PlaySOM: An Alternative Approach to Track Selection and Playlist Generation in Large Music Collections

Michael Dittenbach[1], Robert Neumayer[2], and Andreas Rauber[1,2]

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

[2] Department of Software Technology and Interactive Systems
Vienna University of Technology
Favoritenstrasse 9–11/188, A-1040 Wien, Austria
a0003208@unet.univie.ac.at, andi@ifs.tuwien.ac.at

Abstract. Because of the increasing number of music distributors offering an ever growing number of albums and tracks on the Internet, access methods such as retrieval, interactive exploration or similarity-based search demand more sophisticated technologies than metadata-based approaches currently offer including queries for artists, albums, titles or manually assigned genre information. This requirement also holds for private collections with the tracks being more and more often stored uniformly in a library of, e.g., MP3 or OGG files as opposed to just a few tracks stored on separate media such as vinyl records, tapes or compact discs that have to be changed when playing more than one album.

The *SOM-enhanced JukeBox (SOMeJB)* system provides automatic indexing and organization of music repositories based on perceived sound similarity of the single tracks using a map metaphor for visualization with similar songs being placed into similar regions on the map. In this paper we present the *PlaySOM*, a novel interface allowing to browse a music collection by navigating a map and selecting regions of interest containing similar tracks for playing. This approach offers content-based organization of music as an alternative to conventional playlists, i.e. flat or hierarchical listings of music tracks sorted and filtered by some meta information.

1 Introduction

The requirement of sophisticated methods for organizing digital music repositories is driven by their increasing popularity and size. In a commercial setting the quality of organization and representation of such collections is crucial for satisfying the needs of all three of the participating groups, i.e. users, publishers/retailers as well as artists. Users who can easily find what they are looking for are likely to return using the same service again in the future. Furthermore, offering users additional information about songs or artists that are similar to those they were actually searching for, increases the chance of buying more songs than they actually intended to. Amazon.com has impressively shown that the concept of linking similar items and presenting them to users bears fruits. Please note that this particular linking concept is based on browsing and buying behavior as opposed to content-based similarity. Moreover, the possibility of searching for music similar to already known titles or artists also offers artists that are unknown so far a chance to become popular.

Regarding access to private music repositories the main motivations are most likely to be fun, entertainment and overcoming limitations of current media players. An important aspect of selecting music for listening is the *mood* one is currently in, or maybe the mood one wants to get into. Besides genre, mood (e.g. romantic) is also a main theme of so-called *music samplers* or *compilations*, i.e. albums containing songs of multiple artists adhering to a common motif. Hence, the functionality of generating playlists according to perceived sound similarity is the continuation of the concept of music samplers on a larger scale. This functionality can hardly be achieved by a conventional system with even well-maintained genre information of the tracks, because sound similarity often crosses genre borders. Furthermore, genre information is often attributed to whole albums or even artists and not songs in particular. Consider for example the rather slow and soothing song *Nothing Else Matters* by *Metallica* which would usually be classified as *Metal* or the like. From the perspective of perceived sound similarity, this song is actually closer to the genre *classical music* than to the rest of most *Metallica* songs.

Common query mechanisms for large music collections usually implement text-based metadata searches by keyword, allowing the user to search for a specific term either in the artist, title, or album fields of the songs' metadata. For a collection of audio files, especially MP3 files, a certain set of metadata can be provided, but this is often not the case. However, even though a metadata set kept in good condition makes such text-based queries easy, it does not provide to search by means of similarity. Text-based queries in large repositories require a certain a priori knowledge from the user about the music contained or an enormous manual editorial effort to provide linking between music items (e.g. allmusic[3]). However, those entries may be incomplete, inappropriate, too far oriented towards the personal likes of the contributor, or simply wrong. Therefore relying on those data may lead to results distorted by personal influence, in particular considering genre definitions and missing or wrong values. This might prevent users from finding new titles they had never heard of before.

Additionally, advanced search techniques become more necessary with the growing size of a collection. Although a user might not experience any problems when browsing a collection of a few hundred songs she or he knows quite well, navigating through thousands of songs one is not familiar with, may lead to an intrinsic restriction regarding access to this collection. A user would not be able to gain access to the majority of songs in such collections. Another approach is to provide access via genre hierarchies. This turns out not to be a feasible solution as

[3] `www.allmusic.com`

well because of possible data inconsistencies, the user's acceptance for arbitrarily predefined genre hierarchies and the great effort needed for providing a clearly structured genre hierarchy. Therefore, a mechanism to group music by similarity in combination with innovative access mechanisms is required.

We propose a novel interface, namely the *PlaySOM*, visualizing pieces of music organized on a two-dimensional map in such a way that similar songs are located close to each other with respect to a feature set that is automatically extracted from audio data. We use the feature extraction mechanism of the *SOM-enhanced JukeBox (SOMeJB)* system in order to extract the information from music tracks in MP3 or raw audio format. A *Self-Organizing Map* is used to cluster the numeric representations of the songs that are entirely based on the music itself without the use of any metadata.

Users can interactively explore the resulting two-dimensional map, select areas of interest and play those songs located there in any media player. The map metaphor in combination with the interactive features of the user interface provides an overview of a collection as well as insight into relations between tracks based on sound similarity. A music collection consisting of over 8,000 songs is used to demonstrate the capabilities of the *SOMeJB* system and the *PlaySOM* interface for clustering, visualization and user interaction.

The remainder of this paper is organized as follows. Section 2 briefly reviews the related work followed by a description of the *SOMeJB* system in Section 3 including a brief overview of the feature extraction as well as the *SOM* clustering algorithm. The *PlaySOM* interface is introduced in Section 4 by visualizing a music collection consisting of over 8,000 tracks of a variety of genres. Finally, some conclusions are provided in Section 5.

2 Related Work

A significant amount of research has been conducted in the area of content-based music retrieval (cf. [1, 2]). Methods have been developed to search for pieces of music with a particular melody. Users may express a query by humming a melody, which is then usually transformed into a symbolic melody representation. This is matched against a database of scores given, for example, in MIDI format. For example, research in this direction is reported in [3] and [4]. Other than melodic information it is also possible to extract and search for style information using the MIDI format. Yet, only a small fraction of all electronically available pieces of music are available as MIDI. A more readily available format is the raw audio signal, which all other audio formats can be decoded to. A system where hummed queries are posed against an MP3 archive for melody-based retrieval is presented in [5]. Both melody-based retrieval of music, as well as access to music available in MIDI-format are outside the scope of this paper.

However, this paper focuses on methods extracting style or genre information directly from the audio content, i.e. by indexing e.g. MP3 or WAV files. This kind of similarity-based organization and detection has gained significant interest recently. One of the first works to incorporate psychoacoustic modeling into the feature extraction process and using the *SOM* for organizing audio data is reported in [6]. A first approach, classifying audio recordings into speech, music, and environmental sounds is presented in [7]. A system performing trajectory matching using *SOMs* and MFCCs is presented in [8]. Specifically addressing the classification of sounds into different categories, loudness, pitch, brightness, bandwidth, and harmonicity features are used in [9] to train classifiers. Furthermore, work on similarity-based music retrieval is also reported in [10]. A wide range of musical surface features is used by the *MARSYAS* system [11, 12] to organize music into different genre categories using a selection of classification algorithms. The set of features that are used for clustering the music collection are *Rhythm Patterns* used in the *SOMeJB* system [13].

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 [14]. An implementation of a map-like playlist interface not described in scientific literature is the *Synapse Media Player*[4]. 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.

3 SOM-enhanced JukeBox

3.1 Feature Extraction from Audio Signals

The feature extraction process for the *Rhythm Patterns* is composed of two stages. First, the specific loudness sensation in different frequency bands is computed, which is then transformed into a time-invariant representation based on the modulation frequency. Starting from a standard Pulse-Code-Modulated (PCM) signal, stereo channels are combined into a mono signal, which is further down-sampled to 11kHz. Furthermore, pieces of music are cut into 6-second segments, removing the first and last two segments to eliminate lead-in and fade-out effects, and retaining only every second segment for further analysis. Using a Fast Fourier Transform (FFT), the raw audio data is further decomposed into frequency ranges using Hanning Windows with 256 samples (corresponding to 23ms) with 50% overlap, resulting in 129 frequency values (at 43Hz intervals) every 12 ms. These frequency bands are further grouped into so-called critical bands, also referred to by their unit bark [15], by summing up the values of the power spectrum between the limits of the respective critical band, resulting in 20 critical-band values. A spreading function is applied to account for masking effects, i.e. the masking of simultaneous or subsequent sounds by a given sound. The spread critical-band values are transformed into the logarithmic decibel scale, describing the sound pressure level in relation to the hearing threshold. Since the relationship between the dB-based sound pressure levels and our hearing sensation depends on the frequency of a tone, we calculate loudness levels, referred to as phon, using the equal-loudness contour matrix. From the loudness

[4] `www.synapseai.com`

levels we calculate the specific loudness sensation per critical band, referred to as sone.

To obtain a time-invariant representation, recurring patterns in the individual critical bands resembling rhythm are extracted in the second stage of the feature extraction process. This is achieved by applying another discrete Fourier transform, resulting in amplitude modulations of the loudness in individual critical bands. These amplitude modulations have different effects on our hearing sensation depending on their frequency, the most significant of which, referred to as fluctuation strength, is most intense at 4Hz and decreasing towards 15Hz (followed by the sensation of roughness, and then by the sensation of three separately audible tones at around 150Hz). We thus weight the modulation amplitudes according to the fluctuation strength sensation, resulting in a time-invariant, comparable representation of the rhythmic patterns in the individual critical bands. To emphasize the differences between strongly reoccurring beats at fixed intervals a final gradient filter is applied, paired with subsequent Gaussian smoothing to diminish unnoticeable variations. The resulting 1.440 dimensional feature vectors (24 critical bands times 60 amplitude modulation values) capture beat information up to 10Hz (600bpm), going significantly beyond what is conventionally considered beat structure in music. These *Rhythm Patterns* are further used for data signal comparison.

3.2 Self-Organizing Map

The *Self-Organizing Map (SOM)* is an unsupervised neural network providing a mapping from a high-dimensional input space to a usually two-dimensional output space while preserving topological relations as faithfully as possible [16, 17]. The *SOM* consists of a set of i units arranged in a two-dimensional grid with a weight vector $m_i \in \Re^n$ attached to each unit. 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. Commonly, the Euclidean distance between the weight vector of the unit and the input vector serves 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. Thus, this unit's activation will be even higher the next time the same input signal is presented. Furthermore, the weight vectors of units in the neighborhood of the winner as described by a time-decreasing neighborhood function are modified accordingly, yet to a smaller amount as compared to the winner. This learning procedure finally leads to a topologically ordered mapping of the presented input signals. Consequently, similar input data are mapped onto neighboring regions of the map.

3.3 Visualization Techniques

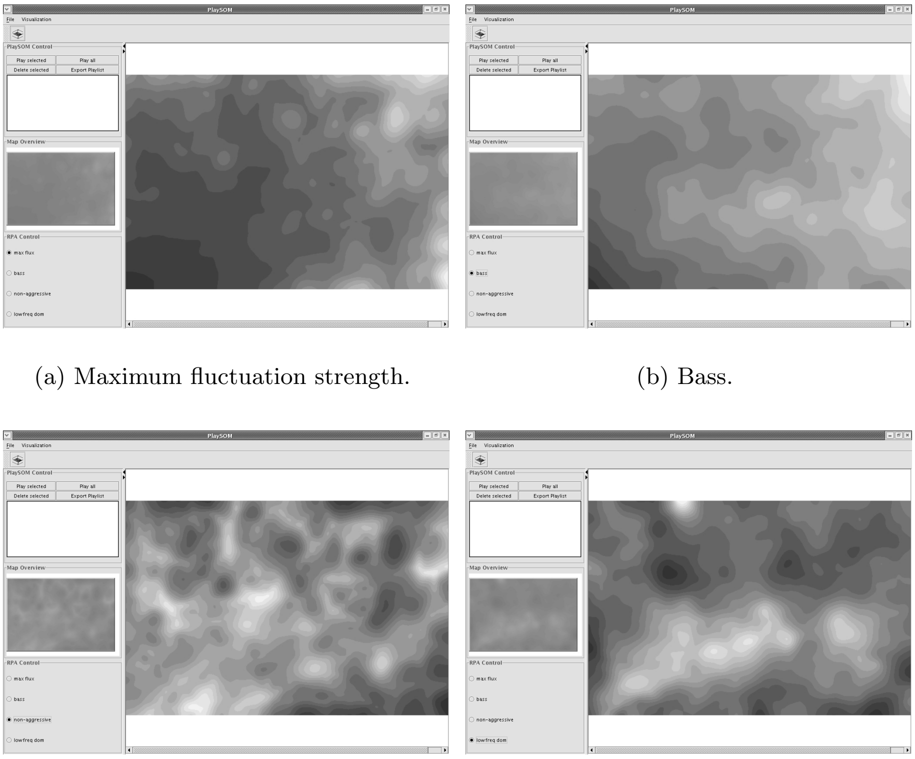
Since the cluster structure of a trained *SOM* is not inherently visible, several visualization techniques have been reported in literature with the most prominent being the *U-Matrix* by Ultsch and Siemon [18]. This technique maps the distances between the weight vectors of adjacent units onto a color 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.

The visualization of component planes is another useful method to gain insight into the structure of a trained *SOM*. Here, only a certain component of the weight vectors is taken into account to color-code the map representation. 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. In the case of *Rhythm Patterns*, four combinations of component planes have been chosen according to psychoacoustic features, because single component planes do not directly translate into psychoacoustic features noticed by the human ear. In particular, *maximum fluctuation strength* evaluates to the maximum value of all vector components representing music dominated by strong beats. Second, *bass* is 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. Third, *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, how much *low frequencies dominate* is measured as the ratio between the five lowest and the five highest critical bands. A more detailed explanation can be found in [19]. Examples of these visualizations are shown in Figure 1 in the next section.

4 PlaySOM Interface and Experiments

In this section we show experimental results of clustering our collection of over 8,000 songs according to perceived sound similarity. Moreover, we present the features of our *PlaySOM* interface for interactive exploration of the audio information space. We have to point out the difficulty of quantitatively assessing the cluster quality due to the highly subjective nature of the data, i.e. the sound similarity of songs. Nevertheless, we will provide an overview of the organization of our collection and pick some sample areas of the map to demonstrate the good results of our approach.

A *Self-Organizing Map* consisting of 60×40 units has been trained using the *Rhythm Patterns* of over 8,000 tracks from variety of genres. The majority of the songs stem from the last couple of decades covering nearly anything from *Soul*, *Pop*, *Punk*, *Alternative*, *Rock* or *Metal*, but the collection also contains some classical pieces from, e.g. *Liszt*, *Smetana* and others. In Figure 1, the *PlaySOM* interface is depicted. The largest part of the user interface is occupied by the interactive map itself. The main elements on the left-hand side are the



(a) Maximum fluctuation strength.

(b) Bass.

(c) Non-aggressiveness.

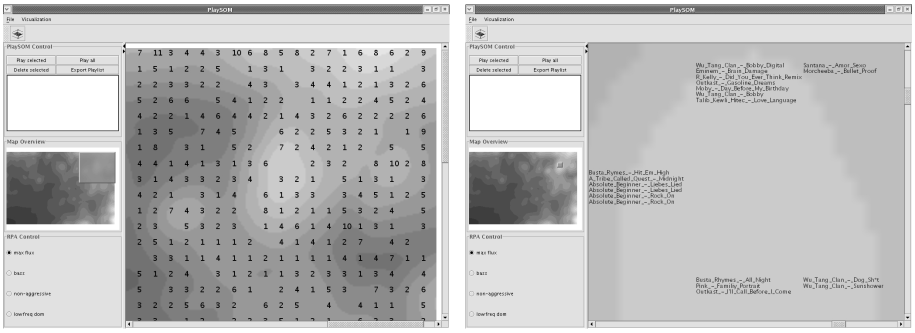
(d) Low frequencies dominant.

Fig. 1. *PlaySOM* interface with different visualizations of Rhythm Patterns.

list of selected tracks, a birds-eye-view on the complete map indicating the area currently visible in main map as well as controls for selecting the visualization.

Figures 1(a)-(d) show the complete map visualizing the four different *Rhythm Patterns* described in the previous section, respectively. 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 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* [20].

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 right-hand side of the map. Especially *Hip Hop* and *Rap* songs exhibit high values and therefore tracks by artists such as the *Wu-Tang Clan* or *Eminem* can be found there. Contrarily, songs with low values are



(a) Upper right corner with the number of songs written on the respective units.

(b) High zoom level showing more detail, i.e. song names on the respective units.

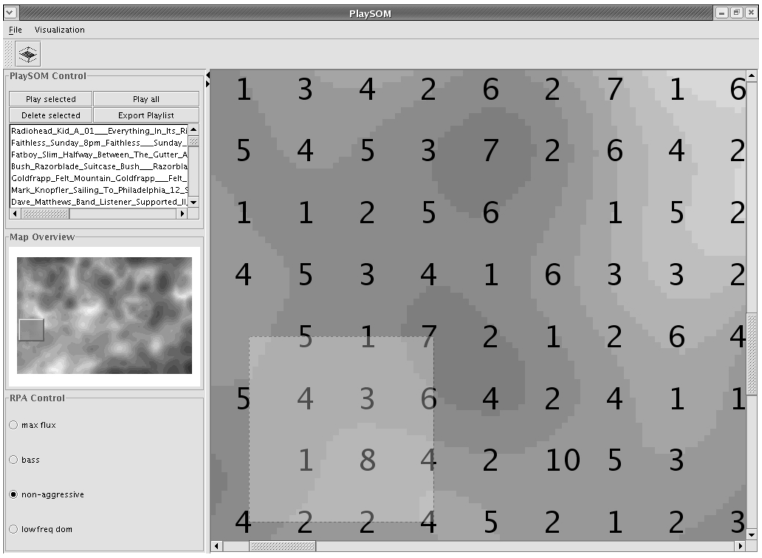
Fig. 2. Depending on the zooming level, different kinds of information are presented depending on the displayed map size.

located in the lower left corner. Some examples of rather tranquil music are the afore mentioned classical pieces, a large number of songs by *Pink Floyd* as well as by *Mark Knopfler*, *Dire Straits*, *Tom Waits*, *Tori Amos* or *Peter Gabriel*.

Figure 1(b) shows that the feature *bass* corresponds to the *maximum fluctuation strength* with high values, again being distributed across the right-hand side of the map but to a larger extent. In Figure 1(c), the majority of clusters containing *non-aggressive* music can be identified in the lower left area of the map as one would expect regarding the distribution of the *maximum fluctuation strength*, which represents music dominated by strong and fast beats. One exception that should be noted is the cluster located on the right-hand side border in the upper half. This area contains mainly songs from the album *Journey to Jah* by *Gentleman* who is attributed the genre of *Roots Reggae*. The songs on this album combine the feature of *bass* with those of *non-aggressiveness*. Finally, a large cluster where *low frequencies dominate* is located in the lower half of the map as shown in Figure 1(d).

The major ways of interacting with the map of the music collection are panning, zooming and selecting. An important characteristic of the map interface is that the level of zooming influences the amount and type of information that is displayed. The more a user zooms into the map, the more details of the mapped data are presented. A zooming sequence is depicted in Figure 2 with a rather coarse overview of the top-right corner of the map in Figure 2(a). At this zoom level, only the number of songs mapped onto the respective units are displayed. By zooming further into the region of interest the actual song titles become visible (see Figure 2(b)).

Currently, pieces of music can be selected by either clicking on single units or by selecting a rectangular region of the map by dragging the mouse. As soon as the map selection changes, the list of selected tracks is updated. It is then



Radiohead_Kid_A_01___Everything_In_Its_Ri
Faithless_Sunday_8pm_Faithless___Sunday_
Fatboy_Slim_Halfway_Between_The_Gutter_A
Bush_Razorblade_Suitcase_Bush___Razorbla
Goldfrapp_Felt_Mountain_Goldfrapp___Felt_
Mark_Knopfler_Sailing_To_Philadelphia_12_S
Dave_Matthews_Band_Listener_Supported_II

○ max flux
○ bass
● non-aggressive
○ lowfreq dom

1	3	4	2	6	2	7	1	6
5	4	5	3	7	2	6	4	2
1	1	2	5	6		1	5	2
4	5	3	4	1	6	3	3	2
	5	1	7	2	1	2	6	4
5	4	3	6	4	2	4	1	1
	1	8	4	2	10	5	3	
4	2	2	4	5	2	1	2	3

Fig. 3. Selection of an area with the list of the respective songs being displayed on the left-hand side of the user interface.

possible to further edit the playlist by deleting single tracks that are unwanted. Any media player supporting playlists in M3U format can then be called for playing the songs selected via the map.

Regarding similarity search, query by example is easily realized by extracting the *Rhythm Patterns* of a sample song, mapping the numerical representation onto the *Self-Organizing Map* and by highlighting the unit with the best-matching weight vector. Our approach can also be combined with metadata-based search in order to identify stylistic regions on the map where songs by a particular artist are located. Again, units containing songs that match a certain metadata-based query can be highlighted to show this information.

5 Conclusions

We have presented the *PlaySOM*, a novel interface to music collections for interactive exploration, track selection and similarity-based search. The audio features of the songs are automatically extracted and used for training of a *Self-Organizing Map*, i.e. a neural network model with unsupervised learning function. The *PlaySOM* offers a two-dimensional map display with similar songs being located in spatially close regions on the map. This approach is especially appealing for large collections that can now be explored by sound similarity rather than by arbitrarily assigned genre information, which often shows to be incorrect or biased by personal preferences. This interface unveils the abundance of music present in large repositories at a glance, which can hardly be seen using conventional text-based search mechanisms.

References

1. Downie, J.: Music information retrieval. In: Annual Review of Information Science and Technology. Volume 37. Information Today, Medford, NJ (2003) 295–340
2. Foote, J.: An overview of audio information retrieval. Multimedia Systems **7** (1999)
3. Bainbridge, D., Nevill-Manning, C., Witten, H., Smith, L., McNab, R.: Towards a digital library of popular music. In Fox, E., Rowe, N., eds.: Proc. ACM Conference on Digital Libraries (ACMDL'99), Berkeley, CA, ACM (1999) 161–169
4. Birmingham, W., Dannenberg, R., Wakefield, G., Bartsch, M., Bykowski, D., Mazzoni, D., Meek, C., Mellody, M., Rand, W.: MUSART: Music retrieval via aural queries. In: Proc. 2nd Ann. Symp. on Music Information Retrieval. (2001)
5. Liu, C., Tsai, P.: Content-based retrieval of mp3 music objects. In: Proceedings of the 10th International Conference on Information and Knowledge Management (CIKM 2001), Atlanta, Georgia, ACM (2001) 506–511
6. Feiten, B., Günzel, S.: Automatic indexing of a sound database using self-organizing neural nets. Computer Music Journal **18** (1994) 53–65
7. Zhang, H., Zhong, D.: A scheme for visual feature based image indexing. In: Proceedings of the IS&T/SPIE Conference on Storage and Retrieval for Image and Video Databases, San Jose, CA (1995) 36–46
8. Spevak, C., Favreau, E.: Soundspotter - a prototype system for content-based audio retrieval. In: Proceedings of the 5. International Conference on Digital Audio Effects (DAFx-02(), Hamburg, Germany (2002)
9. Wold, E., Blum, T., Keislar, D., Wheaton, J.: Content-based classification search and retrieval of audio. IEEE Multimedia **3** (1996) 27–36
10. Welsh, M., Borisov, N., Hill, J., von Behren, R., Woo, A.: Querying large collections of music for similarity. Technical Report UCB/CSD00 -1096, U.C. Berkeley Computer Science Division (1999)
11. Tzanetakis, G., Cook, P.: Marsyas: A framework for audio analysis. Organized Sound **4** (2000)
12. Tzanetakis, G., Cook, P.: Musical genre classification of audio signals. IEEE Transactions on Speech and Audio Processing **10** (2002) 293–302
13. Rauber, A., Frühwirth, M.: Automatically analyzing and organizing music archives. In: Proc. 5th European Conference on Research and Advanced Technology for Digital Libraries (ECDL 2001). LNCS, Darmstadt, Germany, Springer (2001)
14. Logan, B.: Content-based playlist generation: Exploratory experiments. In: Proc. 3rd Ann. Symp. on Music Information Retrieval (ISMIR 2002), France (2002)
15. Zwicker, E., Fastl, H.: Psychoacoustics, Facts and Models. 2 edn. Volume 22 of Series of Information Sciences. Springer, Berlin (1999)
16. Kohonen, T.: Self-organized formation of topologically correct feature maps. Biological Cybernetics **43** (1982) 59–69
17. Kohonen, T.: Self-Organizing Maps. 3rd edn. Volume 30 of Springer Series in Information Sciences. Springer, Berlin (2001)
18. Ultsch, A., Siemon, H.: Kohonen's self-organizing feature maps for exploratory data analysis. In: Proceedings of the International Neural Network Conference (INNC'90), Dordrecht, Netherlands, Kluwer (1990) 305–308
19. Rauber, A., Pampalk, E., Merkl, D.: The SOM-enhanced JukeBox: Organization and visualization of music collections based on perceptual models. Journal of New Music Research **32** (2003) 193–210
20. Pampalk, E.: Islands of Music - Analysis, organization, and visualization of music archives. Journal of the Austrian Soc. for Artificial Intelligence **22** (2003) 20–23