PublicDJ - Music selection in public spaces as multiplayer game

Stefan Leitich, Markus Toth
University of Vienna
Institute for Distributed and Multimedia Systems
Liebiggasse 4/3-4, 1010 Vienna, Austria
stefan.leitich@univie.ac.at, a0025690@univie.ac.at

Abstract. Music is an important tool to achieve a certain atmosphere for social gatherings in public spaces (e.g., Cafés, Pubs, Clubs) . The selection of proper music tracks is often either done by a staff member of such a location or a professional music selector – a disc-jockey (DJ).
The mobility of personal media libraries is increasing. A portable music player has become an everyday item and people are used to have their favorite music with them at any time. Selecting the preferred music for their own makes them at least as professional for themselves, as a "true" professional music selector performing in public. In a usual setting a single person determines what a whole audience is listening to, degrading the listeners to passive consumers. Creating the possibility to allow every person in the auditory getting involved in the music selection process, by using their portable music library, would result in a totally new kind of interactive listening experience in public spaces.
We present our concept for music selection in public spaces as a multiplayer game – PublicDJ – and its prototypical implementation. The concept is based upon a round based multiplayer game, where each player can submit music tracks to a server. The server analyses submitted tracks and selects the best matching track, based on a former announced criteria for playback. Selection criterias can range from high-level manually annotated audio metadata to low-level audio metadata extracted from audio using music information retrieval techniques. This allows the implementation of tasks like "Submit songs of the same genre!" or "Submit songs of the same artist!", which the users have to fulfill and can be used as a steering instrument for the music played.
Our prototype for collaborative music selection in public spaces as a multiplayer game increases interaction and involvement of listeners by providing the possibility of active participation in a previously completely passive experienced procedure.

1 Introduction

"Last night a DJ saved my life" is a famous chorus of a song released in the early 80's, describing the situation of a boring night out, until the disc-jockey (DJ) played a special tune. Thereby, she changed the mood of the people in the club and animated the audience. Maybe not the most profound example, but a good one describing the power of music in influencing the emotional state of human beings – as told by the song's lyrics, as well as by the song itself in clubs all over the world every saturday night. Furthermore, it displays the concentration of this power in the decision of a single person – the DJ – for the whole audience. The desire to be a DJ with this "magic energy-giving power" themselfes may exist in many persons. It's not everyone's favour to spend a lot of money to collect music, stand up in front of an audience, and select or mix music. Nevertheless, everyone who is into music is a music expert in her preferred music domains. Listeners often have a certain idea which song to listen to in which emotional state, through having heard her most liked music over and over again. People are their personal DJ's, as soon as they select the music they want to listen to intentionally, either in company or on their own.

Personal music libraries gain in mobility through new generations of multimedia enabled playback devices, creating the possibility for everyone to carry around her very own music library or at least excerpts of it. These devices (e.g., MP3[1] players, mobile phones, etc.) become more and more wireless network enabled and often provide a runtime environment for custom applications. Those technologies allow the execution of applications with low computational load and the exchange of a certain amount of data within reasonable time.

These are the basic conditions to create the possiblity for everyone to play an active role in the music selection process in a public space, while bluring the boundary between the domain expert (i.e., DJ) and the consumers (i.e., audience).

In the reminder of this paper related work is presented (cf. 2) and the concept of PublicDJ (cf. 3) is introduced with the game principle (cf. 3.1) and song selection criterias (cf. 3.3) implemented so far. A short

[1] MPEG-1 Audio Layer 3

description of the prototype follows (cf. 4, and the paper finishes with intended future work (cf. 5) and conclusions (cf. 6).

2 Related Work

Approaches in the domain of automatic playlist generation are manifold, but mainly aimed on satisfying a single listener's needs by automatically choosing the right sequence of songs. Those approaches share in common the description of songs by low- and/or high-level metadata, which is furthermore used by various algorithms (e.g., graph algorithms, recommender algorithms, etc.) to figure out the right sequence of songs, fitting certain criterias.

The pure audio data, necessary for playback, is analyzed to extract content based features (e.g., [7]). Another approach to describe the audio is manually annotated or automatically generated high-level metadata [2]. Automatically annotated metadata often concerns tracking listening habits [1], observing skipping behavior [11] or simple feedback loops [8]. These techniques are of scientific nature and already being used in real market products (e.g., `last.fm`[2]).

Publications concering music listening as social experience are, for example, the *MUSICtable* [12] introduced by Stavness et al. They use a landscape metaphor to visualize a manually arranged music collection on a table-mounted display, accessible from all sides. Buttons arranged around the display can be pressed by the users involved, to affect the "wind" on the virtual landscape and move the cursor position in a direction, thus controlling the next song to be played. A democratic approach for music choice in public spaces is proposed by O'Hara et al. with the *Jukola* system [10]. The system provides a voting mechanism through various input devices (PDA, Touch-Screen) to enable visitors of a public place (e.g., a pub) to select music democratically, while still allowing control of the music pool selected from. The paper observes the social impact of such a setting and underlines the interest of people taking part in the music selection process.

The following publications possess a definitive aim at a club scenario. These approaches tackle music selection by the audience, or even try to create and manipulate music, through monitoring feedback by multi-modal sensors [3, 4, 13, 6].

[2] `http://last.fm` - A social music platform, recommending music for individual users by tracking it's whole community listening behavior and relevance feedback to suggested songs.

3 Concept

PublicDJ 's concept is to allow every interested visitor of a public space, to bring her own music collection along on a network enabled device, and take part in the selection process of music that is going to be played at this place.

The DJ is responsible for selecting and mixing songs in order to create a certain kind of atmosphere in a public space. Making this decision process a complete democratic one without any kind of control may lead to inadequate abrupt changes in music style and therefore circumventing the creation of an atmosphere at all. The music style could also be developing into a direction undesired for a certain place or time [5]. These requirements have to be considered for the successful design of a system, that strives for the goal of involving diverse people, while simultaneausly creating a certain kind of atmosphere.

3.1 Game Principle

The game principle of PublicDJ is a round based multiplayer game. With the start of each new round users are requested to submit songs to a server that they want to hear, and in their opinion fit a given criteria. The length of a round is determined by the current song playing. After analyzing the received submissions the server determines the best matching song, with respect to the given criteria and the actual song played. The best matching song will be started as soon as a new round is initiated.

Basically, the criterias that have to be fulfilled by submissons are extendable in any way a feature extraction technology exists. The boundaries for low-level features are the computational load for extraction, which has to be performed within a round's duration (i.e., time left of the actual song played, after transmission to the server). For high-level features the requirement is their presence, which is often not given.

3.2 Game Round

As depicted in Fig. 1 a round can be divided into phases. Songs are submitted and analyzed afterwards. The extracted feature attributes are normalized, distances are calculated, and the respective best matching song is selected. Those phases are not visible to users. They can start submitting their selection anytime during the round. If the submission can not be finished during the round, it is canceled and the user is suggested to re-submit her selected song in the next round. Same is valid for the analysis. If a song can not be analyzed totally till the end of the round, the analysis is canceled and the submission is not considered in the selection process. This explains the demand for

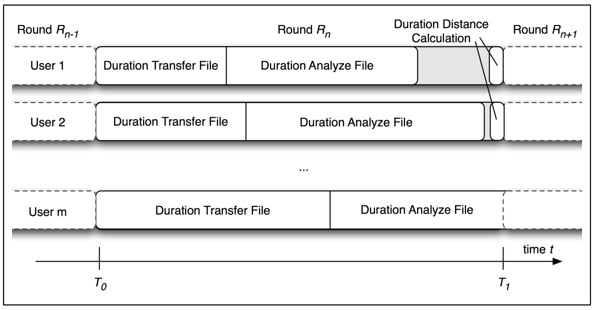

Figure 1: Game round R_n with submitted songs by user 1 to m runnning from T_0 till T_1. Submission m would be ignored, as the analysis is not finished before round end.

estimated time amount calculations on the server for each and every transmission and analysis.

3.3 Song selection criterias

The song selection criterias we tackled for this proof-of-concept, are (1) a simple enqueue mechanism, (2) low-level-, and (3) high-level metadata criterias. In case of the simple enqueue mode (1) all songs submitted by clients are enqueued at the server for playback without any processing. The extracted low-level metadata (2) allows an estimation of perceived similarity between songs. Similiarity in context of the features used can be defined as timbre and rhythm related. High-level metadata (3) allows the comparison of songs regarding, for example, artist, year of creation, or even artwork, if the corresponding feature extraction is able to process images.

Low-level metadata For low-level metadata based selection criterias we used features described in [7]. Those features are established and well performing descriptors for song similarity, as evaluated in the Music Information Retrieval (MIR) community in recent years.

Rhythm Patterns are a time invariant descriptor, which contains the amplitude of modulation for the 24 critical frequency bands, with respect to modulation frequencies. Basically it is extracted by transforming the audio signal with a Fast Fourier Transformation (FFT) into the frequency domain and process the result in psychoacoustically motivated stages (e.g., frequency grouping, loudness leveling, loudness sensation per band). Finally applying a second FFT stage results in the time invariant modulation amplitude information.

The *Statistical Spectrum Descriptor* is a feature vector consisting of seven statistical moments (mean, median, variance, skewness, kurtosis, min- and max-value) for the 24 critical bands. The descriptor captures the frequency characteristics in terms of statistical information about the audio signal, after transforming it into the frequency domain by a FFT and applying frequency grouping into the Bark scale, resulting in a feature vector of 168 elements.

The *Rhythm Histogram* features gather information about the distribution of magnitudes of modulation frequencies in 60 bins summed up for all 24 critical bands. Modulation frequencies are captured in a range from 0 to 10 Hz, segments processed are of 6 sec. duration. This algorithm produces a feature vector with 60 elements, describing "rhythmic energy" per modulation frequency bin.

High-level metadata As we assumed MP3 as the most common audio format for personal music libraries, a metadata extractor for ID3 tags [9] was implemented. ID3 tags are embedded in the MP3 file format, and, if present, can contain annotated song metadata, for example artist name, track title, year of creation, or artwork in terms of an image. The main issue with high-level metadata is, that there is no guarantee in a real world setting for the presence of the information required for song comparison. In case of missing information the metadata extractor has to consider the submitted song as dissimilar.

4 Prototype

The implementation of PublicDJ was done in Java, because it is a platform supported not only by personal computers, but by mobile devices (e.g., mobile phones, PDAs) as well. Communication between the components of PublicDJ is realized using Java's Remote Method Invocation (RMI) interface. The client implementation conforms to IBMs J9 Java Virtual Machine (JVM) to run on a variety of PDAs.

4.1 Architecture

The main component of the system is a central *server*, responsible for controlling the game logic, audio playback, estimating remaining time amounts for transfers and analyses, storing received analysis values, rudimentary display of game stats, and communication between the different parts of the systems. Login and connection status of clients are monitored by the server. Incoming song submissions are received and delegated, in respect to load balancing, to the *analysis server*'s connected. Externalising the feature extraction in arbitrarily instantiations of *analysis servers* ensures scalability. Those extract, in dependency of selected criterias, the regarding metadata. The *administration* and *client* applications are basically the same and allow login and song submission to the system. The *administration* application furthermore allows to adjust server settings remotely regarding the condition, that song submissions have to try to fulfill to be selected as the best matching one for the succeeding round.

5 Future Work

We understand this first prototypical implementation of PublicDJ as a proof-of-concept. The most important aspects we want to implement in the next development steps are the following tasks. (1) Adapting the client interface to an applet or web application, to remove the barrier of requiring the setup of a software component on the client devices. (2) Extend the server with automatic playlist generation capabilities, thereby turning it into an competitor comparable with human users. Depending on this playlist generation algorithm, the server can also gain the possibility to affect the development of the sequence songs are played over time, and provide a certain amount of control to avoid possibly undesired music styles. (3) Finally, we would like to perform a field test of PublicDJ and observe users interacting and using the system, to conclude further improvements and ideas.

6 Conclusion

In this paper we introduced our collaborative music selecting approach – PublicDJ – for public spaces. It allows to submit songs from network enabled audio playback devices for automated relevance evaluated music selection. By making use of various ways of audio metadata extraction, best matching songs for given criterias are determined and selected for playback. The application depicts an interactive system, turning passive listeners into active music selecting protagonists in public spaces.

References

[1] Andreja Andric and Goffredo Haus. Automatic playlist generation based on tracking user's listening habits. *Multimedia Tools Appl.*, 29(2):127–151, 2006.

[2] J.J. Aucouturier and F. Pachet. Scaling up music playlist generation. In *Proceedings of IEEE International Conference on Multimedia and Expo (ICME).* Lausanne, Switzerland, August 2002.

[3] Dave Cliff. hpdj: An automated dj with floorshow feedback. Technical Report HPL-2005-88, Digital Media Systems Laboratories, HP Laboratories, Bristol, UK, May 2005.

[4] Mark Feldmeier and Joseph A. Paradiso. An interactive music environment for large groups with giveaway wireless motion sensors. *Computer Music Journal*, 31(1):50–67, April 2007.

[5] Carrie Gates, Sriram Subramanian, and Carl Gutwin. Djs' perspectives on interaction and awareness in nightclubs. In *DIS '06: Proceedings of the 6th ACM conference on Designing Interactive systems*, pages 70–79, New York, NY, USA, 2006. ACM Press.

[6] Dennis Hromin, Michael Chladil, Natalie Vanatta, Susanne Wetzel, Farooq Anjum, and Ravi Jain. Codeblue: a bluetooth interactive dance club system. In *Proceedings of the IEEE Global Telecommunications Conference GLOBECOM*, pages 2814–2818, December 2003.

[7] Thomas Lidy and Andreas Rauber. Evaluation of feature extractors and psycho-acoustic transformations for music genre classification. In *ISMIR 2005, 6th International Conference on Music Information*, pages 34–41, 2005.

[8] Beth Logan. Content-based playlist generation: Exploratory experiments. In *ISMIR 2002, 3rd International Conference on Music Information*, 2002.

[9] M. Nilsson. Id3 tag version 2.3.0. http://id3.org/d3v2.3.0, February 1999.

[10] Kenton O'Hara, Matthew Lipson, Marcel Jansen, Axel Unger, Huw Jeffries, and Peter Macer. Jukola: democratic music choice in a public space. In *DIS '04: Proceedings of the 2004 conference on Designing interactive systems*, pages 145–154, New York, NY, USA, 2004. ACM Press.

[11] Elias Pampalk, Tim Pohle, and Gerhard Widmer. Dynamic playlist generation based on skipping behavior. In *ISMIR 2005, 6th International Conference on Music Information*, pages 634–637, 2005.

[12] Ian Stavness, Jennifer Gluck, Leah Vilhan, and Sidney Fels. The musictable: A map-based ubiquitous system for social interaction with a digital music collection. In *Entertainment Computing - ICEC 2005, 4th International Conference, Sanda, Japan, September 19-21, 2005, Proceedings*, volume 3711 of *Lecture Notes in Computer Science*, pages 291–302. Springer, 2005.

[13] Ryan Ulyate and David Bianciardi. The interactive dance club: Avoiding chaos in a multiparticipant environment. *Computer Music Journal*, 26(3):40–49, March 2002.