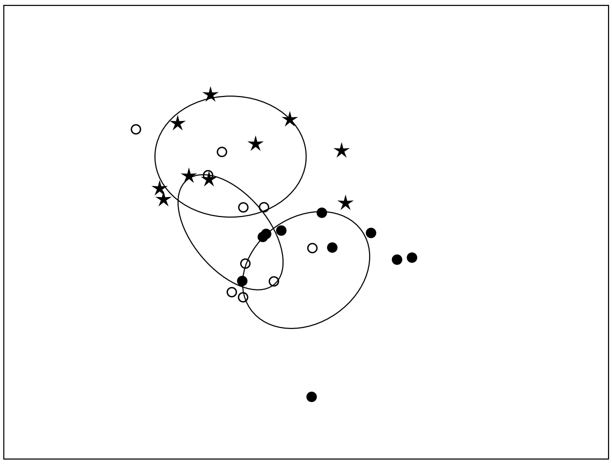

Figure 5.3: A more realistic depiction of clustering in the case of genre: features extracted from the musical content or metadata (apart from the ground truth 'genre' label itself, of course) are not sufficiently discriminating to allow separation of genre clusters.

know the 'right answer', but because there is no *a priori* right answer. People have internal, consistent models of genre (Craft et al. 2007), but they do not necessarily line up perfectly with the industry's labels or with any individual retailer's shop organisation plan. The social aspect of genre suggests a different method, to use community metadata and collaborative filtering as the way to categorise. This can be seen in action in social radio Internet sites such as last.fm or Spotify, which use genre to inform the user's navigation through the collection of available music, and can even be used to build a genre classifier based on words shared in common in text about songs as returned from web searches (Whitman & Rifkin 2002).

5.3 Recommender systems

The primary use of a recommender system is probably in the commercial arena, where the music information retrieval task is to find items which the user would be interested in buying. In this context, there is also a useful evaluation metric for any system purporting to perform this task: success or failure can be quantified in terms of the amount of money spent by users of the system.

Perhaps the most obvious example of such a recommender system, then, is the system generating suggestions of other items to buy at the Amazon[1] online retailer. This is also an interesting system in that it performs no analysis of content or even musical metadata, but works exclusively through association: the website tracks information from a large number of users, remembering details such as users' paths through the collection of available items to purchases and items purchased by the same user; recommendations are then made by establishing a similarity between a given user's profile and some subset of the stored aggregate user data, allowing prediction of suitable items to present for inspection.

One weakness of the system behind Amazon is that its only information about any user is the previous interactions that that user has had with the Amazon service. In particular, Amazon does not know about any purchases made elsewhere, and so

[1] `http://www.amazon.com/`

will naturally suggest items which the user already owns (there is provision for the user to feedback 'I own this'). A more fundamental weakness, from the point of view of music information retrieval if not necessarily from a cost-benefit analysis, is in the fact that the Amazon recommender does not use any musical content or metadata information. This leads to suboptimal treatment of, for example, multiple performances of the same work: given a pair of popular pieces of music (often recorded or covered) there will be many subsets of users with one recording of each pair, which leads Amazon to suggest many instances of the same piece of music if a user shows an interest in any of them.

From the point of view of Music Information Retrieval, Amazon's recommender suffers through its generality: it is applicable to everything from lawnmowers to jewellery. A more targeted recommender is the 'Genius' feature in Apple's iTunes music player, which establishes commonalities between tracks on a user's system with iTunes playlists containing those tracks created by other iTunes users. Again, the commercial motivation is clear: the recommendations generated by the 'Genius' feature come with the ability to purchase recommendations corresponding to tracks not already in the user's collection. Soundbite[2] is another system currently working as an iTunes plugin, similar in nature, but using a summarisation of the musical content of a track ((Levy & Sandler 2006)) rather than the collaboratively-filtered playlist metadata.

The above recommenders use a large, previously acquired set of data, whether from having access to all playlists, purchase data or prior musical feature extraction and analysis: they will therefore only be able to recommend tracks (or however musical works are represented) that exist in their own databases, which consequently are not so useful for exploring niche, targetted collections (or in the case of Apple's iTunes, recommendations based on the Beatles collection). By way of contrast, there exist recommender systems based on navigating a standalone collection, where the emphasis is not in finding new material to offer a user, but in providing a path to navigate through the existing material: a necessary tool in this age of consumer electronic devices with tens of thousands of tracks on them.

There have been many systems designed to visualise and navigate through collections of music, some with more specialised capabilities than others. A recent survey (Donaldson & Lamere 2009) demonstrated over 85 collection visualisers, with different aims and different visual representations, but all aiming to illustrate the range of a collection and in most cases helping to navigate through that collection. Among them are Islands of Music (Pampalk 2001), showing how tracks are grouped by particular acoustic features; the Musicream (Goto & Goto 2005) playlist generator and editor; and the mHashup search interface (Magas et al. 2008) for locating not just tracks but portions of a track by similarity (see the next section) or by geographic location.

5.4 Musical document retrieval

Figure 5.4 illustrates the typical workflow in content-based document retrieval. A database of music 'documents' (usually the audio content, along with whatever metadata is available) has some musical features (such as the **audio spectrogram**, or more usually some transformation of that spectrogram such as a **chromagram**) extracted, and those musical features are stored alongside the documents

[2] http://www.isophonics.net/

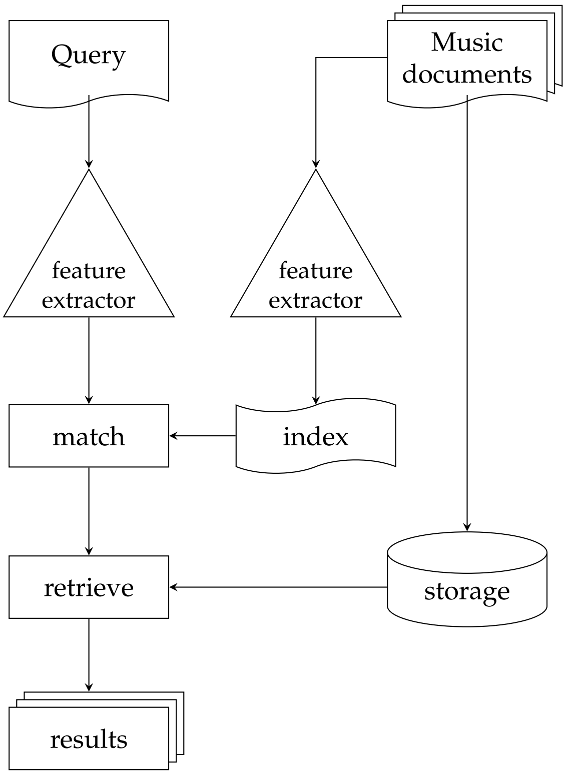

Figure 5.4: An illustration of the typical workflow for content-based document retrieval.

themselves. The retrieval process starts by extracting the same musical features from the query track, resulting in a **feature vector** which can be compared in some distance space with the database document features; if the distance space is indexable, this can be done quickly. Then the retrieval itself merely looks up from storage the documents whose feature vectors are closest to the query's feature vector, up to some maximum distance, and those documents are presented to the user as the results of the query.

This workflow boils down to one of several generic problems: **identity retrieval** for when an exact match is being searched for; the **nearest-neighbour search** for the 'closest' match; *k*-**nearest neighbours** for a *k*-element list of nearest matches; and *r*-**near neighbours**, where near neighbours are only considered for retrieval if their distance from the query is less than *r*.

5.4.1 Music identification

One popular and commonly-used approach to music identification relies on information about the content rather than the content itself. In this case, the documents are not individual tracks but albums, usually in CD form. In this case, used by such online services as freedb[3] and Musicbrainz,[4] the feature extracted, stored and indexed is the disc **TOC** (Table Of Contents), a representation of the start positions and lengths of the tracks on the disc. This feature is highly specific,

[3] Started when the CDDB service was taken proprietary by its owner; see `http://www.freedb.org/`

[4] `http://www.musicbrainz.org`

because it is extremely rare for different albums to share the same lengths of tracks in the same order; thus, an exact match in TOC space identifies an album (and consequently the component tracks) with very high confidence. One weakness of this approach is that slight differences in the generation of CDs, even from the same source audio material, can produce different TOCs, which will then fail to match each other.

Musicbrainz additionally provides identification and metadata retrieval based on an analysis of the musical content of the file, using the AmpliFIND (formerly MusicDNS) acoustic fingerprinting service[5] to retrieve a unique identifier for individual tracks. This allows for more robust album matching, at the cost of higher computational requirements (for computing and matching audio fingerprints rather than merely using the very simple TOC information).

The Shazam service[6] is a purely content-based music identification service, aiming to be robust not only against different pressings of the same CD but also against certain kinds of ambient and digital noise; the recognition service offered even works when the music is transmitted over a mobile phone line from a noisy environment such as a nightclub.

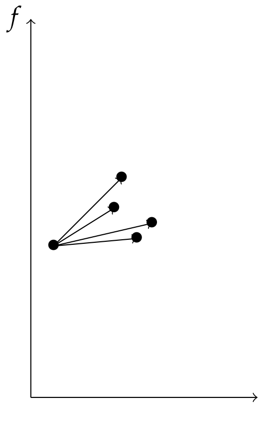

Figure 5.5: A geometric spatial feature, similar to those used in the Shazam music recognition service. The points considered as part of the feature are intensity peaks in the spectrogram, in a frequency range that is carried over telecommunications links such as mobile telephones.

Shazam (Wang 2003, 2006) works by storing an index of groups of intensity peaks in the spectrogram of a track, illustrated schematically in Figure 5.5. The presence or absence of these peaks are not audible in themselves to listeners, but provide a clear marker or fingerprint to the track, as the precise arrangement of those peaks is distinctive. Although the presence of a single such feature is not enough to identify a track with confidence, the specificity of the search can be increased by requiring the query to match not just one such arrangement but two with the same temporal spacing as in the document database (Figure 5.6).

The Shazam identification service is, as described above, robust to ambient and digital noise. However, it will only successfully match against the same *recording* as is present in the Shazam database: live performances or alternate recordings

[5] `http://www.musicip.com/`
[6] `http://shazam.com/`

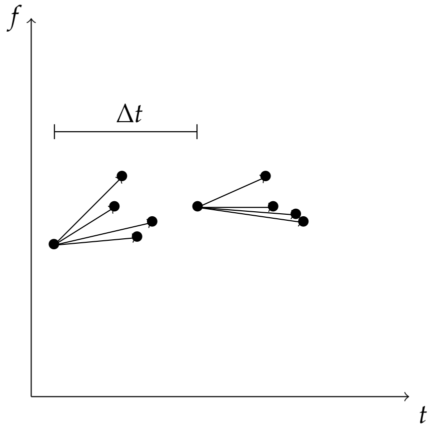

Figure 5.6: The specificity of the Shazam features is increased by requiring not only that the features be the same in query and database index, but also that they be separated in time by the same amount.

(bootlegs from a concert, say) will have a slightly different expressive performance, and so will not be considered to have the same fingerprint. It is for this kind of match, slightly down the specificity scale where we talk about handling multiple versions of the same work, that we need to explicitly perform similarity search, described in the next section.

5.4.2 Content-based similarity search

The systems described in the previous section have the function of searching and retrieving the **identity** of a track, possibly resistant to certain sorts of noise, through a combination of collection metadata and audio fingerprinting. However, many music document retrieval tasks are better described in terms of **similarity** rather than identity: multiple performances of the same work, remixes or mashups of samples from a work; cover versions and variations on a theme will all share more or less of their nature with another of the same class, without being identical. The approach to similar document retrieval is again summarised in Figure 5.4, but here the distance space is not simply defined by an equality predicate but includes the notions of relative closeness.

The concept of similarity is less specific than identity; additionally, the less specific the particular variant of similarity being considered, the more subjectivity comes into play. Therefore, it would be wrong to consider audio similarity search as a single monolithic task; instead, a similarity search must be defined more precisely before it can be performed. Some uses of similarity search include navigation through collections (by association); digital rights management (detecting distorted or sampled copies of copyrighted works); and performance tools through concatenative synthesis, such as in the Soundspotter[7] tool (Casey 2009) – and indeed the classification approach to genre identification described in Section 5.2 can be viewed as a content-based similarity search.

This leads to many different approaches being taken in the general area of similarity search. For some applications, the order in which musical 'events' happen is highly

[7] http://www.soundspotter.org

significant: for example, it would be almost impossible to consider two tracks to be multiple performances of the same work if the notes did not happen in the same temporal order, even if the overall note content was the same; by constrast, one track can be a remix of another while using the musical content of the original in a different sequence.

This difference is mirrored in a pair of significantly different approaches to content-based similarity retrieval: the **bag-of-frames** approach (Aucouturier et al. 2006), where the signal is represented as the long-term distribution of the 'local' (frame-based) acoustic features, discards any information about short-term organisation (such as might be found in musical structures such as bars or phrases); **sequence** approaches, instead, perform matches while keeping the time ordering information, at a cost in computational requirements. This difference in approaches can be characterised as the difference between representing the overall sound of a track, compared with the trajectory taken; the bag-of-frames approach will work well when there is little short-term structure, or when that short-term structure is not relevant to the kind of similarity match being performed.

5.5 Segmentation

In order to perform other analyses of various kinds on musical data, it is sometimes desirable to divide it up into coherent segments; see Figure 5.7 for an illustration. For example, transcription (see the next section) depends on being able to identify individual notes, and any musically ambitious transcription algorithm will also wish to group those notes into bars. Meanwhile, some portable music players attempt to synchronise the music played with the listener's movements (for example, when exercising or running), aiming to select music with the same **tempo** or beats per minute as the repetitive motions. Segmenting at **phrase** or **section** boundaries, identifying high-level musical structure, is of particular use in particular performance tools, such as karaoke machines (Goto 2006),[8] but also for finding a representative segment to use as a 'thumbnail' when a small portion of the track is needed (for example, as a sample in an online music store).

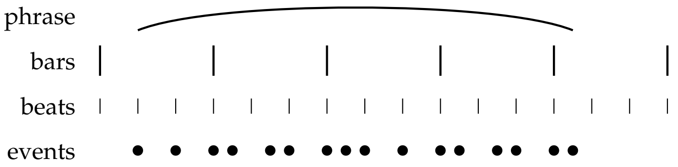

Figure 5.7: An illustration of various levels of possible segmentation: a **phrase** is made up of multiple **note** events, which can be divided into **beats** and **bars**. Importantly, observe that these segmentations do not form a single hierarchy: in this example, the phrase boundary is not aligned with bars.

Onset or event detection is normally phrased as a task in signal processing, where low-level characteristics of the signal are analysed for differences indicative of the introduction of a new signal component (Dixon 2006). However, there should be a distinction made between detecting at what point there is a difference in the signal's characteristics and determining automatically the time at which a listener would

[8] such as in SmartMusicKIOSK, `http://staff.aist.go.jp/m.goto/SmartMusicKIOSK/`

hear the event (Klapuri 1999): the latter can often be relevant for certain applications.

There are many approaches to beat detection, such as in Davies & Plumley (2007); for many kinds of music where there are strong, regular percussive elements, beat detection algorithms can attain a high accuracy, and indeed for the kinds of music which include electronically-generated regular drum tracks (such as many forms of Western popular music), beat detection can be nearly perfect. However, even in such simple cases there is often a simple source of ambiguity in the difference between **tatum** (the fastest regular sequence of events) and **tactus** (the frequency at which people would tap along): often there is a difference of a factor of two, with each tactus beat being composed of two events. Generally, people tend to tap along at speeds between 60 and 180 beats per minute, and music is generally written to conform to that range, but that range does leave scope for differences of opinion as to what the beat actually is.

The identification of musically relevant segments in music is a particularly interesting one: in that in general, a large amount of contextual information must be used to assess what distinguishes different sections from each other. For example, in Western popular music, verses, choruses and other sections are often distinguished from each other by the instrumentation (and hence the timbre) in each section; while in classical Sonata Form, the 'subjects' (segments) are distinguished by their position in the development of the piece. For popular music, a technique (related to the simple approach to genre classification described in Section 5.2) based on classifying individual audio frames to genre labels works adequately, particularly if supplemented by some smoothing (Abdallah et al. 2006) to avoid short sequences of frames being labelled as a section; for classical works, attempting to identify repeated musical material (Rhodes & Casey 2007, Müller & Clausen 2007) is a more fruitful entry point than timbral cues.

5.6 Transcription

Transcribing a musical audio signal into a form of musical notation familiar to practitioners is a seductive goal. Experienced practitioners can perform that task themselves, at least for moderately simple signals, and amateur listeners can memorise and sing along to their favourite tracks. Yet even such apparently simple things such as melody extraction from **polyphonic** audio (multiple sources, as opposed to single source **monophonic** music – not to be confused with **mono** and **stereo** recordings) are beyond the capabilities of current systems and of current understanding.

Even the transcription of a single musical instrument's audio is a difficult task (Cemgil 2004); real musical instruments are all different, have acoustic content different from a pure sinusoid, and allow for expressive performance techniques such as vibrato which further removes their sound from an 'ideal'. The basic idea in analysing monophonic audio is to identify intensity peaks in the spectrogram, and identify high-intensity regions of approximately the same frequency as coming from a note; this hypothesis would be reinforced by finding harmonically-related frequency peaks, which also allow an estimate of the **fundamental frequency** (corresponding to the pitch of the note).

This basic idea for transcription needs considerable refinement for even special cases of polyphonic music, where all the notes have been sampled individually beforehand (Abdallah & Plumbley 2006). The reason that this is such a difficult task is because the human transcriber benefits from years of experience of listening to music generally, and usually the same kind of music as they transcribe; those years of experience provide a strong set of expectations, allowing inference of a clean transcription from the noisy acoustic evidence. To perform transcription automatically to a similar level of accuracy requires encoding of these prior expectations; it appears not to be the kind of problem that can be solved merely by applying more computational resources.

Adequate transcription would have a number of uses, such as in **query by humming** systems, metadata access, and as a compositional or notational tool, as well as a front-end to other music information retrieval systems.[9] However, perhaps surprisingly, even when working from a high-level or **symbolic** representation of music, it can be surprisingly hard to extract information such as the melody (Wu & Li 2007) or chord labels (Rhodes et al. 2007) for that music.

5.7 Summary and learning outcomes

This chapter surveys a number of techniques used in the field of music information retrieval, and discusses the implementation of particular systems intended to perform certain music information retrieval tasks, including strengths and weaknesses of particular approaches.

With a knowledge of the contents of this chapter and its directed reading and activities, you should be able to:

- explain the concept of specificity in the context of music information retrieval
- understand the different ways in which genre labels are generated, and the implications that this has for automatic retrieval of those labels
- describe different forms of recommender systems, suggesting particular approaches for particular recommender applications
- summarise the workflow for content-based similarity retrieval
- describe content-based and metadata-based approaches to music identification and fingerprinting
- describe applications of audio similarity search
- explain the uses of different levels of segmentation, and techniques used to obtain them
- describe a simple method for obtaining a transcription of monophonic music, and the difficulties in general music transcription.

[9] such as the Musipedia database, `http://www.musipedia.org/`