Eurographics/ IEEE-VGTC Symposium on Visualization (2007)
Ken Museth, Torsten Möller, and Anders Ynnerman (Editors)

The CoMIRVA Toolkit for Visualizing Music-Related Data

Markus Schedl Peter Knees Klaus Seyerlehner Tim Pohle

Department of Computational Perception
Johannes Kepler University Linz, Austria
http://www.cp.jku.at

Abstract

We present CoMIRVA*, which is an abbreviation for* Collection of Music Information Retrieval and Visualization Applications*. CoMIRVA is a Java framework and toolkit for information retrieval and visualization. It is licensed under the GNU GPL and can be downloaded from* http://www.cp.jku.at/comirva/*. At the moment, the main functionalities include music information retrieval, web retrieval, and visualization of the extracted information. In this paper, we focus on the visualization aspects of CoMIRVA. Since many of the information retrieval functions are intended to be applied to problems of the field of music information retrieval (MIR), we demonstrate the functions using data like similarity matrices of music artists gained by analyzing artist-related web pages. CoMIRVA is continuously being extended. Currently, it supports the following visualization techniques: Self-Organizing Map, Smoothed Data Histogram, Circled Bars, Circled Fans, Probabilistic Network, Continuous Similarity Ring, Sunburst, and Music Description Map. Since space is limited, we can only present a selected number of these in this paper. As one key feature of CoMIRVA is its easy extensibility, we further elaborate on how CoMIRVA was used for creating a novel user interface to digital music repositories.*

Categories and Subject Descriptors (according to ACM CCS): I.3.4 [Computer Graphics]: Graphics Utilities

Keywords: Information Visualization Toolkit, Web Mining, Music Information Retrieval, User Interface, Java

1. Introduction and Context

Music information retrieval (MIR) has become an important field of research over the past few years. Its main concern is the extraction, analysis, and representation of information that describes various aspects of music. This information can be gained basically from three kinds of sources:

- audio signal of digital music files
- metadata provided by music distributors
- information extracted from the web

While descriptive high-level features based on signal analysis usually describe properties like rhythmical structure, e.g. [Pam01, DPW03], or timbral aspects, e.g. [AP02, Log00], of a piece of music, features based on metadata incorporate, for example, ID3-tags. Information extracted from artists' web pages or gathered by collaborative filtering, in contrast, not only relies on expert opinions like metadata provided by music distributors, but reflects a kind of cultural or community knowledge because it incorporates the opinions of a large number of people, e.g. [WL02, KPW04, KSW05, SKW05a].

Important aspects of MIR include the usage and representation of the extracted information. Usually, the features are used to derive similarities between music artists or pieces of music, which is one of the most important tasks in MIR and is supported by CoMIRVA with a wide variety of functions. Since building MIR applications (e.g. recommender systems, metadata-enriched music players, or intelligent UIs to music repositories) from scratch requires extensive knowledge and is very time-consuming, we have developed CoMIRVA. The aim is to provide a fully functional system for music information retrieval and visualization, which can be used as a toolkit and a framework for building specialized applications like the one presented in Section 5. To achieve this twofold purpose, CoMIRVA is based on an object-oriented design concept, which also facilitates its extensibility.

1.1. Related Systems

There exist several systems for performing MIR and InfoVis tasks independently. However, all of them have more or less serious drawbacks when it comes to combining tasks related to both MIR and InfoVis.

1.1.1. All-in-one Solutions

Professional all-in-one solutions like *Matlab*® can be applied for rapid prototyping (of MIR as well as of InfoVis applications), but suffer from the drawback of a very high price and an even higher complexity. As an alternative to *Matlab*®, a few systems have been developed under the GNU General Public License, for example, *Octave* (`http://www.octave.org/`). However, with respect to visualization functionality and usability, they cannot compete with *Matlab*®.

1.1.2. MIR Toolkits

In regard to their feature extraction functionality, MIR toolkits usually only provide audio-based signal extraction. One of the most popular MIR toolkits is *Marsyas* (`http://opihi.cs.uvic.ca/marsyas/`, [TC00]), which is a framework for rapid prototyping and experimentation and provides functions for retrieval, analysis, and synthesis of audio signals. *jAudio* [MFD05], as part of the *ACE* project (`http://coltrane.music.mcgill.ca/ACE/`), is another signal-based feature extraction tool. A third popular framework that focus on research and application development in the audio and music domain is *CLAM* (`http://www.iua.upf.es/mtg/clam/`).

1.1.3. InfoVis Toolkits

Among the toolkits related to information visualization, the most popular ones include *Pad++* (`http://www.cs.umd.edu/hcil/pad++/`, [BHP*96]), *Piccolo* (`http://www.cs.umd.edu/hcil/piccolo/`, [BGM04]), formerly known as *Jazz* [BMG00], and the *InfoVis Toolkit* (`http://ivtk.sourceforge.net/`, [Fek04]). Pad++ is mainly a toolkit for creating zoomable user interfaces and has been used, for example, to develop InfoVis applications like zoomable web browsers or image browsers. Piccolo is a quite popular 2D graphics framework for developing graphical applications in Java and C#. The InfoVis Toolkit is a relatively recent development which not only provides a set of different visualizations, but also special data structures that are well-suited for a number of information visualization tasks.

1.1.4. Motivation for CoMIRVA

The main motivation for starting to develop CoMIRVA was the high price and very inefficient memory management of *Matlab*® which we used before for music information retrieval and visualization tasks. Furthermore, we wanted to provide an easy to use environment that even novices without programming skills can use.

CoMIRVA is novel in two regards. First, we are not aware of any solution that combines MIR and visualization techniques within one framework. Indeed, CoMIRVA's visualization functions are suited to fulfill the special requirements of tasks related to MIR, for example, special data structures like similarity matrices. Second, to the best of our knowledge, we do not know any MIR system that includes web mining functionalities.

2. General Overview of CoMIRVA

CoMIRVA was designed using the object-oriented design paradigm and was implemented in Java due to its platform independence and its good support for network access which is obviously crucial for web mining. As CoMIRVA is open source, also extensibility was an important requirement.
Since CoMIRVA is not only a framework of MIR and InfoVis algorithms, but also intends to serve as a toolkit, it offers a GUI through which most of the functions provided by the framework are accessible.

2.1. Data Types and Data I/O

CoMIRVA basically provides two data types: *data matrices* and *meta-data vectors*. Data matrices are arbitrarily sized $m \times n$ matrices of double precision values, meta-data vectors are ordered lists of strings that usually describe the rows or columns of a data matrix. Both data matrices and meta-data vectors can be assigned a name under which they are displayed in separate lists in the right part of the GUI (cf. Figure 1)[†]. Naturally, the user can also rename each data item. As a matter of course, CoMIRVA supports loading and saving of data matrices and meta-data vectors from/to standard text files. Moreover, CoMIRVA also provides saving and loading of *workspaces*, i.e. collections of data matrices and meta-data vectors. This facilitates easy restoring of associated data.

2.2. Data Manipulation

As for data manipulation, the current version of CoMIRVA provides *normalization* of data matrices as well as a function which we call *vectorization*. Normalization can be performed linearly or logarithmically to a range whose boundaries are defined by the user (by default to [0,1]). Furthermore, the user can set the scope of the normalization. This determines whether the minimum and maximum values of the complete matrix are mapped to those given by the normalization boundaries, or the minima and maxima are determined for every row or every column separately and therefore also mapped independently row- or column-wise to the defined boundaries. Since we often work with matrices that

[†] Larger versions of the screenshots depicted in this paper are available at `http://www.cp.jku.at/comirva/`.

indicate similarities or distances between each pair of a number of entities, e.g. music artists, normalization in the scope of (independent) similarity vectors (matrix by row or column) is useful. Another task we often have to perform is the decomposition of a similarity matrix into its single similarity vectors. For this purpose, a function which we call vectorization is provided by CoMIRVA. This vectorization can be performed by row or by column and creates as many new 1-dimensional data matrices as rows or columns are present in the original data matrix. As a further convenient feature of the vectorization, each newly created data vector is named after the original data matrix and the description of the respective row or column as given by the meta-data vector (if one is selected). As a result, each similarity vector can easily be assigned a context.

3. Music Information Retrieval

As already mentioned, information about music artists or pieces of music can be gained from the audio signal or from metadata provided by the music distributor or extracted from the web. Since web mining techniques have successfully been applied to MIR problems like similarity measurement [WL02], artist-to-genre classification [KPW04, SKW05a] or automatic lyrics detection [KSW05] and we actively participate in this field of research, we have implemented some simple web mining functionalities. Furthermore, CoMIRVA offers state of the art functions for feature extraction based on the audio signal. The provided functionalities for both types of information retrieval approaches are presented in the following.

3.1. Co-occurrence Analysis

Performing co-occurrence analysis of artist names on web pages was proposed in [ZF04] for finding related artists to a given seed artist and in [SKW05a] for deriving complete similarity matrices which were used for artist-to-genre classification. Given a list of artist names (or arbitrary other entities), CoMIRVA provides a function that uses an arbitrary search engine to estimate the number of web pages containing each artist and each pair of artists. These page counts are inserted in a (symmetric) page count matrix, which is added to CoMIRVA's GUI after the process has finished. Based on such a page count matrix, CoMIRVA can calculate a conditional probability matrix that estimates the probability for the name of an artist (or another entity) A to occur on a web page which is known to mention another artist (or entity) B. This probability matrix represents a similarity matrix which can be used for a wide variety of applications in MIR such as prototypical artist detection [SKW05b,SKW05c] or artist-to-genre classification, e.g. [SKW05a].

3.2. Term Profile Creation

The simple co-occurrence analysis as described above does not take the content of web pages into account as it relies only on the page counts provided by the search engine. However, it is often desirable to analyze the content of web documents. To this end, CoMIRVA offers functions for automatic retrieval of web documents (HTML files), extraction of terms from these (or from other text documents), and calculation of some measures used in text retrieval. We have implemented a special data structure called *Entity Term Profile (ETP)* that uses XML to describe the content of a single document or a set of documents. More precisely, such an ETP contains a list of terms that were automatically extracted from the document(s) as well as the paths to the document(s), which are necessary for using ETPs in interfaces for document search, like our Sunburst implementation (cf. Section 4.3.1). In the case of an ETP describing a set of documents instead of a single one, term occurrences, term frequencies, document frequencies, and the well-established $TF \times IDF$ (term frequency × inverse document frequency) values [SB88] are stored additionally.

3.3. Audio-based Features

Features derived from the audio signal of a piece of music range from very simple low-level properties like zero crossing rate, spectral centroid, or spectral flux to sophisticated high-level descriptors that model the rhythmical or timbral structure of a piece. We have integrated some of the most successful high-level features in CoMIRVA.

The rhythm-based *Fluctuation Patterns* were first presented in [Pam01]. They model the periodicity of the audio signal for a number of critical frequency bands (according to the bark scale) and periodicity intervals (in beats per minute). The outcome of a Fluctuation Pattern calculation on a piece of music is a feature vector whose dimensionality depends on the number of bark intervals and periodicity intervals. To use a set of such feature vectors for defining similarities between pieces of music, e.g. the Euclidean distances between the feature vectors must be calculated.

Furthermore, two different feature extraction algorithms that are based on *Mel Frequency Cepstral Coefficients (MFCCs)* are implemented in CoMIRVA. MFCCs give a coarse description of the envelope of the frequency spectrum and thus, model timbral properties of a piece of music. Since MFCCs are calculated on time invariant frames of the audio signal, usually *Gaussian Mixture Models (GMMs)* are used to model the MFCC distributions of a whole piece of music. Similarity between two pieces of music A and B is then derived by drawing a sample from A's GMM and estimating the probability that this sample was created by B's GMM.

CoMIRVA offers two MFCC-based similarity measures. The first one is described in [APS05] and is called *Aucouturier and Pachet* in CoMIRVA, the second corresponds to [ME05] (called *Mandel and Ellis*). The measures basically differ in terms of the number and type of GMMs and in calculation time.

Given a directory, CoMIRVA recursively searches for MP3 files and calculates the requested audio features.

4. Information Visualization

The implemented functions for information visualization can be categorized according to the type of input data they use. We differentiate between algorithms that work on feature data, those working on similarity matrices or similarity vectors, and those working on special data structures, like term occurrence matrices in the case of our Sunburst visualization of ETPs.

Each visualization provided by CoMIRVA is implemented in its own class, but has to be connected to an instance of the class *VisuPane* which is responsible for double buffering and serves as interface between the individual visualizations and CoMIRVA's GUI. The visualization classes also implement a mouse listener if user interaction is desired, e.g. in the Circled Fans or Sunburst interface.

To each visualization, a *colormap*, i.e. a mapping from a range of values to a range of colors, can be applied. Furthermore, visualizations can be saved as PNG or JPG files for later use. An extension to create EPS output has also been implemented recently by one of our students.

4.1. Visualizations of Feature Data

This kind of data usually represents high-dimensional high-level descriptors. In the case of MIR data, these might be rhythmic or timbral properties of music, cf. Section 3.3.

4.1.1. Self-Organizing Map (SOM)

The *Self-Organizing Map (SOM)*, e.g. [Koh82, Koh01], is a well-established unsupervised neural network that aims at clustering high-dimensional data items in a usually 2- or 3-dimensional space such that similar data items are mapped to similar regions of the target space. CoMIRVA currently supports four different initialization methods: Random, Gradient, Linear [Koh01], and SLC [SLC02]. Furthermore, the size of the SOM grid and the training length can be adjusted by the user or determined automatically using a simple heuristic. Sequential (online) training is supported as well as batch training.

As for the visualization of a SOM grid, after a SOM has been trained, each data item is mapped to the map unit that best represents it. This unit is called the *best matching unit (BMU)*. Determining the BMU for every data item and drawing the SOM grid and the names of the data items on their respective BMU yields visualizations like the one in Figure 1 (without the colorful cluster visualization). This figure shows a SOM trained on web features of music artists. The upper left regions of the SOM contain mainly artists that create quite aggressive music. In the lower right, a peninsula with electronic music can be found. The other artists are mostly mapped to the big islands in the lower left.

4.1.2. Smoothed Data Histogram (SDH)

A visualization approach that emphasizes the data clusters of a SOM is the *Smoothed Data Histogram (SDH)*, proposed in [PRM02]. An SDH estimates the density of the data items over the map. To this end, each data item votes for a fixed number of best matching map units. The selected units are weighted according to the quality of the matching. The votes are accumulated in a matrix describing the distribution over the complete map. After each piece of music has voted, the resulting matrix is interpolated in order to obtain a smooth visualization. Finally, the interpolated matrix is visualized by applying a colormap. An example of an SDH visualization can be found in Figure 1, where the colormap *Islands* was applied to give the impression that clusters of similar artists form islands which rise from the blue sea (the sparse areas of the SDH).

4.2. Visualizations of Similarity Vectors and Matrices

Similarity vectors describe how similar a number of items (e.g. music artists) are to a given one. Similarity matrices indicate the similarity between all pairs of items of a given item set. In the following, we present some visualizations provided by CoMIRVA that help the user to find music artists or pieces of music which are similar to a given one.

4.2.1. Circled Bars

The *Circled Bars* visualization approach offers a simple method to answer questions like: "Which artists produce similar music to that of my favorite artist *A*?". It thus takes a similarity vector as input. Given a seed artist *A*, an adjustable number of most similar artists (according to the used similarity measure) are arranged in a circle. The artists are ordered by their similarity to artist *A*. The similarity values are visualized by filled arcs that vary in length and color corresponding to the applied colormap. Figure 2 shows a sample visualization with artists similar to the Metal band *Stratovarius*. For this figure, the Circled Bars visualization was generated from co-occurrences (cf. Section 3.1), and the colormap *Fire* was applied. Hence, the values in parentheses after the artist names indicate the probability for the respective artist to be found on a web page that is known to mention the seed artist *A*. Since the Circled Bars visualization does not require high computing or graphics capabilities, it may serve as a user interface for small devices with limited screen size, like mobile phones or personal digital assistants.

4.2.2. Circled Fans

The *Circled Fans* visualization is a conceptual extension of the simple Circled Bars. While the Circled Bars only take the nearest neighbors of a given seed artist (or any other entity) into account, the Circled Fans incorporate similarities in a transitive manner.

Given a seed artist *A* whose name is displayed in the center of the visualization, an adjustable number of most similar artists are arranged in a circle around *A* and connected to *A* by edges whose thickness and color correspond to the similarities given by the similarity matrix and the chosen colormap. The thicker the connecting edge, the more similar two artists are. Subsequently, for each of the similar artists of *A*, again, the most similar ones are selected, arranged in a circular arc whose center is the respective parent node, and connected to this parent node by an edge.

The user can adjust the maximum edge thickness, the maximum number of data items on the inner circle and on the outer circular arcs (which we call fans), as well as the angular extent of the fans. Moreover, the Circled Fans support user interaction by redrawing the visualization with a new seed artist *B* whenever the user clicks on the label of an arbitrary artist *B*.

In Figure 3, a screenshot of a Circled Fans visualization with the seed artist *Evanescence* is depicted. In this example, an asymmetric similarity matrix derived from co-occurrences (cf. Section 3.1) was used to define artist similarity, and the colormap *Colorful* was applied. Visualizing such asymmetric similarities is an important area of application of the Circled Fans. For example, the Circled Fans depicted in Figure 3 reveal that the band *Green Day* is mentioned on 53% of the web pages containing the word “Evanescence”, whereas “Evanescence” can only be found on 21% of the web pages that mention “Green Day”. Such information about similarity asymmetries can be used for measuring the popularity of an artist and further for determining which artists are prototypical for a specific genre [SKW05b].

4.2.3. Continuous Similarity Ring (CSR)

The *Continuous Similarity Ring (CSR)* visualization technique, which is described in detail in [SKW05b], uses a graph-based model for illustrating similarities between entities (e.g. music artists) by using one prototype for each of a number of given classes (e.g. music genres). Since prototypical artists are usually very well known, they can serve as reference points for finding similar, but less known artists, e.g. in online music stores.

Given a set of artists and information on which artist belongs to which genre, we determine a prototype for each genre and arrange these prototypes in a circle, cf. Figure 4. Since similar or related prototypes and the genres they represent should be placed close to each other, we formulate a *Traveling Salesman Problem* on the distance matrix generated from the prototypes’ similarity matrix and apply a simple heuristic algorithm. The resulting tour defines the arrangement of the prototypical artists within the circle of prototypes.

Additionally, for each prototypical artist, an adjustable number of its most similar neighbors (according to the used similarity matrix) are shown. To preserve the distances given by the similarity matrix, the neighbors are positioned using a cost-minimizing heuristic. The artists’ vertices are connected by edges whose thickness and color vary according to their similarity values and the colormap applied. For the visualization depicted in Figure 4, the colormap *Fire* was used.

4.3. Visualizations of Term Occurrence Matrices

As already mentioned in Section 3.2, CoMIRVA provides a data type called Entity Term Profile (ETP) which describes a set of documents by various properties relevant for information retrieval tasks. Among other data, an ETP contains a term occurrence matrix that indicates in which documents every term of a given term list occurs.

4.3.1. Sunburst

We use such term occurrence matrices to create a user interface that makes use of the well-established *Sunburst* visualization technique [AH98, SZ00].

Starting with the whole set of documents, more precisely, the ETP that describes this set, the terms with the highest document frequencies are selected and visualized as filled arcs around a centered circle (the root node) that represents the entire document collection. The size (angular extent) of each individual arc is proportional to the document frequency of the associated term, i.e. to the number of documents containing the term. Performing the term selection with respect to document frequencies recursively for all arcs eventually yields a complete Sunburst visualization. Internally, CoMIRVA stores the Sunburst as a tree. Every arc *A* represents a set of documents that contain the term associated with *A* and the terms associated with all arcs that must be traversed on the shortest way to the root node.

The user can define a number of stop criteria to limit the calculation time, the size of the Sunburst, and the number of recursions. CoMIRVA provides the following constraints: maximum sub nodes per node, maximum depth of the tree, minimum angular extent of an individual arc. The font size of the labels, i.e. the terms, is automatically adapted to the angular extent of the arc. Additionally, minimum and maximum values for the font size can be defined by the user.

Since our Sunburst interface is intended to be used for document search, user interaction is provided in two ways. First, clicking with the left mouse button on an arbitrary arc generates a new Sunburst visualization with this arc as root node, i.e. only the documents that are represented by the selected arc are used. Second, a right mouse click on any arc displays a pop-up menu with the locations of the documents represented by the selected arc. The user can then view a document by selecting it from the pop-up menu.

Figure 5 shows a screenshot of a Sunburst interface generated from an ETP of web documents about the music artist *Louis Armstrong*. The values in parentheses indicate the document frequencies. This sample visualization reveals which terms occur in a collection of web documents about *Louis Armstrong*. If the user wants to know, for example, in which documents *Louis Armstrong* and *Miles Davis* are mentioned together, s/he can easily display a list of these documents by clicking on the respective arc as shown in Figure 5. A further click on one of the documents opens it in the standard web browser.

5. Extensibility and Example Applications

Since the beginning of the development of CoMIRVA in late 2004, a number of students have participated in extending the framework. For example, we elaborated a class/interface structure that facilitates extending the audio-based feature extraction algorithms in collaboration with one of our master students, who further implemented two well performing feature extractors which are described in [Pam01, APS05].

Easy extensibility and easy usage in other applications were vital requirements when designing the class structure and interfaces of CoMIRVA. The former is proven by many extensions implemented by us and our students in the context of internships and theses. The latter is demonstrated by *nepTune* [KSPW06], an application that was developed recently and that heavily makes use of functionality provided by CoMIRVA.

nepTune is a novel user interface to music repositories. It combines CoMIRVA's implementations of audio feature extractors, web mining tools, and the SOM and SDH visualization functions.

Given an arbitrary collection of digital music files, nepTune automatically extracts features from the audio signal and trains a SOM on them to form clusters of similar sounding pieces of music. Subsequently, the distribution of the pieces of music on the SOM is determined using an SDH. This SDH is interpreted as a three-dimensional height profile and visualized as a landscape applying a colormap that resembles that of geographical maps. This geographical metaphor, which is called *Islands of Music* [Pam01], yields a landscape where sparse areas are represented by oceans (in blue) whereas clusters with many pieces of music look like mountains (brown and gray) that rise from islands (green). Figure 6 depicts a screenshot taken from the nepTune application. In this case, the landscape was generated from music from the genres Electronic and Metal.

Since similar pieces of music are mapped to similar regions on the landscape by the SOM, the user can intuitively explore his/her own or someone else's music collection by moving through the landscape like in a 3D game. The angle of the viewport is automatically adjusted according to the height of the current position, i.e. if the current position in the landscape is directly in front of a high mountain, the user has the feeling that s/he glances at the top of the mountain; if s/he, in contrast, resides on top of the mountain, the view is adjusted to see which songs are situated at the mountain's foot.

Another project that uses CoMIRVA is *Kifano* (`http://www.kifano.com/`), a music similarity tool in development. Kifano aims at building a recommender system based on audio features which are uploaded by the users.

As for the limitations of CoMIRVA, since it is implemented in Java, the performance of the integrated algorithms is obviously lower than that of native implementations. On the other hand, CoMIRVA offers the advantage of platform independence. Moreover, the integration in the GUI of nearly every function provided by the framework is a quite timecomsuming task and has recently delayed the release of new versions. To cope with this shortcoming, we aim at separating the framework from the GUI.

6. Conclusions and Future Work

In this paper, we presented CoMIRVA, an open source framework and toolkit for music information retrieval and visualization. It combines functions for feature extraction (directly from the audio signal of digital music files as well as from metadata that is derived from the web), special data structures like similarity matrices and Entity Term Profiles, and information visualization approaches that are suited to visualize music-related data.

As for future work, there are many directions into which CoMIRVA should be extended. For example, hierarchical visualization techniques to deal with arbitrarily sized music collections are desirable. Also, time series visualization approaches that describe changes of properties of a piece of music over time should be integrated. Moreover, simple tools like a colormap editor or a statistical editor for data matrices would further increase CoMIRVA's usability. We continuously keep extending CoMIRVA with the help of students that are interested in music information retrieval and information visualization. Eventually, we hope to make CoMIRVA known to a larger community with this paper.

7. Acknowledgments

This research is supported by the Austrian Fonds zur Förderung der Wissenschaftlichen Forschung (FWF) under project number L112-N04 and by the Vienna Science and Technology Fund (WWTF) under project number CI010 (Interfaces to Music). Special thanks are due to all students who support us in extending CoMIRVA.

References

[AH98] Andrews K., Heidegger H.: Information Slices: Visualising and Exploring Large Hierarchies using Cascading, Semi-Circular Discs. In *Proceedings of IEEE Information Visualization 1998* (Research Triangle Park, NC, USA, October 1998), pp. 9–12.

[AP02] Aucouturier J.-J., Pachet F.: Music Similarity Measures: What's the Use? In *Proceedings of the 3rd International Symposium on Music Information Retrieval (ISMIR'02)* (Paris, France, October 2002).

[APS05] Aucouturier J.-J., Pachet F., Sandler M.: "The Way It Sounds": Timbre Models for Analysis and Retrieval of Music Signals. *IEEE Transactions on Multimedia 7*, 6 (December 2005), 1028–1035.

[BGM04] Bederson B. B., Grosjean J., Meyer J.: Toolkit Design for Interactive Structured Graphics. *IEEE Transactions on Software Engineering 30*, 8 (August 2004), 535–546.

[BHP*96] BEDERSON B. B., HOLLAN J. D., PERLIN K., MEYER J., BACON D., FURNAS G. W.: Pad++: A Zoomable Graphical Sketchpad For Exploring Alternate Interface Physics. *Journal of Visual Languages and Computing 7*, 1 (1996), 3–32.

[BMG00] BEDERSON B. B., MEYER J., GOOD L.: Jazz: An Extensible Zoomable User Interface Graphics Toolkit in Java. In *Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology (UIST'00)* (San Diego, CA, USA, November 2000), pp. 171–180.

[DPW03] DIXON S., PAMPALK E., WIDMER G.: Classification of Dance Music by Periodicity Patterns. In *Proceedings of the 4th International Conference on Music Information Retrieval (ISMIR'03)* (Baltimore, MD, 2003), John Hopkins University, pp. 159–166.

[Fek04] FEKETE J.-D.: The InfoVis Toolkit . In *Proceedings of the 10th IEEE Symposium on Information Visualization (InfoVis'04)* (Austin, TX, USA, October 2004).

[Koh82] KOHONEN T.: Self-Organizing Formation of Topologically Correct Feature Maps. *Biological Cybernetics 43* (1982), 59–69.

[Koh01] KOHONEN T.: *Self-Organizing Maps*, 3rd ed., vol. 30 of *Springer Series in Information Sciences*. Springer, Berlin, 2001.

[KPW04] KNEES P., PAMPALK E., WIDMER G.: Artist Classification with Web-based Data. In *Proceedings of the 5th International Symposium on Music Information Retrieval (ISMIR'04)* (Barcelona, Spain, October 2004), pp. 517–524.

[KSPW06] KNEES P., SCHEDL M., POHLE T., WIDMER G.: An Innovative Three-Dimensional User Interface for Exploring Music Collections Enriched with Meta-Information from the Web. In *Proceedings of the 14th ACM Conference on Multimedia 2006* (Santa Barbara, CA, USA, October 2006).

[KSW05] KNEES P., SCHEDL M., WIDMER G.: Multiple Lyrics Alignment: Automatic Retrieval of Song Lyrics. In *Proceedings of 6th International Conference on Music Information Retrieval (ISMIR'05)* (London, UK, September 2005), pp. 564–569.

[Log00] LOGAN B.: Mel Frequency Cepstral Coefficients for Music Modeling. In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR'00)* (Plymouth, Massachusetts, USA, 2000).

[ME05] MANDEL M. I., ELLIS D. P. W.: Song-Level Features and Support Vector Machines for Music Classification . In *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR'05)* (London, UK, September 2005).

[MFD05] MCENNIS D., FUJINAGA C. M. I., DEPALLE P.: jAudio: A Feature Extraction Library . In *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR'05)* (London, UK, September 2005).

[Pam01] PAMPALK E.: *Islands of Music: Analysis, Organization, and Visualization of Music Archives*. Master's thesis, Vienna University of Technology, Austria, December 2001.

[PRM02] PAMPALK E., RAUBER A., MERKL D.: Using Smoothed Data Histograms for Cluster Visualization in Self-Organizing Maps. In *Proceedings of the International Conference on Artifical Neural Networks (ICANN'02)* (Madrid, Spain, August 2002), Springer, pp. 871–876.

[SB88] SALTON G., BUCKLEY C.: Term-weighting Approaches in Automatic Text Retrieval . *Information Processing and Management 24*, 5 (1988), 513–523.

[SKW05a] SCHEDL M., KNEES P., WIDMER G.: A Web-Based Approach to Assessing Artist Similarity using Co-Occurrences. In *Proceedings of the 4th International Workshop on Content-Based Multimedia Indexing (CBMI'05)* (Riga, Latvia, June 2005).

[SKW05b] SCHEDL M., KNEES P., WIDMER G.: Discovering and Visualizing Prototypical Artists by Web-based Co-Occurrence Analysis. In *Proceedings of the Sixth International Conference on Music Information Retrieval (ISMIR'05)* (London, UK, September 2005).

[SKW05c] SCHEDL M., KNEES P., WIDMER G.: Improving Prototypical Artist Detection by Penalizing Exorbitant Popularity. In *Proceedings of the Third International Symposium on Computer Music Modeling and Retrieval (CMMR'05)* (Pisa, Italy, September 2005).

[SLC02] SU M.-C., LIU T.-K., CHANG H.-T.: Improving the Self-Organizing Feature Map Algorithm Using an Efficient Initialization Scheme . *Tamkang Journal of Science and Engineering 5*, 1 (2002), 35–48.

[SZ00] STASKO J., ZHANG E.: Focus+Context Display and Navigation Techniques for Enhancing Radial, Space-Filling Hierarchy Visualizations. In *Proceedings of IEEE Information Visualization 2000* (Salt Lake City, UT, USA, October 2000), pp. 57–65.

[TC00] TZANETAKIS G., COOK P.: MARSYAS: A Framework for Audio Analysis. *Organized Sound 4*, 3 (2000).

[WL02] WHITMAN B., LAWRENCE S.: Inferring Descriptions and Similarity for Music from Community Metadata. In *Proceedings of the 2002 International Computer Music Conference* (Goeteborg, Sweden, September 2002), pp. 591–598.

[ZF04] ZADEL M., FUJINAGA I.: Web Services for Music Information Retrieval. In *Proceedings of the 5th International Symposium on Music Information Retrieval (ISMIR'04)* (Barcelona, Spain, October 2004).

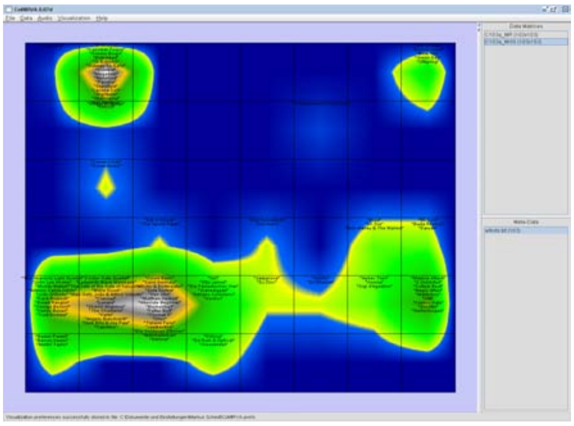


Figure 1: *A Smoothed Data Histogram (SDH) visualization of a Self-Organizing Map (SOM) trained on music features.*

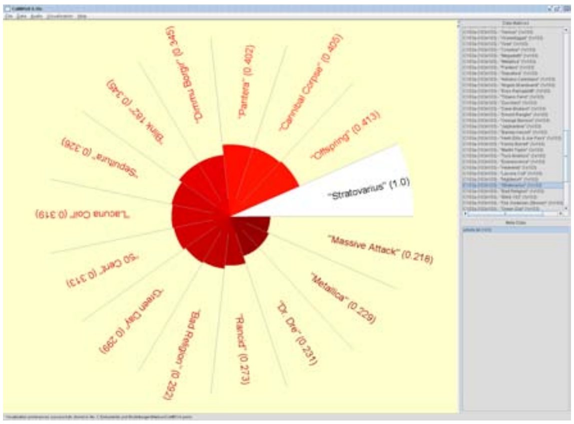
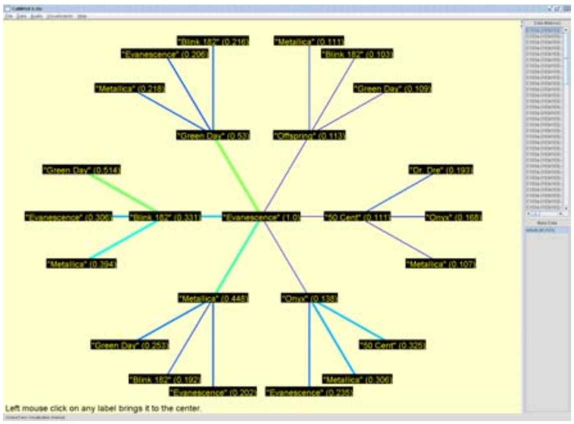


Figure 2: *A Circled Bars visualization of a vector describing similarities between music artists.*



Left mouse click on any label brings it to the center.

Figure 3: *A Circled Fans user interface of a matrix describing similarities between music artists.*

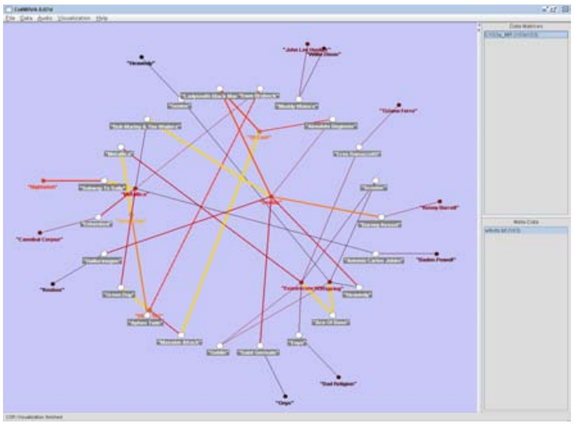


Figure 4: *A Continuous Similarity Ring (CSR) for visualizing prototypical entities.*

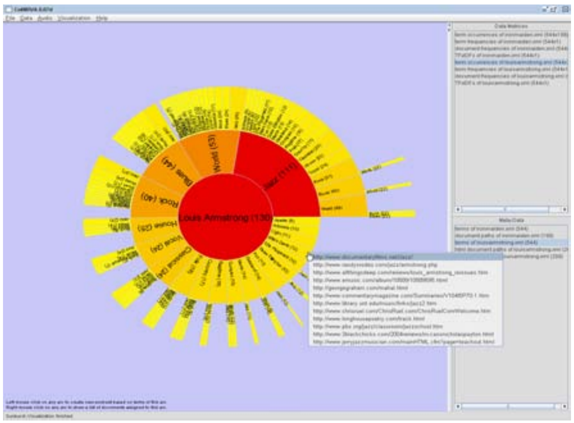


Figure 5: *A user interface for finding (web) documents, which is based on the Sunburst visualization technique.*

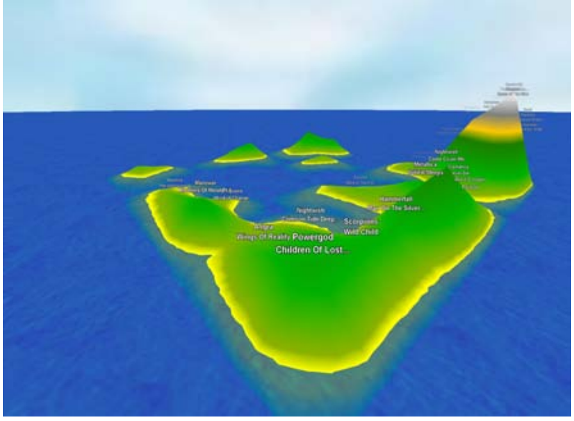


Figure 6: *A screenshot of the nepTune application, depicting a view on a Heavy Metal island.*