AUDIO-BASED MUSIC VISUALIZATION FOR MUSIC STRUCTURE ANALYSIS

Ho-Hsiang Wu and Juan P. Bello
Music and Audio Research Laboratory (MARL), New York University, New York, USA
hhw230,jpbello@nyu.edu

ABSTRACT

We propose an approach to audio-based data-driven music visualization and an experimental design to study if the music visualization can aid listeners in identifying the structure of music. A three stage system is presented including feature extraction, the generation of a recurrence plot and the creation of an arc diagram to visualize the repetitions within a piece. Then subjects are asked to categorize simple forms of classical music with and without audio and visual cues provided. The accuracy and speed are measured. The results show that the visualization can reinforce the identification of musical forms.

1. INTRODUCTION

The detailed study of recorded music is a labor intensive task. This is especially true of the analysis of large audio collections. These collections are difficult to browse given standard catalog and metadata descriptions of music, which provide no information about the musical contents of each recording. Music information retrieval (MIR) research provides a solution to these issues through the development of tools and algorithms that allow for efficient, content-based search and navigation of large music catalogs. In this context, the generation of novel, data-driven and intuitive representations of audio content is necessary to aid the work of musicians and musicologists trying to derive knowledge from these analyses.

Music visualizations have been extensively studied in MIR as ways of helping listeners browse through audio collections and attain a better understanding of their music content [1]. Previous work has mostly concentrated on collection-level visualizations, where tracks are organized according to their similarity or grouped into, e.g., genre or mood categories [2, 3, 4]. Relatively little attention has been paid to the visualization of within-track contents beyond interactive displays based on low-level features [5], or the plotting of feature sequences and intermediate representations (such as self-similarity matrices or other representations used in music structure analysis) that are not necessarily intuitive or informative to non-expert users [6, 4].

In this paper we propose a data-driven approach to the content-based visualization of music structure, based on recurrence analysis of harmonic features. Like previous work on music structure analysis [7, 8, 9], we exploit the existence of patterns of repetition in music. However, we do not assume the common view of music structure as a high-level concatenation of blocks, and thus make no decisions about segmentation boundaries. Instead, we adopt the approach, pioneered in [10] with MIDI data, where data recurrences are visualized by means of arc diagrams, and high-level structure is in the eye of the beholder. The proposed visualization approach is evaluated on its ability to improve the accuracy and speed with which users identify simple forms in classical music.

The remainder of this paper is organized as follows: the details of the visualization approach are described in section 2; the evaluation methodology is discussed in section 3; the results and discussion of the subjective evaluation are presented in section 4; while section 5 presents our conclusions and plans for future work.

2. VISUALIZATION APPROACH

The proposed approach can be subdivided into three main stages: first we extract low-level, harmonic features from the audio signal; second, we project the feature sequence into phase space and compute a recurrence plot from this data; and finally, we generate an arc diagram characterizing the repetitions in the music data stream. The block diagram of the system is shown in Figure 1.

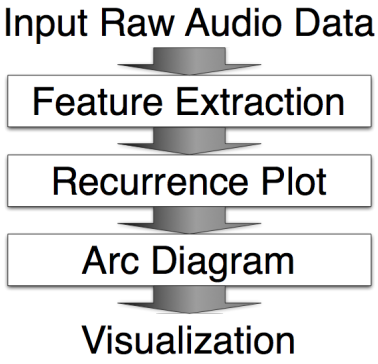

Figure 1. Visualization approach

2.1 Feature Extraction

In our analysis we use chroma features to represent harmonic content in the music signal. Chroma features are commonly obtained from short-term spectral analysis, e.g. via the STFT. For each analysis frame, they represent the signal's energy across the 12 pitch classes of the chromatic scale of western tonal music. The concatenation of these 12-dimensional chroma vectors across time is known as a chromagram.

In our implementation, chroma features are computed via the constant-Q transform [11], a spectral analysis technique in which frequency domain channels are logarithmically spaced, and are thus closely related to the frequency resolution of the human hearing system. First the signal is downsampled to $f_s = 5512.5Hz$. Next, we compute the STFT using a 1024 samples-long Hann window and a hop size of 512. The spectrum is then multiplied by a constant-Q kernel computed with a minimum frequency of 73.42 Hz, a resolution of 36 bins per octave and a 3-octave span. The dimensionality of the chromagram, shown in Figure 2(a), is reduced to 12 bins with a weighted sum across each 3-bin pitch class neighborhood.

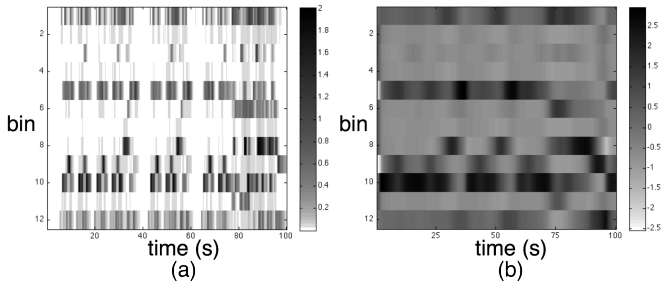

Figure 2. (a) Standard chromagram and (b) chromagram after low-pass filtering, standardization and resampling.

Finally, the features are low-pass filtered, standardized to zero mean and unit variance, and resampled to a resolution of 2 frames per second, as shown in Figure 2 (b).

2.2 Recurrence Plot

A recurrence plot (RP) is a method for analyzing nonlinear dynamic systems [12]. It is visualized as a binary square matrix, in which value ones correspond to pairs of times (indicated by row and column indices) at which a state of the dynamic system recurs. The RP is derived from the so-called phase space, in which all possible states of a system are represented as unique regions. We can consider a chromagram as the output of a nonlinear dynamic system, with each vector corresponding to a trajectory point in a multi-dimensional space. Of course, chromagrams are not produced by dynamic systems, but assuming so allows us to create an intuitive representation able to fully characterize harmonic repetitions in music.

Let us assume a one-dimensional time series $x(t)$, as illustrated in Figure 3(a). We can reconstruct its phase space trajectory using a process known as time-delay embedding. In this process we choose an embedding dimension m, and

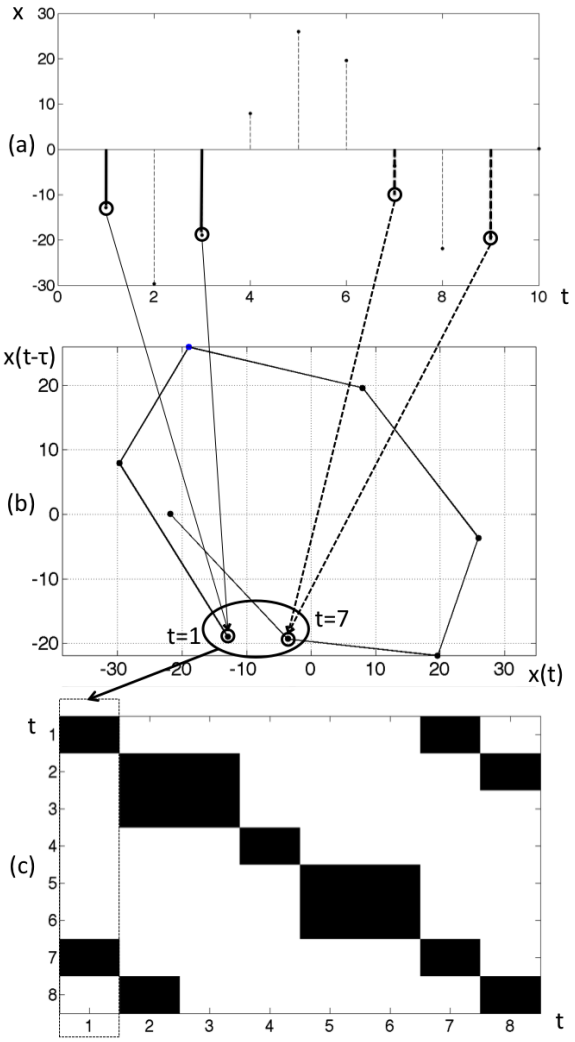

Figure 3. (a) Time series (b) Phase space representation (c) Recurrence plot.

a time delay τ, such that each time t in the series is now represented by a vector obtained by concatenating the values $x(t), x(t-\tau), ..., x(t-(m-1)\tau)$. As can be seen in Figure 3(b), for $m = 2$ and $\tau = 2$, each of these vectors describes a point in the m-dimensional phase space.

Time delay embedding can be also applied to multi-dimensional time series, such as the chromagram $C = \{c_{k,i}\}$, where $i = 1 \ldots N$, the length of the time series, and $k = 1 \ldots 12$, the dimensionality of the chroma vector:

$$\vec{x_c}(i) = (c_{1,i}, c_{1,i-\tau}, \ldots, c_{1,i-(m-1)\tau}, \cdots \\ c_{12,i}, c_{12,i-\tau}, ..., c_{12,i-(m-1)\tau}) \quad (1)$$

We can compute the recurrence plot R for this trajectory, such that $R(i,j) = 1$ if $\vec{x_c}(i)$ and $\vec{x_c}(j)$ are no farther than a distance of ϵ from each other in the phase space, and $R(i,j) = 0$ otherwise. This can be expressed as:

$$R(i,j) = H(\epsilon - \parallel \vec{x_c}(i) - \vec{x_c}(j) \parallel), \\ \vec{x_c}(i) \in \mathbb{R}^m, i, j = m, ..., N, \quad (2)$$

where N is the number of frames in the original time series, ϵ is a threshold value, $\| \cdot \|$ is a norm (e.g. Euclidean norm), and H is the unit step function. Figure 3(c) shows the resulting RP for the phase space in Figure 3(b).

Previous research into applying phase space and recurrence plots to music analysis includes the visualization of expressive timing in piano performance [13] and the state-of-the-art in cover song identification using a variant of RP that can be computed on pairs of songs [14]. Notably, self-similarity matrices, which are widely-used in MIR, are a special case of RP known as *distance* plots [12], computed using $m = \tau = 1$.

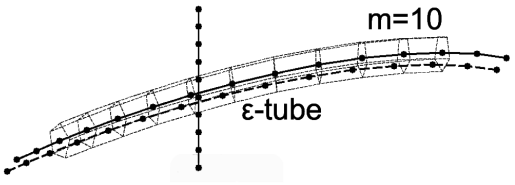

Figure 4. Tangential motion.

Figure 4 serves to illustrate the effect of changing the values of m and τ. The illustration shows a neighborhood of the phase space containing 3 segments of a trajectory: two running in parallel from left to right, and a third one running perpendicular from top to bottom. At the crossover point, when $m = \tau = 1$, it can be seen how all three sections of the trajectory are close enough to be considered recurrences of each other (resulting in values of 1 in the RP). An example RP with $m = 1$ is shown at the top of Figure 5. However, as $(m-1)\tau$ increases, the size of the neighborhood of points that need to be in close proximity to generate a recurrence also increases, weeding out recurrences generated by *tangential* contact. The left column of Figure 5 shows the changes in the RP as m increases. In our visualization experiments, we have heuristically chosen $m = 25$ and $\tau = 1$ as appropriate embedding values.

In both Figure 4 and Figure 5, we can also observe that due to the natural proximity between consecutive points in the trajectory (including between each point to itself), there is an important concentration of activations in the RP along its main diagonal. These activations, however, carry no information about recurrences in the data, and need to be ignored for visualization purposes. A common solution to this problem is to exclude an area of arbitrary width contiguous to the main diagonal by using a *Theiler* window [15]. In our experiments, the Theiler window size is set to 10% of the length of the resampled feature sequence.

Finally, for visualization purposes, it is important to ensure that RPs from different songs contain roughly the same amount of information. In this paper we use recurrence rate (RR), one of several measures commonly used to quantify the properties of RPs [12]. RR measures the density of the plot as a percentage of its recurrences:

$$RR = \frac{1}{N^2} \sum_{i,j=1}^{N} R(i,j) \tag{3}$$

In our implementation RR is used as a threshold on the amount of information we would like to show in the visualization, such that the higher the rate, the denser (and noisier) the plots become. Conversely, the lower the rate, the sparser, and potentially uninformative, they are. After informal experimentation, $RR = 0.2$ was chosen as a good trade-off between both those extremes.

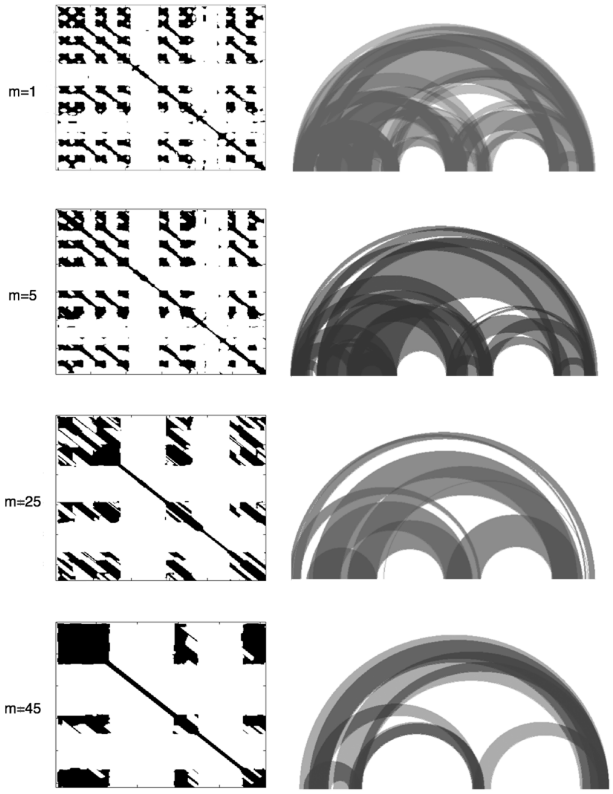

Figure 5. Recurrence plot (left column) and corresponding arc diagram (right column) with different embedding dimension m.

2.3 Arc Diagram

Arc diagrams have been extensively used to represent complex patterns of data recurrence in fields such as biology and physics [10, 16]. They consist of arcs connecting points of repetition in the data stream. Thus, we can simply convert the above recurrence plots into arc diagrams by representing $R_{i,j} = 1$ as an arched line connecting the ith and jth frame of the data sequence. An example diagram can be seen in Figure 6 (a), where the horizontal axis represents discrete time, and $R_{1,9} = R_{2,10} = R_{5,15} = 1$.

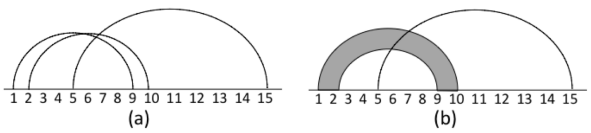

Figure 6. (a) 1 to 1 connection arc diagram (b) Arc diagram with grouping.

To avoid too dense a visualization by repeatedly con-

necting all single arcs, we group neighboring lines into wider arcs. The details of the grouping method are described in Algorithm 1. First we define each group G_s as a vector of four values indexing the arc boundaries, such that $G_s(1)$ and $G_s(2)$ are the left and right boundaries of the leftmost part of the arc, while $G_s(3)$ and $G_s(4)$ are the boundaries of the rightmost part of the arc. Next, we evaluate every point $R_{i,j} = 1$ in the RP, comparing i and j, respectively, with the left and rightmost sets of boundaries of each existing group. If i, j are contiguous to the boundaries of a given group, then they are added to it and the group's boundaries are updated. If i and j do not belong to any existing group, then a new group is created with i and j as the initial boundaries $G_s(i, i, j, j)$. We repeat this process for all points in the RP, until we obtain a final list of groups to be used in the drawing of the arc diagram.

Algorithm 1 Arc diagram grouping method

```
Create a group list G_s(0,0,0,0) with four entries
{G_s(1), G_s(2) leftmost and G_s(3), G_s(4) rightmost part boundaries}

for i ≤ j and i, j = 1 to N do
  if R_{i,j} = 1 then
    for s = 1 to length(G_s) do
      if G_s(3) ≤ j ≤ G_s(4) and 0 ≤ G_s(1) − i ≤ 1 then
        G_s(1) = i
      else if G_s(3) ≤ j ≤ G_s(4) and 0 ≤ i − G_s(2) ≤ 1
      then
        G_s(2) = i
      else if G_s(1) ≤ i ≤ G_s(2) and 0 ≤ G_s(3) − j ≤ 1
      then
        G_s(3) = j
      else if G_s(1) ≤ i ≤ G_s(2) and 0 ≤ j − G_s(4) ≤ 1
      then
        G_s(4) = j
      else
        create new group G_s(i, i, j, j)
      end if
    end for
  end if
end for
```

In the creation of the arc diagram visualization, we have borrowed several implementation strategies from [10]. For example, we use translucent color in order to clearly depict multiple layers of arcs, with color depths (saturation) used to resolve arc overlap in the limited pixel space. Another strategy is to show grouped arcs as a single, wider arc as in Figure 6 (b), rather than as a collection of individual arcs as in Figure 6 (a), where the perceived order of the arc boundaries is reversed. The result of the process can be observed on the right column of Figure 5, which shows a number of examples of arc diagrams for varying values of m. It can be seen how the choice of embedding parameter affects the density and clarity of the visualization.

3. EXPERIMENTAL DESIGN

3.1 Task

We conduct a preliminary experiment to study if these visualizations can help convey information more intuitively and efficiently than simply listening to the music. The task is to identify musical forms as belonging to one of four categories (strophic, binary, ternary and rondo), with and without aid from audio/visual cues.

Four different combinations of visual and audio cues are provided in the experiment in order to compare the influences of each of them on the perception of musical forms. In the first category, both visualization and music are provided. In the second category, only music is provided without any visual cue. In the third category, only visualization is provided without any music playing. In the fourth category, we provide both music and segmentation boundary information obtained using the Echonest API [17].

In order to prevent confusing definitions arising from listeners' different interpretations and understandings of musical form, the four categories to be identified are represented with capital letters indicating the segments which comprise the structure. Strophic form is represented as AA, AAA or AA'. Music in this form contains one main theme which is repeated either with or without slight variations. Binary form is represented as AB or ABAB. Music in this form consists of two main alternating themes, always ending with the second theme. Ternary form is represented as ABC or ABA'. Music in this form has three main themes, or two main themes plus a re-statement of the first theme with or without variations. Rondo form is represented as ABACA, ABACABA or ABACADAEA, where we have a recurring main theme alternating with other different (usually contrasting) themes.

3.2 Methodology

First, the subjects are given descriptions of the task, the explanations of musical forms and the experiment environment. Example tests of each category of audio/visual cue combination are then given to the subjects in order to familiarize them with the system and the experimental process. Next, the subjects are asked to go through forty music pieces with combinations of different forms and visual/audio cues provided. A screen shot of the test environment is shown in Figure 7. Subjects can click on any location on the image they are interested in and listen to the music starting from the corresponding location in the audio track. They can listen to the same piece for as long as they want until they make a decision, at which point they click on one of the check boxes to select a form and move on to the next test. The total length of time they spend on each test is recorded as a measure of the speed of recognition.

The experimental system is implemented using Processing [18]. Processing is a Java-based programming language which provides programmers with quick graphical, visualization and interaction prototyping environment. It also provides certain control abilities for user interface design.

3.3 Subjects

Twenty people were invited to participate in this experiment, ten males and ten females, all over the age of eighteen. Ten of them are trained classical music players and

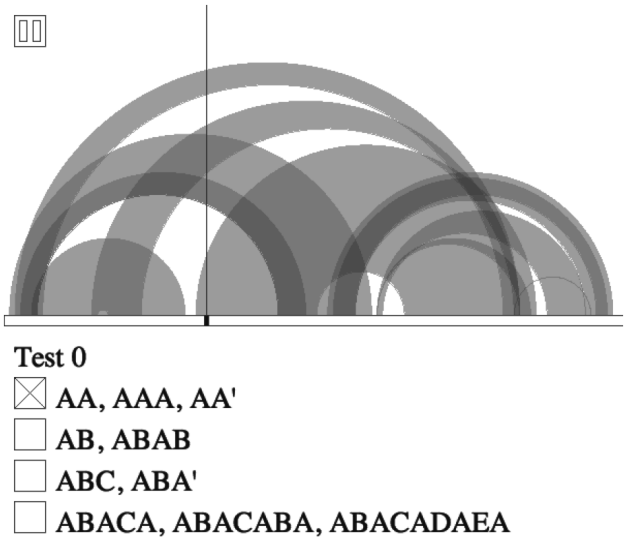

Figure 7. Screen shot of test environment.

ten of them are not. All of them listen to music on a regular basis.

3.4 Test Music

For each of the four forms described above, ten classical music pieces are selected. Strophic form pieces are selected from Lieder of the Classical era. For binary and ternary forms, pieces are chosen from Schumann's "Album for the Young" and Bach's "Notebook for Anna Magdalena Bach". Rondo form pieces are selected according to the examples in the music theory book "The Analysis of Musical Form" [19]. These pieces are mostly from the works of Beethoven and Mozart of the Classical era.

4. RESULTS AND DISCUSSION

The average accuracies and response times for all test are depicted in Table 1. Results are categorized by modes of audio/visual interaction, including a combination of music, arc diagram visualizations and segmentation boundaries.

	Vis/Mus	Mus only	Vis only	Seg/Mus
Accuracy	57.5%	42.5%	37%	45%
Avg time	165s	265s	29s	198s

Table 1. Overall accuracy (%) and average time (seconds) for each category.

The relatively-low levels of classification accuracy illustrate the inherent difficulty of the task, with subjects taking an average of 4 minutes and a half per track to achieve 42.5% accuracy in listening-only tests. It is immediately evident that the use of visual cues (both segmentation boundaries and the arc diagram) improves accuracy and speed. The addition of the proposed visualization is most beneficial, improving accuracy by an average of 15% and reducing analysis time by an average of 100 seconds, significantly better than music-only and music plus segmentation results.

However, visualization-only experiments resulted on worst average accuracy and, unsurprisingly, fastest recognition speed. This indicates that while the arc diagrams help to draw attention to important information in the music signal, they are by themselves not enough to robustly convey information about the musical structure. After tests, subjects informally reported that their preference was to listen to the whole piece before looking for details and finding repetitions. In this context, visualizations helped to guide navigation, but failed to succeed in replacing the music recording as the main information channel.

RE \ GT	Strophic	Binary	Ternary	Rondo
Strophic	52%	14%	8%	17%
Binary	10%	54%	24%	8%
Ternary	11%	17%	56%	7%
Rondo	27%	15%	12%	68%

Table 2. Confusion matrix of the category with both visualization and music.

Table 2 provides a closer look at recognition results using both audio and the proposed visualization. It shows the confusion matrix, with rows representing ground truth values (GT) and columns the subjects' results (RE). Most confusions fall into one of two categories: strophic/rondo confusions and binary/ternary confusions. The former, can be partly attributed to the similarities between the corresponding arc diagrams. Figure 8 shows diagrams for two pieces in strophic (a) and rondo (b) form, respectively. It can be seen how variations of the repeating theme in (a) result on gaps in the visualization that can be easily confused with the representation of an alternating theme, as in the case of (b). This is also partially the result of the density constraints imposed on the diagram, i.e. by definition strophic forms will tend to result in denser diagrams. However, the constraints are necessary to avoid too-noisy or too-sparse diagrams in other styles, and adaptation has proven elusive without prior information about the music. Confusions between binary and ternary forms might be caused by the nuanced and ambiguous difference between AB, ABA and ABA' structures. Thus, more ternary music is miscategorized as binary than the opposite.

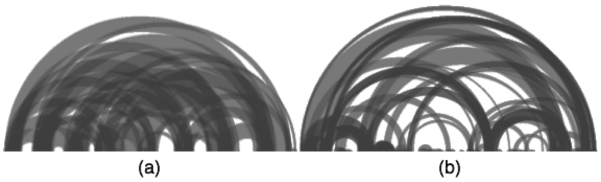

Figure 8. Visualization examples of (a) Strophic form and (b) Rondo form.

5. CONCLUSIONS AND FUTURE WORK

This paper introduces an audio-based, data-driven visualization of music structure, obtained through the use of chroma features, recurrence plots and arc diagrams. Additionally, it presents a preliminary study exploring the ability of the proposed visualizations in helping complex tasks such as music form recognition. Our results indicate that these visualizations can reinforce the identification of musical structures, both reducing the time and increasing the accuracy of the analysis. They provide an efficient way in aiding listeners navigate through music. However, results also indicate that the proposed visualizations are not yet enough, on their own, to convey structural information to users, making them good complements, but not alternative representations, in the analysis of recorded music.

To address these issues, we are currently working on incorporating other musical attributes into the visualization. Previous work has discussed the multi-dimensional nature of musical structure, with important cues provided not only by harmonic and melodic patterns (which the chroma features attempt to characterize), but also by rhythmic and textural characteristics. The data-driven nature of the process means that we can easily compute these visualizations from features such as MFCCs or so-called tempograms [20], which are intended to represent those attributes. However, integrating multiple dimensions into an intuitive diagram is far from trivial and we are actively investigating different color-scheme and layering strategies to solve this problem.

Additionally, we are planning to embrace the multi-scale nature of music, allowing users to interactively navigate across micro and macro readings of the data according to their information needs, e.g. allowing users to zoom into the structure of a song at the phrase-level, to visualize local patterns of recurrence. A long-term goal is to integrate multi-scale analysis at the collection, work and track level.

ACKNOWLEDGEMENTS: This material is based upon work supported by the National Science Foundation (grant IIS-0844654).

6. REFERENCES

[1] J. Donaldson and P. Lamere: "Using Visualizations for Music Discovery," In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR-09)*, Kobe, Japan. October 2009.

[2] E. Pampalk, A. Rauber and D. Merkl: "Content-based Organization and Visualization of Music Archives," In *Proceedings of ACM Multimedia*, pages 570579. ACM 2002.

[3] F. Morchen, A. Ultsch, M. Nocker and C. Stamm: "Databionic Visualization of Music Collections According to Perceptual Distance," In *Proceedings of the International Symposium on Music Information Retrieval (ISMIR-05)*, London, UK. September 2005.

[4] A. Lillie: "MusicBox: Navigating The Space of Your Music," MS thesis, Massachusetts Institute of Technology, MA, USA, 2008.

[5] K. Siedenburg: "An Exploration of Real-Time Visualizations of Musical Timbre," CNMAT, 2009.

[6] P. Toiviainen: "Visualization of Tonal Content with Self-organizing Maps and Self-similarity Matrices," In *Computers in Entertainment (CIE)*, Volume 3, Issue 4, October 2005.

[7] M.Goto: "A Chorus-section Detecting Method for Musical Audio Signals," In *Proceedings of the 2003 IEEE Conference on Acoustics, Speech and Signal Processing*, 2003.

[8] J. Paulus and A. Klapuri: "Music Structure Analysis by Finding Repeated Parts, In *Proc. 1st ACM Audio Music Comput. Multimedia Workshop*, Santa Barbara, CA, Oct. 2006, pp. 5968.

[9] M. Levy, K. Noland and M. Sandler: "A Comparison of Timbral and Harmonic Music Segmentation Algorithms," In *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2007.

[10] M. Wattenberg: "Arc Diagrams: Visualizing Structure in Strings," In *Proceedings of the IEEE Symposium on Information Visualization (InfoVis02)*, 110, 2002.

[11] J. C. Brown and M. S. Puckette: "An Efficient Algorithm for the Calculation of a Constant Q Transform," *J. Acoust. Soc. Am.,* 92, 2698-2701, 1992.

[12] N. Marwan, M. C. Romano, M. Thiel and J. Kurths: "Recurrence Plots For The Analysis of Complex Systems," In *Physics Reports*, Volume 438, Issues 5-6, January 2007, Pages 237-329.

[13] M. Grachten, W. Goebl, S. Flossmann and G. Widmer: "Phase-plane Representation and Visualization of Gestural Structure in Expressive Timing," In *Journal of New Music Research*, Vol. 38, No. 2, pp. 183-195, 2009.

[14] J. Serra, X. Serra and R. G. Andrzejak: "Cross Recurrence Quantification For Cover Song Identification," In *New Journal of Physics*, Volume 11, 2009.

[15] J. Theiler: "Spurious Dimension From Correlation Algorithms Applied to Limited Time-series Data," *Phys. Rev.*, A 34 (3), 2427-2432, 1986.

[16] R. Spell, R. Brady and F. Dietrich: "BARD: A Visualization Tool For Biological Sequence Analysis," In *Proceedings of InfoVis 2003, IEEE*, pp. 219-225, 2003.

[17] http://www.echonest.com/

[18] http://www.processing.org/

[19] J. R. Mathes: *The Analysis of Musical Form*, Prentice Hall, 2006.

[20] K. Jensen: "Multiple Scale Music Segmentation Using Rhythm, Timbre and Harmony," *Journal on Advances in Signal Processing*, EURASIP, 2007.