RWTH Aachen University
Media Computing Group
Prof. Dr. Jan Borchers

Post Desktop User Interfaces
WS2005/2006

Adaptive Selection/Composition of Music

Yang Yang, Markus Kohls

Matrikelnummer 255598, 234348

17.11.05

Tutor: Jan Buchholz

Contents

1 Introduction

1.1 Music in our Everyday Life

Music plays a very important role in our everyday life. It can influence our mood, it can shorten waiting times or just is used in shops to improve sales figures of the retail industry. With the development of digital music and audio compression technologies, the consumer is able to access big archives of different styles of music. While this is leading to more diversity in what consumers are listening to, it creates the need for an improvement of organisational structures in archiving retrieving and playing of Music. Also creators, composers or Disc Jockeys are faced with the increased amount of possibilities that digital music nowadays allows.

1.2 Requirements of Creators and Consumers of Music

With advancing technology, more and more expectations of users can be met. In our paper we deal with systems that accomplish a subset of the requirements of the modern user. The focus of the here presented systems lies in accomplishing the following often encountered tasks:

- The automated selection of music based on one users' likes or dislikes
- The automated selection of music in shared spaces, based on the likes or dislikes of two or more people
- The composition of music without the necessity of learning an instrument (intuitive composition of music)
- Silent notifications of one or more persons in shared spaces without other people recognizing them
- Intuitive Navigation via audio

1.3 Existing Applications for Adaptive Selection/Composition of Music

There are many existing applications, which already address the requirements of the modern consumer. For example, the BMS Business Music System[1] is an automatic music playing and selection system for stores or other commercial places. This system can automatically play background music or announcements according to the time of the day. It has the following features:

- automated music selection and mixing according to set rotation rates and day-part scheduling.
- automatic day time volume changes (when for example your business might be noisier it increases volume).

- mixing of music tracks seamlessly with automatic cue-in and cue-out time scanning

An existing application aimed at the automatic composition of music is called COMPOzE [2]. COMPOzE is a system for the automatic composition of music and was developed with a new constraint programming language Oz. Based on the musical parameters describing musical terms that users input, the system will try to figure out how the given intentions ought to be realized. Then it takes those parameters all together and uses an harmonic progression fix, the metric rhythmic and harmonic structure to create a new music for the user.

1.4 Projects in Recent Research

In this paper, we will examine some other new systems for composition/selection of music that were published in recent years. We will present the method of the realization and the experimental results for every systems. In Section 2 we present 5 different system for selection of music and we classify those 5 systems in 3 parts. One part is a system which selects music only for an individual person. The other part talks about systems which are used in an environment shared by multiple people. Beside those two different kinds of sytems, we will introduce a system named soundpryer, which has the goal of easy sharing of songs. In Section 3 we want to talk about some systems aimed at the automatic composition. The first two use genetic algorithms, while the third is a sensor-based approach. The last 2 papers deal with project, which try to notify users via audio in some way. While it's not automatic song composition per se, it certainly fits in the field of Composition of Music, In Section 4 we'd like to discuss our results, and mention what got our attention, while looking through the papers.

2 Selection of Music

2.1 The Requirement of Selection of Music

With the development of internet technology, music fans are able to quickly download online digital music by using broadband access to the internet. And the portable mp3 players with large storage have made it possible for users to casually carry around thousands of digital songs. More and more shopping markets play popular music to get the customer's attention. Background environmental factors become more important. The problem for the staff is how to decide, which music should be played. Moreover, how in a shared environments choose a music which suitable for as far as possible the most people's taste?

2.2 Various Systems for Selection of Music

2.2.1 Music Retrieval by Melody Style

Before we introduce this system[9] we should first explain what is content-based music retrieval (CBMR). Music content is like a portion of the actual melody of the song, such as pitch and interval contour sequence. The task of CBMR system is to return music objects similar to query in syntactic properties. These approaches provide users the capability to look for music that has been heard. This system is a CBMR system, in which a technique has been developed for querying music by melody style. And with this technique, the user can not only discover the music by melody style which he has heard, but also query mixed style as he or she wish to. If the user wants to retrieve music, for example like romantic music, the first step is to discover the common characteristics of the selected group (romantic music) and the unselected group of music examples respectively. By using the method, the system can make a conclusion, that in romantic music, the melodies are longer, more dramatic and emotional. Moreover, tempos are more extreme. With this conclusion, it finds the discrimination between the characteristics of these two groups. The result of this step is a two-way classifier. At last a ranking function is employed to measure the degree of relevance between a music object and the query style based on the two-way classifier. Given the ranking function, all the music objects in the library are evaluated and a ranking list is produced and output to the user. From this ranking list the user can choose a new kind of music, which sounds like romantic music. On the other hand, if the users want that the returned music objects should be more similar to romantic style but also have a little feeling of modern style the user input is different. For mixed style of music, the only difference lies in the number of music groups. In single style of music selection, there are only two music groups, one for the selected group and the other for the unselected group. But in mixed style of music selection, there are more music groups in the library. (in our example situation, we have two selected group: romantic and modern and another unselected group).The generation of melody style pattern set for these groups of music is similar like a selection of one style of music. Ranking is done by multiplication of the ranking scores respective to the styles.

2.2.2 Music Retrieval Based on Relevance Feedback

The next system[7] which we present efforts also on content-based music retrieval. The goal of this system is to retrieve a specific song from a large music database based on the user's musical preferences and it can choose a song, which the uses are expected to like. As we all know, the music preferences are very ambiguous. The ultimate preference of a song is most likely derived from a complex relation of all musical elements, which is extremely difficult to define, even for the users themselves. In this research the authors developed, based on a tree-structured vector quantization algorithm (TreeQ),a new music retrieval method to find out the user's music preferences. At first, the user will in a learning process hear some sample music. From those samples of learning data the user will select a set of songs, which he or she likes (good songs), and a set of songs, which he or she doesn't like (bad songs). After this, a tree is generated based on those information and a vector representing user preferences can be made by inputting the "good songs" from the learning data set through the tree. And also a vector representing "bad songs" can be generated by simply inputting all "bad songs" from the learning data set through the same tree. Vectors of songs which are not included in the learning data set can also be generated by inputting each song through the tree. Therefore, there may be situations where users are unsatisfied with retrieval results of the system, and feel the need to provide additional information regarding their preferences. The most efficient way to implement this is to collect relevance feedback from the user, and update the user profile based on the collected relevance feedback information. Now after the system gets to know the user's music preferences, it can retrieve music data of the specified musical genre, like jazz or rock.

2.2.3 MUSICFX

Those two systems which we have just proposed are all for the individual music consumer. But in an environment shared by multiple people, for example office or fitness center, how can we select a music which is suitable for as far as possible the most peoples' taste? We have just explained what a CBMR system is. To compare with CBMR systems we present another kind of system for selection of music: a so called MIR system. This is a system,in which the query is based on text-based metadata, like title, artist etc. MusicFX[11] is a group preference arbitration system. It was developed for a fitness center named FX. By using this system it allows the members of this fitness center to influence, but not directly control, the selection of music. The purpose of MusicFX is to adjust the selection of music playing to accommodate the musical preferences of the people working out at any given time. As other systems for selection of music, MusicFX need to know the members' musical preferences. To resolve this problem MusicFX uses a database. Each FX member submits an electronic enrollment form upon first joining the fitness center. The FX member specifies his or her preference for each musical genre. All the fitness center members' preferences will be stored in this database. Every member also has a badge. By swiping this badge across a proximity badge reader, they can login to the MusicFX system. The agent would know the music preference of this member immediately by using the database, and on the same time, it also would calculate how many members are currently present in the fitness center. By using a Group Preference Arbitration Algorithm, the computer will calculate the preference values. These values are then squared in order to widen the gap in selection probabilities between the more popular categories and the less popular categories. Once this group preference value is computed for each category, the list of values is sorted in descending order, such that the most popular category is first and the least popular is last. Based on this list music will be played.

2.2.4 Flytrap

Flytrap[6] is also a MIR system for selection of music. But it is unlike as MusicFX, which requires the users fill out a form to know their music taste. Flytrap works by paying attention to what music people listening to on their computers, and derives a user's musical tastes from observation. Flytrap uses the Real Jukebox[3] to observe the user's musical tastes. This Real Jukebox is installed on each user's personal machine and can gather information about what tracks the user is listening to. The tracks will be recorded in Flytrap's database. So the user's preferences are learned through the act of listening to the music as he or she would normally do. Each user of Flytrap also has a radio frequency ID badge that lets the system know when they are nearby so that the system will automatic pick up his presence (login). After the user login, his music preference is found out by using Flytrap's database. The system will choose a song, which is similar to the one the user often listens to. If another user of Flytrap joins, the system comes up with a song that neither of the users know, yet they both enjoy. This is accomplished by a voting machine, which has the same function as the Group Preference Arbitration Algorithm of MusicFX. Voting is based on artist and genre information. A user's Flytrap agent will give a song a high vote if it's an artist they've listened to previously. Songs that get more votes have a higher probability of being played. Flytrap and MusicFX use a random selection policy to resolve this problem. Flytrap has a Virtual DJ. Virtual DJ decides finally which music should be play in next time. It is based on two rules:

1. Never play two tracks by the same artist in a row and
2. Maintain loose genre coherence across tracks.

The result is a less repetitious playlist, and also one that frequently drifts into new areas (because this rule significantly reduces the number of choices in a given genre available for play).

2.3 SoundPryer

At first SoundPryer[4] is a mobile music sharing tool. It provides the possibility that the users can share music when they are driving a vehicle. SoundPryer is based on a peer-to-peer software architecture and thus provides sharing in a wide variety of mobile settings. To share music in a car, the traditional way is to play it loud to other passengers. Sondpryer provides a new method that sharing is accomplished by streaming MP3 files via WLAN. First a user selects a few music tracks, adds them into a playlist, which later in the car may be played. This playlist will be uploaded to a web server. His or her list will be taken together with other users which have same music tastes. This process is aimed to maximize the possibility that the user has always something to listen to. The functionality of SoundPryer is dependant of the number of co-located peers. After the user activates the playback mechanism, there are four states which are provided by Soundpryer, and the user can choose one of those four states to enjoy the music from the co-located peers. The states are:

- Manual/local: SoundPryer plays and broadcasts a file in the playlist. The user may manually select a peer present in the list of available remote peers.
- Manual/remote: the application receives and plays the broadcast stream from a remote peer. When connectivity is lost, or deliberately terminated, the application resumes to manual/local state.

- Automatic/local: Similar to the manual/local state. However, when a remote peer appears, the playback and broadcasting of a local file is stopped.

- Automatic/remote: When connectivity is lost, a new remote peer is selected automatically. As long as there are at least two peers present, after a fixed interval SoundPryer terminates the current connection and randomly selects a new peer. If no peers are present, operation resumes to the automatic/local state.

2.4 Discussion

We have presented 5 different kinds of systems for selection of music. The first one is a CBMR-based one and retrieves a specific song from a large music database. The second one uses the user's music preferences to choose a song. But they are both used for an individual music user. They are not suitable for a group of people. On the other hand,two systems MusicFX and Flytrap are used in an environment shared by multiple people. But to compare them with the first two, they are not good for the user which want to discover a music that he never heard, although the MusicFX and Flytrap have their own random selection policy. And Soundpryer is a system which select music by choosing the co-located peers which have the same music taste as the usern and plays the music by streaming mp3 files.

3 Composition of Music

With the ongoing development of digital music technology, there were always attempts to automate the composition process. Although the approaches had the same objectives, the ways and means of accomplishing the tasks were very different. We would like to present you 5 different projects to demonstrate the wide field of methods used in automated composing. From those 5 projects, 3 are aimed at automating song composition, and 2 are in the field of composition of music, but have different goals then just composing a song.

3.1 Composition of Music Using Human Evaluation

Many approaches in different fields of science often begin with looking what solutions nature found to solve problems. One interesting field for Computer Sciences is the Genetic Algorithms (GA) approach. By imitating Nature's selection process of the fittest, this project tries to create a song using GA Operations which try to emulate biological evolution processes, like mutating and recombination of DNA. The result of such approaches often depend on how the "fitness" of such individuals (in this case songs) is measured. In this Project[12] it is done manually by humans, and that is why this it's called an "Interactive Genetic Algorithm"-Approach.

3.1.1 Defining the Properties and Creating the Population

To start with a certain number of individuals a lot of preconditions are set fixed to ease the process. The goal is set to create music which has 1 main melody and 4 backing parts. For example the backing parts for that are precomposed, and come from a publicly available book,called "Electone Backing Pattern Book". Also the length of the music is limited, to 4 bars with a time signature 4/4 because a longer song would be more expensive to evaluate. A lot of more assumptions or cognitions on "harmonic Music" are predefined, and limit the range of possible created music. To create new music via GA operations each song takes the role of a chromosome. For creating a Start-Population, the user makes a decision on which music he wants to create (e.g. "cheerful Music"). The System then asks its database (containing the predefined rules of Music Theory) and generates 200 Chromosomes, of which it selects 20 randomly. Then the user comes into play: By evaluating all 20 songs, the GA gets the "fitness"-data of each chromosome, and then applies GA Operations on them. The outcome is again a new generation of 200 chromosomes of which 20 are randomly selected. This circle of operations is repeated until the user is satisfied with his creation.

3.1.2 Determining the Fitness of each Chromosome

The most important part for the outcome of this process is the evaluation. That is why it is interesting to take a closer look to how it's realized in this project. First after listening to a song the user can choose, how he liked a song compared to a reference song, which was chosen by the user as the "best" song a generation before. This is done by a 7 point scale, on which ratings for good and bad go from -3 to 3. Then he also can choose certain parts of a single song, which he very much liked or disliked, and can mark them "good" or "bad". That way, the system knows which passages were very good, and can use them in the new generation. The last thing the user has to do in every evaluation round is again choose the song he liked most as a reference song for the next generation.

3.1.3 Project Results

A small test was made with 20 participants trying to create a "cheerful" song. The data shows that the used GA operations seem to have the expected effect, and evaluation ratings of the songs almost increase every generation round. Also the users said that the new songs met their evaluation feedbacks. But the expected outcome of only "cheerful" music didn't quite come true. Many different styles of melodies were created.

3.2 A Generate and Sense Approach to Automated Music Composition

A very similar approach[8] was chosen by a Team of the University of Augsburg. They also chose a Genetic Algorithm to compose music. For getting some initial Data, a test evaluation was made, in which 10 users evaluated about 1422 automatically generated sound snippets. The users had to say, if the music heard was "disqueting" vs "relaxing", or "pleasant" vs "unpleasant". Also while the sound snippets were heard, several phyisiological signals were taken: a "Electrocardiogram" (ECG), "Electromyogram" (EMG), Galvanic Skin Response (GSR) and Respiration (RESP). Afterwards the team tried to map the measured data to the evaluation results with a so-called Fuzzy Rule Base System. (This means classifying the signals from "very low" to "very high", and then for example define simple if-then-else rules to describe the users feelings, like relaxation). With these two datas (user evaluation and physiological data) a Genetic Algorithm Fitness Function is beeing calculated. The GA Process in princip works like in the approach before. A sample of beat patterns is created, and in every iteration a Fitness Function decides, which candidates are sorted out, and which may reproduce. In a first learning phase, only the user evaluations are used to initially train the system how the physiological data reflects the users mood. Then when the system is trained, the Fitness Function is computed only based on the physiological feedback. That means if a song is meant to be relaxing, and the Fuzzy Rules working on the physiological data also predicted that the song was relaxing, the chromosome gets a high fitness value. With such an approach one is able to map physiological data onto the users feeling and understanding of music.

3.2.1 Results

In a statistical evaluation the GSR-Signal (measures skin secretion) can be identified as a useful indicator for the attributes "disquieting" and "relaxing". To predict positive and negative emotional reactions to music, the EMG (function of nerve routes) measurement is a very promising value. Also influencing the users emotional state was possible. By repeated playing songs of music rating "disquieting" the skin secretion increased, while by playing songs with rating "neutral" or "relaxing" it decreased.

3.3 Sonic City: The Urban Environment as a Musical Interface

Another very different approach was done by a swedish group of scientists, called Sonic City[10]. The aim was to create music by just walking through the city. As this is comprehensible considering the many different impressions and feelings a city awakes in oneself, the technical implementation is hard considering that a human uses all his senses to get an impression in the city, and the generated music should reflect this.

3.3.1 Sensors

Two different kinds of data are used to create the Music. User Data, measured at the user himself, and Environment Data (for example heart rate and temperature). This data can be discrete or continuos, and so the music creation process needs to be prepared for such different kinds of input.

3.3.2 Sound Design and Control

While getting the data, it has to be also decided what music is generated from that and how it can be controlled. An interdisciplanary team of people working at different fields of sound and perception of sound were working on the project to develop this methods. They defined a sound design space, in which events can take place. That means it can be a more foreground or more background event, and musically more ambient or more rhythmical. For each class of event some sounds were created. Also it was decided that the sounds should maintain a close relationship soundwise to the real city soundscape. All the sound content is, when walking through the city, processed in Real Time. Confronted with the question how the sound will be controlled by the user the team invented a profile, which represents the personal affinities of sound creation. The user can choose between highly user controlled music vs. mainly environmental influenced music and can determine the amount of diversion in the generated soundscape (very random vs. very deterministic).

3.3.3 Sound Mapping

When having the data, the sounds and knowing the user's preferences there is only the need to decide how the inputs are actually are mapped on sounds. For that the team defined two abstraction levels on data input and output. The input would be divided in a low level abstraction, which is the pure user and environment sensor data itself and a high level abstraction such as "indoors" or "crossing the street", which is interpreted from the sensor data, Those two abstraction levels are mapped onto their counterparts on the output side. The low level abstraction output side consists of different spectral variables (like envelope or timbre) and also triggering of short musical events. While the continuos input data is more suited to be mapped to those spectral variables, discrete input values are used to trigger those musical events. On the abstract level output side, there are structural composition variables, which are high level layer and structure guidelines. Those are as said before mapped onto and influenced by the high level input.

3.4 Seamless User Notification in Ambient Soundscapes

Not all projects which work on automatic composition of music have the creation of a new song as a goal. There are other demands on systems in that field,too. One example, one can easily imagine, is the notification of someone or a group of people without the other people around recognizing this notification. This is for example usable in stores or banks where security personal needs to be silently alarmed. This can be accomplished by incorporating certain instruments or motifs in the music, which are chosen before, and when a user hears his or her instrument or motif in the music he know what it means, while other people aren't noticing anything.

3.4.1 Soundscape Composition and Prototype

In this actual project[5] in order to create a soundscape, which sounds "complete" to the normal user, an effort was made to create 2 jazz compositions. They sound quite finished to the user when only the core part is played, but also are designed that way, that one can add an additional layer of sound patterns. This optional patterns are the instruments or rhythmic patterns which are used to notify people. The actual prototype was developed with a special audio framework for rooms which is called "SAFIR". They use it to play the core part, and mix the notifying layers when triggered. It is also possible to direct the audio sound depending on the users location. That way the user is not only notified, but knows also which direction the event is coming from.

3.5 Navigation via Continously Adapted Music

Another interesting application for automated music generation is the guidance of a person via continuosly adapted Music[13]. As Navigation Systems in cars are already in use in our everyday life, there is also a need for navigation when going by foot or bike. As many people also like listening to music while going somewhere, this habit can be used to integrate navigation information smoothly into the current music. This is done by shifting the balance to indicate where to go.

3.5.1 Evaluating the Protoype

By constructing a Lab Prototype, called "Ontrack", the approach was tested with a group of 25 subjects in a virtual 3D environment. One group used the system, the other one used a control prototype (called "Map"), in which the music stream isn't adapted in any way but therefore the users get a birds-eye view map from the virtual world. They let each user run three routes in random order. The analysis of the data showed a suprising result: When looking at successful arrivals the needed time with the Ontrack System for the route is statistically seen not different from the Map group. Also a user evaluation showed that the Navigation via Ontrack is seen not that exhausting than with the map.

3.6 Discussion

As one can see, attempts to create artificial sounds with user interaction has many different approaches, goals, and also setbacks. But similarities are recognizable, too. Almost all approaches rely on precomposed or predefined patterns and sounds. While Sonic City uses some city sounds and some preference values for creating music, the Genetic Algorithm approaches also incorporate many results from Music Theory into their construction algorithms of the chromosomes. That is why the Sonic City approach allows to create music out of these bounds and can be considered the more freely, more undetermined approach. On the other Hand while Sonic City music creation is more or less a one time ad-hoc creation when walking through the city, the GA Approaches try, with the help of user interaction, to improve Music again and again each new generation. So as Sonic City really tries to make the city usable as a music instrument which can be played by gestures (i.e. by walking around), the GA approaches can be more or less seen as a "brute-force" music creation, which need much time to evolve. Both Projects intend to map certain moods onto the Music. In Sonic City the impressions of the city were tried to be mapped onto the sound of music, while the GA approaches beforehand tried to make the users select more "cheerful" songs. Sonic City is still very experimental and so the success of the proposed strategy to create music under the impression of the city can't still be decided. The GA approach was able to breed a wide range number of short songs with different moods. So the initial aim to just create cheerful music there seemed to have suffered from the many GA operations and user evaluations that were made. With a different goal the Seamless User Notification is a more pragmatic approach to create music. It also is dependant on precreated songs and sounds, so making use of it in bigger environment with beeing able to notify many different people is very expensive, as every sound has to be created by hand. Also it has to be tested, if the created notify sounds fit to the main songs. Both the notification and navigation approaches share the advantage, that they need not much attention to be used. They can easily be used along the way, that means the seamless notification integrates very smoothly into store or bank atmosphere, so the clients even don't notice anything when the system is used. Also the navigation approach is just very intuitive and easy to use. Users almost don't really notice consciously, that they follow the sound.

4 Summary

Some of the projects are only of scientific use, others already may find actual use in everyday systems. The systems for selecting music for a single user or audio notification applications are more applied. The group preference or automatic song composition approaches tend to be more experimental. But when we look at the potential which lies in every project, one can easily see, that automated processes in digital music are only at its beginnings. Existing commercial companies in digital music business will seek to categorize and improve user experience music more and more. Also we only were able to present a very narrow subset of the users demands and possibilities for other applications which lie in every project. One can only assume what the potential integration of audio technology in every object that surrounds us holds which possibilities.

References

[1] http://www.nch.com.au/bms/.

[2] http://www.iscs.nus.edu.sg/~henz/projects/COMPOzE/.

[3] http://www.real.com/products/realjukebox/index.html?src=home.

[4] F. Axelsson and M. stergren. Sound pryer: Joint music listening on the road, 2002.

[5] Andreas Butz and Ralf Jung. Seamless user notification in ambient soundscapes. In *IUI '05: Proceedings of the 10th international conference on Intelligent user interfaces*, pages 320–322, New York, NY, USA, 2005. ACM Press.

[6] Andrew Crossen, Jay Budzik, and Kristian J. Hammond. Flytrap: intelligent group music recommendation. In *IUI '02: Proceedings of the 7th international conference on Intelligent user interfaces*, pages 184–185, New York, NY, USA, 2002. ACM Press.

[7] Keiichiro Hoashi, Kazunori Matsumoto, and Naomi Inoue. Personalization of user profiles for content-based music retrieval based on relevance feedback. In *MULTIMEDIA '03: Proceedings of the eleventh ACM international conference on Multimedia*, pages 110–119, New York, NY, USA, 2003. ACM Press.

[8] Sunjung Kim and Elisabeth André;. A generate and sense approach to automated music composition. In *IUI '04: Proceedings of the 9th international conference on Intelligent user interface*, pages 268–270, New York, NY, USA, 2004. ACM Press.

[9] Fang-Fei Kuo and Man-Kwan Shan. Looking for new, not known music only: music retrieval by melody style. In *JCDL '04: Proceedings of the 4th ACM/IEEE-CS joint conference on Digital libraries*, pages 243–251, New York, NY, USA, 2004. ACM Press.

[10] G. Lalya, M. Ramia, and H. Lars. Sonic city: The urban environment as a musical interface, 2003.

[11] Joseph F. McCarthy and Theodore D. Anagnost. MUSICFX: an arbiter of group preferences for computer supported collaborative workouts. In *Computer Supported Cooperative Work*, page 348, 2000.

[12] Muneyuri Unehara and Takehisa Onisawa. Composition of music using human evaluation. In *FUZZ-IEEE*, pages 1203–1206, 2001.

[13] Nigel Warren, Matt Jones, Steve Jones, and David Bainbridge. Navigation via continuously adapted music. In *CHI '05: CHI '05 extended abstracts on Human factors in computing systems*, pages 1849–1852, New York, NY, USA, 2005. ACM Press.