THE BEST OF TWO WORLDS: RETRIEVING AND BROWSING

Donncha Ó Maidín

Centre for Computational Musicology and Computer Music
Department of Computer Science and Information Systems
University of Limerick, IRELAND
donncha.omaidin@ul.ie

Mikael Fernström

Interaction Design Centre

Department of Computer Science and Information Systems
University of Limerick, IRELAND
mikael.fernstrom@ul.ie

ABSTRACT

This paper describes the combination of two software systems for work with music corpora in electronic formats. A set of algorithms has been developed in *CPN View* (a class library for representing music scores) that deals with music score processing. These facilitate access to the ever-increasing collections of music corpora [1]. The *Sonic Browser* (a browser that uses sonic spatialization for navigating music or sound databases) has been developed to the proof-of-concept and prototype implementation stage. In previous work it has been demonstrated that with the *Sonic Browser* it is up to 28% faster for users to find a particular melody in a set of melodies, compared to visual browsing [2].

1. INTRODUCTION

This study focuses on issues that arise when searching and browsing large databases of melodic music. Arbitrary mappings can be made between component melodies. One example of a meaningful mapping is that derived from measures of melodic similarity. Each melody can be represented as a node of a weighted graph, with the mappings represented as edges. The key issue addressed here is the provision of mechanisms whereby a user, who has already focused on a specific melody, can explore related melodies. One approach is to represent local regions of the underlying graph visually. This involves mapping the multi-dimensional space from the graph onto a two- or three-dimensional representation. With a purely visual representation, it is difficult to give the user useful clues about which of the multiple related melodies to select. The approach taken in this study is to represent the space both visually and aurally. By adding *direct sonification*, the user can explore this space aurally with a new kind of cursor function that creates an *aura* around the cursor. All melodies within the *aura* are played concurrently using spatialized sound. Each melody occupies a different position in the user's visual and aural space. This allows the user to interactively explore the space both visually and aurally, by shifting the visual and aural perspective under cursor control.

2. Music Information Retrieval

Information retrieval and visualization is about specific fact finding, extended fact-finding, open-ended browsing and exploration of availability of information in large and complex data sets. Today there are numerous musical corpora with thousands of tunes. In musicological research it is getting increasingly difficult to deal with these vast sets of tunes and there is a need for new tools to support the work. But, attempting to do so, we first have to realize how complex existing representations of music are, and that existing algorithms used for text strings may not be appropriate. Alternatively, or as a complement, we can choose to create tools that make it easier for people to explore such data sets.

The general framework used in this paper is Ben Shneiderman's 'mantra' for Information Retrieval: *Overview first, zoom and filter, then details on demand* [3]. To achieve this, we first need a visualization and sonification that provide an overview of the entire data set. We then need the ability to zoom in within the set. The next level is to be able to filter information in the set, i.e. to use different forms of queries to select specific regions of the data set. Finally, we need to have details available about individual objects of the data set whenever required by the user. *CPN View* can be seen as a set of advanced functions for filtering, while the *Sonic Browser* primarily serves the first and last stage - a tool for overview or details on demand.

In Music Information Retrieval it is recognized that we need to search databases of music using melodies as the base for queries. The application areas range from the collection of musical corpora in ethnomusicology, through the development of copyright checking tools, to studies in musical style. It is important that music similarity algorithms have perceptual relevance, i.e. what the algorithm delivers as a measure of similarity or difference between tunes is closely related to a corresponding humans' perception. Issues include how to design mechanisms for modeling human reactions to key transpositions and to syncopation transformations. Comparing tunes is more complex than just comparing strings of text. Melodies carry more multi-dimensional information than ordinary text strings. Examples are pitch, duration, dynamics and timbre. Additionally music score entities exist with scopes that define their meaning. Examples of scoping mechanisms involve clef, key and phrasing.

Music Information Retrieval is therefore a complex and difficult process.

3. Comparing melodic distance

A number of algorithms have been proposed for calculation of melodic difference or similarity. Examples include Dynamic Programming Algorithms [4], Contour Algorithms (CA) and a Geometric Algorithm (GA) [5]. DPA is based on string matching, i.e. the edit distance between a source string and a target string. A 'cost' is calculated in terms of what and how many edit operators that are used to turn source string into a target string. Edit operations such as delete, insert and replace, may be individually weighted. Melodic similarity between a source and a target melody is calculated as the minimal sum of the weights that result from transforming the source melody into the target melody. On the other hand, the CA algorithm involves representing melodies in terms of rising, falling or stationary pitch patterns to some level of granularity, and effectively comparing the resulting strings. GA uses two iterators to compare two melodies of the same length or for a selected melodic fragment to be compared with a melody. The algorithm is also using weights for duration and stress. Equation 1 below outlines the algorithm.

p_{1k}	pitch of the note from the first segment at the k^{th} window
p_{2k}	pitch of the note from the second segment at the k^{th} window
w_k	width of window k
w_{sk}	weight derived from metrical stress for window k
m	number of semitones that the second tune segment is transposed to minimize the difference

$$difference = \sum_{k=1}^{n} \left| p_{1k} - p_{2k} - m \right| w_k w_{sk} \quad (1)$$

The resulting sum represents the melodic distance between two melodies or fragments. However, if we have for example three tunes A, B and C, the distance between A and B and the distance between B and C does not imply that the distance between A and C is the same as the sum of A and B plus B and C. This makes simple visualization difficult.

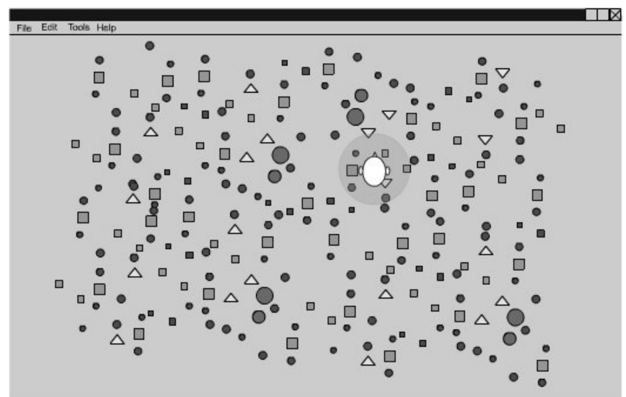
4. Sonic Browsing

Browsing in this context is defined as "an exploratory, information seeking strategy that depends upon serendipity … especially appropriate for ill-defined problems and for exploring new task domains" [6]. This is often the case when searching for melodies in a musical database. People have an outstanding ability to recognize similarities in this domain, which suggests that a good solution should make use of our auditory abilities.

In everyday listening one is often exposed to hundreds of different sounds simultaneously and is still able to pick out important parts of the auditory scene. With musical sounds or melodies, many different factors affect our ability to differentiate and select between the sources. Using instrumental sounds, the timbre, envelope, tonal range and spatial cues support the formation of *auditory streams* [7]. The melodies themselves also assist the formation of streams, as music has its own inherent syntactic and semantic properties [8]. It is also important to note that "cocktail party" effect allows us to switch our attention at will between melodies [9], [10], [11].

With multiple auditory streams it is interesting to note the problem with differences in the individual ability to differentiate between multiple sound sources. A metaphor for a user controllable function that makes it visible to the user is the application of an *aura* [12]. An *aura*, in this context, is a function that indicates the user's range of interest in a domain. The aura is the receiver of information in the domain.

Figure 1. Sonic Browser - Overview



The visual appearance of representations of melodies in a data set can be arbitrarily mapped to properties of the data set. In Figure 1 you have a number of melodies represented by circles, squares and triangles. The shape of an object can be chosen arbitrarily by the user to represent some property of the melodies in the set. Each object can also have color, size and location mapped to properties of the melodies, e.g. melodic distance. The larger gray circle is the *aura*, with a head-shaped cursor in its center. All objects within the *aura* will play concurrently, spatialized through stereo panning. When the user moves the cursor, with the *aura*, the panning of melodies change accordingly. The *aura* can at any time be turned off to pin-point individual tunes. The user can also change the diameter of the *aura* to adjust it to the individual's capability of listening to multiple melodies simultaneously. This feature is also functionally close to zooming in and out of the data set.

By providing fast and direct access to the melodies, users can easily explore a number of tunes in parallel. With tight coupling between the visual and auditory information, users rapidly get a good spatial idea of what objects that are available and how to navigate between them.

In the score view of a melody the user can formulate a query or perform an analysis (Figure 2). A fragment of interest can be selected and the floating control panel used to invoke the analysis.

After analysis of the selected set of melodies, those that matched the query are displayed (Figure 3). The user can again browse the melodies.

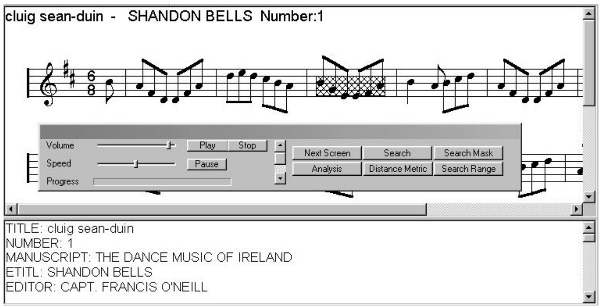


Figure 2. Formulating a query

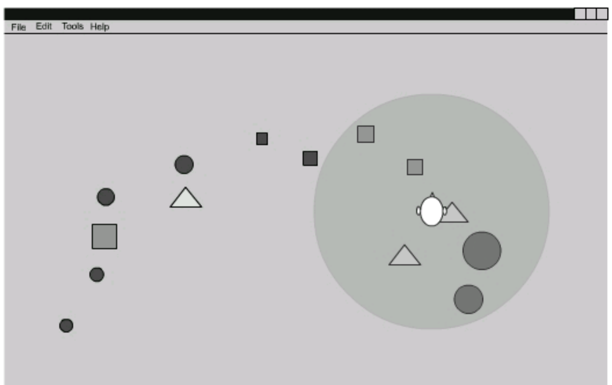


Figure 3. Browsing detail

5. CONCLUSION

In this paper we have presented a prototype of a new system for interactive music information retrieval. The system has powerful algorithms for analyzing melodic distance and an interactive sonification and visualization. The combination of these two approaches facilitates advanced and engaging access to musical databases. The present prototype clearly demonstrates the power of the system, while several additional functions remain to be implemented and tested, such as full zoom in/out, sliders for controlling parametric range selection and more algorithms for musical analysis. Informal user testing has indicated a high degree of satisfaction, while a more formal user testing remains to be done.

6. Future directions

The current version supports music in common music notation. In future versions it would be interesting to experiment with different kinds of music representation, e.g. for timbral music (e.g. Didgerydoo) or music that is using different scales [13].

As the algorithmic comparison of melodies is computationally quite demanding, it might be interesting to separate applications like this into a client-server model with a light-weight client implemented for example in Java. The server would have to create the auditory spatialization in real-time for each client, a goal that appears to be achievable with today's computer power and networking. This would open up the possibility to use applications like this via the Internet.

The fact that melodic distances are difficult to represent in an Euclidean space requires further research into new kinds of visualizations, for example, through relaxed self-organizing layouts [14].

Techniques exist for estimating features like scale, mode, key and tonality, and hence, there is scope for *dynamic queries* to make visualizations even more interactive. Selection boxes and sliders can be added to give the user direct control over the type and range of the selected parameters that control aspects of visibility and audibility [15].

7. REFERENCES

[1] Ó Maidín, D., "Introduction to C.P.N. View", Technical Report UL-CSIS-98-03, University of Limerick, 1998.

[2] Fernström, J. M. and C. McNamara, "After Direct Manipulation - Direct Sonification", Proceedings of ICAD '98, Glasgow, Scotland, Springer-Verlag, 1998.

[3] Shneiderman, B., "Designing the User Interface", Addison-Wesley, Reading, MA, pp139-140, 1997

[4] Kruskal , J.B., "An Overview of Sequence Comparison", in *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*, D. Sankoff and J.B. Kruskal (eds), Addison-Wesley, Reading, Massachusetts, pp. 1-44., 1983.

[5] Ó Maidín, D., "A Geometrical Algorithm for Melodic Difference", in Computing in Musicology - Melodic Comparison: Concepts, Procedures, and Applications, 11, pp. 65-72, Cambridge, Massachusetts: MIT Press, 1998

[6] Marchionini, G., and B. Shneiderman, "Finding facts versus browsing knowledge in hypertext systems." IEEE Computer 19: 70-80., 1988.

[7] Bregman A.S., "Auditory Scene Analysis: The Perceptual Organization of Sound", Cambridge, MA, USA: M.I.T. Press, 1990.

[8] Serafine, M. L., "Music as Cognition: the development of thought in sound", New York, USA, Columbia University Press, 1988.

[9] Arons, B., "A Review of the Cocktail Party Effect", *Journal of the American Voice I/O Societ*y, vol. 12, pp. 35-50, 1992.

[10] Cherry, E.C., "Some Experiments on the Recognition of Speech with One and Two Ears", *Journal of the Acous-tical Society of Americ*a, vol. 25, pp. 975-979, 1953.

[11] Cherry, E.C. and W. K. Taylor, "Some Further Experiments on the Recognition of Speech with One and Two Ears" *Journal of the Acoustical Society of Americ*a, vol. 26, pp. 549-554, 1954.

[12] Benford, S., and C. Greenhalgh, "Introducing Third Party Objects into the Spatial Model of Interaction", Proceedings of BCS HCI '97, Bristol, UK, 1997.

[13] Serman, M., N.J.L. Griffith and N. Serman, "Tracking Monophonic Music for Modelling Melodic Segmentation

Processes", Proceedings of ICMC 2000, 27th August - 1st September, Berlin, pp 217-220, 2000.

[14] Gross, M.H., T.C. Sprenger and J. Finger, "Visualizing Information on a Sphere", Proceedings of the IEEE Information Visualization 97, pp. 11-16, 1997.

[15] Ahlberg, C. and B. Shneiderman, "Visual Information Seeking: Tight coupling of dynamic query filters with starfield displays", Proceedings of ACM CHI '94, Boston, MA, USA, 1993.

Further information and downloads are available at
http://www.csis.ul.ie/cpn/