Appendix B

Project ideas

In this Appendix I provide some information about projects related to music information retrieval based on the course that I have taught at the University of Victoria. First I provide some information about how to plan and organize a group project in MIR. A list of past projects done in the course is provided, followed by some additional suggestions. These are appropriate for class projects of small teams (2-3 students). I am really proud that some of these projects have resulted in ISMIR publications. Another good source of project ideas are the online proceedings of the ISMIR conference.

B.1 Project Organization and Deliverables

Music Information Retrieval is an interdisciplinary field so there is considerable variation in the types of projects one can do. The projects described in this appendix differ in the skills required, type of programming, availability of existing published work, and many other factors. The exact details are usually refined over the course of the term through better knowledge of MIR and interaction with the instructor. The expectations for each project are also adjusted depending on the number of students in the group.

Typically projects have the following components/stages:

1. Problem specification, data collection and ground truth annotation

2. Information extraction and analysis
3. Implementation and interaction
4. Evaluation

At least one of these components/stages should be non-trivial for each project. For example if the project tackles a completely new task then just doing a good problem specification and associated data collection might be sufficient, followed by some baseline analysis using existing tools. Another possibility would be to take an existing task for which there is data and code and build a really nice and intuitive graphical user interface in which case the non-trivial part would be the implementation and interaction stage. A thorough experimental comparison of existing techniques for a particular task would also be a valid project. What should be avoided is just taking existing tools, data for a well known task and just getting some results. Excellent projects which would have a good chance of being accepted as ISMIR submissions typically have more than one of these stages being novel and non-trivial. The exact details are worked out throughout the term based on the project deliverables.

The following deliverables are required for each project. All the written reports should conform to the template used for ISMIR submissions and follow the structure of an academic paper. It is a good idea to read a few ISMIR papers in order to familiarize yourself with both the template and the sections that are typically used. Each successive report builds upon the previous one. There is also a final presentation. In the MIR course at the University of Victoria, the project is worth 60% of the final grade.

1. **Design Specification** (15%)

   The design specification describes the proposed project, provides a timeline with specific objectives, and outlines the role of each team member. In addition it describes the tools, data sets, and associate literature that the group is going to use for the project. This report should be 2-4 pages using the ISMIR template. A proper bibliography of 15-20 references should also be provided.

2. **Progress Report** (15%)

   The progress report extends the design specification with specific information about what has been accomplished so far. Specific objectives achieved are mentioned and deviations from the original plan are discussed. The time line is also revised to reflect the deeper understanding of the project. Additional literature can also be provided at this stage. This report should be 1-2 pages using the ISMIR template in addition to the text of the design specification.

3. **Final Report** (20%)

   The final report describes the system developed, the evaluation methodology, and the role of each member. It also describes which of the objectives were achieved, what has been learned and provides ideas for future work. The target audience is a future student of the course who would like to work on a similar project. The final report should be 2-3 pages of text in addition to the text of the progress report. The final document which will contain parts of the design specification, progress report and final report should be 6-10 pages using the ISMIR template. A bibliography of 20-30 references to literature relevant to the project should also be included.

4. **Presentation** (10%)

   The final presentation is typically done with all of the class attending and is a 10 minute slide presentation summarizing what was done and what was learned. It can also include a short video or demo of the system developed. For groups or students who can not attend physically the final presentation, a 10-minute video or audio recording+slides can be used instead.

B.2 Previous Projects

The following projects have been completed by groups of students in past offering of the Music Information Retrieval course at the University of Victoria. There is some variety in terms of difficulty and novelty but that is intentional as students can pick a project that is a more realistic match with their time commitment to the course. The descriptions are taken directly with minimal editing from the abstracts of the final project reports written by the students.

- **Physical Interfaces for Music Retrieval**

  This project explores integration of tangible interfaces with a music information retrieval system. The framework, called Intellitrance, uses sensor data via midi input data to query and search a library of beats/sound/loops. This application introduces a new level of of control and functionality to the modern DJ and expert musician with a variety of flexible capabilities. IntelliTrance is written in Marsyas, a rapid prototyping and experimentation system that offers audio analysis and synthesis with specific emphasis to music signals and music information retrieval.

- **World Music Classification**

  Classification of music intro a general genre is a difficult task even for most humans; it is even more difficult to automate by a computer. Typically music would have attached to it meta-data that would include its genre classification, but as music databases become more prolific, this information is essentially missing. Automatic classification becomes essential for the organizing of music databases. In this project, we classify the genre of international music from a specific area and train a classifier to predict the genre of an unknown song given that it is from the same region. For this project we chose Turkish music as our region of interest.

- **Audio Caricatures**

  We have set out to tackle three problems. The first is how to accurately detect pitch in acoustic songs in wav format. The second is how to translate this information into a high level form that can be read and used by music production programs. And finally we want to create a caricature of the original wav file by excluding certain instruments. The result of our attempt at these problems has been Kofrasi, a perl program that uses text files produced by the pitch detection program, which bridges the gap between analyzing the original wav files and creating the final caricature.

- **An Interactive Query-by-Humming System**

  The challenge to create an effective and robust front-end for a query-by-humming system is a familiar topic in the field of Music Information Retrieval. This paper discusses the application we have created as a means to explore this problem. It first looks at the issues involved in creating an effective user interface and examines our own solution, specifically in relation to how it reconciles ease of use with maximal user control.

- **SIREN: A Sound Information Retrieval ENgine**

Multimedia players and the Internet have significantly altered the way people listen to and store music libraries. Many people do not realize the value of thier personal collections because they are unable to organize them in a clear and intuitive manner that can be efficiently searched. Clearly, a better way of fetching audio files and creating rich playlists from personal music collections needs to be developed utilizing MIR techniques. We have developed SIREN, a multimedia player that has been optimized to perform this task.

- **MIXMASTER: Automatic Musical Playlist Generation**

  Mixmaster is a playlist creator. Start and end songs are specified, as well as a time length. Then, a playlist is created by finding songs that approach the same similarity as the specified end song, while not deviating greatly from the original start and that last as long as the user wanted.

- **Musical Instrument Classification in Mixture of Sounds**

  This project attempts to evaluate the possibility of classifying musical instruments from an amplitude panned stereo or multi-channel mix. First the signal from each individual instrument is isolated by the frequency domain upmix algorithm proposed by Avendano and Jot. Then various features are extracted from the isolated instrument sound. The last step is to classify these features and try to determine what instrument(s) are present in the mixture of sound. The results show that the feature extraction and classification produce decent performance on single instrument, and as a whole, the classification system also produces reasonable result considering the distortion and loss of information introduced by unmix.

- **Ostitch: MIR applied to musical instruments**

  The paper discusses the use of MIR in computer music instruments. This paper proposed and implements a performance time MIR based instrument (Ostitch) that produces ”audio mosaics” or ”audio collages”. Buffering, overlapping and stiching (audio concatenation) algorithms are discusse‘d - problems around these issues are evaluated in detail. Overlapping and mixing algorithms are proposed and implemented.

- **Extracting Themes from Symbolic Music**

  This project investigates extracting the dominant theme from a MIDI file. The problem was broken into two tasks: track extraction and theme extraction. Two algorithms were developed for both tasks and a combination of these algorithms were tested on various MIDI files.

- **Singer Gender Identification**

  Automatic detection of whether a recording contains a male or female voice is an interesting problem. Ideally a computer could be programmed to recognize whether a singer is male or female in both monophonic and polyphonic audio samples. Artificial neural networks are a common technique for classifying data. Given enough training data and sufficient time to train, a neural network should be able to classify the gender of a voice in a recording with fairly high accuracy. This report outlines our results in building an artificial neural network that can determine the gender of a voice in both monophonic and polyphonic recordings.

- **Music information retrieval techniques for computer-assisted music mashup production**

  A music mashup (also know as a bootleg, boot, blend, bastard pop, smashup, and cut-up) is a style of music where two or more songs are mixed together to create a new song. Traditionally, mashup artists use trial and error, record keeping of tempos and keys of their song collection, and meticulous audio editing to create mashups. For this research project I will investigate the use of music information retrieval techniques to assist this production process by automatically determining tempo and keys of songs as well determining the potential of two songs to mix together by using these features in addition to other features such as frequency distribution.

- **Automatic Classification and Equalization of Music**

  For this project we designed a program for a stereo system which automatically classifies the current music being played and changes the equalization to best match the genre of music.

- **Location- and Temporally-Aware Intelligent Playlists**

  As location-aware mobile devices (PMPs, media-centric cell phones, etc.) are becoming increasingly ubiquitous, developers are gaining location data based on GPS, cellular tower triangulation, and Wi-Fi access point triangulation for use in a myriad of applications. I propose to create a proof-of-concept location-aware intelligent playlist for mobile devices (e.g. an Apple iPod Touch) that will generate a playlist based on media metadata (such as tag, BPM, track length, etc.) and profiling the user's preferences based on absolute location as well as route of travel.

Route-of-travel based information could be useful in determine a user's listening habits while in transit to common destinations ("What does the user like listening to on the way to UVic from home?"), but another major factor missing is the time of travel. Does the user like listening to something mellow on the way to UVic in the morning, but prefer something more energetic while heading to UVic in the afternoon? If the user is heading downtown in the afternoon to shop, do they listen to the same music as when heading downtown at 10pm to hit a club? By profiling the user by absolute location, path, destination, and time, more interesting and appropriate playlists may be created using a history of the user's preference than a shotgun approach of using one playlist for the entire day without context.

- **Guitar Helper**

  A system is developed that detects which notes are being played on a guitar and creates a standard guitar tab. The system should work for both single notes and multiple notes/chords. An extension to the system would be to have the system perform accompaniment in the form of beats and/or chords. This will require beat detection and key detection.

- **RandomTracks : Music for sport. Corey Sanford**

  Random tracks is a GUI that allows the user to select a genre of music, or several genres of music to be played amongst each other. These genres can be played for a set period of time (for example perhaps 45 minutes of ambient music, or 3 minutes of punk rock). The specific problem this program will be built for is compiling a playlist for boxing sparring rounds. In gyms, music plays songs of various genres (mostly metal/rock) with the low points of the song sometimes coinciding with the more intense parts of training (the end of a 3 minute round). This program - randomtracks - can choose to play the most "intense" 3 minutes from a song that fits the genre specified by the user. As an example, the user may want to specify: A: 3 minutes, Rock. Increasing intensity. B: 1 minute, Ambient. Random start position. Alternate A and B. (Could also randomize A and B).

- **Robot Drummer Clap Tracking: Neil MacMillan**

  Build a clap sensor - this will offload the signal processing from the Arduino to another chip and eliminate the need to move around the robot's stepper motor wiring (the stepper is connected to the analog input pins). Measure the software delays and physical delays between the stimulus and the drummer's strike at various velocities, to get a table of delays that can be used to predict the next beat. Modify the robot's firmware to accept the clap sensor's digital output and predictively play along with a simple, regular beat pattern. Program the robot to play along with more sophisticated patterns (e.g. varations in timing and velocity).

B.3 Suggested Projects

The following are some representative ideas for projects for this class. They are listed in no particular order.

- **Feature Extraction on Multi-Core Processors**

  Feature extraction forms the basis of many MIR algorithms. It is a very computationally intensive process. However it has low memory requirements and is very straightforward to parallelize. The goal of this project is to explore how modern multi-core processor can be utilized to provide more efficient feature extraction. Experiments to find out what is the best granularity will be conducted using the Marsyas software framework.

- **Genre classification on MIDI data**

  Genre classification has mostly been explored in the audio domain for some time. More recently algorithms for genre classification based on statistics/features over symbolic data such as MIDI files have appeared in the literature. The goal of this project would be to recreate some of the existing algorithms and investigate alternatives.

- **A framework for evaluating similarity retrieval for music**

  A variety of similarity-based retrieval algorithms have been proposed for music in audio-format. The only way to reliably evaluate content-based similarity retrieval is to conduct user studies. The goal of this project is to build a framework (possibly web-based) that would allow different algorithms for audio similarity to be used and evaluated by users. The main challenge would be to design the framework to be flexible in the way the algorithms are evaluated, the similarity measure, the presentation mode etc.

- **Sensor-based MIR**

  One of the less explored areas in MIR is the interface of MIR systems to the user. As more and more music is available in portable digital music players of various forms and sizes we should envision how MIR can be used on these devices. This project is going to explore how sensor technology such as piezos, knobs, sliders can be used for browsing music collections, specifying music queries (for example tapping a query or playing a melody), and for annotation such onset detection and beat locations.

- **Key finding in polyphonic audio**

  There has been some existing work on key finding on symbolic scores. In addition, pitch-based representations such as Chroma vectors or Pitch Histograms have been shown to be effective for alignment, structural analysis and classification. This project will explore the use of pitch-based representations in order to identify the key in polyphonic audio recordings.

- **Query-by-humming front-end**

  The first stage in a QBH system is to convert a recording of a human singing, humming or whistling into either a pitch contour or note sequence that can then be used to search a database of musical pieces for a match. A large variety of pitch detection algorithms have been proposed in literature. This project will explore different pitch detection algorithms as well as note segmentation strategies

- **Query-by-humming back-end**

  Once either a pitch contour or a series of notes have been extracted they can be converted to some representation that can then be used to search a database of melodies for approximate matches. In this project some of the major approaches that have been proposed for representing melodies and searching melodic databases will be implemented.

- **ThemeFinder**

  In order to search for melodic fragments in polyphonic music it is necessary to extract the most important ”themes” of a polyphonic recording. This can be done by incorporating knowledge from voice leading, MIDI instrument labels, amount of repetition, melodic shape and many other factors. The goal of this project is to implement a theme finder using both techniques described in the literature as well as exploring alternatives.

- **Structural analysis based on similarity matrix**

  The similarity matrix is a visual representation that shows the internal structure of a piece of music (chorus-verse, measures, beats). By analyzing this representation it is possible to reconstruct the structural form of a piece of music such as AABA.

- **Drum pattern similarity retrieval**

  Drums are part of a large number of musical pieces. There are many software packages that provide a wide variety of drum loops/pattern that can be used to create music. Typically these large drum loop collections can only by browsed/searched based on filename. The aim of this project is to explore how the actual sound/structural similarity between drum patterns can be exploited for finding drum loops that are ”similar”. Accurate drum pattern classification/similarity can potentially lead to significant advances in audio MIR as most of recorded music today is characterized by drums and their patterns.

- **Drum detection in polyphonic audio**

  Recently researchers have started looking at the problem of identifying individual drum sounds in polyphonic music recordings such as hihat, bass drum etc. In this project, students will implement some of these new algorithms and explore variations and alternative approach. A significant part of the project will consists of building tools for obtaining ground truth annotations as well as evaluating the developed algorithms.

- **Content-based audio analysis using plugins**

  Many of the existing software music players such as WinAmp or itunes provide an API for writing plugins. Although typically geared toward spectrum visualization these plugins could pottentially be used as a front-end for feature extraction, classification and similarity retrieval. This project will expore this possibility.

- **Chord-detection in polyphonic audio**

  Even though polyphonic transcription of general audio is still far from being solved a variety of pitch-based representations such as chroma-vectors and pitch histograms have been proposed for audio. There is some limited research on using such representations potentially with some additional knowledge (such as likely chord progression) to perform chord detection in polyphonic audio signals. The goal of this project is to explore possibilities in that space. Jazz standards or beatles tunes might be a good starting point for data.

- **Polyphonic alignment of audio and MIDI**

  A symbolic score even in a ”low” level format such as MIDI contains a wealth of useful information that is not directly available in the acoustic waveforms (beats/measures/chords etc). On the other hand most of the time we are interesting in hearing actual music rather than bad sounding MIDI files. In polyphonic audio alignment the idea is to compute features on both the audio and MIDI data and try to align the two sequences of features. This project will implement some of the existing approaches to this problem and explore alternatives and variations.

- **Music Caricatures**

  Even though we are still a long way from full polyphonic transcription music information retrieval are increasingly extracting more and more higher-level information about audio signals. The idea behind this project is to use this information to create musical ”caricatures” of the original audio using MIDI. The only constrain is that the resulting ”caricature” should somehow match possibly in a funny way the original music.

- **Comparison of algorithms for audio-segmentation**

  Audio segmentation referes to the process of detecting when there is a change of audio ”texture” such as the change from singing to instrumental background, the change from an orchestra to guitar solo, etc. A variety of algorithms have been proposed for audio segmentation. The goal of this project is to implement the main approaches and explore alternatives and variants.

- **Music Information Retrieval using MPEG-7 low level descriptors**

  The MPEG-7 standard was recently proposed for standarizing some of the ways multimedia content is described. Part of it describes some audio descriptors that can be used to characterize audio signals. There has been little evaluation of those descriptors compared to more other feature front-ends proposed in the literature. The goal of this project is to implement the MPEG-7 audio descriptors and compare them with other features in a variety of tasks such as similarity retrieval, classification and segmentation.

- **Instrumentation-based genre/style classification**

  The type of instruments used in a song can be a quite reliable indicator of a particular musical genre. For example the significant presense of saxophone probably implies a jazz tune. Even though these rules always have exceptions they still will probably work for many cases. The goal of this project is to explore the use of decision trees for automatically finding and using such instrumentation-based rules. A significant part of the project will consist of collecting instrumentation annotation data.

- **Template-based detection of instrumentation**

  The goal of this project is to detect what (and maybe when) instruments are present in an audio recording. The goal is not source separation or transcription but rather just a presense/absence indicator for particular instruments. For example from minute 1 to minute 2 there is a saxophone, piano and drums playing after which a singer joins the ensemble would be the output of such a system. In order to identify specific instruments templates will be learned from a large database of examples and then adapted to the particular recording.

- **Singing-voice detection**

  Detecting the segments of a piece of music where there is singing is the first step in singer identification. This is a classic classification problem which is made difficult by the large variety of singers and instrumental backgrounds. The goal of this project is to explore various proposed algorithms and feature front-ends for this task. Specifically the use of phasevocoding techniques for enhancing the prominent singing voice is a promising area of exploration.

- **Singer Identification**

  The singer identity is major part of the way popular music is characterized and identified. Most listeners that hear a piece they haven't heard before can not identify the group until the singer starts singing. The goal of this project is to explore existing approaches to singer identification and explore variations and alternatives.

- **Male/Female singer detection**

  Automatic male/female voice classification has been explored in the context of the spoken voice. The goal of this project is to first explore male/female singer detection in monophonic recordings of singing and then expand this work to polyphonic recordings.

- **Direct manipulation music browsing**

  Although MIR for historical reasons has been mostly focused on retrieval a large part of music listening involves browsing and exploration. The goal of this project is to explore various creative ways of browsing large collections of music that are direct and provide constant audio feedback about the user actions.

- **Hyperbolic trees for music collection visualization**

  Hyperbolic trees are an impressive visualization technique for representing trees/graphs of documents/images. The goal of this project is to explore the potential of using this technique for visualizing large music collections. Of specific interest is the possibility adjustment of this technique to incorporate content-based music similarity.

- **Playlist summarization using similarity graphs**

  Similarity graphs are constructed by using content-based distances for edges and nodes that correspond to musical pieces. The goal of this project is to explore how this model could be used to generate summaries for music playlists i.e a short duration representation (3 seconds for each song in the playlist) that summarizes a playlist.

B.4 Projects that evolved into publications

Some of the best previous projects in the MIR course evolved into ISMIR publications (typically with some additional work done after the completion of the course). The corresponding publications are a good starting point and hopefully will inspire you to work hard on your projects with the eventual goal of an ISMIR publication.

Drum transcription for audio signals can be performed based on onset detection and subband analysis [101]. There are some interesting alternative ways of specifying a music query beyond query-by-example and query-by-humming. One possibility is beat-boxing [?] and another is various interfaces for specifying rhythm [54]. Tabletop displays provides interesting possibilities for collaboration and intutive music browsing [46]. Stereo panning information is frequently ignored in MIR in which typically stereo signals are converted to mono. However stereo panning information is an important cue about the record production process and can be used for classifying recording production style [100]. Smart phones contain location and acceleration sensors that can be used ot infer the context of user activites and create personalized music based on the occassion [77]. Audio analysis can be used as an empirical analytical tool to study how DJs select and order tracks in electronic dance music.