TECHNOLOGY, REAL-TIME NOTATION, AND AUDIENCE PARTICIPATION IN FLOCK

Jason Freeman   *Mark Godfrey*

Music Department
College of Architecture
Georgia Tech
840 McMillan ST
Atlanta, GA 30332-0456 USA

ABSTRACT

Flock is a full-evening work for saxophone quartet, dancers, audience participation, electronic sound, and video. In this paper, we discuss the project's design and implementation and evaluate its success with respect to audience surveys from the premiere performances.

1. INTRODUCTION

In Flock (2007), we attempt to reconcile the growing cultural shift toward collaborative models of content creation with the few-to-many model that dominates live musical performance. Technology, for us, is a powerful tool through which to more strongly link the activities of composers, performers, and listeners.

Flock uses novel computer vision and real-time notation systems to delay content creation until the moment of each performance, so that the music can reflect the activities of each show's performers and audience members. Music notation, electronic sound, and video animation are all generated in real time based on the location of musicians, dancers, and audience members as they stand up, move around, and interact with each other.

2. BACKGROUND

2.1. Audience Participation

Flock follows in the tradition of musical works that invite the audience to shape the music performed by live musicians. In Kevin Baird's *No Clergy* [1], computer software stochastically generated notation for each performer based on votes cast by audience members on laptop computers. McAllister et al [6] developed a performance environment in which audience members drew notation on a PDA's touch screen for musicians to play. And Wulfson, Barrett, and Winter [8] created *LiveScore*, in which gallery visitors adjusted knobs on physical controllers to adjust the parameters of a stochastic music notation algorithm.

2.2. Real-time Notation

All of these works rely upon real-time notation systems to dynamically generate visual scores for musicians to read. Such systems provide a powerful tool for connecting musicians and audiences: audiences generate input that drives a software algorithm, and the algorithm generates real-time notation that directs the musicians' performance. Other composers, such as Nick Didkovsky [3], have incorporated real-time notation into works without audience participation; Art Clay [4] uses location as a mechanism to generate the notation, just as in Flock.

2.3. Multi-player Gaming

Though Flock is not overtly competitive, it is still inspired by multiplayer games staged in physical venues. Some of these invite seated audience members to play collaboratively by holding up a colored paddle [2] or shifting in their seats [5], while others take place in a mixed reality space combining location-aware devices with physical game play in a venue, neighborhood, or city.

3. DESIGN AND IMPLEMENTATION

3.1. Positioning System

Flock's positioning system tracks the locations of one hundred audience members, four dancers, and four saxophonists within the space; each person is represented as a single point on the two-dimensional plane of the venue's floor. The system identifies each saxophonist with a unique label but does not distinguish among other points. Our implementation, a computer vision approach developed in Cycling '74's Jitter with Jean-Marc Pelletier's cv.jit library and our own tracking objects, offered us high scalability, fast installation time, and low cost.

We ceiling-mounted a single firewire camera with a fisheye lens; our software then performed lens correction to account for most of the fisheye distortion. This approach was more robust than using multiple cameras with minimal lens distortion; our image skew and stitching algorithms increased setup time, reduced frame rates, and created small image boundary errors.

Each audience member and dancer wore a baseball hat with a white LED sphere to facilitate tracking. Camera iris, shutter, and exposure settings, combined with a luminosity threshold filter, isolated the LED lights on the image, and connected components analysis detected blobs and their centroids. Though we had not originally wanted the audience to wear lighted hats, this improved reliability under variable lighting conditions with minimal calibration, eliminating problems caused by shadows and clumps of people. The novelty of the hats also helped the audience get excited to participate. And we preferred the white LEDs to an infrared solution; in our experimentation, we needed far more and far brighter IR targets in order to track at the same distances and camera skews.

Figure 1. Two saxophonists wear hats with colored LED spheres for computer vision tracking.

Each saxophonist wore a hat (Figure 1) with a unique LED color in its sphere. (Pattern-based identification was not practical because of the extreme skew as targets moved to the edge of the image.) A chromakey produced a "distance image" to represent the similarity of each source pixel to each target color. At this stage, simple blob detection was inadequate, as any noise introduced ambiguity as to the path of each target. Instead, we implemented a particle filter to track each saxophonist.

The particle filer employed importance sampling from a set of particles moving stochastically across an image from frame to frame: the closer a particle to its target, the more likely it was to survive into the next frame. In this way, samples that were statistically insignificant died out, moving the entire system of particles towards the saxophonist. The algorithm reported the location as the weighted centroid of all particles. Based on the approach developed in [7], our custom-built particle filter used the output of the chromakey as the probability distribution from which to sample, giving us control over the noise tolerance by adjusting the chromakey parameters.

The entire algorithm operated at approximately 6 fps on our 2.4 GHz MacBook Pro. While this was generally adequate for our needs, analyzed motion became jerky when targets moved extremely quickly; we compensated with a smoothing filter in the video animation and some connected line segments in the music notation.


Figure 2. PocketPC PDAs are mounted on each saxophone to display the music notation.
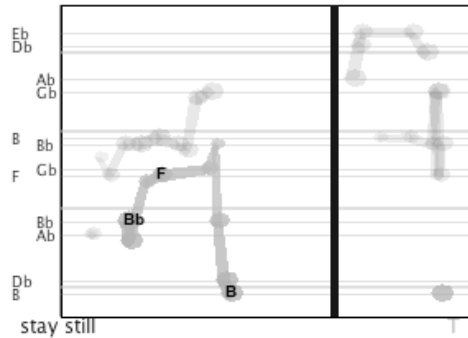
3.2. Music Notation

The position data drove the generation of real-time music notation for each saxophonist. Our Java software, built with Nick Didkovsky's Java Music Specification Language (JMSL) [3], rendered images that were sent wirelessly to PocketPC devices mounted on each player's instrument using marching-band lyres (Figure 2). Our PocketPC software, also written in Java,

decompressed and displayed the images, maintained time sync, and rendered a measure-position bar.
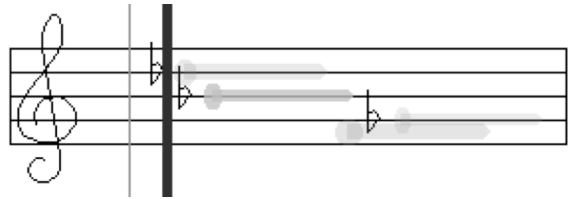
Figure 3. Staff-based music notation. The musician plays the Bb, which is rendered in green. The remaining notes, rendered in pink, show what the other saxophonists play.

Figure 4. Graphical music notation. The musician plays the lower-left contour, which is rendered in green. The remaining gestures, rendered in pink, show what the other saxophonists play.

3.2.1.   Notation Styles

Our software used a variety of approaches to render music notation. In rhythmically sparse sections, it displayed notes on a conventional music staff (Figure 3). Though pitches were exact, time was proportional; horizontal lines extended from solid noteheads to indicate the approximate duration of each note.

Rhythmically denser music was difficult to cleanly render and sight read on a staff, so for these passages, we switched to a graphical view (Figure 4) that emphasized gestural contours. Notes appeared on a time-pitch grid; successive octave registers and a handful of notes were labeled with exact pitches. Diagonal lines often connected nearby noteheads.

Both the staff-based and graphical views displayed only a single measure of music at a time, due to limited screen size and the need to minimize the delay between the music's rendering and its performance. But musicians still had to read ahead. To accommodate, the notation sometimes updated regularly throughout each measure, so that musicians could see the next measure by looking behind the scrolling measure-position bar. Other times, each new measure began rendering, from left to right, about a second before it started, but the notation did not change mid-measure. While the mid-measure changes of the former strategy were sometimes visually distracting, the latter strategy sometimes left musicians with too little time to respond to material at the beginning of the measure. So each approach was used for different sections of the piece, based on the density of musical material and its rate of change.

The size and brightness of noteheads, and of lines connecting noteheads, indicated their dynamic. Articulations were drawn above noteheads as in conventional notation. Each musician's own notes were rendered in green; the other players' music was rendered in a transparent pink. Textual cues instructed musicians how to move and interact on stage and indicated the progression through the structure of the piece.

3.2.2. Data Mapping

When the saxophone quartet was featured alone, each musician generated his own notation via a simple mapping of his absolute location in the space: his x position mapped to the measure position of the note onset, and his y position mapped to its pitch.

For much of the piece, though, dancers and audience members generated notation based on their locations relative to each musician. Each saxophonist was the center of his own polar coordinate system, and any person within a maximum radius generated a note; the distance between them determined the note's pitch, while the angle determined its measure position. In this manner, as more participants got closer to a musician, his notation included more events per measure.

In many sections of the performance, notation was also generated from a history of data going back several seconds; participants essentially left trails on the notation as they moved. This helped to create more continuous gestures in the notation, build up more complex and interesting musical textures, and facilitate more gradual musical change from measure to measure.

3.3. Electronic Sound

While Flock's focus was on the acoustic sound created by the saxophonists, some sections used electronic sound to underscore the saxophones or to create music while they were offstage. In these sections, each audience member's position generated a single synthesized note per measure: x corresponded to measure position and y to pitch.

Our software, written in Cycling '74's Max/MSP, relies on CNMAT's spectral manipulation objects and on Brad Garton's RTcmix external for physical models of plucked string and struck percussion instruments. Small variations in audience positions varied the timbres, dynamics, and envelopes of the sounds, and each note was diffused over an eight-speaker system according to the person's location.

3.4. Structure and Organization

Our biggest challenge in developing Flock was not the technical implementation but the structural framework for the participation of the four musicians, four dancers, and one hundred audience members and the progression of that framework over the work's 60-minute duration.

3.4.1. Musical Structure

The pitches notated for the saxophonists and synthesized in the electronic sound were quantized to a pitch set. Over the course of each performance, the software moved through a circular progression of 28 pitch sets; adjacent sets always differed by a single pitch class. With graphical notation, the saxophonists did not limit themselves exclusively to the notes in the pitch set, but the labeled notes indicating important points of arrival always fell within the set.

The performance was divided into five sections performed without pause: solo saxophone, saxophone quartet, quartet with dancers, audience with dancers, and everyone. Together, these sections created a progression from sparser to denser musical textures, from fewer to more participants, from organized movement to more chaotic behavior, and from fully-notated music to guided improvisation. Each individual section functioned as a small arch by moving up and then down through these same continuums; the peak of the arch for each successive section was higher than the previous.

It is difficult to describe the music itself, since the musicians' response to the notation had a tremendous effect on style and content. At the premiere performances in Miami, the music ranged from pointilistic bursts and slowly-changing drones to rhythmically dense textures full of sudden register shifts, undulating arpeggios, and multiphonics, calling to mind influences ranging from Terry Riley and Steve Reich to Ornette Coleman and Evan Parker.

3.4.2. Dancers and Audience Members
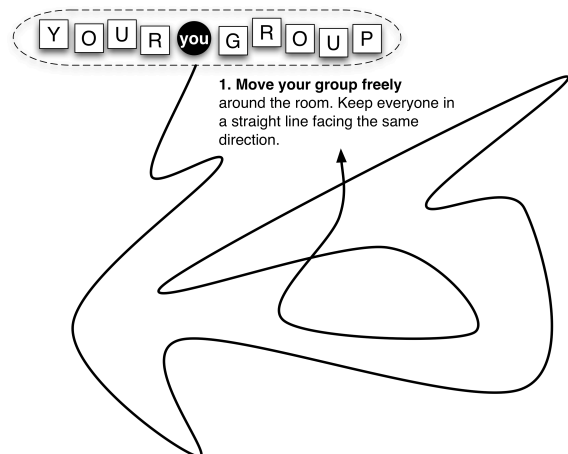
Figure 5. One of the instruction cards given by the dancers to audience members.

During test runs of Flock before its premiere, we discovered that the audience needed guidance in their participation or they became frustrated. To address this, we invited four dancers to assist in the performance.

Initially, the audience remained seated as the saxophonists and dancers performed, introducing the key musical and conceptual ideas for the piece. Over the remainder of the performance, the dancers invited audience members out of their seats, organized them into groups, guided them through movements, and handed out instruction cards (Figure 5) directing those groups to move, interact, merge, and divide in various ways. By the end of the performance, the audience was finally free to move around on their own.

During our test runs, we also found that the creativity of participants was inversely proportional to their number. So the dancers limited the number of simultaneous people on stage, constantly encouraging new people to participate and others to sit down and be spectators. In some sections, there were only a handful of audience members on stage; in others, there were as many as thirty people.

4. EVALUATION AND DISCUSSION

4.1.1. Audience Surveys

Approximately 400 audience members attended five performances of Flock in Miami in December 2007. We asked every audience member to fill out a short survey about their experience; fifty people did so. Survey respondents indicated their feelings about twelve statements with a numerical response (1=strongly disagree; 5=strongly agree), and several added free-response comments. Most respondents were not trained as musicians and did not regularly listen to experimental music; their interpolated median responses to these statements were 1.32 and 1.77, respectively.

While they did have fun (4.61) and enjoy participating (4.58), audience members were divided over whether they had been creative (3.40) and whether the performance would have been different without them (3.20). (We did observe much creativity, including spontaneous ballroom dancing and conga lines.) While many audience members understood what was expected of them (3.92) and how they helped shape the performance (3.58), several wished that the experience had been more thoroughly explained beforehand.

Respondents expressed conflicting feelings about the role of the dancers and instruction cards in organizing their participation. One person never realized that "I didn't have to do just what the dancers told me to – that I could move around." Another complained that it "felt very unorganized, chaotic, unstructured." Someone else noted that "even with the points at which it wasn't quite clear what to do…the ambiguity was part of the fun."

4.1.2. Future Work

Our technical systems for Flock are robust, but we continue to develop the interactive framework and structure as we prepare for upcoming performances. We have learned that we need to explicitly explain more to the audience from the beginning, that gestural and even whispered instructions from the dancers are often clearer and more interesting than written ones, and that we must constantly monitor the pacing of the performance. More generally, we continue to seek a balance between organized and free movement, between more transparent mappings and more compelling music; and between the audience's role as participants and spectators.

These are challenging problems, in particular because experimentation necessitates the presence of a full audience. But this is also part of the reason this project excites us: each performance brings fresh people, new social dynamics, and new insights.

6. REFERENCES

[1] Baird, K. "Real-Time Generation of Music Notation via Audience Interaction Using Python and GNU Lilypond." *Proceedings of the 2005 International Conference on New Interfaces for Musical Expression (NIME)*, Vancouver, 2005.

[2] Carpenter, L., and Carpenter, R. "Audience Participation." *Ars Electronica: Facing the Future*, ed. T. Druckrey. Cambridge: MIT Press, 1999.

[3] Didkovsky, N. "Recent Compositions and Performance Instruments Realized in the Java Music Specification Language." *Proceedings of the 2004 International Computer Music Conference*, Miami, 2004.

[4] Gutknecht, J., Clay, A., and Frey, T. "GoingPublick: Using Realtime Global Score Synthesis." *Proceedings of 2005 International Conference on New Interfaces for Musical Expression (NIME)*, Vancouver, 2005.

[5] Maynes-Aminzade, D., Pausch, R., and Seitz, S. "Techniques for Interactive Audience Participation." *Proceedings of the IEEE International Conference on Multimodal Interfaces*, Pittsburgh, 2002.

[6] McAllister, G., Alcorn, M., Strain, P. "Interactive Performance with Wireless PDAs." *Proceedings of the 2004 International Computer Music Conference*, Miami, 2004.

[7] Nummiaro, K., Koller-Meier, E., and Van Gool, L. "An Adaptive Color-Based Particle Filter." *Image and Vision Computing*, 21, 2003.

[8] Wulfson, H., Barrett, G., and Winter, M. "Automatic Notation Generators." *Proceedings of the 2007 International Conference on New Interfaces in Musical Expression (NIME)*, New York, 2007.