

# Tree Music: Composing with GAIA

Judith Shatin  
McIntire Dept. of Music  
Box 400176  
University of Virginia  
011-434-924-3052  
[shatin@virginia.edu](mailto:shatin@virginia.edu)

David Topper  
McIntire Dept. of Music  
Box 400176  
University of Virginia  
011-434-924-7355  
[topper@virginia.edu](mailto:topper@virginia.edu)

## ABSTRACT

In this report, we discuss *Tree Music*, an interactive computer music installation created using GAIA (Graphical Audio Interface Application), a new open-source interface for controlling the RTcmix synthesis and effects processing engine. *Tree Music*, commissioned by the University of Virginia Art Museum, used a wireless camera with a wide-angle lens to capture motion and occlusion data from exhibit visitors. We show how GAIA was used to structure and navigate the compositional space, and how this program supports both graphical and text-based programming in the same application. GAIA provides a GUI which combines two open-source applications: RTcmix and Perl.

## Keywords

Composition, new interfaces, interactive systems, open source, Real time audio, GUI controllers, video tracking

## 1. INTRODUCTION

Interactive pieces take a wide variety of approaches. Often, they involve response to the surrounding environment. An intriguing example is Garth Paine's *Reeds*, an ecoacoustic piece in which weather data is collected from pods floating on a lake, analyzed, and used to control sound synthesis software that creates sound that emerges from the "reeds" installed in the lake.<sup>5</sup> Other examples include Ambrose Field's *Aqueduct Zero*, a room-responsive environment in which combinations of environmentally-based sounds seep into spaces as they are vacated by exhibit visitors.<sup>3</sup>

Other approaches involve spectators as performers. Todd Winkler's installation *Magic Mirrors* captures spectators' movements via video tracking, and they are mapped and reflected in dynamically changing visual images.<sup>11</sup> *ToneTable*, described by John Bowers, also works with spectator movement. Here, however, they move virtual objects above a virtual liquid field, with sounds associated with each object.<sup>1</sup>

Another genre focuses more specifically on the individual performer. Examples include Matthew Burtner's *S-Morphe-S* (2002) in which his performance on the *MetaSax* is used to control a physical model of a singing bowl,<sup>2</sup> and Atau Tanaka's *BioMuse*, which draws data from his own muscle movement to develop and transform the unfolding music in *Rail*.<sup>10</sup>

These pieces all require software for tracking, analyzing and mapping complex data. The most common programs for such interactive analysis and mapping include MAX-MSP, PD<sup>7</sup> and Super Collider. There are a variety of tracking programs, such as VNS2, David Rokeby's program.<sup>8</sup> Custom devices abound as well, developed by composers, such Butch Rován's data glove<sup>9</sup>,

or by researchers, such Joe Paradiso's dance shoes.<sup>6</sup> Often these devices link to one of the programs mentioned above, as is the case with Rován's glove and MAX-MSP.

GAIA provides a new open-source application that combines a robust GUI environment with the functionality of RTcmix<sup>4</sup> Perl scripts. Unlike other similar applications, GAIA supports both GUI and script-based control in the same program. It is capable of supporting the kinds of composition described above, and those detailed below.

*Tree Music* responds to the changing environment created by visitors to a sculpture exhibit. Video tracking is the primary interface between the audience/performer and the system; motion and occlusion data are captured via movement by exhibit visitors. Occlusion represents the percentage of pixel change from a starting frame (in this case the gallery without people) to the current frame. Motion is calculated as the percentage of pixel change from the previous frame. GAIA uses this information in a hierarchy of responses. These include a) performance of mini set-pieces; b) dynamic creation and performance of micro phrase-ensembles; c) interactive musical response to people's movement, with particulars depending on which interactive region is operative and d) extended polyrhythmic performance during extended periods with no change of motion or occlusion (i.e. when the exhibit area is empty for more than five minutes). The exhibit visitors thus become collaborators in the creation of a dynamic compositional interface.

## 2. GAIA

GAIA is a graphical network-style application for controlling musical processes. Specific structures (called objects) are created on a blank screen (called a canvas) and connected together like objects in a multi-node graph. Signals are sent from parents to children. The basic concept is similar to that employed by other graphical network based control applications (e.g., Max/MSP, PD, Open Music, Alsa Modular Synth, GLAME). GAIA differs from such applications on several levels. It is open source, and distributed under the GNU General Public License (GPL). Synthesis methods are not "sub patches" within the program, but are instead RTcmix "instruments." Perhaps the most notable difference is GAIA's incorporation of Perl directly within the graphical structure. Users can combine any level of graphical or scripted control. GAIA offers an API for third party developers to facilitate the creation of new objects and features within the program. Recent work has focused on video processing, now part of the main GAIA distribution. Video data is broadcast from the video-tracking object when it receives an input signal.

GAIA objects fall into three general categories: objects specific to GAIA, individual RTcmix commands and Perl scripts. GAIA-specific objects include the standard range of sliders, data boxes, timers, 2D graph plots, and logical operators. Singular RTcmix statements can be sent either to a running RTcmix engine via a TCP socket connection, or directly via the RTcmix “imbed” module. The most powerful feature of GAIA is its use of Perl within a GUI environment.

Combining RTcmix and Perl with a graphical front end represents a new paradigm. Here, Perl does not parse text files. Instead, it parses and evaluates a buffer in real time, maintaining a persistent state during the entire run of the application. Perl provides built-in functionality for this.

Perl objects are represented on the screen by a special purpose widget containing input and output connection boxes, as well as show and hide buttons which access a text editor which spawns external to the main GAIA canvas. Within a script, there are special reserved variables: \$inX and \$outX, where X corresponds to the respective input or output connection box number. GAIA also uses Perl for logical expression evaluators. These objects, called Perl signal condition objects, are similar to Perl text buffer objects. The main difference is that they take input data and are evaluated for boolean true/false condition. If true, the signal and data are passed along to the object's outputs.

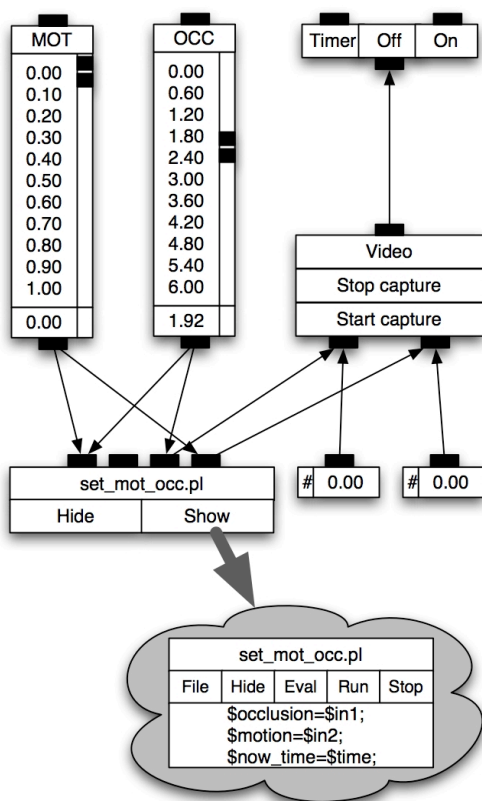


Figure 1: Snapshot of GAIA’s video-tracking object

### 2.1 Video Processing

By reading pixels from a live video feed it is possible to implement computer vision algorithms to control audio synthesis

and effects processing. GAIA uses a video-tracking object to read in pixel data from a camera, then send out several different types of data which can be used by the rest of the running GAIA process. Currently this object performs frame difference calculations from a start frame (called occlusion) and from the previous frame (called motion). The video tracker object takes a trigger as input, and sends out visual analysis data of the incoming video stream.

With each signal sent to the video tracking object, a single frame is read and analyzed. Currently up to thirty frames per second can be processed, but this is only a hardware limitation imposed by commercial video equipment. Theoretically, more sophisticated video-frame grabbers can also be used to achieve an even higher data acquisition rate.

### 3. TREE MUSIC DESIGN

Work on the conceptual design of *Tree Music* raised numerous compositional questions. How could the music respond to the sculpture it was to accompany? What types of response should the installation make to exhibit visitors? What should happen if the room is empty for an extended period? How should the timing of and type of response relate? How could various compositional states be monitored? Answers to these questions flowed from an initial decision regarding sound sources, and from prototyping with GAIA.

First, a sound palette was created from recordings of the sculptor at work, thus embodying the sculptural process in the sound. Families of sounds were then developed, ranging from the recognizable sounds of saw, chisel and broom, to sounds whose processing breaks all ties to their sources.

Second, tiered levels of system response were organized. These range from a relatively-fixed system, in which the computer picks from a collection of mini set-pieces and plays them, to the more spontaneous creation of ensembles of micro-phrases, to interactive sections in which the visitors more overtly effect the sonic environment. To move between these levels, and to control the timing between them, a system of states and holds was created in GAIA. Interactive monitoring of changes in motion and occlusion generated results which were then interpreted to choose when and which section to enter. And, depending on which section, different compositional behaviors resulted.

Finally, a type of music was created that would sonically respond to the sculpture and exhibit space only during extended human absence. Humans might hear the resultant music from other rooms, or not. It is only during certain of these timespans that the voices of the sculptor and composer emerged, replacing the missing human component and creating an intermingled sonic representation of the installation’s creators.

The composition of *Tree Music* involved four phases: 1) sound source development; 2) sectional design of the relatively fixed, open and interactive regions, as well as ways to trigger them and create multiple paths within and between them; 3) construction of a top-level control system that would link and time motion between the sections. 4) testing and scaling of the program on site. The latter was particularly tricky, because a substantial range in the number of potential visitors (ca. 1 – 20) needed to be taken into account. There were two testing periods in the museum during which data was gathered for this scaling. RTcmix was used for the sound source development and GAIA for the rest of these tasks.

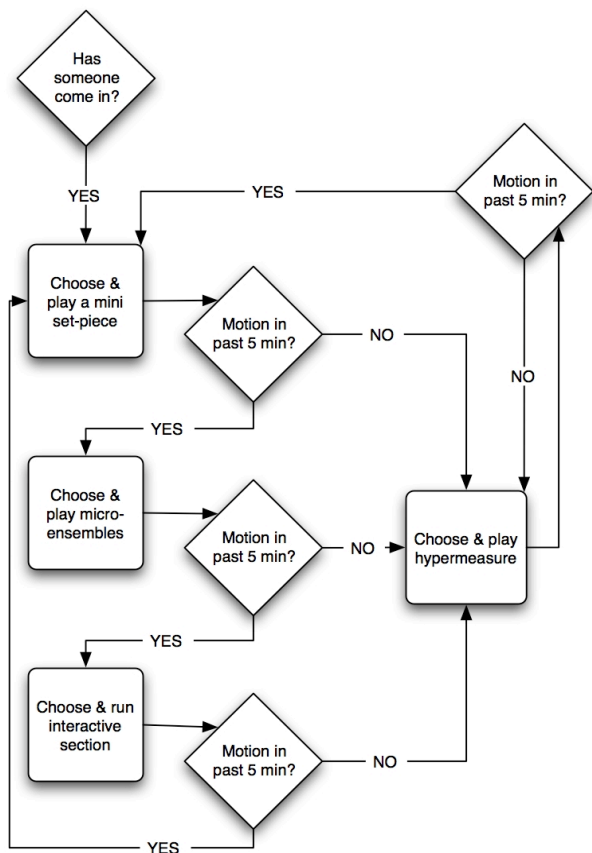


Figure 2: Tree Music Flow Chart

### 3.1 Sectional Organization

Figure 1 displays the four main sections. Each has its own performance mode, with pauses of controlled-random duration between them. And, if there is no change in motion in the room for five minutes, the music shifts to Hypermeasure mode, as described below. Figure 3 shows the GAIA config for *Tree Music*.

*Lead-follow*: When someone first enters the space, or re-enters after a silence of at least five-minutes, one of a collection of mini set-pieces is triggered. Sharply contrasting sonic profiles were created for these, so that repeated experiences of the piece would yield strikingly different encounters. These set pieces have a duration of one-two minutes. After one is played, the camera continues monitoring change in motion and occlusion. If none is detected after five minutes, the program jumps to the Hypermeasure mode; otherwise it continues to the *Solo* section after a brief pause, whose exact length differs within a set range.

*Solo-Ensemble*: After a brief pause (4-8 seconds), if people continue to move around the space, micro-phrase ensembles are performed, with controlled-random choices of soundfiles and layering based on low, medium and high density families of soundfiles. The resultant ensembles, again configured for a wide variety of listening experiences, are currently set to durations of one-half to three minutes.

*Interactive*: Continued movement sensed after section two completes triggers an interactive mode. One of four interactive segments is activated. Parameters such as pitch, register and timbral effects will change depending on the change in motion and occlusion in the room. There are four interactive scripts that respond to motion. One is chosen at random when the “interactive state” signal is broadcast. The next time through the system, a different script will be chosen, until all have been performed. One script triggers a soundfile and processes it through a comb filter. As motion increases, the register changes. A second script increases the amplitude level in response to increased motion. A third combines effects by changing both register and strength of a comb filter. A fourth interactive section changes both comb filter register and tempo in conjunction with movement.

*Hypermeasure*: An absence of motion or occlusion change for five minutes triggers one of a collection of extremely extended polyrhythms, which unfold over timespans ranging from five to thirty-two minutes. Each has particular soundfiles associated with it. It is during these polyrhythms, for example, that the artists can be heard speaking, leaving the most overt traces of the project creators in the absence of other people, and, in a way, standing in for them.

### 4. FUTURE DIRECTIONS

GAIA is a new application, but one with strong potential. The ability to harvest data from multiple devices makes this a system that will serve interactive pieces that respond to elements of the environment, as well as those based on elements of a performer’s actions. One can imagine a host of future applications for GAIA, as well as the development of a broad community of users.

A main focus in the short term is on the creation of new objects. There are many yet to be explored, including interactive graph plotters. More video tracking algorithms are also planned, such as color sensing, object recognition and positioning. As a corollary to GAIA’s new video processing object, a video output object is currently under development.

In addition, while RTcmix has been successfully ported to OSX, work remains to be done on support for X windows based libraries like the GTK canvas, before GAIA can be ported to OSX.

GAIA can be downloaded via ftp from the Virginia Center for Computer Music at: <http://presto.music.virginia.edu/pub/gaia>

### 5. ACKNOWLEDGMENTS

Thanks to sculptor Emilie Brzezinski for her collaborative spirit; to the helpful staff of the University of Virginia Art Museum; and to the Museum Director Jill Hartz and the Virginia Commission for the Arts for providing the opportunity to develop this work.

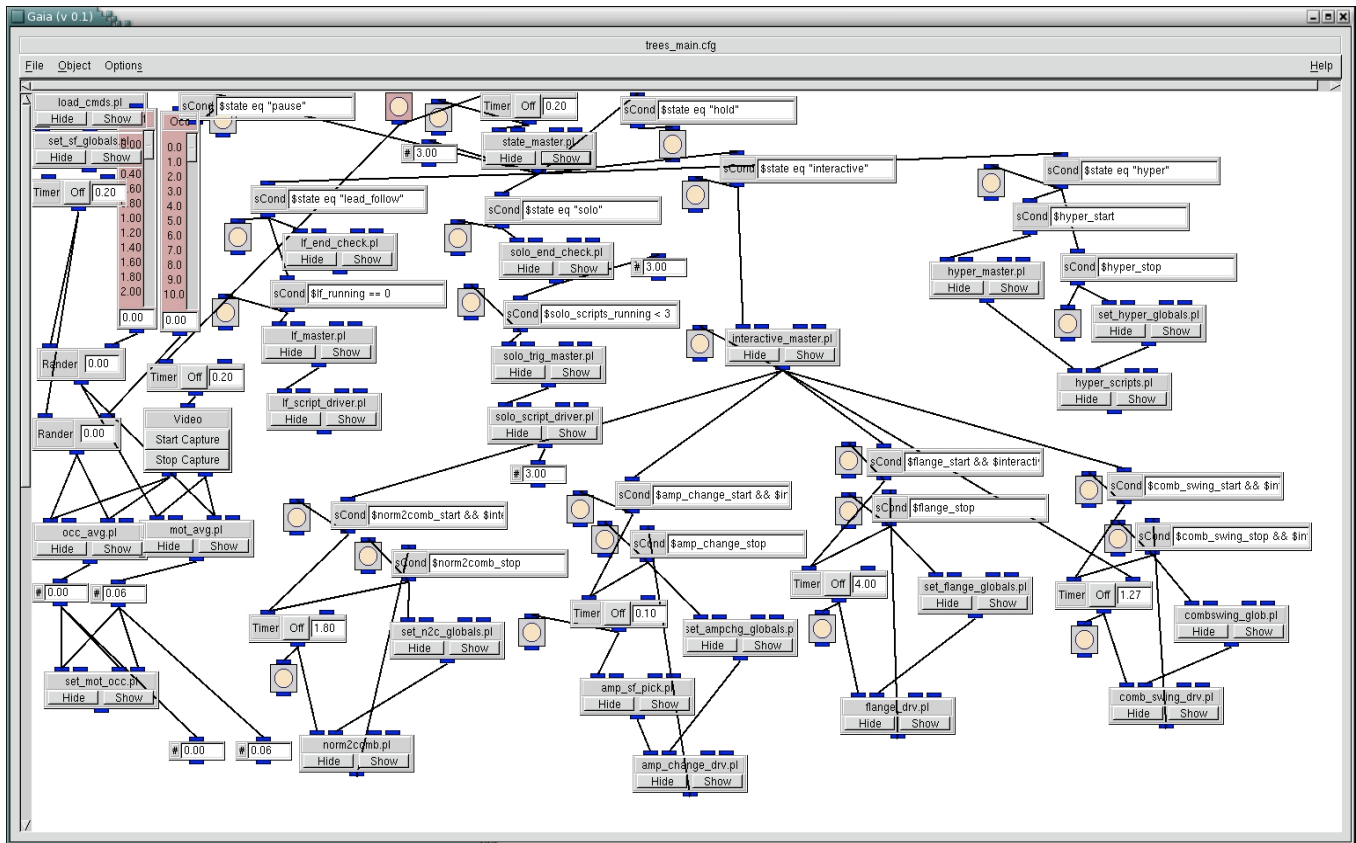


Figure 3: Tree Music Config

## 6. REFERENCES

- [1] Bowers, J. *TONETABLE: A Multi-User, Mixed-Media, Interactive Installation*, Proceedings of the COST G-6 Conference on Digital-Audio Effects (DAFX01) (Limerick, Ireland 2001)
- [2] Burtner, M., Serafin, S., and Topper, D. Real-Time Spatial Processing and Transformations of a Singing Bowl, *Proceedings, 5<sup>th</sup> Int. Conference on Digital Audio Effects (DAFX-02)* (Hamburg, Germany 2002), 63-69.
- [3] Field, A. <http://www-users.york.ac.uk/~aef3/homesite/aquaduct.html> (n.d.). Retrieved 1/3/04.
- [4] Garton B. G., and Topper, D. RTmix -- Using CMIX in Real Time, *Proceedings of the International Computer Music Conference (ICMC '97)* (Thessaloniki, Greece, 1997).
- [5] Gye, L. *Music of the Pods: Garth Paine's Reeds*, <http://www.realtimearts.net/rt40/gye.html> (n.d.). Retrieved 12/14/03.
- [6] Paradiso, J. A, *Electronic Music: New Ways to Play*, Media Laboratory, MIT. IEEE Spectrum, 1997. <http://www.spectrum.ieee.org/select/1297/muse.html>
- [7] Puckette, M. Pure Data, *Proceedings, International Computer Music Conference (ICMC '96)* (San Francisco, 1996), 269-272.
- [8] Rokeby, D. <http://homepage.mac.com/davidrokeby/softVNS.html> (n.d.) Retrieved 12/18/03.
- [9] Rován, B. <http://www.soundidea.org/rovan/index.html> (n.d.) Retrieved, 1/17/04.
- [10] Tanaka, A. <http://www.gyoza.com/ate/atau/html/index2.html> (n.d.) Retrieved, 1/14/04.
- [11] Winkler, T. [http://www.brown.edu/Departments/Music/faculty/winkler/installations/magic\\_mirrors/index.html](http://www.brown.edu/Departments/Music/faculty/winkler/installations/magic_mirrors/index.html) (n.d.). Retrieved 1/3/04.