Using Psycho-Acoustic Models and Self-Organizing Maps to Create a Hierarchical Structuring of Music by Sound Similarity

Andreas Rauber*
Dept. of Software Technology
Vienna Univ. of Technology
A-1040 Vienna, Austria
andi@ifs.tuwien.ac.at

Elias Pampalk
Austrian Research Institute for
Artificial Intelligence
A-1010 Vienna, Austria
elias@ai.univie.ac.at

Dieter Merkl
Dept. of Software Technology
Vienna Univ. of Technology
A-1040 Vienna, Austria
dieter@ifs.tuwien.ac.at

ABSTRACT

With the advent of large musical archives the need to provide an organization of these archives becomes eminent. While artist-based organizations or title indexes may help in locating a specific piece of music, a more intuitive, genre-based organization is required to allow users to browse an archive and explore its contents. Yet, currently these organizations following musical styles have to be designed manually.

In this paper we propose an approach to automatically create a hierarchical organization of music archives following their perceived sound similarity. More specifically, characteristics of frequency spectra are extracted and transformed according to psycho-acoustic models. Subsequently, the Growing Hierarchical Self-Organizing Map, a popular unsupervised neural network, is used to create a hierarchical organization, offering both an interface for interactive exploration as well as retrieval of music according to perceived sound similarity.

1. INTRODUCTION

With the availability of high-quality audio file formats at sufficient compression rates, we find music increasingly being distributed electronically via large music archives, offering music from the public domain, selling titles, or streaming them on a pay-per-play basis, or simply in the form of on-line retailers for conventional distribution channels. A core requirement for these archives is the possibility for the user to locate a title he or she is looking for, or to find out which types of music are available in general.

Thus, those archives commonly offer several ways to find a desired piece of music. A straightforward approach is to use text based queries to search for the artist, the title or some phrase in the lyrics. While this approach allows the localization of a desired piece of music, it requires the user to know and actively input information about the title he or she is looking for. An alternative approach, allowing users to explore the music archive, searching for musical styles, rather than for a specific title or group, is thus usually provided in the form of genre hierarchies such as *Classical, Jazz, Rock*. Hence, a customer looking for an opera recording might look into the *Classic* section, and will there find - depending on the further organization of the music archive - a variety of interpretations, being similar in style, and thus possibly suiting his or her likings. However, such organizations rely on manual categorizations and usually consist of several hundred categories which involve high maintenance costs, in particular for dynamic music collections, where multiple contributors have to file their contributions accordingly. The inherent difficulties of such taxonomies have been analyzed, for example, in [22]. Another approach taken by on-line music stores is to analyze the behavior of customers to give those showing similar interests recommendations on music which they might appreciate. For example, a simple approach is to give a customer looking for pieces similar to *Für Elise* recommendations on music which is usually bought by people who also purchased *Für Elise*. However, extensive and detailed customer profiles are rarely available.

The *SOMeJB*, i.e. the *SOM-enhanced Jukebox* system, outlined in [26], facilitates exploration of music archives without relying on further information such as customer profiles or predefined categories. It does not require the availability of detailed, high-quality meta-data on the various pieces of music, or musical scores. Rather, we rely on the sound information, present in the form of any acoustical wave format, as it is available e.g. from CD tracks or MP3 files. Based on the sound signal we extract low-level features based on frequency spectra dynamics, and process them using psycho-acoustic models of our auditory system. The resulting representation allows us to calculate to a certain degree the perceived similarity between two pieces of music. We use this form of data representation as input to the *Growing Hierarchical Self-Organizing Map* (*GHSOM*) [6], an extension to the popular self-organizing map [13]. This neural network provides cluster analysis by mapping similar data items close to each other on a map display. Specifically, the *GHSOM* is capable of detecting hierarchical relationships in the data, and thus produces a hierarchy of maps representing various styles of music, into which the pieces of music are organized.

The remainder of this paper is organized as follows. Section 2 briefly reviews the related work. The feature extraction process is presented in detail in Section 3, followed by a description of the principles and training procedure of the *Self-Organizing Map*, and the *Growing Hierarchical Self-Organizing Map* in Section 4. We then describe experimental results, using both a reduced collection of 77 pieces of music, as well as a larger archive consisting of 359 pieces in Section 5. Finally, in Section 6 some conclusions are drawn.

2. RELATED WORK

A vast amount of research has been conducted in the area of content-based music and audio retrieval. For example, methods have been developed to search for pieces of music with a particular melody. The queries can be formulated by humming and are usually transformed into a symbolic melody representation, which is matched against a database of scores usually given in MIDI format. Research in this direction is reported in, e.g. [1, 2, 10, 16, 28]. Other than melodic information it is also possible to extract and search for style information using the MIDI format. For example, in [4] solo improvised trumpet performances are classified into one of the four styles: *lyrical, frantic, syncopated*, or *pointillistic*.

The MIDI format offers a wealth of possibilities, however, only a small fraction of all electronically available pieces of music are

*Part of this work was done while the author was an ERCIM Research Fellow at IEI, Consiglio Nazionale delle Ricerche (CNR), Pisa, Italy.

available as MIDI. A more readily available format is the raw audio signal to which all other audio formats can be decoded. One of the first audio retrieval approaches dealing with music was presented in [35], where attributes such as the pitch, loudness, brightness and bandwidth of speech and individual musical notes were analyzed. Several overviews of systems based on the raw audio data have been presented, e.g. [9, 18]. However, most of these systems do not treat content-based music retrieval in detail, but mainly focus on speech or partly-speech audio data, with one of the few exceptions being presented in [17], using hummed queries against an MP3 archive for melody-based retrieval.

Furthermore, only few approaches in the area of content-based music analysis have utilized the framework of psychoacoustics. Psychoacoustics deals with the relationship of physical sounds and the human brain's interpretation of them, cf. [37]. One of the first exceptions was [8], where psychoacoustic models are used to describe the similarity of instrumental sounds. The approach was demonstrated using a collection of about 100 instruments, which were organized using a *Self-Organizing Map* in a similar way as presented in this paper. For each instrument a 300 milliseconds sound was analyzed and steady state sounds with a duration of 6 milliseconds were extracted. These steady state sounds can be regarded as the smallest possible building blocks of music. A model of the human perceptual behavior of music using psychoacoustic findings was presented in [30] together with methods to compute the similarity of two pieces of music. A more practical approach to the topic was presented in [33] where music given as raw audio is classified into genres based on musical surface and rhythm features. The features are similar to the rhythm patterns we extract, the main difference being that we analyze them separately in 20 frequency bands.

Our work is based on first experiments reported in [26]. In particular we have redesigned the feature extraction process using psychoacoustic models. Additionally, by using a hierarchical extension of the neural network for data clustering we are able to detect the hierarchical structure within our archive.

3. FEATURE EXTRACTION

The architecture of the *SOMeJB* system may be divided into 3 stages as depicted in Figure 1. Digitized music in good sound quality (44kHz, stereo) with a duration of one minute is represented by approximately 10MB of data in its raw format describing the physical properties of the acoustical waves we hear. In a preprocessing stage, the audio signal is transformed, down-sampled and split into individual segments (steps P1 to P3). We then extract features which are robust towards non-perceptive variations and on the other hand resemble characteristics which are critical to our hearing sensation, i.e. rhythm patterns in various frequency bands. The feature extraction stage can be divided into two subsections, consisting of the extraction of the specific loudness sensation expressed in *Sone* (steps S1 to S6), as well as the conversion into time-invariant frequency-specific rhythm patterns (step R1 to R3). Finally, the data may be optionally converted, before being organized into clusters in steps A1 to A3 using the *GHSOM*. The feature extraction steps are further detailed in the following subsections, with the clustering procedure being described in Section 4, with the visualization metaphor being only touched upon briefly due to space considerations.

3.1 Preprocessing

(**P1**) The pieces of music may be given in any audio file format, such as e.g. MP3 files. We first decode these to the raw *Pulse Code Modulation* (PCM) audio format.

(**P2**) The raw audio format of music in good quality requires huge amounts of storage. As humans can easily identify the genre of a piece of music even if its sound quality is rather poor we can safely reduce the quality of the audio signal. Thus, stereo sound quality is first reduced to mono and the signal is then down-sampled from

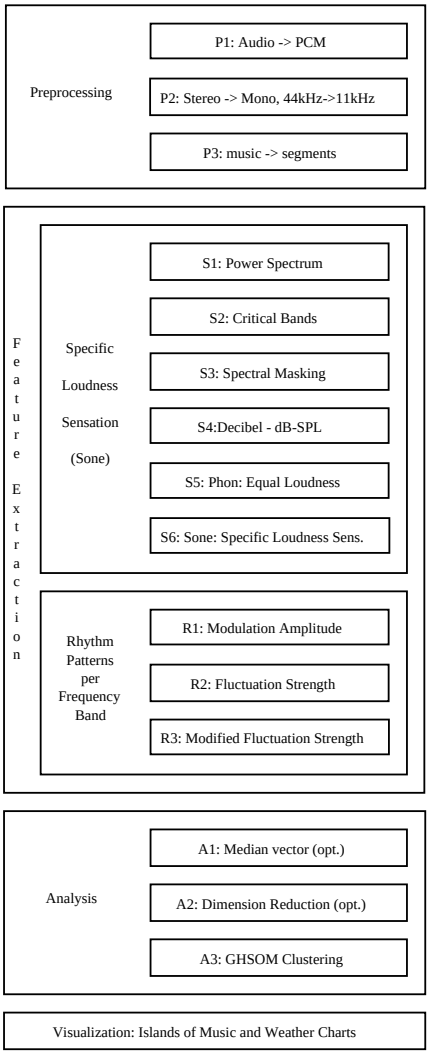


Figure 1: System Overview: preprocessing, 2-stage feature extraction, cluster analysis and visualization

44kHz to 11kHz, leaving a distorted, but still easily recognizable sound signal comparable to phone line quality.

(**P3**) We subsequently segment each piece into 6-second sequences. The duration of 6 seconds (2^{16} samples) was chosen heuristically because it is long enough for humans to get an impression of the style of a piece of music while being short enough to optimize the computations. However, analyses with various settings for the segmentation have shown no significant differences with respect to segment length. After removing the first and the last 2 segments of each piece of music to eliminate lead-in and fade-out effects, we retain only every third of the remaining segments for further analysis. Again, the information lost by this type of reduction has shown insignificant in various experimental settings.

We thus end up with several segments of 6 seconds of music every 18 seconds at 11kHz for each piece of music. The preprocessing results in a data reduction by a factor of over 24 without losing relevant information, i.e. a human listener is still able to identify the genre or style of a piece of music given the few 6-second sequences in lower quality.

3.2 Specific Loudness Sensation - Sone

Loudness belongs to the category of intensity sensations. The loudness of a sound is measured by comparing it to a reference sound. The 1kHz tone is a very popular reference tone in psychoacoustics,

and the loudness of the 1kHz tone at 40dB is defined to be *1 Sone*. A sound perceived to be twice as loud is defined to be 2 *Sone* and so on. In the first stage of the feature extraction process, this specific loudness sensation (Sone) per critical-band (Bark) in short time intervals is calculated in 6 steps starting with the PCM data.

(**S1**) First the power spectrum of the audio signal is calculated. To do this, the raw audio data is first decomposed into its frequencies using a *Fast Fourier Transformation (FFT)*. We use a window size of 256 samples, which corresponds to about 23ms at 11kHz, and a Hanning window with 50% overlap. We thus obtain a Fourier transform of 11 / 2 kHz, i.e. 5.5 kHz signals.

(**S2**) The inner ear separates the frequencies and concentrates them at certain locations along the basilar membrane. The inner ear can thus be regarded as a complex system of a series of band-pass filters with an asymmetrical shape of frequency response. The center frequencies of these band-pass filters are closely related to the critical-band rates, where frequencies are bundled into 24 critical-bands according to the *Bark* scale [37]. Where these bands should be centered, or how wide they should be, has been analyzed through several psychoacoustic experiments. Since our signal is limited to 5.5 kHz we use only the first 20 critical bands, summing up the values of the power spectrum within the upper and lower frequency limits of each band, obtaining a power spectrum of the 20 critical bands for the segments.

(**S3**) Spectral Masking is the occlusion of a quiet sound by a louder sound when both sounds are present simultaneously and have similar frequencies. Spectral masking effects are calculated based on [31], with a spreading function defining the influence of the j-th critical band on the i-th being used to obtain a spreading matrix. Using this matrix the power spectrum is spread across the critical bands obtained in the previous step, where the masking influence of a critical band is higher on bands above it than on those below it.

(**S4**) The intensity unit of physical audio signals is sound pressure and is measured in *Pascal* (Pa). The values of the PCM data correspond to the sound pressure. Before calculating *Sone* values it is necessary to transform the data into decibel. The decibel value of a sound is calculated as the ratio between its pressure and the pressure of the hearing threshold, also known as dB-SPL, where SPL is the abbreviation for sound pressure level.

(**S5**) The relationship between the sound pressure level in decibel and our hearing sensation measured in *Sone* is not linear. The perceived loudness depends on the frequency of the tone. From the dB-SPL values we thus calculate the equal loudness levels with their unit Phon. The *Phon* levels are defined through the loudness in dB-SPL of a tone with 1kHz frequency. A level of 40 *Phon* resembles the loudness level of a 40dB-SPL tone at 1kHz. A pure tone at any frequency with 40 *Phon* is perceived as loud as a pure tone with 40dB at 1kHz. We are most sensitive to frequencies around 2kHz to 5kHz. The hearing threshold rapidly rises around the lower and upper frequency limits, which are respectively about 20Hz and 16kHz. Although the values for the equal loudness contour matrix are obtained from experiments with pure tones, they may be applied to calculate the specific loudness of the critical band rate spectrum, resulting in loudness level representations for the frequency ranges.

(**S6**) Finally, as the perceived loudness sensation differs for different loudness levels, the specific loudness sensation in *Sone* is calculated based on [3]. The loudness of the 1kHz tone at 40dB-SPL is defined to be 1 Sone. A tone perceived twice as loud is defined to be 2 *Sone* and so on. For values up to 40 *Phon* the sensation rises slowly, increasing at a faster rate afterwards.

Figure 2 illustrates the data after each of the feature extraction steps using the first 6-second sequences extracted from *Beethoven, Für Elise* and from *Korn, Freak on a Leash*. The sequence of *Für*

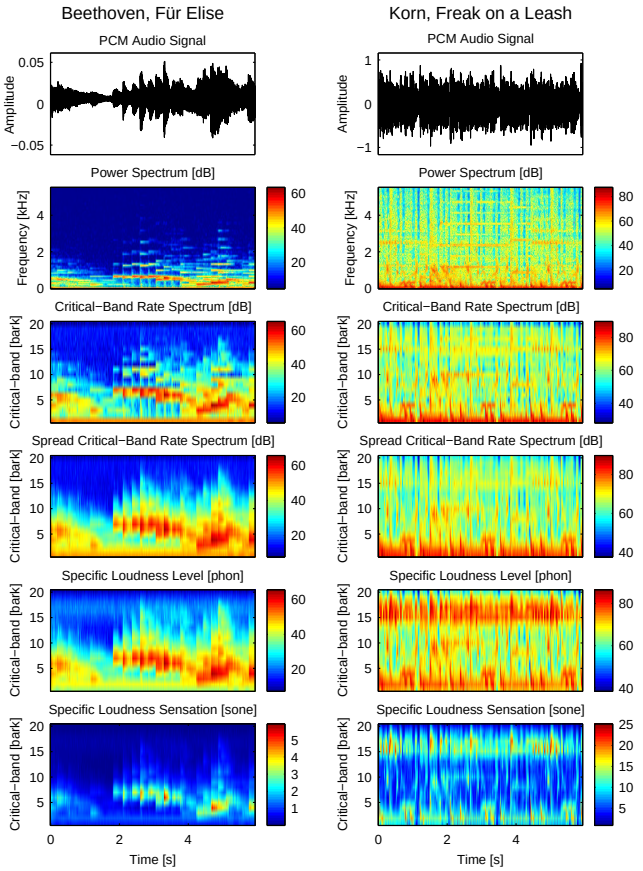


Figure 2: Steps S1-S6: from the 11kHz PCM audio signal to specific loudness per critical-band

Elise contains the main theme starting shortly before the 2nd second. The specific loudness sensation depicts each piano key played. On the other hand, *Freak on a Leash*, which is classified as *Heavy Metal/Death Metal*, is quite aggressive. Melodic elements do not play a major role and the specific loudness sensation is a rather complex pattern spread over the whole frequency range, whereas only the lower critical bands are active in *Für Elise*. Notice further, that the values of the patterns of *Freak on a Leash* are up to 18 times higher compared to those of *Für Elise*.

3.3 Rhythm Patterns

After the first preprocessing stage a piece of music is represented by several 6-second sequences. Each of these sequences contains information on how loud the piece is at a specific point in time in a specific frequency band. Yet, the current data representation is not time-invariant. It may thus not be used to compare two pieces of music point-wise, as already a small time-shift of a few milliseconds will usually result in completely different feature vectors. In the second stage of the feature extraction process, we calculate a time-invariant representation for each piece of music in 3 further steps, namely the frequency-wise rhythm pattern. These rhythm patterns contain information on how strong and fast beats are played within the respective frequency bands.

(**R1**) The loudness of a critical-band usually rises and falls several times resulting in a more or less periodical pattern, also known as the rhythm. The loudness values of a critical-band over a certain time period can be regarded as a signal that has been sampled at discrete points in time. The periodical patterns of this signal can then be assumed to originate from a mixture of sinuids. These sinuids modulate the amplitude of the loudness, and can be calculated by a Fourier transform. The modulation frequencies, which

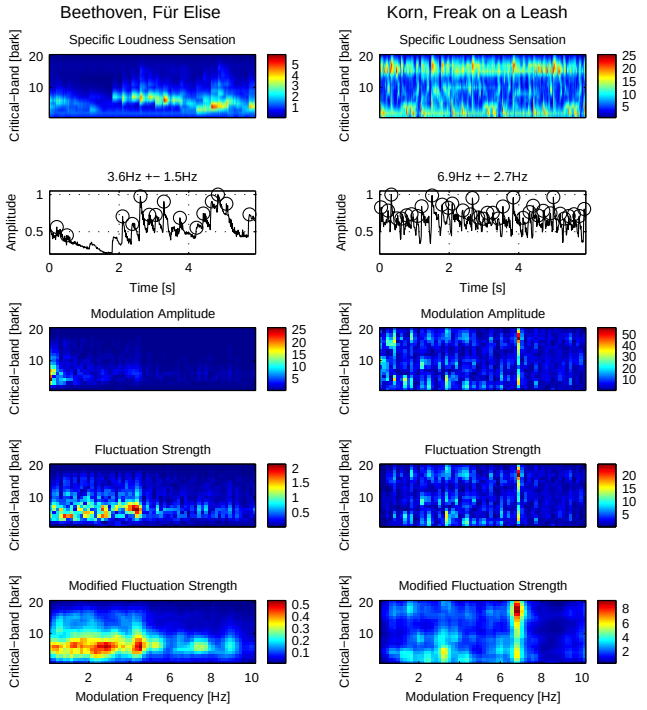


Figure 3: Steps R1-R3: from loudness sensation to modified fluctuation strength

can be analyzed using the 6-second sequences and time quanta of 12ms, are in the range from 0 to 43Hz with an accuracy of 0.17Hz. Notice that a modulation frequency of 43Hz corresponds to almost 2600bpm. Thus, the amplitude modulation of the loudness sensation per critical-band for each 6-second sequence is calculated using a FFT of the 6-second sequence of each critical band.

(**R2**) The amplitude modulation of the loudness has different effects on our sensation depending on the frequency. The sensation of *fluctuation strength* is most intense at a a modulation frequency of around 4Hz and gradually decreases up to 15Hz. At 15Hz the sensation of *roughness* starts to increase, reaches its maximum at about 70Hz, and starts to decreases at about 150Hz. Above 150Hz the sensation of hearing *three separately audible tones* increases. It is the fluctuation strength, i.e. rhythm patterns up to 10Hz, which corresponds to 600 beats per minute (bpm), that we are interested in. For each of the 20 frequency bands we obtain 60 values for modulation frequencies between 0 and 10Hz. This results in 1200 values representing the fluctuation strength.

(**R3**) To distinguish certain rhythm patterns better and to reduce irrelevant information, gradient and Gaussian filters are applied. In particular, we use gradient filters to emphasize distinctive beats, which are characterized through a relatively high fluctuation strength at a specific modulation frequency compared to the values immediately below and above this specific frequency. We further apply a Gaussian filter to increase the similarity between two rhythm pattern characteristics which differ only slightly in the sense of either being in similar frequency bands or having similar modulation frequencies by spreading the according values. We thus obtain modified fluctuation strength values that can be used as feature vectors for subsequent cluster analysis.

The second part of the feature extraction process is summarized in Figure 3. Looking at the modulation amplitude of *Für Elise* it seems as though there is no beat. In the fluctuation strength subplot the modulation frequencies around 4Hz are emphasized. Yet, there are no clear vertical lines, as there are no periodic beats. On the other hand, note the strong beat of around 7Hz in all frequency bands of *Freak on a Leash*. For an in-depth discussion of the characteristics of the feature extraction process, please refer to [23, 24].

4. HIERARCHICAL DATA CLUSTERING

Using the rhythm patterns we apply the *Self-Organizing Map* (*SOM*) [13], as well as its extension, the *Growing Hierarchical Self-Organizing Map* (*GHSOM*) [6] algorithm to organize the pieces of music on a 2-dimensional map display in such a way that similar pieces are grouped close together. In the following sections we will briefly review the principles of the *SOM* and the *GHSOM*, followed by a description of the last steps of the *SOMeJB* system, i.e. the cluster analysis steps A1 to A3 in Figure 1.

4.1 Self-Organizing Maps

The *Self-Organizing Map* (*SOM*), as proposed in [12] and described thoroughly in [13], is one of the most distinguished unsupervised artificial neural network models. It basically provides cluster analysis by producing a mapping of high-dimensional input data onto a usually 2-dimensional output space while preserving the topological relationships between the input data items as faithfully as possible. In other words, the *SOM* produces a projection of the data space onto a two-dimensional map space in such a way, that similar data items are located close to each other on the map.

More formally, the *SOM* consists of a set of units i, which are arranged according to some topology, where the most common choice is a two-dimensional grid. Each of the units i is assigned a model vector m_i of the same dimension as the input data, $m_i \in \Re^n$. In the initial setup of the model prior to training, the model vectors are frequently initialized with random values. However, more sophisticated strategies such as, for example, Principle Component Analysis, may be applied. During each learning step t, an input pattern $x(t)$ is randomly selected from the set of input vectors and presented to the map. Next, the unit showing the most similar model vector with respect to the presented input signal is selected as the winner c, where a common choice for similarity computation is the Euclidean distance, cf. Expression 1.

$$c(t) : ||x(t) - m_c(t)|| = \min_i \{||x(t) - m_i(t)||\} \qquad (1)$$

Adaptation takes place at each learning iteration and is performed as a gradual reduction of the difference between the respective components of the input vector and the model vector. The amount of adaptation is guided by a monotonically decreasing learning-rate α, ensuring large adaptation steps at the beginning of the training process, followed by a fine-tuning-phase towards the end.

Apart from the winner, units in a time-varying and gradually decreasing neighborhood around the winner are adapted as well. This enables a spatial arrangement of the input patterns such that alike inputs are mapped onto regions close to each other in the grid of output units. Thus, the training process of the self-organizing map results in a topological ordering of the input patterns. According to [27] the self-organizing map can be viewed as a neural network model performing a spatially smooth version of k-means clustering. The neighborhood of units around the winner may be described implicitly by means of a neighborhood-kernel h_{ci} taking into account the distance – in terms of the output space – between unit i under consideration and unit c, the winner of the current learning iteration. A Gaussian may be used to define the neighborhood-kernel, ensuring stronger adaption of units close to the winner. It is common practice that in the beginning of the learning process the neighborhood-kernel is selected large enough to cover a wide area of the output space. The spatial width of the neighborhood-kernel is reduced gradually during the learning process such that towards the end of the learning process just the winner itself is adapted.

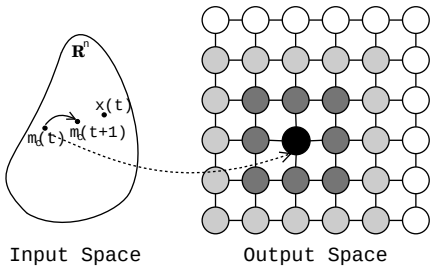


Figure 4: SOM Training: model vector adaption

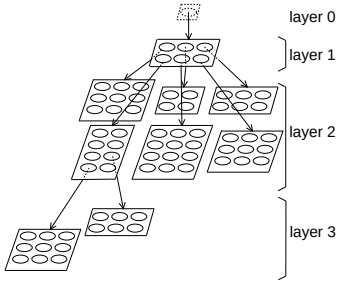


Figure 5: *GHSOM* architecture

In combining these principles of self-organizing map training, we may write the learning rule as given in Expression (2), with α representing the time-varying learning-rate, h_{ci} representing the time-varying neighborhood-kernel, x representing the currently presented input pattern, and m_i denoting the model vector assigned to unit i.

$$m_i(t+1) = m_i(t) + \alpha(t) \cdot h_{ci}(t) \cdot [x(t) - m_i(t)] \qquad (2)$$

A simple graphical representation of a self-organizing map's architecture and its learning process is provided in Figure 4. In this figure the output space consists of a square of 36 units, depicted as circles, forming a grid of 6×6 units. One input vector $x(t)$ is randomly chosen and mapped onto the grid of output units. The winner c showing the highest activation is determined. Consider the winner being the unit depicted as the black unit labeled in the figure. The model vector of the winner, $m_c(t)$, is now moved towards the current input vector. This movement is symbolized in the input space in Figure 4. As a consequence of the adaptation, unit c will produce an even higher activation with respect to the input pattern x at the next learning iteration, $t+1$, because the unit's model vector, $m_c(t+1)$, is now nearer to the input pattern x in terms of the input space. Apart from the winner, adaptation is performed with neighboring units, too. Units that are subject to adaptation are depicted as shaded units in the figure. The shading of the various units corresponds to the amount of adaptation, and thus, to the spatial width of the neighborhood-kernel. Generally, units in close vicinity of the winner are adapted more strongly, and consequently, they are depicted with a darker shade in the figure.

Being a decidedly stable and flexible model, the SOM has been employed in a wide range of applications, ranging from financial data analysis, via medical data analysis, to time series prediction, industrial control, and many more [5, 13, 32]. It basically offers itself to the organization and interactive exploration of high-dimensional data spaces. One of its most prominent application areas is the organization of large text archives [15, 19, 29], which, due to numerous computational optimizations and shortcuts that are possible in this NN model, scale up to millions of documents [11, 14].

However, due to its topological characteristics, the SOM not only serves as the basis for interactive exploration, but may also be used as an index structure to high-dimensional databases, facilitating scalable proximity searches. Reports on a combination of $SOMs$ and R*-trees as an index to image databases have been reported, for example, in [20, 21], whereas an index tree based on the SOM is reported in [36]. Thus, the SOM combines and offers itself in a convenient way both for interactive exploration, as well as for the indexing and retrieval, of information represented in the form of high-dimensional feature spaces, where exact matches are either impossible due to the fuzzy nature of data representation or the respective type of query, or at least computationally prohibitive, making them particularly suitable for image or music databases.

4.2 The GHSOM

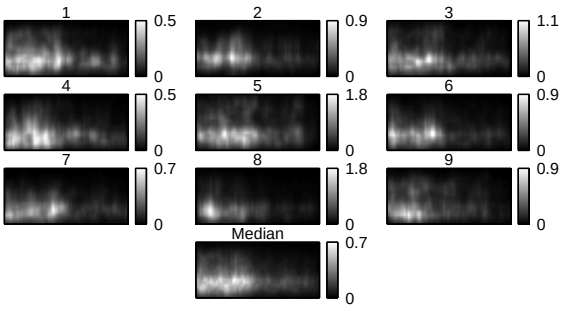
The key idea of the *Growing Hierarchical Self-Organizing Map* [6] is to use a hierarchical structure of multiple layers, where each layer consists of a number of independent $SOMs$. One SOM is used at the first layer of the hierarchy, representing the respective data in more detail. For every unit in this map a SOM might be added to the next layer of the hierarchy. This principle is repeated with the third and any further layers of the $GHSOM$.

Since one of the shortcomings of SOM usage is its fixed network architecture we rather use an incrementally growing version of the SOM. This relieves us from the burden of predefining the network's size which is rather determined during the unsupervised training process. We start with a layer 0, which consists of only one single unit. The weight vector of this unit is initialized as the average of all input data. The training process basically starts with a small map of, say, 2×2 units in layer 1, which is self-organized according to the standard SOM training algorithm.

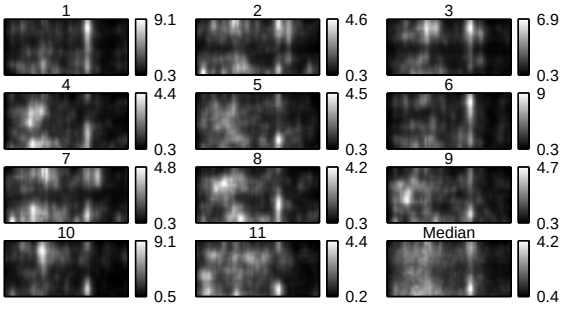
This training process is repeated for a fixed number λ of training iterations. Ever after λ training iterations the unit with the largest deviation between its weight vector and the input vectors represented by this very unit is selected as the error unit. In between the error unit and its most dissimilar neighbor in terms of the input space either a new row or a new column of units is inserted. The weight vectors of these new units are initialized as the average of their neighbors.

An obvious criterion to guide the training process is the *quantization error* q_i, calculated as the sum of the distances between the weight vector of a unit i and the input vectors mapped onto this unit. It is used to evaluate the mapping quality of a SOM based on the *mean quantization error* (MQE) of all units in the map. A map grows until its MQE falls below a certain fraction τ_1 of the q_i of the unit i in the preceding layer of the hierarchy. Thus, the map now represents the data of the higher layer unit i in more detail.

As outlined above the initial architecture of the $GHSOM$ consists of one SOM. This architecture is expanded by another layer in case of dissimilar input data being mapped on a particular unit. These units are identified by a rather high quantization error q_i which is above a threshold τ_2. This threshold basically indicates the desired granularity level of data representation as a fraction of the initial quantization error at layer 0. In such a case, a new map will be added to the hierarchy and the input data mapped on the respective higher layer unit are self-organized in this new map, which again grows until its MQE is reduced to a fraction τ_1 of the respective higher layer unit's quantization error q_i. Note that this does not necessarily lead to a balanced hierarchy. The depth of the hierarchy will rather reflect the diversity in input data distribution which should be expected in real-world data collections. Depending on the desired fraction τ_1 of MQE reduction we may end up with either a very deep hierarchy with small maps, a flat structure with large maps, or – in the extreme case – only one large map. The growth of the hierarchy is terminated when no further units are available for expansion.



(a) Beethoven, Für Elise



(b) Korn, Freak on a Leash

Figure 6: The rhythm patterns of *Beethoven, Für Elise* and *Korn, Freak on a Leash* and their medians

A graphical representation of a $GHSOM$ is given in Figure 5. The map in layer 1 consists of 3×2 units and provides a rough organization of the main clusters in the input data. The six independent maps in the second layer offer a more detailed view on the data. Two units from one of the second layer maps have further been expanded into third-layer maps to provide sufficiently granular input data representation. By using a proper initialization of the maps added at each layer in the hierarchy based on the parent unit's neighbors, a global orientation of the newly added maps can be reached [7]. Thus, similar data will be found on adjoining borders of neighboring maps in the hierarchy.

4.3 Cluster analysis of music data

The feature vectors extracted according to the process described in Section 3 are used as input to the $GHSOM$. However, some further intermediary processing steps may be applied in order to obtain feature vectors for pieces of music, rather than music segments, as well as to, optionally, compress the dimensionality of the feature space as follows.

(**A1**) Basically, each segment of music may be treated as an independent piece of music, thus allowing multiple assignment of a given piece of music to multiple clusters of varying style if a piece of music contains passages that may be attributed to different genres. Also, a two-level clustering procedure may be applied to first group the segments according to their overall similarity. In a second step, the distribution of segments across clusters may be used as a kind of *finger print* to describe the characteristics of the whole piece of music, using the resulting distribution vectors as an input to the second-level clustering procedure [26].

On the other hand, our research has shown, that simply using the median of all segment vectors belonging to a given piece of music, results in a stable representation of the characteristics of this piece of music. We have evaluated several alternatives using Gaussian mixture models, fuzzy c-means, and k-means pursuing the assumption that a piece of music contains significantly different rhythm patterns. However, the median, despite being by far the simplest technique, yielded comparable results to the more complex methods. Other simple alternatives such as the the mean proved to be too vulnerable with respect to outliers.

The rhythm patterns of all 6-second sequences extracted from *Für Elise* and from *Freak on a Leash* as well as their medians are depicted in Figure 6. The vertical axis represents the critical-bands from *Bark* 1-20, the horizontal axis the modulation frequencies from 0-10Hz, where *Bark* 1 and 0Hz is located in the lower left corner. Generally, the patterns of one piece of music have common properties. While *Für Elise* is characterized by a rather horizontal shape with low values, *Freak on a Leash* has a characteristic vertical line around 7Hz. To capture these common characteristics within a piece of music the median is a suitable approach. The median of *Für Elise* indicates that there are common but weak activities in the range of 3-10 *Bark* with a modulation frequency of up to 5Hz. The single sequences of *Für Elise* have many more details, for example, the first sequence has a minor peak around 5 *Bark* and 5Hz modulation frequency. However, the main characteristics, e.g. the vertical line at 7Hz for *Freak on a Leash*, as well as the generic activity in the frequency bands are preserved.

(**A2**) Furthermore, the 1200-dimensional feature space may be compressed using Principle Component Analysis (PCA). Our experiments have shown that a reduction down to 80 dimensions may be performed without much loss in variance. Yet, for the experiments presented in this paper we use the uncompressed feature space.

(**A3**) Following these optional steps, a $GHSOM$ may be trained to obtain a hierarchical map interface to the music archive. Apart from obtaining hierarchical representations, the $GHSOM$ may also be applied to obtain flat maps similar to conventional $SOMs$, or grow linear tree structures.

(**Visualization**) The resulting maps offer themselves as interfaces to explore a music archive. Yet advanced cluster visualization techniques based on the SOM, such as the *U-Matrix* [34], may be used to assist in cluster identification. A specifically appealing visualization based on *smoothed data histograms (SDH)* [25] are the *Islands of Music*, which use the metaphor of geographical maps, where islands resemble styles of music, to provide an intuitive interface to music archives. Furthermore, attribute aggregates are used to create *Weather charts* that help the user to understand the sound characteristics of the various areas on the map. For a detailed discussion and evaluation of these visualizations, see [24].

5. EXPERIMENTS

In the following sections we present some experimental results of our system based on a music archive made up of MP3-compressed files of popular pieces of music from a variety of genres. Specifically, we present in more detail the organization of a small subset of the entire archive, consisting of 77 pieces of music, with a total playing time of about 5 hours, using the $GHSOM$. This subset, due to its limited size offers itself for detailed discussion. We furthermore present results using a larger collection of 359 pieces of music, with a total playing length of about 23 hours. In both cases, each piece is represented by 1200 features which describe the dynamics of the loudness in frequency bands. The experiments, including audio samples, are available for interactive exploration at the $SOMeJB$ project homepage at `http://www.ifs.tuwien.ac.at/~andi/somejb`.

5.1 A GHSOM of 77 pieces of music

Figure 7 depicts a $GHSOM$ trained on the music data. On the first level the training process has resulted in the creation of a 3×3

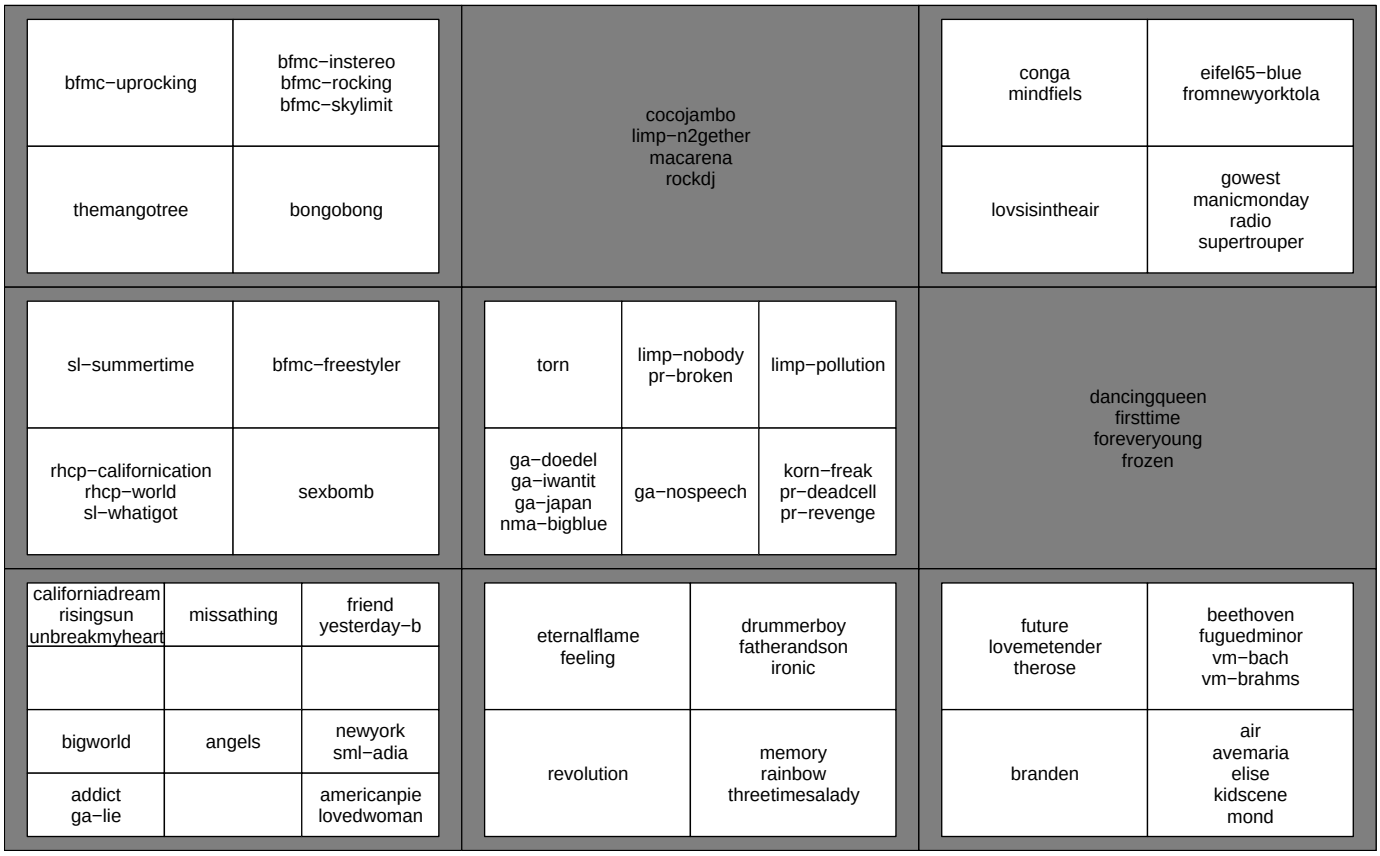


Figure 7: GHSOM of music collection

map, organizing the collection into 9 major styles of music. The bottom right represents mainly classical music, while the upper left mainly represents a mixture of Hip Hop, Electro, and House by *Bomfunk MCs (bfmc)*. The upper-right, center-right, and upper-center represent mainly disco music such as *Rock DJ* by *Robbie Williams (rockdj)*, *Blue* by *Eiffel 65 (eiffel65-blue)*, or *Frozen* by *Madonna (frozen)*. Please note, that the organization does not follow clean "conceptual" genre styles, splitting by definition, e.g. *HipHop* and *House*, but rather reflects the overall sound similarity.

Seven of these 9 first-level categories are further refined on the second level. For example, the bottom right unit representing classical music is divided into 4 further sub-categories. Of these 4 categories the lower-right represents slow and peaceful music, mainly piano pieces such as *Für Elise (elise)* and *Mondscheinsonate (mond)* by *Beethoven*, or *Fremde Länder und Menschen* by *Schumann (kidscene)*. The upper-right represents, for example, pieces by *Vanessa Mae* (vm), which, in this case, are more dynamic interpretations of classical pieces played on the violin. In the upper-left orchestral music is located such as the as the end credits of the film *Back to the Future III (future)* and the slow love song *The Rose* by *Bette Midler (therose)*, exhibiting a more intensive sound sensation, whereas the lower right corner unit represents the *Brandenburg Concerts* by *Bach (branden)*.

Generally speaking, we find the softer, more peaceful songs on this second level map located in the lower half of the map, whereas the more dynamic, intensive songs are located in the upper half. This corresponds to the general organization of the map in the first layer, where the unit representing Classic music is located in the lower right corner, having more aggressive music as its upper and left neighbors. This allows us, even on lower-level maps, to move across map boundaries to find similar music on the neighboring map following the same general trends of organization, thus alleviating the common problem of cluster separation in hierarchical organizations.

Some interesting insights into the music collection which the *GHSOM* reveals are, for example, that the song *Freestyler* by *Bomfunk MCs* (center-left) is quite different then the other songs by the same group. *Freestyler* was the groups biggest hit so far and, unlike their other songs, has been appreciate by a broader audience. Generally, the pieces of one group have similar sound characteristics and thus are located within the same categories. This applies, for example, to the songs of *Guano Apes (ga)* and *Papa Roach (pr)*, which are located in the center of the 9 first-level categories together with other aggressive rock songs. However, another exception is *Living in a Lie* by *Guano Apes (ga-lie)*, located in the lower-left. Listening to this piece reveals, that it is much slower than the other pieces of the group, and that this song matches very well to, for example, *Addict* by *K's Choice*.

5.2 A GHSOM of 359 pieces of music

In this section we present results from using the *SOMeJB* system to structure a larger collection of 359 pieces of music. Due to space constraints we cannot display or discuss the full hierarchy in detail. We will thus pick a few examples to show the characteristics of the resulting hierarchy, inviting the reader to explore and evaluate the complete hierarchy via the project homepage.

The resulting *GHSOM* has grown to a size of 2×4 units on the top layer map. All 8 top-layer units were expanded onto a second layer in the hierarchy, from which 25 units out of 64 units total on this layer were further expanded into a third layer. None of the branches required expansion into a fourth layer at the required level-of-detail setting. An integrated view of the two top-layers of the map is depicted in Figure 8. We will now take a closer look at

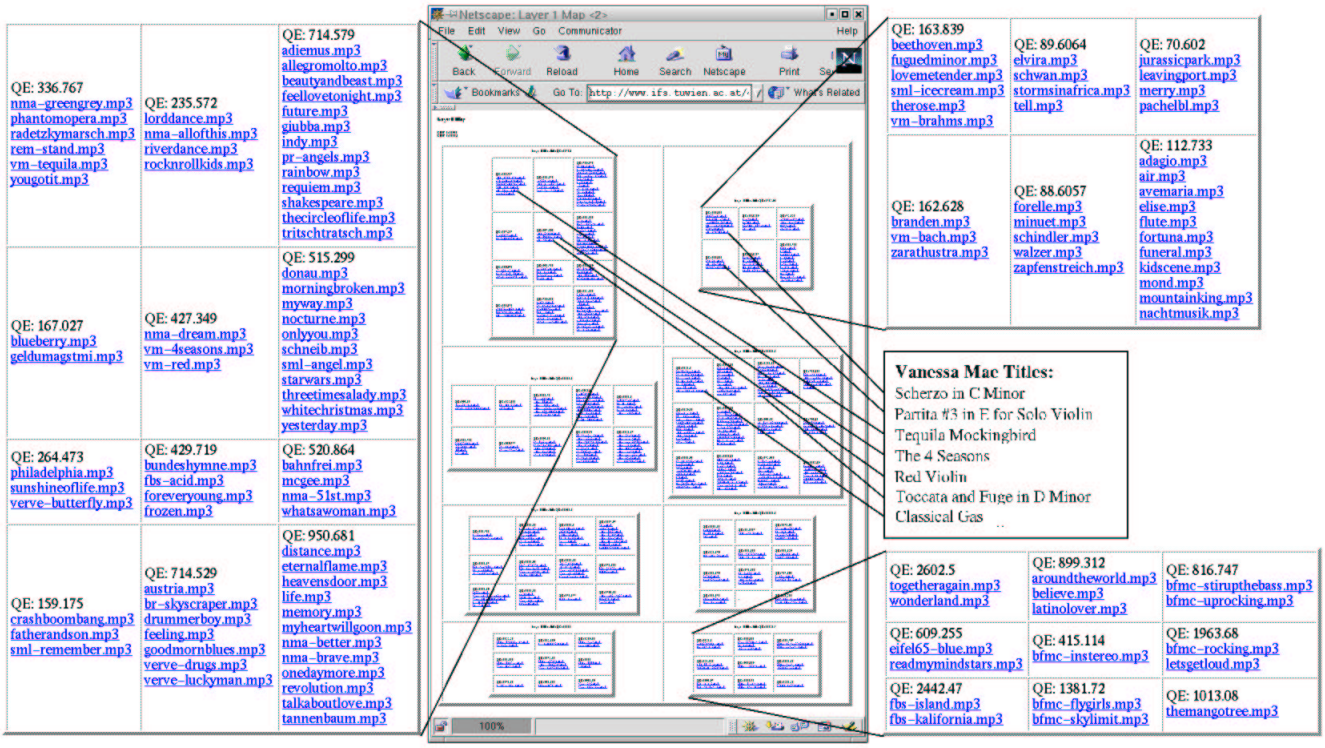


Figure 8: GHSOM of the larger music collection (359 pieces)

some branches of this map, and compare them to the respective areas in the *GHSOM* of the smaller data collection depicted in Figure 7

Generally, we find pieces of soft classical music in the upper right corner, with the music becoming gradually more dynamic and aggressive as we move towards the bottom left corner of the map. Due to the characteristics of the training process of the *GHSOM* we can find the same general tendency at the respective lower-layer maps. The overall orientation of the map hierarchy is rotated when compared to the smaller *GHSOM*, where the classical titles were located in the bottom right corner, with the more aggressive titles placed on the upper left area of the map. This rotation is due to the unsupervised nature of the *GHSOM* training process. It can, however, be avoided by using specific initialization techniques if a specific orientation of the map were required.

The unit in the upper right corner of the top-layer map, representing the softest classical pieces of music, is expanded onto a 3×2 map in the second layer (expanded to the upper right in Figure 8). Here we again find the softest, most peaceful pieces in the upper right corner, namely part of the sound-track of the movie *Jurassic Park*, next to *Leaving Port* by *James Horner*, *The Merry Peasants* by *Schumann*, and *Canon* by *Pachelbel*. Below this unit we find further soft titles, yet somewhat more dynamic. We basically find all titles that were mapped together in the bottom right corner unit of the *GHSOM* of the smaller collection depicted in Figure 7 on this unit, i.e. *Air*, *Ave Maria*, *Für Elise*, *Fremde Länder und Menschen (kidscene)* and the *Mondscheinsonate*. Furthermore, a few additional titles of the larger collection have been mapped onto this unit, the most famous of which probably are *Die kleine Nachtmusik* by *Mozart*, the *Funeral March* by *Chopin* or the *Adagio* from the Clarinet Concert by emphMozart.

Let us now take a look at the titles that were mapped onto the neighboring units in the previously presented smaller data collection. The *Brandenburgische Konzerte*, located on the neighboring unit to the right in the first example, can be found in the lower left corner of this map, together with, for example, *Also sprach Zarathustra* by *Richard Strauß*. Mapped onto the upper neighboring unit in the smaller *GHSOM* we had titles like the *First Movement of the 5th Symphony* by *Beethoven*, or the *Toccata and Fuge in D Minor* by *Bach*. We find these two titles in the upper left corner of the 2-layer map of this *GHSOM*, together with two of the three titles mapped onto the diagonally neighboring unit in the first *GHSOM*, i.e. *Love me Tender* by *Elvis Presley*, and *The Rose* by *Bette Midler*, which are again soft, mellow, but a bit more dynamic. The third title mapped onto this unit in the smaller *GHSOM*, i.e. the sound track of the movie *Back to the Future III* is not mapped into this branch of this *GHSOM* anymore. When we listen to this title we find it to have mainly strong orchestral parts, which have a different, more intense sound than the soft pieces mapped onto this branch, which is more specific with respect to very soft classical titles as more of them are available in the larger data collection. Instead, we can find this title on the upper right corner in the neighboring branch to the left, originating from the upper left corner unit of the top-layer map. There it is mapped together with *The Beauty and the Beast* and other orchestral pieces, such as *Allegro Molto* by *Brahms*. We thus find this branch of the *GHSOM* to be more or less identical to the overall organization of the smaller *GHSOM* in so far as the titles present in both collections are mapped in similar relative positions to each other.

Due to the topology preservation provided by the *GHSOM* we can move from the soft classical cluster map to the left to find somewhat more dynamic classical pieces of music on the neighboring map (expanded to the left in Figure 8). Thus, a typical disadvantage of hierarchical clustering and structuring of datasets, namely the fact that a cluster that might be considered conceptually very similar is subdivided into two distinct branches, is alleviated in the *GHSOM* concept, because these data points are typically located in the close neighborhood. We thus find, on the right border of the neighboring map, the more peaceful titles of this branch, yet more dynamic than the classical pieces on the neighboring right branch discussed above.

Rather than continuing to discuss the individual units we shall now take a look at the titles of a specific artist and its distribution in this hierarchy. In total, there are 7 titles by *Vanessa Mae* in this collection, all violin interpretations, yet of distinctly different style. Her most "conventional" classical interpretations, such as Brahm's *Scherzo in C Minor (vm-brahms)* or Bach's *Partita #3 in E for Solo Violin (vm-bach)* are located in the classic-cluster in the upper right corner branch on two neighboring units on the left side of the second-layer map. These are definitely the most "classical" of her interpretations in the given collection, yet exhibiting strong dynamics. Further 3 pieces of Vanessa Mae (*The 4 Seasons* by Vivaldi, *Red Violin* in its symphonic version, and *Tequila Mockingbird*) are found in the neighboring branch to the left, the former two mapped together with *Western Dream* by *New Model Army*. All of these titles are very dynamic violin pieces with strong orchestral parts and percussion.

When we look for the remaining 2 titles by *Vanessa Mae*, we find them on the unit expanded below the top right corner unit, thus also neighboring the classical cluster. On the top-left corner unit of this sub-map we find *Classical Gas*, which starts in a classical, symphonic version, and gradually has more intensive percussion being added, exhibiting a quite intense beat. Also on this map, on the one-but-next unit to the right, we find another interpretation of the *Toccata and Fuge in D Minor* by Bach, this time in the classical interpretation of *Vanessa Mae*, also with a very intense beat. The more "conventional" organ interpretation of this title, as we have seen, is located in the classic cluster discussed before. Although both are the same titles, the interpretations are very different in their sound characteristic, with *Vanessa Mae*'s interpretation definitely being more pop-like than the typical classical interpretation of this title. Thus, two identical titles, yet played in different styles, end up in their respective stylistic branches of the *SOMeJB* system. We furthermore find, that the system does not organize all titles by a single artist into the same branch, but actually assigns them according to their sound characteristics, which makes it particularly suitable for localizing pieces according to ones likings independent of the typical assignment of an artist to any category, or to the conventional assignment of titles to specific genres.

In spite of these desired characteristics, however, several weaknesses remain, especially when titles, that may be very similar in terms of their beat characteristics in the various frequency bands, are mapped together, yet derive from very different genres and are immediately associated with those genres. This refers, for example, to titles where the language is a specific characteristic, such as several German-language songs in our collection. Furthermore, in some cases like the previously-mentioned *Western Dream* by *New Model Army*, which is mapped together with titles by *Vanessa Mae*, the rhythmic properties might be similar, yet the perceived sound is still distinctively different because of the strong vocal parts. Even if the acoustic background shares some similarities over long distances of the title, the rhythmic vocal parts are perceived much stronger. This points towards the necessity to incorporate additional features to better capture sound characteristics. Furthermore, in some cases like these it might be advisable to use the two-stage clustering approach outlined in [26], as for some titles the variance of sound characteristics of segments is rather large. When taking a look at the mapping of the respective segments of *Western Dream* in another experiment we find 3 segments of it to be located in a more classical sub-branch, whereas the other segments are located in the more dynamic, aggressive branches of the hierarchy.

Further units depicted in more detail in Figure 8 are the bottom right unit representing the more aggressive, dynamic titles. We leave it to the reader to analyze this sub-map and compare the titles with the ones mapped onto the upper left corner map in Figure 7.

6. CONCLUSIONS

We have presented the *SOM-enhanced Jukebox (SOMeJB)*, a system for content-based organization and visualization of music archives. Given pieces of music in raw audio format a hierarchical organization is created where music of similar sound characteristics is mapped together. Our system thus enables a user to browse through the archive, searching for music representing a particular style, without relying on manual genre classification.

Rhythm patterns in various frequency bands are extracted and used as a descriptor of perceived sound similarity, incorporating psychoacoustic models during the feature extraction stage. The *GHSOM* automatically identifies the inherent structure of the music collection and offers an intuitive interface for genre browsing. Furthermore, by mapping a piece of music representing a "query" onto the map structure, the user is pointed to a location within the map hierarchy, where he or she will find similar pieces of music. We evaluated our approach using a collection of about 23 hours of music and obtained encouraging results. Future work will mainly deal with improving the feature extraction process. While the presented features offer a simple but powerful way of describing the music, additional information is required to better capture sound characteristics that go beyond frequency-specific beat patterns, focusing e.g. on the timbre and instrumentation. Furthermore, more abstract features are necessary to explain the organization principles to the user.

While the current evaluation allows for an intuitive analysis of the system's performance, a more formal evaluation is desired. We thus plan to perform a user study allowing us to evaluate both users' expectations towards such a system as well as to obtain feedback on the perceived quality of the current approach.

7. ACKNOWLEDGMENTS

Part of this research has been carried out in the project Y99-INF, sponsored by the Austrian Federal Ministry of Education, Science and Culture (BMBWK) in the form of a START Research Prize. The BMBWK also provides financial support to the Austrian Research Institute for Artificial Intelligence. The authors wish to thank Simon Dixon, Markus Frühwirth, and Werner Göbel for valuable discussions and contributions.

8. REFERENCES

[1] D. Bainbridge, C. Nevill-Manning, H. Witten, L. Smith, and R. McNab. Towards a digital library of popular music. In *Proc of the ACM Conf on Digital Libraries (ACMDL'99)*, pages 161–169, Berkeley, CA, August 11-14 1999. ACM.

[2] W. Birmingham, R. Dannenberg, G. Wakefield, M. Bartsch, D. Bykowski, D. Mazzoni, C. Meek, M. Mellody, and W. Rand. MUSART: Music retrieval via aural queries. In *Proc of the Annual Symposium on Music Information Retrieval (ISMIR 2001)*, Bloomington, ID, October 15-17 2001.

[3] R. Bladon. Modeling the judgement of vowel quality differences. *Journal of the Acoustical Society of America*, 69:1414–1422, 1981.

[4] R.B. Dannenberg, B. Thom, and D. Watson. A Machine Learning Approach to Musical Style Recognition. In *Proc. of the Int'l Computer and Music Conf (ICMC97)*, pages 344–347, Thessaloniki, Greece, Sept. 25-30 1997.

[5] G. DeBoeck and T. Kohonen, editors. *Visual Explorations in Finance*. Springer Verlag, Berlin, Germany, 1998.

[6] M. Dittenbach, D. Merkl, and A. Rauber. The growing hierarchical self-organizing map. In *Proc of the Int'l Joint Conf on Neural Networks (IJCNN 2000)*, pages 15 – 19, Como, Italy, July 24-27 2000. IEEE Computer Society.

[7] M. Dittenbach, A. Rauber, and D. Merkl. Recent advances with the growing hierarchical self-organizing map. In *Proc of the Workshop on Self-Organizing Maps*, Advances in Self-Organizing Maps, pages 140–145, Lincoln, England, June 13-15 2001. Springer.

[8] B. Feiten and S. Günzel. Automatic indexing of a sound database using self-organizing neural nets. *Computer Music Journal*, 18(3):53–65, 1994.

[9] J. Foote. An overview of audio information retrieval. *Multimedia Systems*, 7(1):2–10, 1999.

[10] A. Ghias, J. Logan, D. Chamberlin, and S. B.C. Query by humming: Musical information retrieval in an audio database. In *Proc of the ACM Int'l Conf on Multimedia*, pages 231–236, San Francisco, CA, November 5 - 9 1995. ACM.

[11] S. Kaski. Fast winner search for SOM-based monitoring and retrieval of high-dimensional data. In *Proc of the Int'l Conf on Artificial Neural Networks (ICANN99)*, pages 940–945. IEE, September, 7.-10. 1999.

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

[13] T. Kohonen. *Self-organizing maps*. Springer-Verlag, Berlin, 1995.

[14] T. Kohonen. Self-organization of very large document collections: State of the art. In *Proc of the Int'l Conf on Artificial Neural Networks*, pages 65–74, Skövde, Sweden, 1998.

[15] T. Kohonen, S. Kaski, K. Lagus, J. Salojärvi, J. Honkela, V. Paatero, and A. Saarela. Self-organization of a massive document collection. *IEEE Transactions on Neural Networks*, 11(3):574–585, May 2000.

[16] N. Kosugi, Y. Nishihara, T. Sakata, M. Yamamuro, and K. Kushima. A practical query-by-humming system for a large music database. In *Proc of the ACM Int'l Conf on Multimedia*, pages 333–342, Marina del Ray, CA, 2000. ACM.

[17] C. Liu and P. Tsai. Content-based retrieval of mp3 music objects. In *Proc of the Int'l Conf on Information and Knowledge Management (CIKM 2001)*, pages 506 – 511, Atlanta, Georgia, 2001. ACM.

[18] M. Liu and C. Wan. A study of content-based classification and retrieval of audio database. In *Proc of the Int'l Database Engineering and Applications Symposium (IDEAS 2001)*, Grenoble, France, 2001. IEEE.

[19] D. Merkl and A. Rauber. Document classification with unsupervised neural networks. In F. Crestani and G. Pasi, editors, *Soft Computing in Information Retrieval*, pages 102–121. Physica Verlag, 2000.

[20] K. Oh, Y. Feng, K. Kaneko, A. Makinouchi, and S. Bae. SOM-based R*-tree for similarity retrieval. In *Proc of the Int'l Conf on Database Systems for Advanced Applications*, pages 182–189, Hong-Kong, China, April 18-21 2001. IEEE.

[21] K. Oh, K. Kaneko, and A. Makinouchi. Image classification and retrieval based on wavelet-som. In *Int'l Symposium on Database Applications in Non-Traditional Environments (DANTE'99)*, pages 164–167, Kyoto, Japan, November 28-30 1999. IEEE.

[22] F. Pachet and D. Cazaly. A taxonomy of musical genres. In *Proc of the Int'l Conf on Content-Based Multimedia Information Access (RIAO 2000)*, Paris, France, 2000.

[23] E. Pampalk *Islands of Musik: Analysis, Organization, and Visualization of Music Archives*. Master's thesis, Vienna University of Technology, 2001.

[24] E. Pampalk, A. Rauber, and D. Merkl. Content-based organization and visualization of music archives. In *Proc of ACM Multimedia 2002*, Juan-les-Pins, France, December 1-6 2002. ACM.

[25] E. Pampalk, A. Rauber, and D. Merkl. Using smoothed data histograms for cluster visualization in self-organizing maps. In *Proc of the Int'l Conf on Neural Networks (ICANN 2002)*, Madrid, Spain, August 27-30 2002. Springer.

[26] A. Rauber and M. Frühwirth. Automatically analyzing and organizing music archives. In *Proc of the European Conf on Research and Advanced Technology for Digital Libraries (ECDL 2001)*, Darmstadt, Germany, Sept. 4-8 2001. Springer.

[27] B. Ripley. *Pattern Recognition and Neural Networks*. Cambridge University Press, Cambridge, UK, 1996.

[28] J. Rolland, G. Raskinis, and J. Ganascia. Musical content-based retrieval: An overview of the Melodiscov approach and system. In *Proc of the ACM Int'l Conf on Multimedia*, pages 81–84, Orlando, FL, 1999. ACM.

[29] D. Roussinov and H. Chen. Information navigation on the web by clustering and summarizing query results. *Information Processing and Management*, 37:789 – 816, 2001.

[30] E. Scheirer. *Music-Listening Systems*. PhD thesis, MIT Media Laboratory, 2000.

[31] M. Schröder, B. Atal, and J. Hall. Optimizing digital speech coders by exploiting masking properties of the human ear. *Journal of the Acoustical Society of America*, 66:1647–1652, 1979.

[32] O. Simula, P. Vasara, J. Vesanto, and R. Helminen. The self-organizing map in industry analysis. In L. Jain and V. Vemuri, editors, *Industrial Applications of Neural Networks*, Washington, DC., 1999. CRC Press.

[33] G. Tzanetakis, G. Essl, and P. Cook. Automatic musical genre classification of audio signals. In *Proc Int'l Symposium on Music Information Retrieval (ISMIR)*, Bloomington, Indiana, October 15-17 2001.

[34] A. Ultsch and H. Siemon. Kohonen's self-organizing feature maps for exploratory data analysis. In *Proc of the Int'l Neural Network Conf (INNC'90)*, pages 305–308, Dordrecht, Netherlands, 1990. Kluwer.

[35] 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.

[36] H. Zhang and D. Zhong. A scheme for visual feature based image indexing. In *Proc of the IS&T/SPIE Conf on Storage and Retrieval for Image and Video Databases*, pages 36–46, San Jose, CA, February 4-10 1995.

[37] E. Zwicker and H. Fastl. *Psychoacoustics, Facts and Models*, volume 22 of *Series of Information Sciences*. Springer, Berlin, 2. edition, 1999.