

HIERARCHICAL ORGANIZATION AND VISUALIZATION OF DRUM SAMPLE LIBRARIES

Elias Pampalk, Peter Hlavac*

Austrian Research Institute for
Artificial Intelligence (OeFAI)
Vienna, Austria
elias@oefai.at

Perfecto Herrera

Music Technology Group
Institut Universitari de L'Audiovisual
Universitat Pompeu Fabra, Barcelona, Spain
pherrera@iua.upf.es

ABSTRACT

Drum samples are an important ingredient for many styles of music. Large libraries of drum sounds are readily available. However, their value is limited by the ways in which users can explore them to retrieve sounds. The available schemes for organizations those collections are either insufficient to hold a large number of samples, rely on cumbersome manual classification, are inconsistent, and error-prone.

In this paper, we present a new approach for automatically structuring and visualizing large sample libraries using only audio input. We create a hierarchical user interface for efficient exploration and retrieval which is based on a similarity measure for drum sounds and self-organizing maps.

1. INTRODUCTION

Digitized drum samples are an important ingredient in music production for many styles of contemporary music. However, finding the best samples for a drum loop can be a difficult and very time-consuming task.

Countless sample CDs with drum sounds are available on the market. Each one uses its own way and criteria to organize and label the hundreds of samples it contains. Furthermore, an increasing amount of samples are available directly from the Internet. Each source has different naming conventions and supplies metadata (instrument type, diameter, settings, recording environment,...) of variable quality. This makes it difficult to integrate samples from different sources into a single collection with a content-based organization, and practically limits the size of sample collections artists and producers work with.

Currently basically two approaches are used to organize samples from different sources. The first approach is to classify samples on the first level by the instrument (tom, bass drum, snare, hi-hat, cymbal, etc.) and on the second level by the CD source (name of the sample CD or manufacturer). The second approach is to classify samples by their source on the first level, and instrument on the second level. Usually both approaches are used in parallel.

Suppliers have recently started to address the difficulties of managing large sample libraries by integrating them into virtual instruments which offer advanced search mechanism combined with a graphical user interface (e.g. Stylus from Spectrasonic, Groove Agent from Steinberg, PLP 120 from Best Service). However,

* Part of this work was done while the author was a visiting researcher at MTG-IUA.

no system is available for drum sounds which supports similarity-based exploration and retrieval beyond metadata queries.

In this paper we leave aside the metadata and focus solely on the audio signal of a sample and its similarity to others. We hierarchically organize large sample libraries according to this similarity measure. This allows exploration using queries such as: "Find something that sounds more like this sample than that one".

To automatically create such an organization we adapted an auditory model. Furthermore, we used clustering algorithms to create summaries of the collection and visualize the hierarchical structure. To demonstrate our approach we implemented a HTML-based interface allowing the user to explore the hierarchical structure of the sample collection. First demonstrations of our approach to prospective users gave very positive feedback and pointed out some interesting issues for future research.

The remainder of this paper is organized as follows. In the next Section we review related work. In Section 3 we present the similarity measure we use for drum sounds. In Section 4 we discuss the self-organizing map algorithm. In Section 5 we describe the user interface and how we create it. In Section 6 we discuss first feedback from users. Finally, in Section 7 we conclude our work.

2. RELATED WORK

A vast amount of research on organizing and structuring sounds has been published. Three main directions are relevant to our work, namely: (1) instrument identification and classification, (2) timbre spaces, and (3) user interfaces to sound collections.

Instrument identification and classification (for an overview see [1]) is relevant for several reasons. For one it would be a valuable extension to the work presented in this paper to automatically label sounds and use this information in the interface. Furthermore, most approaches to instrument classification can be generalized to classify according to almost any concept. Thus, in an ideal case drum sounds could be classified into bright, thick, heavy, or any other user defined categories. Furthermore, features extracted from an audio signal which are useful for classification are also likely to be useful in developing similarity measures which form the basis of our work. Very promising results on classification of drum sound have recently been published in [2, 3]. However, since we rely on similarity measures, the results from instrument classification cannot be applied directly to our work.

Interesting models for the similarity of percussive instruments have been the outcome of research on timbre spaces and perceptual similarities in general. For example, Lakatos [4] suggests a

3-dimensional timbre space for percussive sounds. In particular, he suggests that the physical dimensions which are correlated with the 3 most salient perceptual dimensions are log-attack time, spectral centroid, and temporal centroid. These physical dimensions are also used in the MPEG-7 description format as descriptors for timbre [5]. However, in first experiments we conducted the quality of these descriptors was insufficient to distinguish fine details in the samples as required for our task.

Another interesting aspect of research on timbre spaces is that the similarity relationships of the sounds are usually visualized in 2 or 3 dimensions using multi-dimensional scaling. One of the earliest approaches to use a self-organizing map (SOM) [6] to study timbre spaces by visualizing sound collections was presented by Cosi et al. [7]. An auditory model is used to compute the similarity between 19 relatively different sounds (e.g., flute, oboe, piano, organ, ...) with pitch C4. Although this previous work differs from our work with respect to the number and type of samples, the same principles apply.

Our approach is mainly inspired by the work of Feiten et al. [8, 9] where a SOM-based interface to efficiently access sample collections was proposed. Analogue to [7] an auditory model to compute the similarity of approximately 100 synthesized samples is used as input to a self-organizing map to visualize the collection. Despite differences in number and type of samples, our main contribution to this direction of research is an adapted similarity measure which was optimized using results from preliminary drum listening tests. In addition, we present a new hierarchical extension to the SOM-based interface to deal with large collections.

A quite different user interface to find sounds is the Sonic Browser [10]. The main idea of the Sonic Browser is to browse audio files by listening to them simultaneously in a stereo-spatialized sound scape. However, feedback from our targeted users indicated that even small overlaps (e.g. open hi-hat fading out while snare starts playing) are irritating and should be avoided.

3. SIMILARITY MEASURE FOR DRUM SOUNDS

In general it is difficult to predict when a human listener will consider two drum sounds to be similar. Similarity depends on the context, which aspects the listener is focusing on, and subjective impressions. Sounds can be described in measurable dimensions such as attack time or spectral centroid but are commonly described with vaguely defined adjectives such as dark, fat, punchy, deep, thick, crispy, etc.

However, compared to other instrument sounds there are several simplifications which make the similarity easier to compute. First of all, drums allow fewer variations in pitch than instruments in general. Thus, we consider changes in frequency equally important to changes in the loudness envelope over time. Secondly, the temporal loudness contour usually has a relatively simple shape. In particular, to some extent we can ignore effects such as vibrato or other loudness modulations which are very common for other types of instruments.

Our approach to compute the similarity of drum sounds is based on [8] where the main idea is to interpret sonograms as vectors and use a distance metric to compute distances in the vector space. Sonograms are computed taking some aspects of the auditory system into account. As input we use 44kHz mono samples. Samples longer than 500ms are truncated. We use a FFT with 23ms windows, weighted with a Hann function, and 12ms overlap to obtain the spectrogram. To model the frequency response

of the outer and middle ear we use the formula proposed by Terhardt [11],

$$A_{dB}(f_{\text{kHz}}) = -3.64 f^{-0.8} + 6.5 \exp(-0.6(f - 3.3)^2) - 10^{-3} f^4. \quad (1)$$

The main characteristics of this weighting filter are that the influence of very high and low frequencies is reduced while frequencies around 3–4kHz are emphasized (see Figure 1).

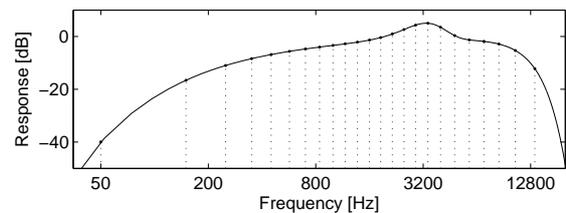


Figure 1: Terhardt's outer and middle-ear model. The dotted lines mark the center frequencies of the 24 critical-bands.

Subsequently the frequency bins of the STFT are grouped into 24 critical-bands according to Zwicker and Fastl [12]. The conversion between the bark and the linear frequency scale can be computed with,

$$Z_{\text{bark}}(f_{\text{kHz}}) = 13 \arctan(0.76f) + 3.5 \arctan(f/7.5)^2. \quad (2)$$

The main characteristic of the bark scale is that the width of the critical-bands is 100Hz up to 500Hz, and beyond 500Hz the width increases nearly exponentially (see Figure 1 where the dotted lines appear almost equally spaced beyond 500Hz on the log-scaled frequency axis).

We calculate spectral masking effects according to Schroeder et al. [13] who suggest a spreading function optimized for intermediate speech levels. The spreading function has lower and upper skirts with slopes of +25dB and -10dB per critical-band. The main characteristic is that lower frequencies have a stronger masking influence on higher frequencies than vice versa. The contribution of critical-band z_i to z_j with $\Delta z = z_j - z_i$ is computed by,

$$B_{dB}(\Delta z_{\text{bark}}) = +15.81 + 7.5(\Delta z + 0.474) - 17.5 \left(1 + (\Delta z + 0.474)^2\right)^{1/2}. \quad (3)$$

We calculate the loudness in sone using the formula suggested by Bladon and Lindblom [14],

$$S_{\text{sone}}(l_{\text{dB-SPL}}) = \begin{cases} 2^{(l-40)/10}, & \text{if } l \geq 40\text{dB}, \\ (l/40)^{2.642}, & \text{otherwise.} \end{cases} \quad (4)$$

After these steps each sample is described by a sonogram in the dimensions time ($f_s = 86$ Hz), frequency (24 critical-bands with the unit bark), and loudness (measured in sone) with a maximum length of 500ms. Examples for sonograms are shown in Figure 2.

The use of different metrics to compare sonograms was studied in [15] where based on data from listening tests the authors

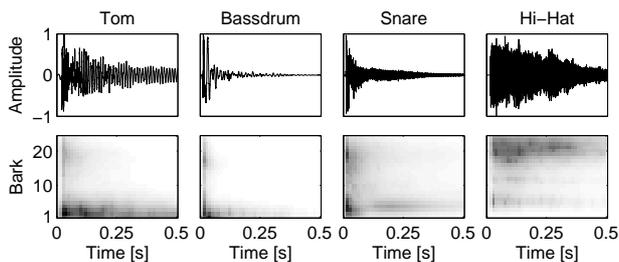


Figure 2: Four typical samples and their sonograms. The grayscale of the sonograms in the 2nd row are normalized so that white equals 0 sone and black equals 27 sone.

come to the conclusion that a Minkowski metric with $p = 5$ produces best results on synthesized harmonic samples. In first experiments we could not confirm these findings for drum samples, thus, we have resorted to the use of the Euclidean distance.

A main problem when using sonograms is the sensitivity to time shifts which requires some sort of temporal alignment. For example, comparing 2 versions of the same sample where one version is shifted by 20ms could yield a distance larger than the distance between two perceptually different samples. In [9] an approach is proposed where a SOM is trained on steady state sounds (with approximately 6ms duration) extracted from the samples. Subsequently, the samples are represented by trajectories on the SOM. The sounds are then compared by computing the distance of their trajectories using the city-block distance and aligning them temporally so that the distance is minimal. The main advantage of using the SOM is to optimize the computations, however, the computing power available today allows us to directly align the sonograms.

To align two sonograms we compute the distances between them while shifting them against each other in the range of 50ms. We then take the minimum of these distances as the distance of the samples. Figure 3 gives an example for such a direct alignment.

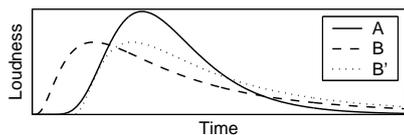


Figure 3: Illustration of the temporal alignment assuming there is only one frequency band. B is aligned with A which results in B' with a minimum distance to A.

4. SELF-ORGANIZING MAP

The SOM [16, 6] is a useful algorithm mainly to visualize very high-dimensional data. In previous work we have applied it to organize and visualize music collections [17, 18]. In this paper we use 1-dimensional SOMs to hierarchically structure the samples and a 2-dimensional SOM for visualization. The SOM consists of units which have a topological order (usually a 2-dimensional rectangular grid, referred to as map). Each of these units is assigned a model vector in the data space. The model vectors can be initialized in various ways, basically a random initialization is sufficient.

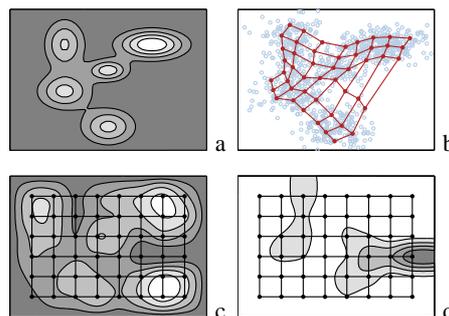


Figure 4: Illustration of the SOM. (a) The probability distribution from which the sample was drawn. (b) The model vectors of the SOM. (c) The SDH and (d) the U-matrix visualizations.

Each sound sample is assigned to the unit which has the most similar model vector (best matching unit). Thus, each sound is mapped to a location on the map (which is usually used for visualization).

The main objective of the SOM is to map similar data items (i.e., sound samples) to units close to each other. This is achieved by iteratively optimizing the topology and quantization error. The quantization error is optimized by adapting the model vector of each unit so that it better represents the samples assigned to it. This is identical to k-means clustering. The topology is preserved by taking the neighborhood of each unit into consideration when adapting the model vectors. The model vector of each unit is adapted not only to fit the directly assigned samples, but also the samples of neighboring units. The size of the neighborhood which is taken into consideration is decreased gradually during training. The final size of this neighborhood together with the number of map units (map size) are the two main parameters of the SOM which control how much freedom the SOM has to adapt to the data.

Figure 4 illustrates some important characteristics of the SOM. Samples are drawn from a 2-dimensional probability density function. A 2-dimensional (8x6) SOM is trained so that the model vectors adapt to the topological structure of the data. There are two important characteristics of the non-linear adaptation. First, the number of data items mapped to each unit is not equal. Especially in sparse areas some units might represent no data items. Second, the model vectors are not equally spaced. In particular, in sparse areas the adjacent model vectors are relatively far apart while they are close together in areas with higher densities.

Both characteristics can be exploited to visualize the cluster structure of the SOM using smoothed data histograms (SDH) [19] and the U-matrix [20], respectively. The SDH visualizes how many items are mapped to each unit. The smoothing is controlled by a parameter. The U-matrix visualizes the distance between the model vectors. The SDH visualization (Figure 4(c)) shows the cluster structure of the SOM. Each of the 5 clusters are identifiable. The U-matrix mainly reveals that there is a big difference between the clusters in the lower right and the upper right.

5. USER INTERFACE

The HTML-based user interface (see Figure 5) that we have developed consists of two parts. The upper part is only text-based and the lower part is mainly graphical. In the following we describe

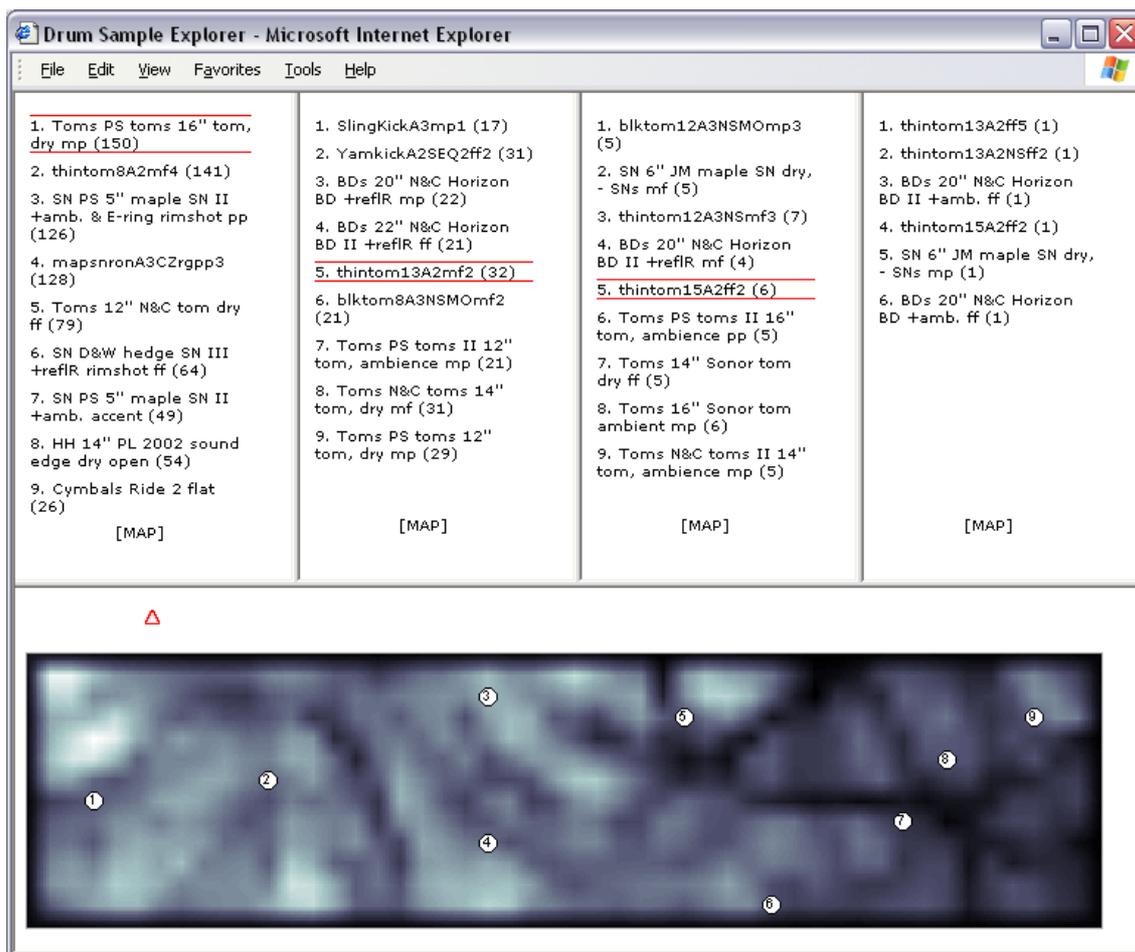


Figure 5: Screenshot of HTML-based user interface.

the ideas and concepts behind both of them. The main idea is to give the user first a overview of the different samples available and then rapidly narrow down the search with each input from the user. A demonstration is available online without the audio files due to copyright restrictions.¹

5.1. Text Interface

The intention of the text-based interface is to create a very simple interface which would allow the user to navigate in the sample collection with closed eyes using only a few keyboard keys. The basic functionality would be using up and down keys to listen to the next or previous sample, right to listen to more similar and left to listen to less similar samples. However, in the HTML interface this functionality is available only for usage with the mouse.

In Figure 5 the four columns represent the four levels of the hierarchical structure. The first level is the leftmost column, the fourth level the rightmost. In this case, the user has selected the first sound on the first level (requesting more of this kind). Each of the 9 choices are typical sounds for the sub-branches they represent. On the second and third level the 5th sample was selected

leaving a final set of 5 samples in the fourth column. If the user is not satisfied with this set it is always possible to make different choices at higher levels in the hierarchy and explore other branches.

The first level is a rough summary of the collection based on 9 samples. The number 9 was arbitrarily chosen manually. It is a trade-off of using as many samples as possible to describe the collection as accurately as possible on one side, and using as few samples as possible to create a good summary. In future work it might be interesting to investigate determination of this number automatically for each node in the tree.

The 9 samples are determined using a 1-dimensional SOM with 9 units. The motivation for using a SOM is to order the samples in a meaningful way. In particular, adjacent neighbors on the list should be similar to each other. In this case we have a rough order of toms (1,2,4,5), snares (3,6,7), hi-hat (8), and cymbals (9). Alternatively, k-means clustering could be used in combination with a traveling salesman algorithm to sort the clusters.

Each of the 9 (parent) samples on the first level represent a subset of the collection (children). Each subset includes all samples which are best represented by the respective parent. The number of these children is displayed in brackets next to the name of the parent. Furthermore, each parents subset is enlarged by 50%

¹<http://www.oefai.at/~elias/dafx04>

to include children which are not best represented but are nevertheless similar. For example, in Figure 5 on the first level 150 samples are best represented by the first parent. The subbranch includes these 150 plus an addition 75 which are best represented by the second parent but are located on the boundary to the first parent. This overlap in the hierarchy ensures that samples which are similar to two parents can be found more easily.

For each set of children a SOM is trained until the number of children in a set is smaller than 13. Thus, the depth of the branches of the tree are not fixed to a specific number. In the experiments we will discuss later the depth reached a maximum level of 4 and a minimum level of 3.

There are several alternatives to create hierarchical structures. In previous work [21] we have used the Growing Hierarchical SOM (GHSOM) [22]. The main reason for not using the GHSOM in this work is because it would not be possible to create the visualizations described below. However, the GHSOM and variations [23] use heuristics which would automatically determine the number of parent samples. These heuristics might be suitable choices for future work.

5.2. Graphical Interface

The intention of the graphical user interface is to give the user more information on the automatically created organization. The GUI is tightly coupled with the text-based interface. The visualization is based on one large 2-dimensional (48×12) SOM which is trained on the whole collection. The location of the 9 samples (determined in the text-based part) are marked on the map with numbers. The user can move the mouse over the samples to get a tooltip with the sounds filename displayed and hear the sounds. If the user wants to find more of the same then it is possible to click on the sounds to descend to the next level.

The image in the lower part of Figure 5 is the SOM where the 9 samples of the first level marked. Note the order which is created by the 1-dimensional SOM. The gray-shadings in the background are a smoothed data histogram and indicate how many items are located in the different areas. The SDH smoothing parameter is automatically adjusted for each hierarchy level to create rough summaries on higher levels, and more detailed summaries on lower levels. In particular, the parameter was calculated based on the square root of samples on the respective level.

In previous work we used the SOM and SDH to visualize music collections using a metaphor of islands of music [17]. Clusters found by the SDH were visualized as islands. In this work, we use the SDH mainly to visualize which part of the collection is currently being considered in a branch of the hierarchy.

Figure 6 illustrates how the set of samples is narrowed down as the user makes selections. On the first level (see map in Figure 5) the whole collection is visualized on the map and the parents are spread across it. On the second level the area only covers about one quarter of the whole collection. Finally, on the fourth level the parents are so close to each other that they overlap in the figure.

6. FIRST EVALUATIONS & DISCUSSION

For the experiments we used (mostly dry, or mixed with some ambience or reflection room) drum samples from 2 sample CDs from different vendors. In total the collection had 817 distinct samples. We computed the HTML interface described above and informally demonstrated it to 5 prospective users who use and search for drum

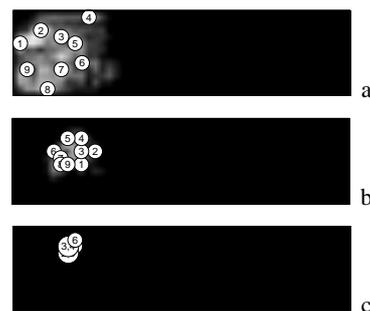


Figure 6: Maps belonging to (a) column 2, (b) column 3, and (c) column 4 of Figure 5.

samples on a daily basis. In general feedback was very positive. We observed the following.

(1) Given the choice the users always preferred the graphical interface. One user explained that producing music is a creative process. Unconventional search mechanisms are more likely to come up with something unpredictable. However, the text interface was also considered to be very useful. Mainly because some of the displayed file names contained very useful information (e.g., *SN 6" mixed with reflection room rimshot hard*). The request was made to better integrate this information into the graphical interface.

(2) Generally, the better integration of metadata was a major request. Especially filters to focus only on one instrument are missing. However, filtering instruments might be a suboptimal solution in cases where samples from different instruments can sound very similar such as a bass drum mixed with a reflection room and snares. One solution might be to use a color coding to indicate the instrument type of the parent in the GUI. Another option might be to use component planes analogue to the weather charts in the islands of music metaphor [17]. The weather charts are laid over the SOM and display in which areas there is a high or a low concentration of a specific property.

(3) A very interesting point which was brought up is that 3 hierarchical levels would be sufficient instead of 4. Mainly this can be explained by weaknesses of the similarity measure we use. Although, the correlation of the similarity measure and user ratings from preliminary A-B drum listening tests is around 0.8, the similarity measure is far from optimal. In particular, the level of detail which our organization creates is not supported by the similarity measure. For example, in Figure 5 in the 4th column there is a mix of toms, bass drums, and even one snare drum. Another reason why 3 levels could be sufficient is that in manually created organizations it is very common to have significantly more than 12 samples in the lowest levels of the hierarchy (some users mentioned they usually have 30 or more samples in directories on the lowest level).

(4) Another important point brought up in the interviews was that every artist uses his own vocabulary to describe samples. Although it would not be necessary to adapt to each artist's individual vocabulary it would be very useful to classify samples according to words such as dark, thick, crispy, etc. even if the meaning of these words needs to be more or less arbitrarily predefined. This additional information could be visualized as mentioned in (2).

Other points brought up were (5) the lack of a zoom function, (6) the restriction to mono sound, (7) the selected parent should

be marked on the lower level hierarchy, (8) a good system should allow very fast browsing using as few clicks as possible to support the creative process.

7. CONCLUSIONS

We have presented a system to automatically organize and visualize drum sound collections hierarchically. The similarity measure we use has several limitations, however, it seems to be sufficient for drum sounds and their specific characteristics. First feedback we got from prospective users was very positive although some modifications were suggested. In future work will focus on improving the similarity measure. Currently we are conducting listening tests for drum sounds to gather data to optimize the various parameters in the similarity measure. Furthermore, we plan to implement a VST plugin to integrate the system into sequencers such as Cubase or Logic.

8. ACKNOWLEDGEMENTS

This research was supported by the EU projects HPCRN-CT-2000-0015 MOSART² and FP6-507142 SIMAC³. The Austrian Research Institute for Artificial Intelligence is supported by the Austrian Federal Ministry for Education, Science, and Culture and by the Austrian Federal Ministry for Transport, Innovation, and Technology. Furthermore, the authors wish to thank the researchers and students at the MTG for participating in the drum listening tests and Marx Zsolt for valuable feedback.

9. REFERENCES

- [1] P. Herrera, X. Amatriain, E. Battle, and X. Serra, "Instrument segmentation for music content description: a critical review of instrument classification techniques," in *Proceedings of the International Symposium on Music Information Retrieval (ISMIR'00)*, Plymouth, MA, 2000.
- [2] P. Herrera, A. Yeterian, and F. Gouyon, "Automatic classification of drum sounds: A comparison of feature selection methods and classification techniques," in *Proceedings of the International Conference of Music and Artificial Intelligence*. 2002, pp. 69–80, Springer.
- [3] P. Herrera, A. Dehamel, and F. Gouyon, "Automatic labeling of unpitched percussion sounds," in *Proceedings of the Audio Engineering Society, 114th Convention*, Amsterdam, Netherlands, 2003.
- [4] S. Lakatos, "A common perceptual space for harmonic and percussive timbre," *Perception and Psychophysics*, vol. 62, no. 7, pp. 1426–1439, 2000.
- [5] G. Peeters, S. McAdams, and P. Herrera, "Instrument sound description in the context of MPEG-7," in *Proceedings of the International Computer Music Conference (ICMC'00)*, Berlin, Germany, 2000, ICMA.
- [6] T. Kohonen, *Self-Organizing Maps*, Springer, 2001.
- [7] P. Cosi, G. De Poli, and G. Lauzzana, "Auditory modeling and self-organizing neural networks for timbre classification," *Journal of New Music Research*, vol. 23, pp. 71–98, 1994.
- [8] B. Feiten, R. Frank, and T. Ungvary, "Organisation of sounds with neural nets," in *Proceedings of the International Computer Music Conference (ICMC'91)*, San Francisco, CA, 1991, pp. 441–444, ICMA.
- [9] B. Feiten and S. Günzel, "Automatic indexing of a sound database using self-organizing neural nets," *Computer Music Journal*, vol. 18, no. 3, pp. 53–65, 1994.
- [10] M. Fernström and E. Brazil, "Sonic browsing: An auditory tool for multimedia asset management," in *Proceedings of the International Conference on Auditory Display (ICAD'01)*, Espoo, Finland, 2001, pp. 132–135.
- [11] E. Terhardt, "Calculating virtual pitch," *Hearing Research*, vol. 1, pp. 155–182, 1979.
- [12] E. Zwicker and H. Fastl, *Psychoacoustics, Facts and Models*, Springer, Berlin, 2nd edition, 1999.
- [13] M. R. Schroeder, B. S. Atal, and J. L. Hall, "Optimizing digital speech coders by exploiting masking properties of the human ear," *Journal of the Acoustical Society of America*, vol. 66, no. 6, pp. 1647–1652, 1979.
- [14] R.A.W. Bladon and B. Lindblom, "Modeling the judgment of vowel quality differences," *Journal of the Acoustical Society of America*, vol. 69, no. 5, pp. 1414–1422, 1981.
- [15] B. Feiten and S. Günzel, "Distance measure for the organization of sounds," *Acustica (Research Notes)*, vol. 78, no. 3, pp. 181–184, 1993.
- [16] T. Kohonen, "Self-organizing formation of topologically correct feature maps," *Biological Cybernetics*, vol. 43, pp. 59–69, 1982.
- [17] E. Pampalk, A. Rauber, and D. Merkl, "Content-based organization and visualization of music archives," in *Proceedings of the ACM Multimedia*, Juan les Pins, France, 2002, pp. 570–579, ACM.
- [18] E. Pampalk, S. Dixon, and G. Widmer, "Exploring music collections by browsing different views," *Computer Music Journal*, vol. 28, no. 2, pp. 49–62, 2004.
- [19] E. Pampalk, A. Rauber, and D. Merkl, "Using smoothed data histograms for cluster visualization in self-organizing maps," in *Proceedings of the International Conference on Artificial Neural Networks (ICANN'02)*, Madrid, Spain, 2002, pp. 871–876.
- [20] A. Ultsch and H. P. Siemon, "Kohonen's Self-Organizing Feature Maps for Exploratory Data Analysis," in *Proceedings of the International Neural Network Conference (INNC'90)*, Dordrecht, Netherlands, 1990, pp. 305–308, Kluwer.
- [21] A. Rauber, E. Pampalk, and D. Merkl, "The SOM-enhanced JukeBox: Organization and visualization of music collections based on perceptual models," *Journal of New Music Research*, vol. 32, no. 2, pp. 193–210, 2003.
- [22] M. Dittenbach, A. Rauber, and D. Merkl, "Uncovering hierarchical structure in data using the growing hierarchical self-organizing map," *Neurocomputing*, vol. 48, no. 1-4, pp. 199–216, October 2002.
- [23] E. Pampalk, G. Widmer, and A. Chan, "A new approach to hierarchical clustering and structuring of data with self-organizing maps," *Journal of Intelligent Data Analysis*, vol. 8, no. 2, pp. 131–149, 2004.

²<http://www.diku.dk/musinf/mosart/>

³<http://www.semanticaudio.org>