# *e-Motion: Our Reality* - 3D Motion Capture and Sonorization Via Two Cameras

**Bradford Blackburn**

University of Tampa, Department of Music, Tampa, FL, USA

bblackburn@ut.edu

***Abstract.*** *This paper will present details about a two camera motion capture system that facilitates the tracking of movement in 3D. The system was developed for use in e-Motion: Our Reality, a real-time interactive music, dance, and video installation. Practical solutions to problems encountered with a two camera system are discussed. The specific sonorization techniques for dance used in the installation are also examined.*

## Synopsis of the Project

e-Motion: Our Reality was an inter-disciplinary collaboration presented at the Krannert Art Museum, on the campus of the University of Illinois, Urbana-Champaign. It took place in conjunction with the “Here and Now” exhibition which featured regional artists and their work. The collaborators on this project included Bradford Blackburn (music composer), Elizabeth Johnson (dancer), Hank Kaczmarski (engineer), Ya-Ju Lin (dancer), Jessica Ray (dancer), Benjamin Schaeffer (programmer), Cho-Ying Tsai (dancer), and Luc Vanier (choreographer). Hank Kaczmarski is the Director of the Integrated Systems Laboratory at the Beckman Institute of the University of Illinois, Urbana-Champaign and coordinated the visual virtual reality portions of the project including the ten camera motion capture system, and the inclusion of 3D images of artworks being displayed in the museum. Ben Schaeffer, a research programmer for ISL, wrote the software that allowed the dancers to manipulate 3D visual imagery in real-time. Luc Vanier, a professor of dance at the University of Illinois, and a choreographer with a deep interest in motion capture technology, worked with the dancers to develop movement that tested and utilized the capabilities of both the graphic and musical virtual reality systems [Vanier, Kaczmarski, Chong, Blackburn, Williams, and de Velder 2003].

The developing choreography inspired new approaches for interfacing with the dancer’s movement and thus a circle of feedback was quickly established between dancers, visual programmers, and the interactive music design. As a result, the project was continuously growing in sophistication, nuance, and organicism with each day’s work. The museum visitors were able to watch this process happen up close, and view the work in progress at daily showings where they were given demonstrations of the technology and were invited to ask questions. For the collaborators this was a great opportunity to get feedback from audience members about their reactions to the technology, and to gain a greater understanding of how people interpret the various relationships between human and machine in an interactive performance.

## Synopsis of the Music

The music for e-Motion was not "composed" in the traditional sense, but was instead a design for an interactive performance space that facilitated the sonorization of the dancer's. motion. Unlike previous work I have undertaken utilizing similar technology, the music for e-Motion differed because it used a three-dimensional motion capture space for the control of the music. To make this possible, it was necessary to use a minimum of two video cameras whose combined perspectives formed a 90° angle when placed on adjacent sides of the marley floor (where the dancer was performing). By overlapping the view of both cameras in this way, the dancer's movement could be viewed in any direction: front-to-back, side-to-side, up-and-down, etc., as opposed to simply looking at their motion in 2D. The output from each camera was analyzed in real-time on separate computers (logically dubbed "Computer A" and "Computer B" for their respective associations with Camera A and Camera B). The data from the analysis of each camera's output was translated into MIDI (Musical Instrument Digital Interface) data using Max (a graphic object programming language for interactive computer music) and then realized in musical sound using an external synthesizer. The sound was diffused according to the dancer's location amongst four speakers arranged quadraphonically around the motion capture space.

## Video Analysis Process

The video analysis and motion tracking program which provided data concerning the dancer's movements was a third-party program written for the Max programming language called Cyclops, authored by Eric Singer [Singer 2002]. It allows for a digitized video signal to be processed in real time in a variety of ways. It works by dividing up each new frame of the digitized video into a grid of a predetermined size (for e-Motion I chose to use an 8x8 grid containing sixty-four blocks for reasons I will explain later). Within each coordinate block contained in the grid the user can include a "zone" function, which designates that a particular kind of analysis is to occur for the block containing the zone. With each successive frame the pixels contained within the block are summed together to produce an average shade or color. Depending on whether any change has occurred, and or what type of analysis has been specified for the zone, a value may be sent out. The values, along with an ounce of imagination, can then be used to control an infinite variety of processes within Max.

The 3D motion capture space for the music was that area where the separate views of the two cameras overlapped, in other words, the shared space from two different perspectives. An area where the cameras' views did not overlap, but yet was still visible to one of the cameras, was considered to be part of the peripheral 2D motion capture space.

In the arrangement used in e-Motion both cameras formed a nearly perpendicular angle with each other, therefore it is possible to think of the 3D motion capture space as similar in shape to a cube (even though its actual geometry was closer to an asymmetric polygon due to the fact that the diameter of the view seen by the video camera widened exponentially with distance from the lens), see Figure 1.

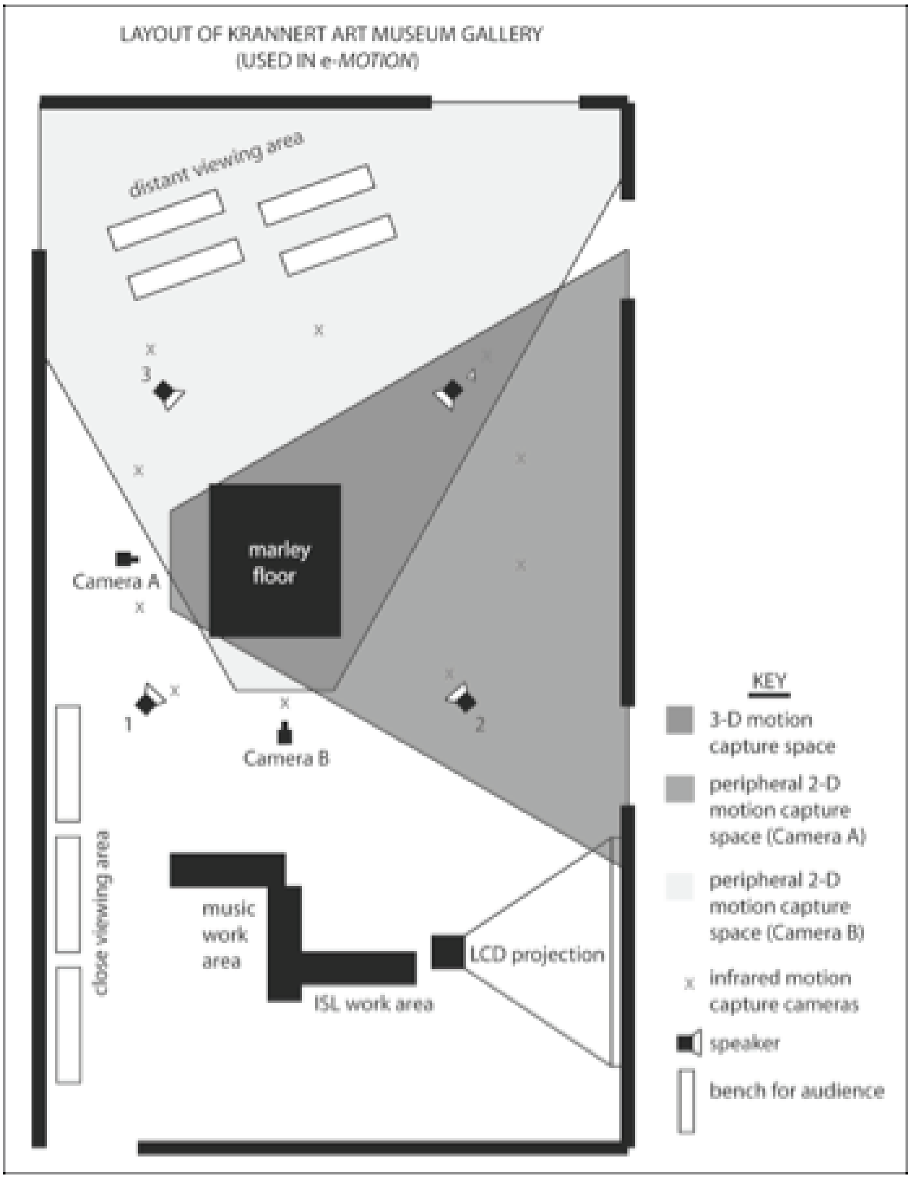

Figure 1. Camera arrangement

For each camera, the 8x8 grid within Cyclops was divided up into four quadrants of equal dimensions (4x4, or 16 blocks each). The zones assigned to each of these blocks were numbered 1 to 64 and distributed so that modular arithmetic could be used to determine which quadrant any particular zone number, that was currently registering a change in values, belonged to (quadrant 1 contained the series: 1, 5, 9...; quadrant 2 contained the series: 2, 6, 10...; quadrant 3 contained the series 3, 7, 11...; quadrant 4 contained the series: 4, 8, 12...). For example, if zone #34 was registering a change in values, than 34 would be divided by 4 to produce a remainder of 2 thus indicating that the change was occurring in quadrant 2. The quadrants for Camera A (A1, A2, A3, and A4) when combined with those of Camera B (B1, B2, B3, and B4) produced an invisible arrangement of 8 cubic sectors within the 3D motion capture space, see Figure 2.

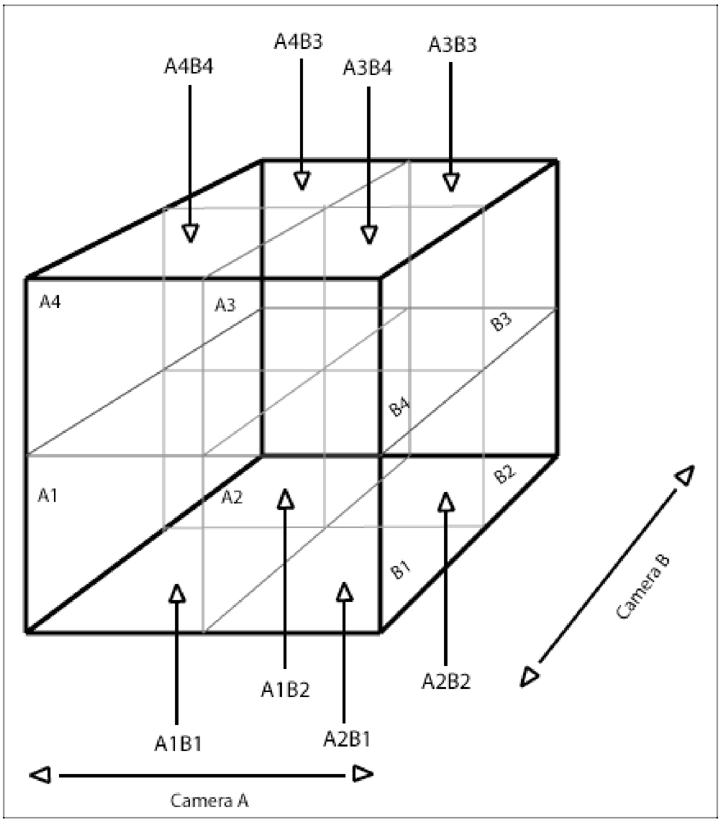

Figure 2. Camera quadrants

The central horizontal axis of both cameras was aimed at the waistline of the dancer when they were standing upright. This allowed for the dancer to isolate their control of the music between their upper and lower bodies or avoid triggering the upper sectors all together by staying below the horizontal axis'. The central vertical axis for Camera A corresponded to the division between stage left and stage right, and the central vertical axis for Camera B corresponded to the division between upstage and downstage. Since these vertical axis' along with the border of the 3D motion capture space and the peripheral 2D motion capture space were invisible to the dancer (except through sound) gaffer’s tape was applied to the floor to delineate the boundary locations. The locations of the central horizontal axis' however, were left to the estimation of the dancer. By comparing the analysis from the images of both cameras it was possible to determine the dancer's general location within the eight sector cubic space (assuming they were not spiraling somewhere in the center where they might trigger all eight sectors simultaneously).

The particular kind of analysis process used for all 128 zones (total between Cameras A&B) was a difference threshold analysis on a grayscale-converted image. Whereby if a change in the average shade value for a particular block, in comparison to the value of the same block for the previous frame, exceeded a given threshold, then a +1 value would be sent out for a shade change towards the white end of the spectrum, and a -1 value would be sent out for a shade change towards the black end of the spectrum. The threshold for the process was set just high enough that only physical motion within the motion capture space would produce sufficient changes in light values that were capable of triggering an output of values from Cyclops. Using the modular operation for quadrant differentiation (previously discussed) it was possible to track the total motion occurring over a specified period of time for a particular quadrant.

The two computers shared much of the same code for kinesthetic analysis and sector differentiation. However, Computer A was allocated the additional control function of sending a synchronization pulse to itself and Computer B every two seconds. With each new pulse the total kinesthetic activity for each sector in the past two seconds was calculated. This allowed for at least a minimum of kinesthetic activity to be recorded and used as musical information, while keeping latency low enough to be relatively inconspicuous. In my experience I have found a little bit of latency actually desirable in setting music to a visual image, whether dance or film, for a variety of reasons. For one, it approximates the way we experience visual and sonic stimuli in the natural world. Another reason is that the interpretation of musical sound events is a much slower and more abstract experience compared to our instant ability to recognize visual stimuli. If the natural relationship is reversed, then the dancer would appear to be following the music and the sense of interactivity would be lost. (Enter the classic rule of effective film scoring, the orchestra swells a moment after the kiss of dramatic culmination.)

## Motion to Music Interrelationships

One of the fundamental questions at the heart of an interactive performance is always how direct and palpable are the relationships between two interacting forces. How clear should the relationships be for the uninformed viewer? Naturally any answer to this question presupposes a lot about the potential audience. In the case of e-Motion the expectation was that most visitors to the exhibition would be seeing this technology for the first time, and would not have preconceived notions about what an interactive dance performance should be. It seemed likely that many would saunter through the museum at a fairly steady pace without pausing to observe any particular exhibit for very long. Since I did indeed want to make the viewer aware of the fact that the dancer had control over the music at some level, and keeping in mind my notion of who the average visitor would be, I sought to make these relationships as clear as possible and chose to establish readily observable 1:1 correspondences between the dancer and the music. In previous interactive dance performances where I have used more convoluted algorithmic processes for creating interaction, the majority of audience members have been almost entirely oblivious as to how the interaction was taking place, even when I have written extensive program notes to explain the kinds of interaction that were occurring. Which begs the question—why bother with the live interactive technology at all, when a recording of anything other than a static sine wave tone might achieve the same result through pure and simple chance? Therefore, using clearly presented connections seemed the best choice for the project.

The 1:1 correspondences that were employed included the dancer's kinesthetic motion as a control for amplitude, duration, and total note events. As the dancer's total kinesthetic motion for each of the eight quadrants was calculated every two seconds, the value from this calculation would replace the previous as the new amplitude for all note events triggered by the dancer's motion within the same quadrant. This new amplitude would only be reached one and a half seconds after it became the new value, in the intervening time there would be a gradual ramp of values to smooth out any sharp transitions that were the result of taking a tally of the kinesthetic motion every two seconds instead of in smaller increments.

Although the general term "amplitude" is being used here to refer to the acoustical loudness of the actual sound, what it really refers to in the context of the algorithms employed are MIDI attack velocity messages; these messages may control (among other things): loudness, timbre, proximity, modulation, etc. Since MIDI attack velocity messages are in a range of 128 possible values, a simple multiplication operation was used to scale the total kinesthetic motion to a usable quantity. Kinesthetic motion was also used in a similar manner to control the duration of any notes being triggered within the same quadrant. However, unlike the calculation for amplitude, duration was considered in inverse proportion to the total kinesthetic energy for the quadrant so that the more motion that occurred the shorter the durations became.

Note events were triggered by changes in zone values (the result of Cyclops detecting changes in light intensity through its difference threshold analysis) and were therefore the direct result of the dancer's motion, see Figure 3.

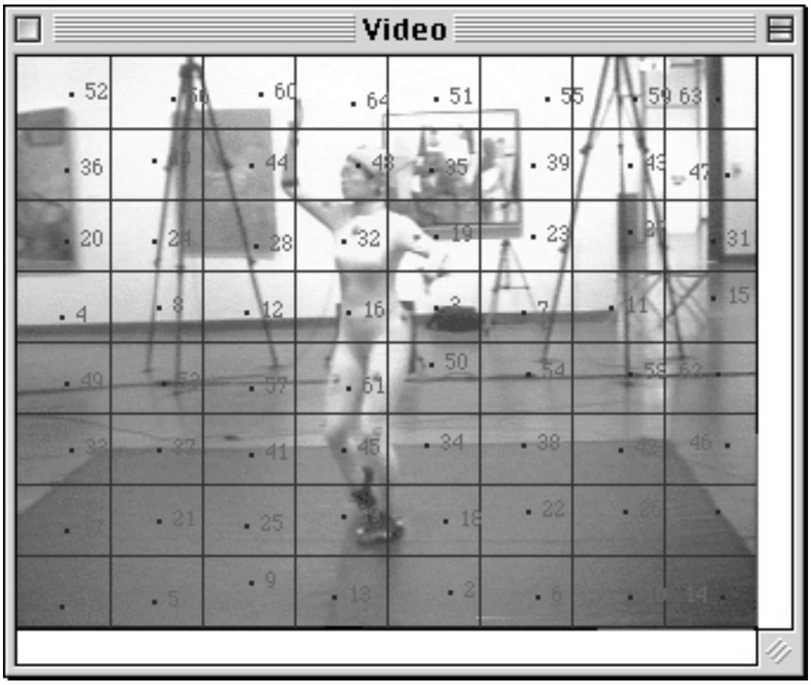

Figure 3. Grid view of dancer

By moving with isolation and poise the dancer could initiate note events very precisely, and by making larger sweeps and gestures they could create huge washes of sound. In fact, it was fascinating to hear how a conscientious dancer could sound distinctly different from an untrained mover; inevitably they sounded less random.

Additionally, the dancer's location was used to control pitch, timbre, and sound diffusion. Whenever the dancer's movement triggered a zone, the index number for the triggered zone became the MIDI note number of the note event corresponding to the change in zone values. This meant, for example, that zone #55 for Camera A (located in quadrant 3) would correspond to MIDI note 55 (unless it was transposed for esthetic purposes, as will be explained later). Assuming that the MIDI note numbers are then being filtered through a standard twelve tone equal-tempered tuning, it is already possible to predict the prevailing harmonic quality the music will have by simply looking at the distribution of zone numbers on the grid. With each quadrant being built from an integer series based on increments of four, the numbers contained within the quadrant will be mapped to members of the same augmented chord, albeit in various octaves. Furthermore, since the dancer's motion most frequently occurs as a trajectory through the same area (they don't disappear from one quadrant and reappear in another) it follows that distinct collections of augmented triads (in various register distributions) will occur and be audible. In a way, this system resembles a giant 3D pitch lattice that can be played by moving within it [Fonville 1991]. Although the preponderance of augmented chords was actually an accidental byproduct resulting from trying to find an efficient solution to the problem of locating the dancer's position in the motion capture space using a modular operation, it turned out that the resulting "neo-impressionist" sounding harmonic quality resonated in a satisfying way, esthetically speaking, with the character of the exhibition and the ambient sound of the museum itself. There was also the added bonus of facilitating palpable connections between pitch space and physical space on a perceptual level. In future projects it would be easy to circumvent this particular mapping by redistributing the zones within the grid, or creating a separate algorithm for generating pitch.

The dancer's location was also mapped to timbre in a 1:1 correspondence. Each of the eight quadrants (between Cameras A&B) were associated with a particular MIDI channel, and each MIDI channel was assigned a particular sound on the external synthesizer. With four separate audio outputs on the synthesizer the sounds were distributed in isolation to one of four speakers arranged in a quadrophonic array around the motion capture space. Each sound was also sent to a subwoofer to achieve added bass resonance. The mapping of the particular quadrants to the four speakers was done so that the dancer's location would be paralleled by the sound diffusion (via the activation of a particular timbre in a fixed location). The way it appeared to the viewer was that the sound seemed to follow the dancer through the space, and the timbre changed depending on their location.

## Idio-synchro-sies

The decision to use timbres available from an external synthesizer instead of synthesizing the sound in real-time within the computer was primarily done for practical reasons. With both computers heavily tied up with the video analysis and algorithmic processes in Max, using MIDI to control outboard gear for sound generation was an efficient solution compared with adding a third computer to handle the signal processing tasks.

Within the external synthesizer (an Ensoniq TS-12) there were two sets of eight sounds each selected and programmed in advance, like a palette of colors that could be called up at will. One of the sets featured essentially familiar acoustic instruments, while the other was a hybrid of electronic and more obscure ethnic instruments. Each set produced distinctly different results, for example the electronic sounding set had sounds that would not decay automatically. This resulted in the occasional inadvertent pedal tone as the dancer tripped up the process before a note-off message could be sent to the synthesizer. The result turned out to be a desirable accident since it contrasted well with the more percussive quick-decaying acoustic sound set.

In order to create additional variety it was sometimes effective to switch one or the other cameras off. By doing so, the dancer was able to move in at least one trajectory where their movement would not trigger musical events, or would trigger them only minimally. This provided a satisfying thinning of the texture periodically, which had the effect of clearing the air.

Another method for achieving variety was to transpose the zone index numbers for one of the grids by some degree in order to shift the pitch material up or down. With extreme transpositions, there were interesting artifacts which resulted from using notes at the peripheral extremes of the synthesizer's sound sample map (eg., key noises were mapped to register extremes in some cases).

Ultimately, both these tools for variation were triggered automatically on cycles that were out of phase with each other. Computer B was given the task of toggling at random between one of the three permutations for the on/off status of Cameras A&B (1. A-on B-on, 2. A-on B-off, 3. A-off B-on) using the arbitrary time interval of 37 seconds. Computer A transposed the zone index numbers within a 128 note range at random every 51 seconds, and Computer B transposed its zone index numbers every 60 seconds. These automatic processes were allowed to run unhindered producing a gradually evolving kaleidoscope of endless possible combinations; except during the daily showings, where a manual override would be used to allow more direct control over the pacing of the performance.

## Technical Challenges

One of the biggest challenges for the project was getting both computers to communicate with each other through MIDI. With a very assorted collection of gear including a MINI Macman interface, and a Tascam US-428 controller, it was possible to jerry-rig a system that seemed to work fairly well, with the occasional MIDI port overload. Every so often, when the dancer's movement became extremely active for an extended period of time, the MIDI port would choke on the flood of data causing the OMS MIDI driver application to freeze-up. This was easy to observe when children were allowed to play in the motion-capture space; with their zealous enthusiasm and unmitigated energy they proved to be among the best extreme "beta-testers" for the system. To get around this problem, a set of filtering subroutines ("speedlim" objects) were inserted into the Max algorithms to insure that the data would not overload the MIDI port.

Other problems that were encountered mostly concerned the video and its analysis. In trying to get accurate data about the dancer's location, it was crucial that the cameras were locked down, did not change their view, and were not otherwise adjusted in anyway. Once this was fairly secure, it was observed that the angle of the track lighting on the ceiling was creating long shadows from the dancer's body. To circumvent this problem, the lights were angled to be more perpendicular to the floor and focused around the center of the motion capture space. The settings within Cyclops had to be fine-tuned as well, for example, the threshold value couldn't be too sensitive or minute changes in ambient light intensity could be recorded as motion, and conversely the threshold value had to be sensitive enough to respond to subtle movements by the dancer. In order to gain increased sensitivity, the decision was made early on to switch from a 5x5 grid in Cyclops to an 8x8 grid. This proved to be a magic number since it allowed the dancer's body, when they were standing in a central location, to be divided up into enough segments to capture their movement in isolation. It also kept the processing time low and cut down on latency; and as an added bonus allowed for a direct numerical correlation to MIDI (which uses 128 as its range of values).

## Observations

Perhaps one of the hardest things for dancers to get used to when working with this technology is the feeling of control that they are suddenly empowered with, since it is an aberration from the traditional relationship of music and dance. To quote one of the dancers from e-Motion, it might very well be "too much control". The default role of music in dance is to drive the dancers along or to fill a void left by the starkness of movement without words. It may seem discomforting then to have the music suddenly change from a static monolith to a malleable mirror. But it is precisely this ambiguity between having control and being surprised by the unexpected, which creates the opportunity for an authentic interactive performance. Both the dancer and the system become equal partners in this exchange with the dancers seeking to achieve greater accuracy in musical results by acquiring mechanical precision; and the composer, working vicariously through the system, seeking to undermine the regularity of the music with engineered humanist spontaneity. Like a new environment, an interactive music performance space will seem strange and exotic to the dancer when they first enter it. In a sense, the music will "play them" for as long as it takes the dancer to understand the result of their movement before its execution. Eventually though, their command of the environment will be complete and the scale will be tipped in the other direction. The moment of true interaction is the ephemeral state of equilibrium that happens in between.

## References

Fonville, J., “Ben Johnston’s Extended Just Intonation: A Guide for Interpreters,” Perspectives of New Music, 1991, Vol. 29, No. 2 (Summer, 1991) pp. 106-137.

Singer, E., Cyclops (Software), 2000-2001. Commercially released in February 2002. Available from: http://www.cycling74.com/downloads/cyclops

Vanier, L, Kaczmarski, H., Chong, L., Blackburn, B., Williams, M., de Velder, A., "Connecting the Dots: The Dissection of a Live Optical Motion Capture Animation Dance Performance", New London (CT): Biennial Symposium for Arts and Technology, 2003 [cited 2009 Aug 8]. Available from: http://www.isl.uiuc.edu/Publications/final%20dance1.pdf

Winkler, T., Composing Interactive Music: Techniques and Ideas Using Max, Cambridge (MA): MIT Press, 1998.