

Daniel Han

ISE599

Feb 19, 2004

Summary review of Real-Time Tonal Pattern Extraction and Interactive 3D Visualization

Chew, E and François, Alexandre R.J.

This paper describes the structural components and implementation of MuSA.RT, a 3-D visual interactive tool for examining and analyzing tonal patterns in music. MuSA.RT is a real-time application that takes as input MIDI values (usually from a keyboard) and places them on a model for examining tonality called the Spiral Array. As the notes are played on a midi controller, the midi events are processed and displayed in the spiral array model. Here, relevant information can be readily seen such as the present center effect (for both chord and key) and the view can also be modified via game controller. The MuSA.RT application is implemented using François' SAI framework for integrating varying types of multimedia efficiently and easily.

SAI is software architecture that is suitable for processing generic data streams, and in the case with MuSA.RT, this would be the MIDI events. Quite simply, what this software architecture is a set of programming structures and libraries that aid in design and implementation of a software system. MuSA.RT can be conceptually divided into two parts, tonal processing and graphics rendering. For the tonal processing, the MIDI I/O and computing the C.E. algorithm are the important functions. An example of how this works is as follows. Within the tonal analysis subsystem the MIDI input uses an Input cell which serves as the gateway to the entire software system. This cell produces various pulses, a component within in SAI that encapsulates the MIDI information. This is necessary because the system needs a way handle the input for processing later on. From here, the information is relayed to the other cells for more analysis. For graphics rendering, 3-D visualization and user input in modifying the view are the essential components. The functionality of these are also incorporated to different cells such as the Rendering cell and the Camera control cell. In contrast to the cell components, the Spiral Array is implemented as a node. Instead of processing information, the node contains information such major/minor key, triads and other pitch relationships. Thus this component is naturally tied to the midi input system and the center effect algorithm subsystem.

From the point of a software application, the task of implementing the MuSA seems extremely daunting and time-intensive. Given a model for tonality and certain relationships in of the system, creating a real-time application in addition to a visualization implementation from scratch would make it easy to leave it as simply a white paper on the subject rather than actually implementing it. However, the flexibility that the SAI framework offers allows implementation to be more controlled and more feasible.