IEEE ICMCS 1997. June
Ottawa

MusiKalscope: A Graphical Musical Instrument

Sidney Fels, Kazushi Nishimoto and Kenji Mase
ATR Media Integration & Communication Research Laboratories
Seika-cho, Soraku-gun, Kyoto, 619-02, JAPAN
fels,knishi,mase@mic.atr.co.jp +81 774 95 1448

Abstract

This paper introduces a new multi-media system for musical and graphical expression called MusiKalscope. *Inside of MusiKalscope are two sub-systems: RhyMe and the Iamascope, connected by a third sub-system called the Graphical Musical Instrument Interface (GMII). RhyMe is a system which supports computer assisted jazz improvisation. RhyMe provides* functional *actuator maps for a performer in contrast to tonal actuator maps typically found in traditional instruments. The Iamascope is a computer-based kaleidoscope which allows the user to be inside a kaleidoscopic image which can be viewed in real-time for visual expression. The GMII connects the two sub-systems by providing a virtual drum set which allows the performer to play unencumbered inside the Iamascope. The functional map used to control RhyMe is also used to control aspects of the imagery in the Iamascope to provide mood to match the music played.*

1. Introduction

Imagine a system where a performer creates art by having a trumpet on her right hand and a paint brush on her left hand and using them simultaneously. Intuitively, it seems that this means of creating art, while possible, is fairly impractical when it comes to issues of controllability and ease of mastering for expression[1]. However, we believe that usual multi-media art generating systems have urged performers to do just such things. Our contention is that, for an artist to create good artwork with individual media such as paint or music is difficult enough. Further, to create them simultaneously is nearly impossible.

To make simultaneous production possible in a multi-media computer supported art system, we believe that it is essential that the performer's cognitive load be reduced by using several support mechanisms. Care must be exercised, though, to avoid excessive support. The supporting method must not obstruct performers' creativity and should leave enough room for performers to show their creativity and expression. In this sense, we believe that multi-media art systems which have several media completely controlled automatically are not very suitable to artists. To realize a good multi-media art generating system, we suggest supporting a minimum "good enough" quality as well as keeping sufficient room for performers' creativity in all of the media. In this paper we introduce MusiKalscope, which we argue is an example of such a system.

During performances using multi-media systems which allow performers to control imagery and music it is typical for a performer to concentrate on either producing good music while sacrificing image quality, or, for a performer to focus on controlling imagery while ignoring the music being played. The MusiKalscope attempts to address this imbalance in attention and its adverse effects on the quality of a performance. Our solution is based upon functional operation of the *whole* multi-media system rather than one part of it. Specifically, we designed the system to enable a performer to control the "colour" of the whole performance and not just the "melody" of the music or the details of the computer graphic image.

In the MusiKalscope, each action performed by the performer controls all media. Thus, even if one of the actions is targeted to a specific media, such as producing music, the other media will follow according to the *function* of the action for the specified medium. The MusiKalscope does this by defining a mechanism for the performer's actions to adjust the "colour" of a piece of music rather than the melody while a song plays and makes the same actions control the "colour" of the computer graphic image. At the same time the movement and dance of the performer during his performance with the rhythm of the music translates di-

[1] Of course, using this system is itself a means of expression.

0-8186-7819-4/97 $10.00

rectly to the aesthetic feeling provided by the visual imagery.

Other systems related to the MusiKalscope have been created. These include systems such as: Brush de Samba [5], "Cindy" [2], DanceSpace [1] and "MUSE" [6]. Brush de Samba most closely relates to our work. Brush de Samba is a system to generate music and CG simultaneously. In Brush de Samba, a performer draws a picture using a drawing pad. Music is automatically generated based on the pen position data. The performers generally focus on the image they are drawing rather than the music which is generated, and hence, the system is biased for graphical input making it difficult to control the music quality in the system. In contrast, "Cindy" is musically biased. In this system, performers play music and an animated character, named "Cindy" dances along with the music played. The style of dance is controlled by the style of music played. Control is exercised by changing musical parameters such as: the number of notes played, playing a single tone or chord and the beat. In DanceSpace, the performer's dance is captured by a video camera and used to control music and computer graphics. The movement of the performer is mapped such that the dancer's hands and feet control virtual musical instruments. The pitch of the music is controlled by the dancer's head height. At the same time, computer graphics are created and controlled by the dancer's motion. A coloured outline of the dancer's body is successively represented. With DanceSpace, various music styles can be played; however, the sounds generated are always continuously ascending or descending, significantly impacting the quality of the music generated. Finally, MUSE provides a musical interface to control a computer generated character. MUSE interprets the music played as an emotional context which is displayed by the MUSE character. The musical grammar is described in [6]. Of interest, is that the computer graphics coordinate with the musical sounds; however, the grammar limits the musical scope when using MUSE.

In creating the MusiKalscope, three main goals were important:

1. good balance between the quality of computer graphics and music generated
2. novices can easily achieve a reasonable quality of music and imagery
3. with more training, enhanced expression is possible; thus, no performance ceiling is imposed

The next sections describe the MusiKalscope in detail.

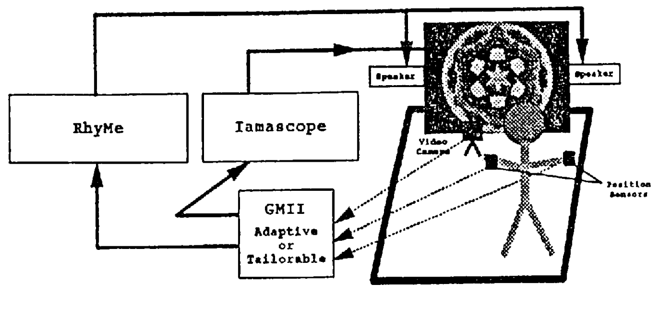

Figure 1. Block diagram of MusiKalscope.

2. Overview of MusiKalscope

The MusiKalscope consists of three sub-systems: the graphical musical instrument interface (GMII), RhyMe and the Iamascope as shown in figure 1. The GMII provides the input interface between the performer and the two other sub-systems. Two main input devices are used by the GMII, a video camera and a Polhemus Fastrak. The video camera input is fed directly to the Iamascope to be used for the imagery. The output of the Iamascope is displayed on a large video screen (170") in front of the performer so they may see the imagery they are producing. The Polhemus Fastrak data is analysed to implement a virtual drum interface to RhyMe and the Iamascope. The name of the virtual drum pad that the user strikes is passed to RhyMe (see section 4). RhyMe then plays the particular note appropriate for that drum pad at that moment in time during the song (according to jazz analysis of the song being performed). The output from RhyMe is played on speakers mounted on the sides of the large video screen. Further, RhyMe dictates whether the virtual drum pad struck corresponds to a chord tone or a tension tone. Currently, this mapping is determined apriori and is embedded in the GMII which sends the appropriate signal to change the appearance of the Iamascope. When a tension note is played the Iamascope image becomes bluer. When a chord note is played the Iamascope returns to its normal colour. The faster a virtual drum pad is played the brighter the Iamascope image becomes. All changes in colour and brightness to the Iamascope image will gradually return to the "normal" state if the user stops playing.

In the next sections, each of the subsystems is described in detail. First, the interactive kaleidoscope, Iamascope, is discussed. The Iamascope is a *graphical* instrument which allows the performer to play with visual imagery derived from his body. Second, the RhyMe system is discussed. The RhyMe system is a computer supported jazz improvisational system. The key feature of RhyMe is that it allows a performer to focus on playing keys[2] which correspond to controlling the localized (in time) colour of a specific piece being played. Hence, the instrument keys used are mapped to functional aspects of musical performance rather than particular tones. Finally, the two systems are integrated with the graphical musical instrument interface. This system provides the performer with an input device to control the Iamascope and RhyMe.

3. The Iamascope Sub-system

Kaleidoscopes have captured imaginations all over the world since they were first invented by D. Brewster in 1816. The Iamascope is an *interactive* kaleidoscope, which uses computer video and graphics technology. In the Iamascope, the performer becomes the object inside the kaleidoscope and sees the kaleidoscopic image on a large screen (170") in real time. The Iamascope is an example of using computer technology to develop art forms. As such, the Iamascope does not enhance functionality of some device or in other words, "do anything", rather, its intent is to provide a rich, aesthetic visual experience for the performer using it *and* for people watching the performance. The Iamascope is more than a mirror-based kaleidoscope put in front of a video camera since the types of reflections possible with the computing machinery are more extensive than are possible with mirrors such as asymmetric reflections and different tiling patterns.

From the perspective of balanced, minimum quality multi-media, the Iamascope possess one important quality; for the most part, anything placed inside it will look beautiful. The symmetries involved with the mirror reflections appeal to many people as atested to by the popularity of kaleidoscopes since they were invented. Thus, as part of a multi-media system, the Iamascope provides a reasonable minimum level of aesthetic quality allowing novices inside the Iamascope to produce beautiful images. However, as a performer learns to move their body to manipulate the kaleidoscopic image they are able to achieve greater forms of expression. Hence, the Iamascope provides an upward pathway for achieving highly skilled forms of expression through visual imagery.

A block diagram of the Iamascope is shown in figure 2. For input, the Iamascope uses a single video camera connected to a video board with a drain to texture memory. Output from the Iamascope is displayed on a video monitor. In our current implementation, the video image from the camera is placed in texture memory and then the appropriate part of the video

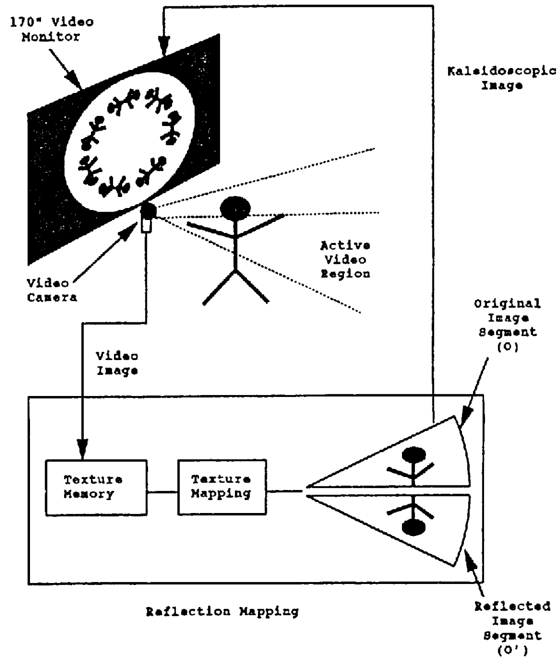

Figure 2. Block diagram of the Iamascope. The system runs at 30 fps.

image (currently a "pie" slice also referred to as a segment) is selected to form the original image (O) which is used to create the desired reflections (O'). A multi-polygonal circle is drawn upon which the appropriate textures (original or reflected) are drawn alternately. The necessary reflections for the Iamascope are simulated with texture hardware providing frame rates of 30 frames per second. This frame rate provides low-latency, high bandwidth control of the kaleidoscopic image supporting a sense of intimacy with the Iamascope. An example of a single frame is shown in figure 3.

The Iamascope used in MusiKalscope is based on the reflections found in a two-mirrored kaleidoscope. However, in the Iamascope used in the MusiKalscope a pie slice (segment) from the original video image is used instead of a triangular slice typical of two-mirror kaleidoscopes. Thus, if the arc angle of the slice is an even integer divisor of 360 degrees a circular image is formed using the alternation of the original image and its mirrored reflection (as shown in figure 2). The even integer multiple arc angle is required so that alternation of the original image with its reflection will exactly fill the circle as shown in figure 2. For example, if we use a 30 degree pie slice then there will be 12 segments which make up the circular image. The odd segments will have the original image and the even ones will have the mirror reflection. Three aspects of this method

[2]The term key here refers to any actuator on any type of musical instrument.

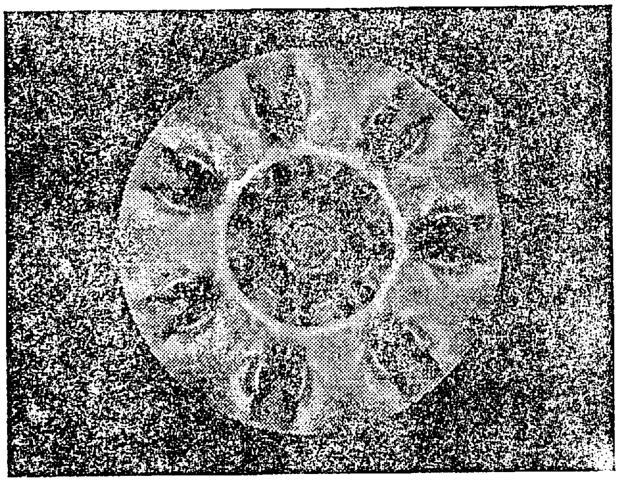

Figure 3. Example of a single frame from the Iamascope. Note: only monochrome is shown here, normally 24 bit colour is used.

provide a beautiful effect. First, as the segments exactly fill the circle and there is always a reflected image paired with the original slice the boundaries of each segment will exactly line up without any perceivable discontinuity. Second, since a "pie" slice is used as the original there is a singularity at the centre of the image. This singularity allows the users movement to be perceived relative to the outside edge of the circle and the centre. Third, the pie slice allows for different visual scales to be used from the image. The outer edge of the slice captures a large area of the video image while towards the center of the image only a small area is captured for the reflections. By placing objects close to the centre of the slice it is more difficult to recognize it in the kaleidoscope image allowing for more abstract forms of expression in the image[3].

Additional controls are available which are exploited by the MusiKalscope system. The Iamascope has controls for image background mixing colour, image brightness, arc angle, slice angle rotation speed etc. The MusiKalscope uses the image background mixing colour and brightness controls to match the mood of the Iamascope image with improvisational sounds that the performer makes using RhyMe. Refer to section 5 for how these controls are exploited.

4. The RhyMe Sub-system

Typically, in multi-media art creation which uses music, the performer cannot concentrate only on playing music. However, even if a well trained musician concentrates *only* on playing music it is still quite difficult to generate good quality music. Therefore, in many cases of multi-media art creation, the musical medium may not have particularly good quality.

There are at least two ways to avoid this problem. One way is to apply a "definitional or casual music" concept which is often seen in some contemporary classical music. In this genre, a musical work consists of all of sounds generated casually. That is, all sounds produced are defined to be part of the music regardless of their origin. Thus, by this definitional argument, in a multi-media art creation system we can ignore what kind of instruments are used to play sounds and the aural effect from the composition of sounds created during the performance is called music. Needless to say, this is a very dangerous approach. Even in the casual music genre observered in contemporary classical music, the casualness is designed and structured to some extent and is not completely random. Generally speaking, composition of completely random sounds does not make good quality music.

An alternative strategy in multi-media art work is to apply structure to a performance artificially. In MusiKalscope we use a computer to control the structure of the performance thereby applying our own definition of what is music to the performance. We call this "computer supported improvisation". Working from this philosophy we restrict the type and structure of the music which can be created during a performance; however, we are rewarded by allowing the performer easier access to musically appealing performance within these confines.

In this research, we focused on supporting jazz improvisation, in particular "Be Bop", one of the styles of jazz. Improvisation is a critical component of jazz. For unaccustomed listeners, jazz improvisation performances may sound like random compositions. However, if studied, these performances can be analysed as conforming to some well-defined jazz improvisational theory. Well-trained jazz players are always analysing the playing song in their brain, and compose improvisation pieces by using notes that are dictated by the particular jazz theory. By applying the theory, the piece can have a jazzy atmosphere and the player can express "colour" of the song derived from its chord progression structure.

There are two difficulties in playing jazz improvisation based upon theory: first, it is difficult for people to master the theory, and second, it is also difficult to reflect the knowledge to the piece during the performance in real-time even if one has mastered the theory knowledge. To overcome the first difficulty, several systems have been developed which automatically analyse

[3] Likewise, the user can also move closer or farther from the video camera to get different scaling effects.

songs using a knowledge base of particular jazz theories [1][5] [4] [3]. Such systems can show users theoretically determined notes at each time of songs which correspond to different jazz effects. By referring to the results of the analysis, performers can play improvisation using theoretically correct notes. However, since jazz music has complicated structure, it is still quite difficult for non-professional people to play improvisation even if referring to the analysis result. To make matters worse, it is even more difficult to use this type of system in a multi-media art performance where attention may need to shift to other media representations in real-time.

To overcome the difficulties encounted from these systems during live performance we employed the following method: first, a song which is to be performed is automatically analysed based on a particular jazz theory. Then, based on the analysed results, only notes which are in the available note scale are assigned to playing positions of the musical instrument being used in the performance (see section 5 for the mapping between notes and positions in the virtual drum set). Thus, the performer plays one part of the jazz piece being performed. The jazz piece plays continuously while the performer joins in with improvisation assisted by RhyMe.

Figure 4 shows a block diagram of the RhyMe subsystem. RhyMe consists of the song database, the automatic analyse module and the note-position mapper. The song database stores chord progression data of several musical pieces. Sample data of a piece looks like:

$$\cdots \mathbf{(Dm7,2)\ (G7,2)\ (C6,4)} \cdots$$

Each entry consists of the chord name and the number of beats. The above example corresponds to the following score : D-minor 7th for 2 beats, G dominant 7th for 2 beats, and then C 6th for 4 beats. Data for one song is used as input to the analyse module.

The automatic analyse module has a musical theory knowledge base. Currently, this knowledge base is constructed based on Berklee-theory, a well-known jazz theory (mainly "Be Bop" style jazz). Based on this knowledge, this module analyses the chord progression (from the input data) and determines what kind of scales are available at each chord regarding the "context" of the chord progression. The whole analysis process is complex, but the idea can be illustrated by using the example above. In analysing the data above, first, the analyse module looks for a "dominant 7th" chord and assumes it to be a "V7". As a result, the tonal-center, i.e. the note corresponding to "I", is assumed. In this example, "G7" is a dominant 7th chord and hence note "C" is assumed as the tonal-center. Second, the module interprets the chord progression data which is notated absolutely as relative by using the assumed tonal-center. In this example, the chord-progression is interpreted as follows: IIm7-V7-I6. Third, the module matches the interpreted chord-progression with one of the "chord-progression-patterns" which the module has as apriori knowledge. In this example, this chord progression matches with "major II-V-I" chord-progression, which is a typical and basic chord-progression-pattern. Fourth, if the module can find a matched chord-progression-pattern, the module decides the assumed tonal-center is correct, and outputs available note scales for each chord by looking up the chord-scale map. In this example, the following correspondent scales are output: II – dorian, V – mixolydian/altered 7th/whole tone, etc., I – ionian, etc. Finally, the relative note name is re-interpreted as an absolute note. From these results, the available note scales of the example are determined as shown in figure 4, i.e. D dorian, G mixolydian or G altered 7th or G whole tone scale or etc., and C ionian.

These analysed results are input into the *note-position* mapper. This module maps notes of a scale at each time point in the song onto the playing positions of the musical instrument. The mapping is changed while notes are synchronized with the progression of the song. With the MusiKalscope, the instrument used is a virtual drum machine (see section 5.1) which has 7 active zones. Notice, that the root zone corresponds to the root note of the currently available note scale (for example, a "G" note in figure 4) while the "II" and "III" active zones correspond to the "II" and "III" note of the current available note scale (in figure 4, "A" and "B" note, respectively). For example, if a performer plays the "VII" zone, the actual note played is an "F" as shown in figure 4. However, when the available note scale is changed to "D dorian" by the progression of the song, the note played by striking the "VII" zone becomes "C". The whole RhyMe system uses MIDI, thus, the synchronization is done by the MIDI timing clock data. Using this scheme, either chord tone zones (root, III, V, VII) can be played or tension note zones (II, IV and VI) corresponding to the relative position of the note in the current available note scale. The key point to remember though, is that the actual note played is dictated by RhyMe.

When using conventional musical instruments, performers must always judge which notes are theoretically correct or not and which note has which colour by thinking about the chord progression of the song and the performers' own theoretical knowledge. However, by using RhyMe, the performer does not have to determine them at any moment in playing a song. Since only

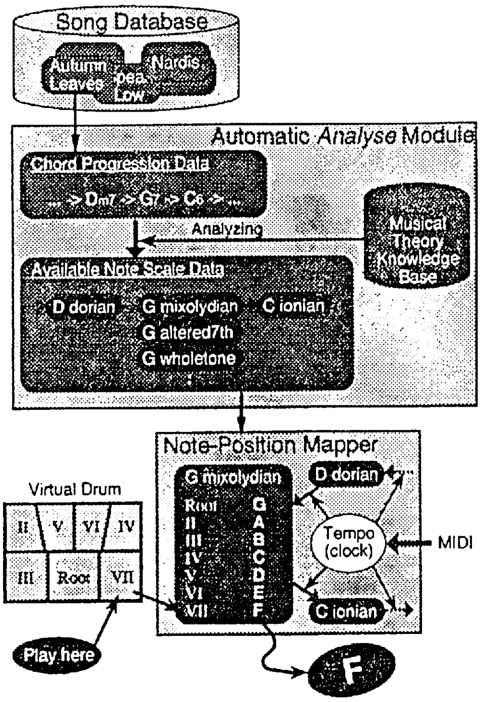

Figure 4. Block diagram of the RhyMe sub-system.

theoretically correct notes are mapped onto the musical instrument at any time in the song, the performer need not to be conscious of the chord progression and the theory. Furthermore, notes are mapped onto zones adequately classified (i.e., chord tone or tension note). The main result of this is that the performer's cognitive load to create music can be drastically reduced and at the same time he can easily maintain a reasonable quality of music throughout the piece. Further, the system allows for easy access for the performer to express colour throughout the song. From this reasoning, RhyMe is a suitable sub-system for generating music in a real-time multi-media art performance system such as MusiKalscope.

The music generating methods used in DanceSpace [1] and Brush de Samba [5] also theoretically analyse songs similar to the technique used in RhyMe. As such, in these systems, a performer can play theoretically correct notes without needing conscious attention at each moment in the song. However, in these system, no obvious discrete playing positions are provided to performers to control the music. In particular, performers controlled only successive ascending or descending phrases. Therefore, it is very difficult for performers to compose phrases which they want to play. The RhyMe system provides discrete playing positions which can be played using instruments like keyboards, virtual drums or other MIDI instrument, thus, performers can easily choose specific notes (which are always theoretically correct) and compose phrases quite freely.

5. GMII: Bringing the Pieces Together

The graphical musical instrument interface (GMII) in the MusiKalscope links the two subsystems, RhyMe and the Iamascope together. The GMII provides a set of virtual drums (see section 5.1) which the performer uses to play with RhyMe and control some parameters of the Iamascope image. With RhyMe (see section 4) the global colour of a song is determined by the chord progression of the song. However, the performer can play different tones with different velocities to control local (in time) colour, for example, playing a chord tone, playing a tension tone, playing strongly or playing weakly. For the MusiKalscope, we mapped chord tone and tension notes of the available scale onto different virtual drum pad zones. Therefore, players can easily choose to play a chord tone or a tension note to control the local colour of the playing. The virtual drum set provided by the GMII provides 7 tailorable active zones corresponding to either chord tone or tension notes (each corresponding to an offset from the root tone note). The default configuration of the active zones are shown in figure 5. The functions associated with each zone are activated when the performer makes a down-up motion of their hands in the zone.

Control of the Iamascope has been mapped according to the functions that RhyMe uses for the active zones. The mapping has been chosen apriori based on artistic license. Thus, for zones which produce tension notes according to RhyMe, the Iamascope image is made to provide more visual tension. For this effect we change the background mixing colour to be bluer for a sense of foreboding. The more tension zones struck, the bluer the Iamascope image looks. If the performer strikes a chord zone then the Iamascope image immediately returns to its normal, non-blue state to provide a sense of resolution. If the performer doesn't strike any key the Iamascope image gradually returns to its normal state over time. The velocity that the performer strikes the virtual drum controls the brightness of the Iamascope image. For fast strikes the Iamascope image lights up strongly in a flash and gradually returns to its normal brightness.

5.1. A Simple Virtual Drum

Many different instruments can be used to control RhyMe; in fact, any instrument with 7 keys or more can be used. For the MusiKalscope it is important that the performer's hands and arms be unencumbered so that

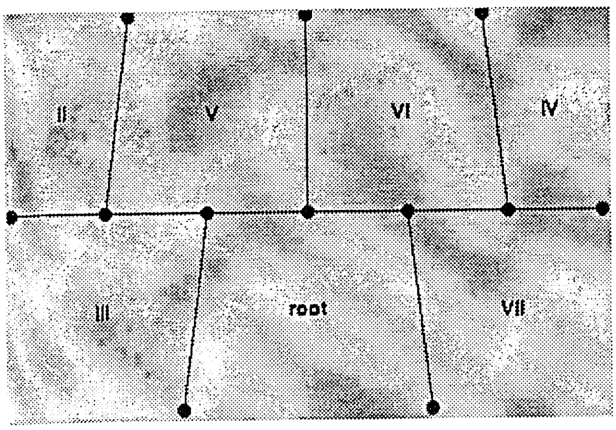

Figure 5. The seven active zones of the virtual drum.

they can move freely in front of the Iamascope at the same time as they play with RhyMe so they can control the visual imagery at the same time as the musical performance. To do this, we decided to use a virtual drum set where the performer's hand/arm motions are measured using a Polhemus 3-space Fastrak system. These magnetically based trackers measure the 6 degrees of freedom (position and orientation) of a receiver relative to a source. In the MusiKalscope, two receivers are used; one for each hand. The performer attaches the receivers to their arms using straps allowing them to move their arms and hands freely. Down-up movements of their arms strike the virtual drum pads according to the mapping shown in figure 5. The active zones are currently situated horizontally (parallel to the floor) in front of the fixed source. We plan to attach an additional receiver on the performer's body so that the active zones are relative to the performer and not a stationary point in front of the Iamascope.

To implement the virtual drums the velocity of the performer's arm is measured. An idealized speed profile of a performer moving their hand down and then up is shown in figure 6. Notice, that as the performer moves their hand down from stationary the speed increases. Then, as they prepare to turn their hand around the speed of their arm decreases. As their hand begins to go upward again, the speed of their arm reaches it's minimum (ideally, it should be 0). In our virtual drum system as soon as the change in direction is detected we consider the zone to be activated[4]. The particular name of the zone that was activated is passed to RhyMe and the Iamascope controls.

One subtle point about this virtual drum scheme is that the change in direction does not correspond exactly to the drum metaphor. Often, a drum is struck using a balistic downward motion of the performer's hand. Contact with the drum is used to change the direction of the performer's hand. However, with the virtual drum there are no surfaces to contact. Instead, the performer's own muscles have to change the direction. Using a drum metaphor, the user expects the system to respond *during* the balistic downward stroke of their arm. However, with the virtual drum activation occurs when their hand changes direction which occurs somewhat later than the balistic stroke. This introduces a perceptual lag in the system stemming from the wrong metaphor. A closer metaphor might be a conductor's baton. We are currently investigation other sensors, (such as accelerometers, infrared batons and vision systems) and other techniques to match the perceptual striking position with that of the virtual drum metaphor.

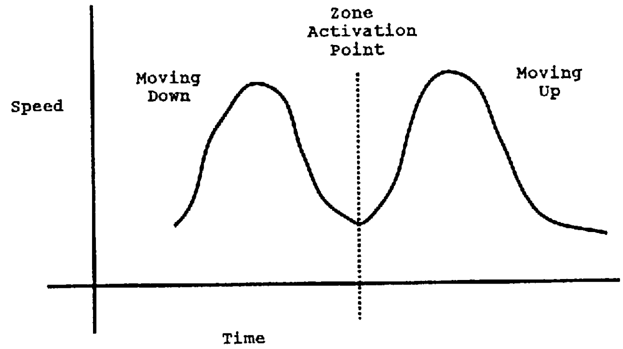

Figure 6. Idealized graph showing a performer moving his arm down and then up.

6. Discussion of MusiKalscope

An earlier version of the MusiKalscope[5] was displayed in Kyoto, Japan. During this display several hundred people used the MusiKalscope, providing opportunity to observe both performers and audiences. Two important points were noted: first, novice performers tended to focus either on producing music or producing visual imagery but not both, second, audiences often enjoyed the combined visual and musical performance, but had difficulty appreciating and understanding the relationship between the performer and the performance they were hearing and seeing.

The fact that performers attended either to producing music with RhyMe or playing with visual imagery is not surprising. In particular, the main point of

[4]The performer must also have exceeded a minimum downward speed threshold before the change in hand direction can activate a zone.

[5]The earlier version only had a geographic link between RhyMe and the Iamascope, that is performers used the virtual drum in front of the Iamascope. The zones activated had no effect on the Iamascope image in contrast with the current system.

MusiKalscope is to allow performers to only focus on one aspect at a time. The complimentary medium always maintains a reasonably good quality level even if it is ignored or played poorly. However, either system responds appropriately to allow improved expression as the performer becomes skilled. From this point of view, the system was successful.

Unfortunately, sometimes audience members were not aware of the connection between what the performer was doing and the imagery and/or music created. This difficulty probably arises for different reasons for the different media. For the visual aspect, the kaleidoscopic images bear little resemblance to the performer as only a small slice of the whole image is used in the reflections. Thus, the movement and changes in the image are not obviously attributed to the performer. Once an audience member becomes a performer by stepping into the MusiKalscope though, the correspondence is obvious and lasting. Perhaps the solution is familiarity with the novel art form or better introduction. For the musical performance, correspondence between the performer's actions and the music is difficult because there is always background music being played which doesn't change at all with the performer's actions. Further, lag introduced by the virtual drums (see 5.1) exasperated the problem further (both for audience and performer). Improving the lag situation will help, however, there is an inherent tradeoff between providing a minimum quality level of music and allowing the performer control over the music establishing a strong link between actions and sounds.

Finally, one frequent comment from performers was that there was too much lag between the virtual drum strike and the sound produced. As discussed in section 5.1 this is probably due to the incorrect metaphor. Work is continuing to improve this situation.

7. Conclusions and Directions

We have three main goals for developing the MusiKalscope as a multi-media art system. The three goals are:

1. obtain good balance between the quality of computer graphics and music generated;
2. enable novices to achieve a reasonable quality of music and imagery easily;
3. allow an upward path for skill acquisition so experts can become more expressive with the MusiKalscope.

These goals were mostly achieved by combining two systems which each have a reasonable minimum quality even if ignored completely. The RhyMe system will continue to play pleasant music regardless of user input. It can never play particularly bad sounding music (in its genre); but, with user input can sound even better. Likewise, the Iamascope, like a standard kaleidoscope, generally produces beautiful images no matter what is put in it. However, the Iamascope allows the performer freedom to express herself by moving inside it. The expressiveness of the image is under the control of the performer and her imagination. Thus, both subsystems are balanced with respect to quality.

Currently, we are improving three main areas to further accomplish the goals set out. First, we want to make the system easier to learn to use skillfully. Currently, even though novice user are producing reasonable quality music and images, the progression to express themselves well is not as fast as we would like. The main difficulty stems from the awkward input device of the virtual drum set. We plan to modify this device either by using different sensors or improving the recognition techniques we are using. Second, currently we have only a small selection of songs that have been analysed for use in RhyMe. Some song are more suited to be accompanied with visual imagery. We are planning on increasing the available selections. Third, it is an iterative process determining which aspects of the Iamascope best fit a particular genre of music. We are investigating different ways to control the Iamascope to enhance the feeling it provides along with matching the particular "colour" of the music being performed.

References

[1] S. Flavia. Choreographing media for interactive virtual environments. Master's thesis, Media Arts and Sciences, M.I.T., 1996.

[2] M. Goto and Y. Muraoka. A virtual dancer "Cindy" — interactive performance of a music-controlled CG dancer —. In *Proc. of Lifelike Computer Characters*, page 65, October 1996.

[3] K. Hirata. Towards formalizing jazz piano knowledge with a deductive object-oriented approach. In *Proc. of Artificial Intelligence and Music*, pages 77–80. IJCAI workshop, August 1995.

[4] D. Horowitz. Representing musical knowledge in a jazz improvisation system. In *Proc. of Artificial Intelligence and Music*, pages 16–23. IJCAI workshop, August 1995.

[5] A. Kotani and P. Maes. An environment for musical collaboration between agents and users. In *Proc. of Lifelike Computer Characters*, page 54, September 1995.

[6] N. Tosa and R. Nakatsu. Life-like communication agent – emotion sensing character "MIC" and feeling session character "MUSE". In *Proc. of International Conference on Multimedia Computing and Systems*, pages 12–19. IEEE, June 1996.