PlaySOM and *PocketSOMPlayer*, Alternative Interfaces to Large Music Collections

Michael Dittenbach
eCommerce Competence Center
iSpaces Group
Donau-City Strasse 1
A-1220 Wien, Austria
michael.dittenbach@ec3.at

Robert Neumayer
Vienna University of Technology
Department of Software Technology
and Interactive Systems
Favoritenstr. 9-11 / 188
A - 1040 Vienna
neumayer@ifs.tuwien.ac.at

Andreas Rauber
Vienna University of Technology
Department of Software Technology
and Interactive Systems
Favoritenstr. 9-11 / 188
A - 1040 Wien, Austria
andi@ifs.tuwien.ac.at

ABSTRACT

With the rising popularity of digital music archives the need for new access methods such as interactive exploration or similarity-based search become significant. In this paper we present the *PlaySOM*, as well as the *PocketSOMPlayer*, two novel interfaces allowing to browse a music collection by navigating a map of clustered music tracks and to select regions of interest containing similar tracks for playing. The *PlaySOM* system is primarily designed to allow interaction via a large-screen device, whereas the *PocketSOMPlayer* is implemented for mobile devices, supporting both local as well as streamed audio replay. This approach offers content-based organization of music as an alternative to conventional navigation of audio archives, i.e. flat or hierarchical listings of music tracks that are sorted and filtered by meta information.

Keywords: User Interaction, Music Collections, Information Discovery and Retrieval, Audio Clustering, Audio Interfaces, Mobile Devices.

1 Introduction

The increasing popularity and size of digital music repositories drives the need for advanced methods to organize those archives for both private as well as commercial use.

The ability to offer users information about songs or artists that are similar to the ones they were actually searching for, holds a great market potential as recommendation engines have proven. The linking of similar products to results of customer searches bears fruits, even when it is not based on the similarity of products themselves, but on the buying behavior of other customers, like Amazon.com has impressively shown. Therefore commercial music vendors could particularly profit from organization of music archives based on sound-similarity.

It is an intrinsic need for them to offer high-level user interfaces to their repositories to satisfy their customers' needs. Search methods based on track similarity, such as query-by-example, offer alternatives to keyword based searches that avoid the downside of having to rely on manually assigned metadata. A system that offers searches that rely on metadata only, can never meet these more sophisticated needs (although this functionality is, of course, a basic prerequisite for any online music store).

The motivations for organizing private music collections are most likely fun and entertainment by overcoming the limitations of conventional media players. Similarity-based organization of music archives allows users to explore pieces of music that are similar to ones they know and like. Moreover, it provides a clear and easy navigation for music collections the users are familiar with and allows users to abstract from manually assigned genre information which is, at least in private collections, often inappropriate or simply missing.

Overcoming traditional genre boundaries can improve search results, e.g. concerning tracks from samplers or movie soundtracks which do not have any (reliable) genre assigned at all. Further, single songs that are very different from the rest of an album could distort the result for a query if relying on genre information, because usually all songs of an album are assigned to the same genre. This could lead to problems for albums containing remixes or rather inhomogenous songs. Concerning the access to rapidly growing and changing collections, the similarity-based organization is much more satisfying than conventional search methods because users do not have to know new songs by name but get them in results of their usual queries. This problem gets more important with the growing size of a collection. The browsing of a few hundred songs a user knows well might not be much of a problem using metadata, but navigating through thousands of songs one is not familiar with may lead to restrictions, preventing the user from gaining access to the majority of songs.

This paper describes two novel interfaces for accessing music collections. The music tracks are organized spatially on a two-dimensional map display based on the similarity of the extracted sound features. We will show how this map metaphor is used to provide convenient access to music repositories and how such an organization of songs can be used for playlist generation and interactive exploration for both desktop applications and mobile devices.

Our work focuses on private collections and types of interaction itself, commercial aspects are not considered in this context.

The remainder of this paper is structured as follows. Section 2 briefly reviews the related work followed by an introduction to the fundamentals of the *Self-Organizing Map*, a neural network-based clustering algorithm and the *Rhythm Patterns* feature extraction model that were used for our experiments in Section 3. We then describe the experimental results of clustering the collection of the ISMIR04 genre contest and describe the *PlaySOM* and *PocketSOMPlayer* user interfaces in detail in Section 4. Finally, Section 5 provides some conclusions.

2 Related Work

Scientific research has particularly been conducted in the area of content-based music retrieval (cf. (Downie, 2003; Foote, 1999)). Recently, content analysis for similarity-based organization and detection has gained significant interest. Numerous feature sets have been proposed and evaluated, all of which can basically be used for the system presented in this paper as a basis for organizing audio by sound similarity. Loudness, pitch, brightness, bandwidth, and harmonicity features are used in (Wold et al., 1996) to train classifiers. The *MARSYAS* system (Tzanetakis and Cook, 2000, 2002) uses a wide range of musical surface features to organize music into different genre categories using a selection of classification algorithms. The set of features that are used for clustering a music collection in this paper are *Rhythm Patterns* used in the *SOMeJB* system (Rauber et al., 2002).

Regarding intelligent playlist generation, an exploratory study using an audio similarity measure to create a trajectory through a graph of music tracks is reported in (Logan, 2002). Furthermore, many applications can be found on the Internet that are not described in scientific literature. An implementation of a map-like playlist interface is the *Synapse Media Player*[1]. This player tracks the user's listening behavior and generates appropriate playlists based on previous listening sessions and additionally offers a map interface for manually arranging and linking pieces of music for an even more sophisticated playlist generation. Another example of players offering automatic playlist generation is the *Intelligent Multimedia Management System*[2] which is based on tracking of the user's listening habits and recommends personalized playlists based on listening behavior as well as acoustic properties like BPM or a song's frequency spectrum. *Audioscrobbler*[3] is a rather classic recommendation engine, identifying classes of users based on registered individuals, who share the information about their listening habits, and recommending playlists according to users with similar listening behaviors. A novel interface particuarly developed for small-screen devices, was presented in (Vignoli et al., 2004). This artist map interface clusters pieces of audio based on content features as well as metadata attributes using a a spring model algorithm. The need for advanced visualization to support selection of audio tracks in ever larger audio collectionis also addressed in (Marc Torrens, 2004), where different representation techniques grouping audio by metadata attributes using Tree-Maps and a disc visualization is presented.

[1] www.synapseai.com

[2] www.luminal.org

[3] www.audioscrobbler.com

3 Self Organizing Maps for Clustering Audio Collections

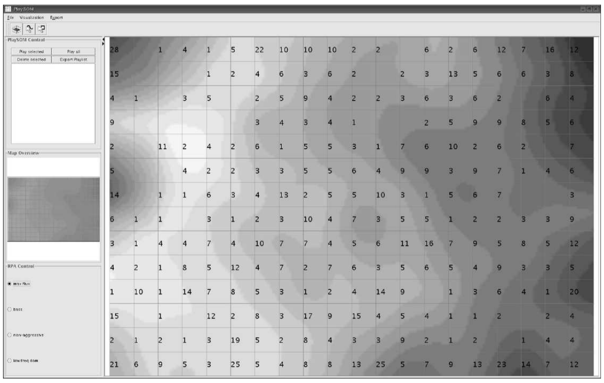
3.1 Self-Organizing Map

In order to organize audio by sound similarity a range of clustering algorithms can be employed. One particularly suitable model is the *Self-Organizing Map (SOM)*, an unsupervised neural network that provides a mapping from a high-dimensional input space to usually two-dimensional output space (Kohonen, 1982, 2001). During that process topological relations are preserved as faithfully as possible. A *SOM* consists of a set of i units arranged in a two-dimensional grid, each attached to a weight vector $m_i \in \Re^n$. Elements from the high-dimensional input space, referred to as input vectors $x \in \Re^n$, are presented to the *SOM* and the activation of each unit for the presented input vector is calculated using an activation function. The Euclidean distance between the weight vector of the unit and the input vector is very commonly used as the activation function. In the next step the weight vector of the unit showing the highest activation (i.e. the smallest Euclidean distance) is selected as the 'winner' and is modified as to more closely resemble the presented input vector. Pragmatically speaking, the weight vector of the winner is moved towards the presented input signal by a certain fraction of the Euclidean distance as indicated by a time-decreasing learning rate α. Consequently, the next time the same input signal is presented, this unit's activation will be even higher. Furthermore, the weight vectors of units neighboring the winner, as described by a time-decreasing neighborhood function, are modified accordingly, yet to a smaller amount as compared to the winner. The result of this learning procedure is a topologically ordered mapping of the presented input signals in two-dimensional space. Accordingly, similar input data are mapped onto neighboring regions of the map. A *SOM* can be trained using all kinds of feature sets. For our experiments we will use the *Rhythm Patterns* features as input data.

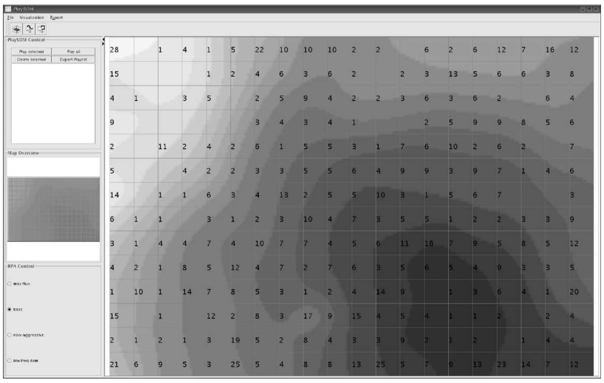
3.2 Audio Feature Extraction Using Rhythm Patterns

The feature extraction process consits of two main stages, incorporating several psycho-acoustic transformation (Zwicker and Fastl, 1999). First the specific loudness sensation in different frequency bands is computed. This is then transformed into a time-invariant representation based on the modulation frequency.

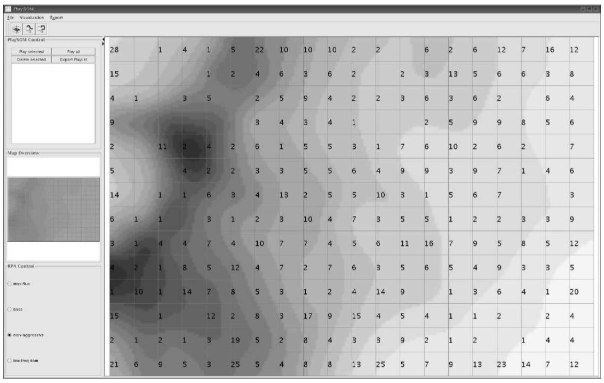
The audio data is decomposed into frequency bands, which are then grouped according to the Bark critical-band scale. Then, we calculate loudness levels, referred to as phon using the equal-loudness contour matrix, which is subsequently transformed into the specific loudness sensation per critical band, referred to as sone.



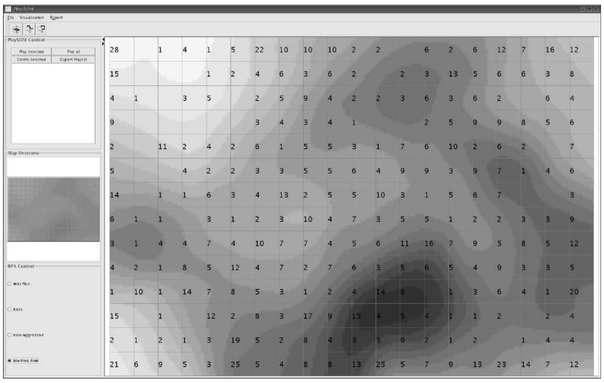
(a) Maximum fluctuation strength.



(b) Bass.



(c) Non-aggressiveness.



(d) Low frequencies dominant.

Figure 1: *PlaySOM* interface with different visualizations of Rhythm Patterns.

To obtain a time-invariant representation, recurring patterns in the individual critical bands are extracted in the second stage of the feature extraction process. These are weighted according to the fluctuation strength model, followed by the application of a final gradient filter and gaussian smoothing. The resulting 1.440 dimensional feature vectors capture rhythmic information up to 10Hz (600bpm). These *Rhythm Patterns* are further used for the comparison and similarity measurement of data signals. More detailed descriptions of this approach can be found in (Rauber et al., 2003)

3.3 Visualization Techniques of the SOM

Due to the fact that the cluster structure of a trained *SOM* is not inherently visible, several visualization techniques have been developed, the most prominent being the *U-Matrix* (Ultsch and Siemon, 1990). Here, the distances between the weight vectors of adjacent units are mapped onto a color palette with the result of homogeneous clusters, i.e. the weight vectors of neighboring units have rather small distances, being colored differently from cluster boundaries with larger distances between the respective units' weight vectors.

Another useful method that provides insight into the structure of a trained *SOM* is the visualization of component planes, i.e. individual features. Only a single component of the weight vectors is used to color-code the map representation. This information can also be overlayed onto other visualizations using Gradient Fields (Pölzlbauer et al., 2005). In other words, the values of a specific component of the weight vectors are mapped onto a color palette to paint units accordingly allowing to identify regions that are dominated by a specific feature.

Since single component planes do not directly translate into psychoacoustic sensation noticed by the human ear, the *Rhythm Patterns* uses four combinations of component planes according to psychoacoustic characteristics (Pampalk et al., 2002). More precisely, *maximum fluctuation strength* evaluates to the maximum value of all vector components representing music dominated by strong beats. *bass* denotes the aggregation of the values in the lowest two critical bands with a modulation frequency higher than 1Hz indicating music with bass beats faster than 60 beats per minute. *Non-aggressiveness* takes into account values with a modulation frequency lower than 0.5Hz of all critical bands except the lowest two. Hence, this feature indicates rather calm songs with slow rhythms. Finally, the ratio of the five lowest and highest critical bands measures in how far *low frequencies dominate*. These characteristics can be used to color the resulting map, providing weather-chart kind of visualizations of the music located in different parts of the map. Figure 1 shows examples for all four kinds of visualizations.

4 PlaySOM and PocketSOMPlayer

We present two interfaces to digital music collections that are based on the *Self-Organizing Map* clustering algo-

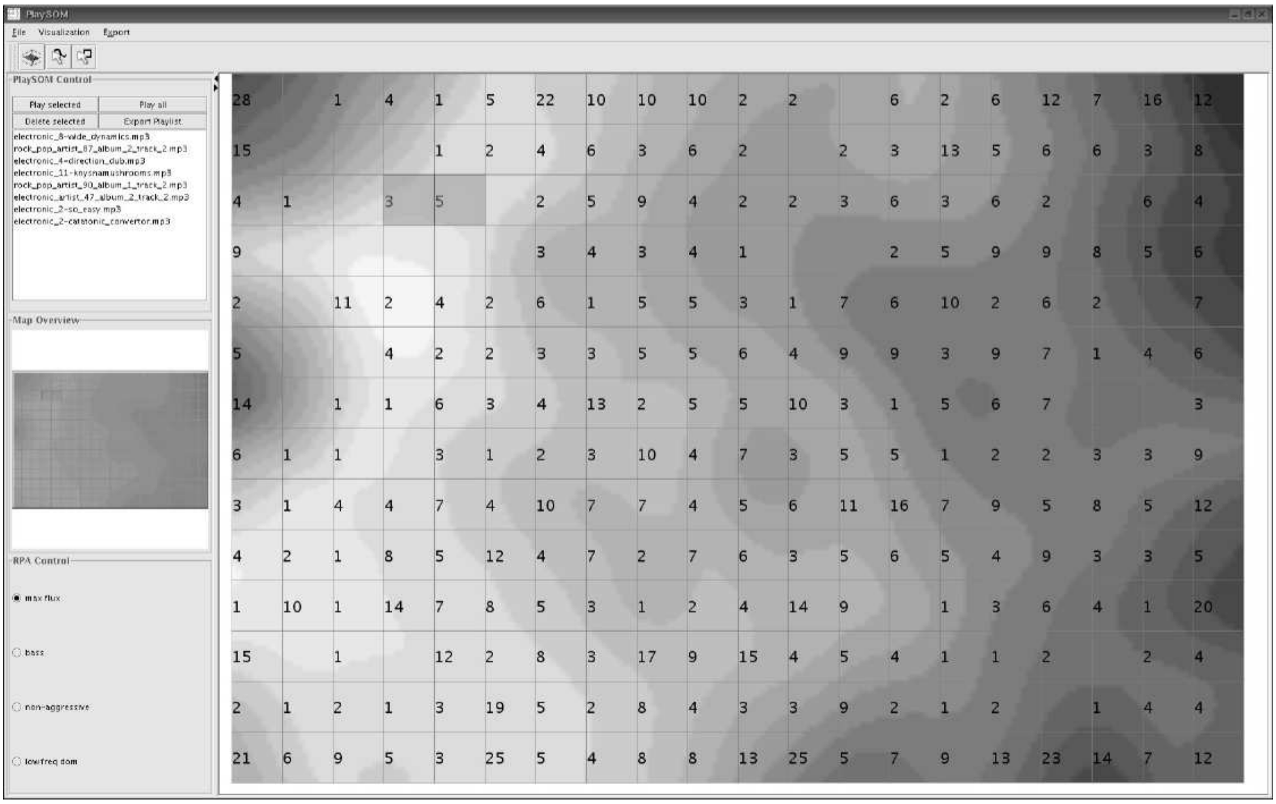


Figure 2: The main *PlaySOM* interface.

rithm and allow interactive exploration of music collections according to feature similarity of audio tracks. The *PlaySOM* and *PocketSOMPlayer* applications both enable users to browse collections, select tracks, export playlists as well as listen to the selected songs. The *PlaySOM* presents a full interface, offering different selection models, a range of visualizations, advanced playlist refinement, export to external player devices or simply playback of selected songs. The *PocketSOMPlayer*, on the other hand, offers a slim version of the desktop application, optimized for the *PocketPC* platform, implemented for an iPaq using Java and SWT to be used in a streaming environment.

The upcoming sections show the experimental results of the clustering of the music collection used for the 2004 ISMIR genre contest. Furthermore we will present the *PlaySOM* and *PocketSOMPlayer* interfaces in detail and demonstrate their feasibility for interactive exploration of that collection.

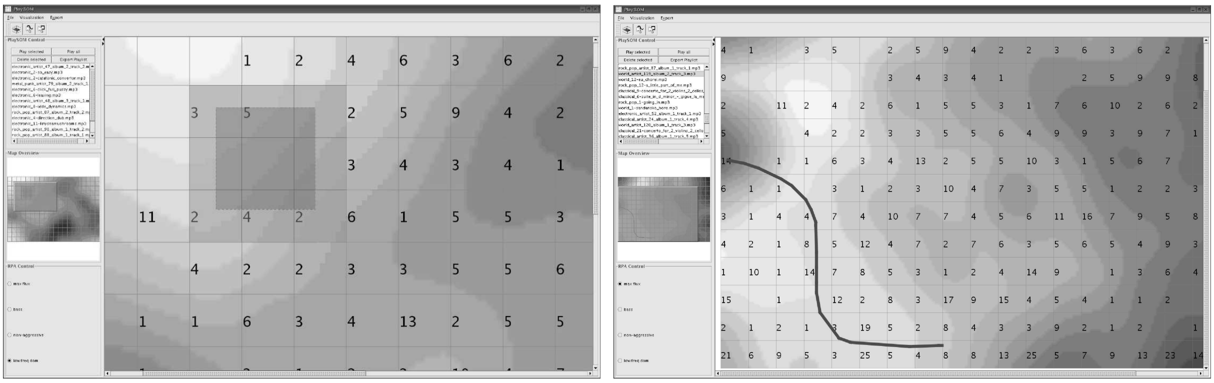
4.1 Data Collection and Trained SOM

The audio collection used in the ISMIR 2004 genre contest consists of 1458 titles, organized into 6 genres, the major part of which is *Classical* music (640), followed by *World* (244), *Rock_Pop* (203), *Electronic* (229), *Metal_Punk* (90) and *Jazz_Blues* (52). Yet, these genre labels are only used as an indicator during evaluation, as the kind or ofganization provided by the SOM is intended also to overcome the restrictions of manually assigned genre information. The *Rhythm Patterns* of that collection of songs were extracted and its songs were mapped onto a *Self-Organizing Map* consisting of 20×14 units.

The assessment of clustering quality is generally difficult due to the highly subjective nature of the data and the broad spectrum of individual similarity perception. We still try to provide an overview of the map-based organization of this collection and pick some sample areas of the map to demonstrate the results based on the interfaces.

Figures 1(a)-(d) show the complete map visualizing the four different *Rhythm Patterns* sub-groups described in the previous section. The visualizations provide an important clue to the overall organization of the map and offer starting points for interactive exploration depending on the characteristics of music one is interested in. A linear gray scale comprising 16 colors from dark gray to white representing feature values from low to high is used for printing purposes. (For on-screen use, we emphasize the map metaphor by using a fine-grained color palette ranging from blue via yellow to green reflecting geographical properties similar to the *Islands of Music* (Pampalk, 2003)).

The organization of the songs according to the *maximum fluctuation strength* feature is clearly visible in Figure 1(a) where pieces of music having high values are located primarily on the left-hand side of the map. Especially *Metal_Punk* and *Rock_Pop* as well as some of the *Electronic* songs that are less bass-dominated can be found there. Contrarily, songs with low values are located on the map's right-hand side. Some examples of rather tranquil music are tracks belonging to the genres *Classic* or *World* as well as single *Pop_Rock* songs.



(a) Rectangle selection without preserving track order.

(b) Trajectory selection preserving track order.

Figure 3: The *PlaySOM* interface, its selection models and playlist contents.

Figure 1(b) shows that the feature *bass* is concentrated on the upper left corner and basically consists of bass-dominated tracks belonging to *Electronic* genre. This cluster is the most homogenous on the map (along with a cluster of classical music) according to genre tags, almost no other genres are found in this area.

In Figure 1(c), the majority of clusters containing *non-aggressive* music can be identified on the right-hand side of the map as one would expect regarding the distribution of the *maximum fluctuation strength*, which represents music dominated by strong and fast beats. The most prominent members of that clusters are the afore mentioned *World* and *Classical* tracks. Most of the *Jazz_Blues* songs are located at the lower middle of the map, representing the transition from *non-agressive* music and high values for *maximum fluctuation strength* on the other hand. Finally, a small cluster where *low frequencies dominate* is located in the upper left of the map as shown in Figure 1(d) and corresponds to the results of *bass* setting, leading to low values in this region.

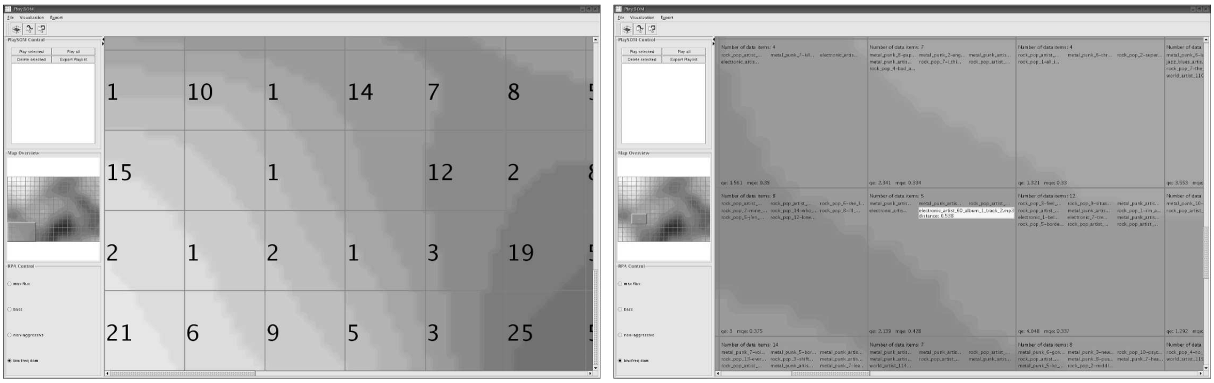
The different types of classical music are a good example of similarity-based clustering that overcomes genre boundaries. Whereas many songs from operas are located on the lower left-hand side of the map, many other tracks that also belong to the *Classical* genre, but sound very different from operas are located on the upper right. This mapping is based on the fact that many songs from the *World* genre are very much alike slow pieces of classical music, but rather different from operas, a possibility which is not captured by static genre assignments at all.

4.2 PlaySOM

The *PlaySOM* allows users to interact with the map mainly by panning, semantic zooming and selecting of tracks. Users can move across the map, zoom into areas of interest and select songs they want to listen to. They can thereby browse their private collections of a few thousand songs, generating playlists based on track similarity instead of clicking through metadata hierarchies, and either listening to those selected playlists or exporting them for later use. Users can abstract from albums or genres which often leads to rather monotonous playlists often consisting of complete albums or many songs from one genre. This approach enables users to export playlists based on track not on metadata similarity or manual organization.

The main *PlaySOM* user interface is shown in Figure 2. Its largest part is covered by the interactive map on the right, where squares represent single units of the SOM. Controls for selecting different visualizations and exporting the map data and the current visualization for the *PocketSOMPlayer* are part of the menubar on the top. The left hand side of the user interface contains (1) a playlist of currently selected titles, (2) a birds-eye-view showing which part of the potentially very large map is currently depicted in the main view on the right and (3) controls for the currently selected visualization (as demonstrated by the different settings of the *Rhythm Patterns* in Figure 1).

The icons on the upper left allow the user to switch between the two different selection models and to automatically fit the map to the current screen size. Figure 3 depicts the interaction models that are currently supported by the *PlaySOM*. The rectangular selection model allows the user to drag a rectangle and select the songs belonging to units inside that rectangle without preserving any order of the selected tracks. This model is used to select music from one particular cluster or region on the map. Figure 3(a) depicts the selection of a cluster of songs located at the upper left part of the map mainly belonging to the *Electronic* genre, comprising single tracks from *Rock_Pop* and *Metal_Punk* without any specific order. On the other hand, the line selection model allows users to draw trajectories and select all songs belonging to units beneath that trajectory. Figure 3(b) shows a selection of tracks and the according transitions between those genres along the trajectory. The dark region located at the beginning of the trajectory at the left middle of the figure mainly consists of *Electronic* tracks and represents high values in the *maximum fluctuation strength* set of features. Further along the trajectory, the playlist continues with a few more lively and dynamic songs belonging to the *Rock_Pop* and *Metal_Punk* genres, represented by the lighter region, before it turns back to rather tranquil music from the *Classical* genre. In this case the sequence of selected units is of particular importance, because this line chooses a variety of songs according to their position on the map, i.e. their similarity. Hence the line selection model makes it possi-



(a) Low level of detail - the number of songs mapped are written on the respective units.

(b) High zooming level - song names are displayed on the respective units.

Figure 4: Semantic zooming and its impact on the displayed data.

ble to generate playlists including smooth transitions between clusters of tracks. This might be of specific interest when browsing very large music collections or when rather long playlists shall be generated (for example if a playlist for several hours should be generated and several changes in musical style shall occur over time, similar to an *auto-dj* functionality).

Once a user has selected songs and refined the results by manually dropping single songs from the selection, those playlists can be listened to on-the-fly or exported for later use on the desktop machine or even other platforms like PDAs or Multimedia Jukeboxes if the collection is served via a streaming environment.

Another vital aspect of the interface is that it supports semantic zooming, i.e. the zooming level influences the amount and type of data displayed. As outlined in Figure 4, the higher the zooming level, the more information is displayed ranging from information about the number of songs mapped to a particular unit (Figure 4(a)) to detailed information about the tracks (Figure 4(b)), i.e. artist- and trackname. Furthermore, the main *PlaySOM* application can easily and efficiently be used on a Tablet PC and used as a touch screen application because of its portable Java implementation (a live demo is shown in 6(b)).

4.3 PocketSOMPlayer

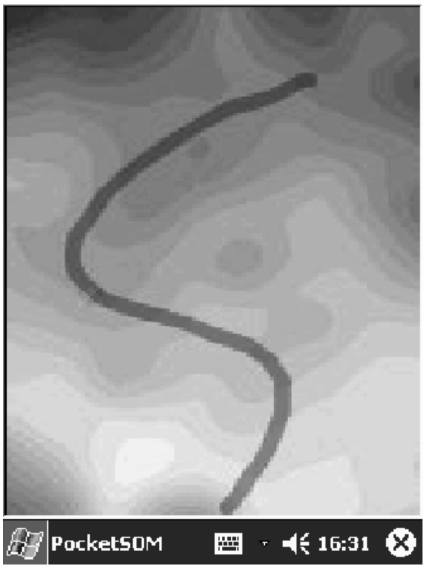
The *PocketSOMPlayer* application offers similar but simplified functionality as the *PlaySOM* being designed for mobile devices such as PDAs or Smartphones. Therefore it only provides the basic functionality of selecting by drawing trajectories and a simplified refinement section, omitting means to zoom or pan the map. Its operational area is likely to be a client in a (wireless) audio streaming environment for entertainment purposes. Regarding the current memory restrictions of PDAs, the use of a streaming server as music repository seems even more appealing than for the desktop application. Nevertheless, the mobile interface could be synchronized with its desktop pendant to take the role of a mobile audio player within the PDA's memory limits.

Figure 5(a) shows the *PocketSOMPlayer*'s main interface, a trajectory selection with an underlying map. Its user refinement view which allows the user to modify the previously selected playlist before listening to the result is depicted in Figure 5(b). (Due to the anonymized format of the ISMIR collection we emphasized on genres instead on individual track names. In real application scenarios, filenames or ID3-tag information would be used for displaying information on the map.) The main panel allows the user to draw trajectories and to select the units underneath those trajectories. All songs mapped to the selected units are added to the playlist. The user refinement panel pops up as soon as a selection is finished and provides similar functionality as the *PlaySOM*'s playlist controls, namely the user can delete single songs from the playlist to refine her/his selection. The resulting playlist can then be played, retrieving the MP3s either from the local storage or a streaming server.

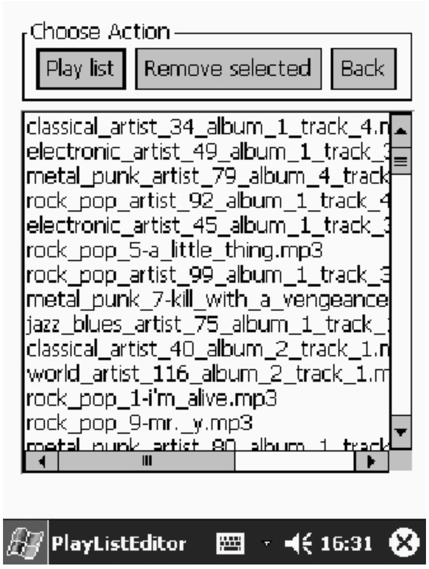
Figure 6(a) shows the *PocketSOMPlayer* running on an iPaq PDA without a trajectory selection. The map describes a music repository located on a streaming server running on another machine, accessible via WLAN, in contrast to keeping the music files locally (note that labels are manually assigned to clusters according to the most prominent genres in this example). Selecting tracks via drawing of trajectories on a touch schreen is straightforward, easy to learn and intuitive as opposed to clicking through genre hierarchies and therefore particularly interesting for mobile devices and their handling restrictions.

5 Conclusions and Future Work

We presented the *PlaySOM*, a novel user interface to map representations of music collections created by training a *Self-Organizing Map*, i.e. a neural network with unsupervised learning function using automatically extracted feature values for clustering of audio files. The interface allows user interaction and interactive exploration based on those maps, which was described in detail in our experiments. The *PlaySOM* offers a two-dimensional map with spatial organization of similar tracks and is especially appealing for large or unknown collections, which could hardly be browsed by metadata search only. The application allows users to browse their collections by simi-



(a) The *PocketSOMPlayer*'s main panel showing a trajectory selection.



(b) *PocketSOMPlayer* user refinement panel.

Figure 5: The *PocketSOMPlayer* interface showing different interaction views.

(a) The *PocketSOMPlayer* application running on an iPaq PDA.

(b) *PocketSOMPlayer* running on a Tablet PC.

Figure 6: Both presented interfaces running on an iPaq and Tablet PC respectively.

larity and therefore find songs similar to ones they know by name in contrast to metadata-based approaches. Moreover, we introduced a PDA application offering similar functionality, showing that alternative approaches to music organization are feasible for mobile devices as well. Both user interfaces are well suited for interactive exploration of collections of digital music because of their different levels of interaction like semantic zooming or on-the-fly playlist generation.

Future work will mainly deal with further development of interfaces for mobile devices, especially concentrating on their use in streaming environments. Therefore the combination of such clients with centralized music repositories, offering tighter integration of feature extraction and online exchange of stored information about tracks such as the well known ID3 tags, is going to be evaluated. Moreover, the desktop interface may be extended by more sophisticated methods for playlist generation such as automatic smooth transitions between clusters.

In addition, user studies might be of great help to measure the quality of the *Self-Organizing Map* clustering in combination with the *Rhythm Patterns* feature extraction as the automated quality assessment is very difficult as mentioned before. The current *PlaySOM* implementation is well suited for such studies and on-the-fly evaluation of specific areas on the maps as described in our experiments.

ACKNOWLEDGEMENTS

Part of this work was supported by the European Union in the 6. Framework Program, IST, through the DELOS NoE on Digital Libraries, contract 507618, and the MUSCLE NoE on Multimedia Understanding through Semantics, Computation and Learning, contract 507752.

References

J. Stephen Downie. *Annual Review of Information Science and Technology*, volume 37, chapter Music Information Retrieval, pages 295–340. Information Today, Medford, NJ, 2003.

Jonathan Foote. An overview of audio information retrieval. *Multimedia Systems*, 7(1):2–10, 1999. ISSN 0942-4962. doi: http://dx.doi.org/10.1007/s005300050106.

Teuvo Kohonen. *Self-Organizing Maps*, volume 30 of *Springer Series in Information Sciences*. Springer, Berlin, 3rd edition, 2001.

Teuvo Kohonen. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43: 59–69, 1982.

Beth Logan. Content-based playlist generation: Exploratory experiments. In *Proceedings of the 3rd Annual Symposium on Music Information Retrieval (ISMIR'02)*, pages 295–296, Paris, France, October 13-17 2002.

Josep Lluis Arcos Marc Torrens, Patrick Hertzog. Visualizing and exploring personal music libraries. In *Proceedings of the 5th International Conference on Music Information Retrieval (ISMIR'04)*, pages 421–424, Barcelona, Spain, October 10-14 2004.

E. Pampalk. Islands of Music - Analysis, organization, and visualization of music archives. *Journal of the Austrian Society for Artificial Intelligence*, 22(4):20–23, 2003.

Elias Pampalk, Andreas Rauber, and Dieter Merkl. Content-based Organization and Visualization of Music Archives. In *Proceedings of the ACM Multimedia (MM'02)*, pages 570–579, Juan les Pins, France, December 1-6 2002. ACM.

Georg Pölzlbauer, Michael Dittenbach, and Andreas Rauber. A visualization technique for self-organizing maps with vector fields to obtain the cluster structure at desired levels of detail. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN'05)*, pages 1558–1563, Montreal, Canada, July 31 - August 5 2005. IEEE Computer Society. ISBN 0-7803-9049-0.

Andreas Rauber, Elias Pampalk, and Dieter Merkl. Using psycho-acoustic models and self-organizing maps to create a hierarchical structuring of music by musical styles. In *Proceedings of the 3rd International Symposium on Music Information Retrieval (ISMIR'05)*, pages 71–80, Paris, France, October 13-17 2002. ISBN 2-84426-166-3.

Andreas Rauber, Elias Pampalk, and Dieter Merkl. The SOM-enhanced JukeBox: Organization and visualization of music collections based on perceptual models. *Journal of New Music Research*, 32(2):193–210, 2003.

George Tzanetakis and Perry Cook. Marsyas: A framework for audio analysis. *Organized Sound*, 4(30):169–175, 2000.

George Tzanetakis and Perry Cook. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302, July 2002.

Alfred Ultsch and Hans Peter Siemon. Kohonen's self-organizing feature maps for exploratory data analysis. In *Proceedings of the International Neural Network Conference (INNC'90)*, pages 305–308. Kluwer, 1990.

Fabio Vignoli, Rob van Gulik, and Huub van de Wetering. Mapping music in the palm of your hand, explore and discover your collection. In *Proceedings of the 5th International Conference on Music Information Retrieval (ISMIR'04)*, pages 409–414, Barcelona, Spain, October 10-14 2004.

E. Wold, T. Blum, D. Keislar, and J. Wheaton. Content-based classification search and retrieval of audio. *IEEE Multimedia*, 3(3):27–36, Fall 1996.

Eberhard Zwicker and Hugo Fastl. *Psychoacoustics, Facts and Models*, volume 22 of *Series of Information Sciences*. Springer, Berlin, 2 edition, 1999.