AISB 2004 Convention:

Motion, Emotion and Cognition

AISB The Society for the Study of Artificial Intelligence and the Simulation of Behaviour

Proceedings of the AISB 2004

COST287-ConGAS Symposium on

Gesture Interfaces for Multimedia Systems

29 March – 1 April, 2004

ICSRiM, University of Leeds, Leeds LS2 9JT, UK

www.leeds.ac.uk/aisb www.icsrim.org.uk

AISB 2004 Convention

29 March – 1 April, 2004
ICSRiM, University of Leeds, Leeds LS2 9JT, UK
www.leeds.ac.uk/aisb www.icsrim.org.uk

Proceedings of the AISB 2004 COST287-ConGAS Symposium on Gesture Interfaces for Multimedia Systems

Published by

The Society for the Study of Artificial Intelligence and the Simulation of Behaviour

http://www.aisb.org.uk

ISBN 1 902956 37 4

Contents

The AISB 2004 Convention

On behalf of the local organising committee and all the AISB 2004 programme committees, I am delighted to welcome you to the AISB 2004 Convention of the Society for the Study of Artificial Intelligence and the Simulation of Behaviour (SSAISB), at the University of Leeds, Leeds, UK.

The SSAISB is the oldest AI society in Europe and it has a long track record of supporting the UK AI research community. This year, the underlying convention theme for AISB 2004 is "*Motion, Emotion and Cognition*", reflecting the current interest in such topics as: motion tracking, gesture interface, behaviours modelling, cognition, expression and emotion simulation and many others exciting AI related research topics. The Convention consists of a set of symposia and workshop running concurrently to present a wide range of novel ideas and cutting edge developments, together with the contribution of invited speakers:

- Prof Anthony Cohn
  Cognitive Vision: integrating symbolic qualitative representations with computer vision;
- Prof Antonio Camurri
  Expressive Gesture and Multimodal Interactive Systems;
- Dr David Randell
  Reasoning about Perception, Space and Motion: a Cognitive Robotics Perspective; and
- Dr Ian Cross
  The Social Mind and the Emergence of Musicality,

not to mention the many speakers invited to the individual symposia and workshop, who will made the Convention an exciting and fruitful event.

The AISB 2004 Convention consists of symposia on:

- Adaptive Agents and Multi-Agent Systems;
- Emotion, Cognition, and Affective Computing;
- Gesture Interfaces for Multimedia Systems;
- Immune System and Cognition;
- Language, Speech and Gesture for Expressive Characters; and the
- Workshop on Automated Reasoning.

The coverage is intended to be wide and inclusive all areas of Artificial Intelligence and Cognitive Science, including interdisciplinary domains such as VR simulation, expressive gesture, cognition, robotics, agents, autonomous, perception and sensory systems.

The organising committee is grateful to many people without whom this Convention would not be possible. Thanks to old and new friends, collaborators, institutions and organisations, who have supported the events. Thanks the Interdisciplinary Centre of Scientific Research in Music (ICSRiM), School of Computing and School of Music, University of Leeds, for their support in the event. Thanks to the symposium chairs and committees, and all members of the AISB Committee, particularly Geraint Wiggins and Simon Colton, for their hard work, support and cooperation. Thanks to all the authors of the contributed papers, including those which were regretfully not eventually accepted. Last but not least, thanks to all participants of AISB 2004. We look forward to seeing you soon.

Kia Ng
AISB 2004 Convention Chair
ICSRiM, University of Leeds,
School of Computing & School of Music,
Leeds LS2 9JT, UK
kia@kcng.org www.kcng.org

Proceedings of the AISB 2004 COST287-ConGAS Symposium on Gesture Interfaces for Multimedia Systems (GIMS)

Symposium Preface

The aim of this symposium is to explore emerging technologies and applications in gesture controlled multimedia systems, expressive gesture analysis, modelling and synthesis; to increase awareness and exchange views and experiences in the use of gesture interfaces and interactive multimedia technologies, with particular interests in performing arts. Topic of interests include but not limited to gesture sensing, tracking, analysis of expressive gesture, modelling, synthesis/simulation of expressive gesture, animation, mapping strategies and others. The symposium is designed to have an informal atmosphere to promote discussion and also to offer the opportunity to provide live demonstrations.

Thanks to Antonio Camurri for the plenary keynote presentation on "*Expressive Gesture and Multimodal Interactive Systems*". The papers presented here were carefully selected by multiple anonymous peer reviews. Thanks to the members of the ConGAS GIMS programme committee for their fast and thorough reviews. Thanks to all the authors for their contributions and cooperation, and to the Convention organisers for making the event possible.

This Symposium is supported by the COST287-ConGAS project (www.cost287-congas.org), and the MUSICNETWORK project (www.interactivemusicnetwork.org). Thanks to COST-TIST, European Science Foundation (ESF), and IST European Commission for the supports.

Kia Ng, University of Leeds

Motion and emotion using home-made digital musical instruments

Daniel Arfib, Jean-Michel Couturier, Loic Kessous*
*LMA-CNRS
31 chemin Joseph Aiguier 13402 Marseille
arfib@lma.cnrs-mrs.fr

Abstract

A combination of hardware and software using a gestural control and an audio system cannot be called a musical instrument till its musical use is not proven. In this article, after a general description of gestural system, we will focus on certain realisations of home-made digital instruments. For each of them we will focus on the repertoire of gestures, and then the musical use it provides. As a matter of conclusion the special link between motion and emotion will be evoked in these particular implementations.

1 Making a musical instrument

Before building from scratch what will be a digital musical instrument, it is good to have a though about what is gestural control of musical synthesis systems. At one end, we have a algorithm that can calculate a sonic signal. This algorithm is governed by data, and when these data evolve with time, we can hear (or not) what is called in musicology a musical gesture. This is a cognitive process where the combination of the physiology of the ear, the cognitive process of perception, and the cultural inheritance (be it of a new trend) mix to make us hear an energy, a dynamism, an harmony or whatever is a movement in the sound. These things are the object of a special discipline, the machine listening. On the other hand (if one can say so) we have different sensors, and when physical gestures are done, we get some specific data. In fact our intention is translated into data, which can preserve or not the entirety of this intention. The intermediary part, named mapping is the way to connect the physical data and the sonic data. We have proposed (Arfib et al., 2002b)to use an intermediate layer where the intention process is retrieved form the gestural data, and connects thewith psychoacoustic data related to what we hear from a sound. This is not always so trivial, but we should keep in mind one thing: the best we define this layer, the easiest way the feedback or the graphical interface will be.

1.1 The design of instruments

All the instruments we have designed use Max-Msp patches on a Macintosh, where gesture data comes from peripherals linked to the machine. A special care is taken about the mapping, but also the repertoire of gestures and the musical possibilities of these instruments that will be seen in next sections. As these instruments have already been described elsewhere, only a short presentation will be made in this article.

The peripherals can be of different kind, and some taxonomy of these sensors have been tempted by different authors. Two main division are to us very important:

- free / not free: either the user wear or touches something or his free movements are captured (usually via a camera). The devices we use are equipments.

- Using a surface or not: tablets are surface oriented, while gloves are not. The pointing fingers are a special case while they can be of both kind.

The dynamic/non dynamic status of the mapping provided in the instrument is especially important. When we introduce a dynamical system in between the gestural data and the synthesis one, we can say that we no more control the sound but we control the dynamical system included in the sound. It makes a big difference in terms of gestures, every gesture is followed by the response of this system.

1.2 the repertoire of gestures

One extremely important point using a digital musical instrument is the panorama of gestures that it permits. These gestures are first akin to the peripherals we use: surfaces induce scratching movements, gloves induce movements such as to take, aso. But musically, it is important to see these three kind of movements:

- selection: we take an object, or select a patch number.

- activation: we trigger this object and this gives an event which can last over time but is governed only by its initial values

- modulation: the data evolve with time, and follow the gesture.

These big gestures are the basis of the architecture of many sound systems. But the expressivity one can add when interpreting a score mostly comes from added values such as nuances, small glides of frequency (appoggiatura, trill, portamento) or vibrato. Nuances by themselves can be seen from two points of view: either it is a climate change, or it is an accentuation given at a time where something else happens.

2 Home-made instruments and associated gestures

We will now see how each of our home-made instruments has its own territory of gestures, a notion which is at the basis of the concept of a true instrument.

2.1 The scangloves

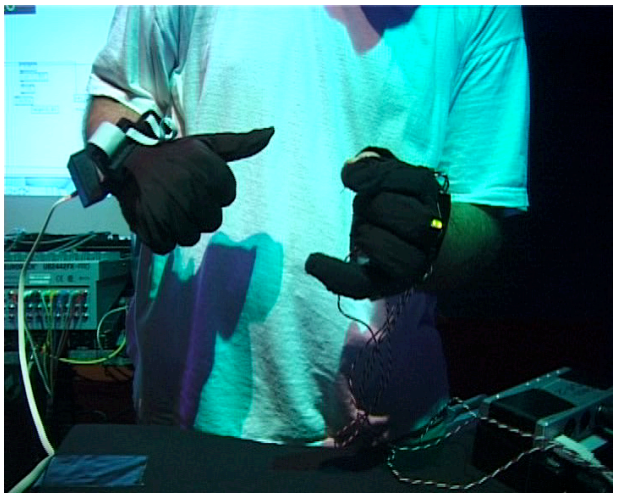


Figure 1: the two scanned gloves

The scan gloves (described in another article in this gims symposium) are a combination of two gloves linked to a scanned synthesis algorithm(Verplank et al., 2000). Once again it is a bimanual control (Kessous and Arfib, 2003)where one hand uses a sign recognition algorithm to provide pitch, whereas the second one both triggers (as a plectrum) and modulates the sonic signal (as the aftertouch of a keyboard).

Here we have a gesture of selection from one hand and a decision/modulation gesture from the other. The non-referred hand gives the pitch and it is only when it is triggered that it is activated again, and then modulated. This can be compared with the guitar, from which it takes its metaphor: one hand touches the strings while the other one uses a plectrum. However the second hand also acts as a distort effecter so that it is again a combined gesture, very akin to the after touch of keyboard. This is possible by the way of tweaking a pressure sensor between a thumb and index finger, a way very sensitive for the human being. We see here that ergonomics has its importance, notwithstanding the fact that there is a musical choice behind every decision we make.

2.2 The voicer

The voicer links a vocal synthesis program with a gesture caption using both hands. The separation between the source signal and the filter algorithm allows a good discrimination and combination of musical gestures with an easy link to the devices. The mapping itself uses a special function to translate circular data on a tablet to pitch and an interpolation scheme for the calculation of filters simulating vowels (Kessous and Arfib, 2003).

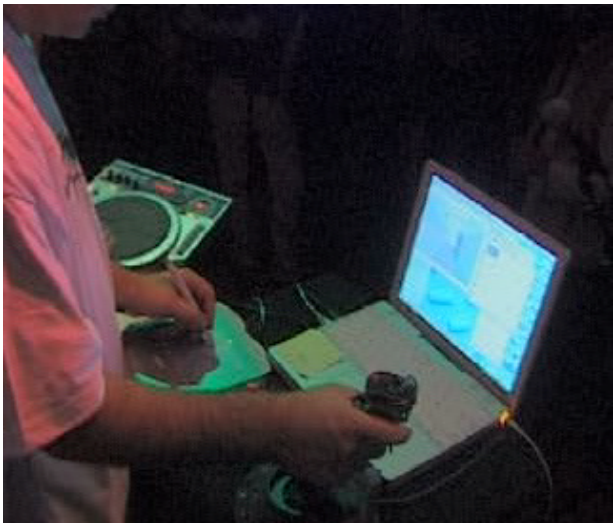


Figure 2: the voicer in action

The gestures that are strictly necessary are of two kinds: the circular coordinates of position of the stylet on the tablet gives the pitch, while the position of the joystick gives the vowel in the interpolation plane. But gestures are not static, and it is worth noting that it is the movement including the gestures that renders a movement in the sound. We will take three examples of these special gestures:

- continuous melodies are given by moving the stylet around a circle. When vibrato is needed, the special configuration of the mapping between the angle and the pitch allows to use the separation between two notes as a glide, a trill or vibrato.

- As the amplitude depends upon the stylet pressure, a melody can be built in a detached manner by using an inking gesture together with the pointing of different notes. A very good combination is to use the mechanical feedback of the joystick to provide movements with the other hand always coming back to the centre. This allows a real virtuosity with one hand while the other one only modulates slightly the vocality.

- Reversely it is possible to ply strict vowels while phrasing melodies each phrase being colored by a different vowel, such as the exposition of a theme by different colorations.

2.3 The filtering string

The filtering string uses a scanned synthesis algorithm (Arfib et al., 2002a)for the drive of an equaliser. The gestural data controls the scanned cord, in such a way that

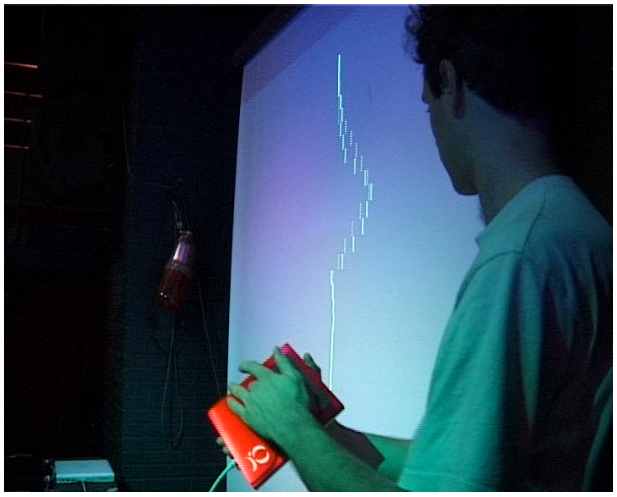


Figure 3: a demonstration of one tactex gesture

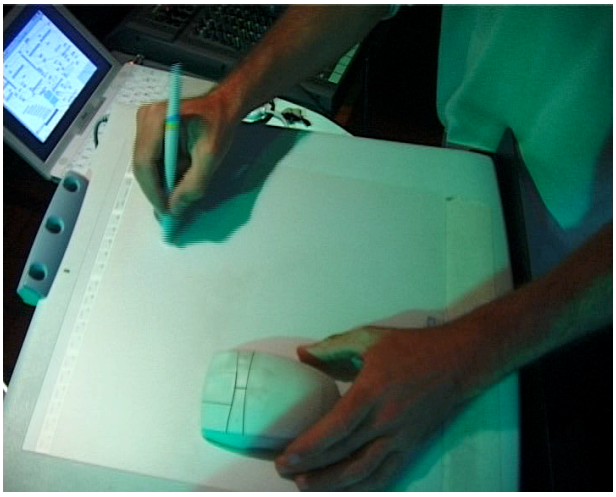


Figure 4: the photosonic emulator

a dynamical sytem is in between the two: by the way of forces, a movement is induced in this cord, which shape finally serves as a template for the equalisation system. A particularity, that we will see effective in the gesture range is that it uses a multipoint tablet for the introduction of these forces.

The dynamic characteristics influence much the gestures that are possible with such an algorithm. While one hand has a static mapping (similar to the voicer) for the scaning frequency, the other one uses a multipoint device to induce forces in the algorithm. This means that the dynamic of the gesture will be enhanced by the dynamic of the algorithm. Putting fingers in a static configuration on the Tactex tablets establishes a fixed sound (after a while) and specific movements will change dramatically the play. Here are some possible movements

- construct a form by moving the hand, sliding each finger in a gentle way

- construct a form by moving the hand, sliding each finger in a gentle way

- play more rapidly, in a kind of random fashion, in order to introduce new forces

- at a point it can even be conceived as a percussion instrument, where the action of impulses forces is followed by the response of the instrument.

Of course it is difficult to talk about gestures (videos are evidently better for this purpose) but this gives the hint that a repertoire of gesture can be build according to the dynamic aspect of the instrument itself.

2.4 The photosonic emulator

The photosonic emulator (Arfib and Dudon, 2002) takes benefit of a A3 Wacom tablet to capture the data from two surface sensors. This bi-dimensional data drives an algorithm combining the navigation in a sonic data base of rings (with some blend between them) anda very effective filtering which enhances and colours the sound according to the gestures. The mapping is quite straightforward: one hand governs the navigation and the other one the filtering action.

The repertoire of gestures has been demonstrated with the initial photosonic synthesiser and the same gestures are kept in the emulator. From a general point of view, the division between the work of both hands is governed by two principles

- navigation in a data base of rings can be perceived either as a melody or a climate, depending the speed, and the precision of hold we will have concerning these rings

- filtering cannot be seen as an independent feature. As an exemple extracting one harmonic from a ring depends upon the sonic content of this ring, so that the filtering gesture has to adapt to the work of the other hand.

New gestures may also happen from the interaction of gesture and sound listening. If vibrato gesture is quite intuitive (a vertical movement on the filtering device gives a vibrato) articulation such as arches spirals and circles, shut down with octaviation can be described using the navigation scheme because of two features: one coordinate corresponds to amplitude, and a specific button on the stylet of the tablet allows for octaviation, hence a possible rhythm.

2.5 Interfacing with pointing fingers.

Though it is not in itself an instrument, the concept of pointing fingers (Couturier and Arfib, 2003)is interesting, as it is a device that can replace other peripherals to select, trigger or modulate signals., bring some of its idiosyncrasies.

The pointing fingers allow to select (assigning an object by pointing at it), activate (via a trigger button placed either at the end of the finger or on the side of it) and modulate / navigate. It has been used both with the scanned

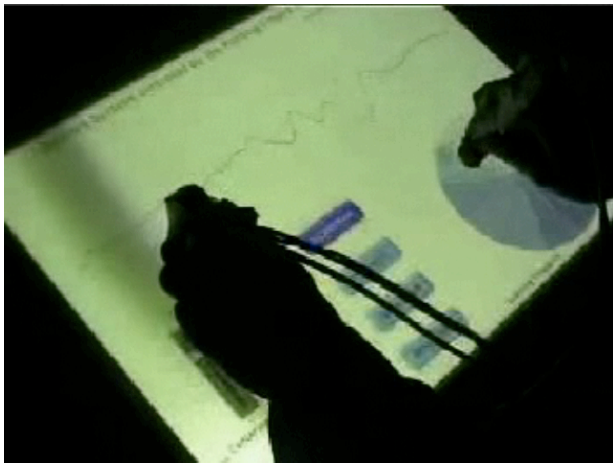


Figure 5: the pointing fingers in action

synthesis algorithm and the photosonic emulator. The fact that the visual inerface is just under the movement makes it clearly a very responsive device for the feedback, making a step towards the design of new musical instruments in virtual reality systems.

3 So what is a digital music instrument?

So far we have assumed that a good mapping and the good choice of devices allows a repertoire of gestures that caracterise the gestural control of systhesis algorithm. Now the main question comes. When can it be called a digital music instrument?

3.1 Acoustic and digital instruments

Let us see first what happens with acoustical instruments: they are called instruments because a performer can go on stage, play different scores in different styles, and also adapt his play to the other players. Some instruments like piano, which have a very rough in term of control (just hit a key with a specific velocity) are in the meantime fabulous instruments for the experimentation of harmony, so that one can say that a pianist is an orchestra of fingers. Another instrument, guitar, has proved to be a way bring orchestration features in the very play of the instrument.

Though it may be risky, let us try to see how gestural control can become musical instrument: we must first name it and use it as such; so go on stage, play with it and with others. We must be able to play a range of scores, for exemple a melodic trajectory, or an harmonic path, or a timbral evolution. Though we may play by heart (or intuition) at least we must be able to provide a way to use the unstrument, and have a pedagogy for it. Maybe be we still are a little bit far of this, but at least the most important features are there.

Every of our instruments has been played on stage, with scores that we have written for ourselves. And from this experience we know more how to provide a real identification for this instrument: it can be an identification by the dynamic of the sound, or the recognition of known repertoire of sounds. This is of course helped if there is some kind of melody or harmony in the structure so that it is even possible to provide a regular score. But also timbre for example is able to provide a real dynamism, a musical gesture, so that the purpose is to open new possibilities without closing too many others. As an example the use of scales in these new instruments is fine, and even more when one knows how to restrict oneself to what may bring the groove of a band.

So one general rule for the introduction of new instruments is to find the adequation of these instrument towards the musical objectives: the combination of a melodic structure together with a possible modulation/articulation of timbre may bring new ways to play (for example deplacing the perception focus on timbre allows to play fast and approximate melodies). The dynamic aspect must always been taken in account: gestures are not postures and even a clarinettist playing one note is always modulating it.

3.2 Evaluation in term of Computer Human Interfaces? motion and emotion

This leads to an open question: is it possible to evaluate a musical instrument, in ways similar to human-computer interfaces? First we must define the context of this evaluation, tasks are not always similar to expression for example. The criteria of musicality should be defined in a more precise way, the same for the ergonomy or the practical manipulability. One answer would be: how easy is it to explore the sonic universe that is suggested by the instrument.

One should say again and again that designing a digital musical instrument is not only a matter of mapping or devices, it is the adequation of the sound with the movement one can make, so the emotion that one wants to express corresponds with the sonic result and its emotional content. Inadequations though they may be fruitful as challenges must not break the fragile link between what one plays and what one hears. In a word it should be possible to incorporate new instruments so that they make part of us. Looking at videos (on rehearsal or during concerts) gives very good hints on the way these instruments allow the imagination to be connected in real time, which is a good sign for the future developments.

4 Conclusion

The title motion and emotion is hard to tackle directly. An alternative strategy has been taken: starting from the simple, algorithms and gestures, we have scrutinised the

missing link, the mapping, as the key point to provide a good repertoire of gestures. From there it has been possible to reintroduce the emotion as a way to play real music. And this music gives plenty of information on the movement used, influencing back again the design, the ergonomics of the instrument making.

References

Daniel Arfib, Jean-Michel Couturier, and Loic Kessous. Gestural stategies for specific filtering processest. *Proceedings of DAFx02 conference, Hamburg*, pages 1–6, 2002a.

Daniel Arfib, Jean-Michel Couturier, Loic Kessous, and Vincent Verfaille. Strategies of mapping between gesture data and synthesis parameters using perceptual spaces. *Organised Sound*, 7(2):127–144, 2002b.

Daniel Arfib and Jacques Dudon. A digital emulator of the photosonic instrument. *Proceedings of the 2002 Conference on New Instruments for Musical Expression*, NIME-02:128–131, 2002.

Jean Michel Couturier and Daniel Arfib. Pointing fingers: Using multiple direct interactions with visual objects to perform music. *Proceedings of the 2003 Conference on New Interfaces for Musical Expression (NIME-03), Montreal, Canada*, NIME-03:184–187, 2003.

Loic Kessous and Daniel Arfib. Bimanuality in alternate instrument design. *Proceedings of the 2003 Conference on New Interfaces for Musical Expression (NIME-03), Montreal, Canada*, pages 2–14, 2003.

Bill Verplank, Max Mathews, and Robert Shaw. Scanned synthesis. *Proceedings of the 2000 International Computer Music Conference, Berlin, Zannos editor, ICMA*, pages 368–371, 2000.

Sensing Systems for Interactive Architecture

Bert Bongers
MAAS Lab
Maastricht, The Netherlands
bertbon@xs4all.nl

1 Introduction

Architecture is becoming increasingly dynamic. After decades of developments, partly enabled by the use of computers in the design process, buildings have appeared with free form curves, suggesting motion, as a frozen movement. There is a vision of 'liquid architecture', ranging from an "architecture that is more of time than of space" (de Sola Morales, 1997), the combination of water and dynamics (Oosterhuis, 1996) to the inclusion of the "fourth dimension" in the Trans-architecture from the digital to the real world (Novak, 1991). A real time moving architecture is emerging, and in fact perhaps architecture was never meant to be static.

Electronic systems, communication technologies, computers and networks form the nervous system of the architectural body. By giving this nervous system sense organs (sensors) and hands and feet (actuators) and a brain (computer) it becomes possible to 'interactivate' a space (Bongers, 2002, 2003). An Interactivated Space is an environment which interacts with the people that are in it. Interactivated Spaces sense the activity of people, and (re)act through a variety of displays: auditory, visual, kinetic, haptic.

Already there are many dynamic elements and materials in buildings such as elevators, sliding doors, air flow, heating, sound and light (Travi, 2001). Technologies are now becoming available to enable dynamic structures. To give meaning to these dynamics, to control it and to interact with it, sensing systems are needed that facilitate the connection between people and their technological environment (Bongers, 2004). In this paper some recent developments in the field of architecture are described.

2 Projects

In this section several architectural projects are described involving dynamic elements and interaction. In these projects I developed sensor systmes and interaction engineering.

A merging of disciplines is taking place. It is crossing the traditional organisation along sensory modalities, bringing in knowledge from the fields of music (the ear, performative, time based, intimate), video (the eye, time based, screen size), architecture, and new technological developments.

2.1 Water Pavilion

The first project I was involved in was the Water Pavilion, built in 1997 as part of the Delta Works in The Netherlands. This interactive building is very well known and documented extensively elsewhere (Zellner, 1999) (Schwartz, 1999) (Jormakka, 2002). It was designed by the Dutch architects Lars Spuybroek and Kas Oosterhuis, with a team of other experts from music, visual art, and technology.

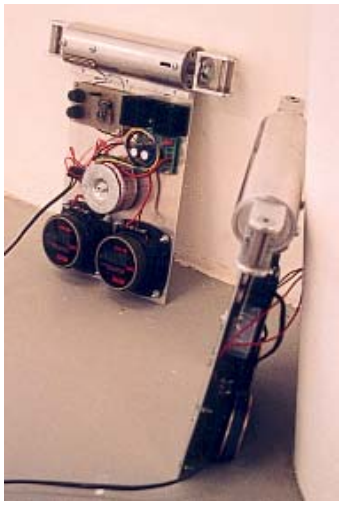
The sensors used were photocells, detecting the presence and motion of the audience in a certain area, touch sensors to be pushed by the hand or to be stepped on, pulling sensors, and the 'Surfboard' which was essentially a large joystick (custom developed based on infrared proximity sensors that translated the three Degrees-of-Freedom (DoF's) into electrical signals). These sensors would influence the computer generated projections of virtual worlds and surfaces, sound spatialisation and light movements. Water flows and environmental light were modulated as well, making use of all the traditionally dynamic materials (Harris, 2002).

In addition to the sensors mentioned, an electronic weather station captured and relayed environmental conditions such as wind speed and direction to the system, influencing the behaviour of the space.

2.2 Deep Surface

For an architectural exhibition in Hilversum, The Netherlands in 1999 of Lars Spuybroek, a lightweight curved projection surface was designed by him which floated in space suspended by connections to the columns of the actual building (Spuybroek, 1999). The aim was to make a sonification of the structural tension between the

floating structure and the resident architecture, and the minimal changes in it, through two oscillator / amplifier units.



The oscillators produced a pure sine wave with a base frequency 800 Hertz and a constant volume. The frequency was influenced by the structural tensions, measured by two custom built tension sensors. The sensors consist of a moving part (which is connected to the ropes of the structure) inside a spring, fitted in an aluminium tube. The moving part, the piston, then is connected to a slide potentiometer. The sensors are sensitive around 400 kg (mechanically adjustable by changing the spring compression). The two oscillators were voltage controlled with the potentiometer of the sensor. The combined sound resulted in a beating frequency due to the detunings.

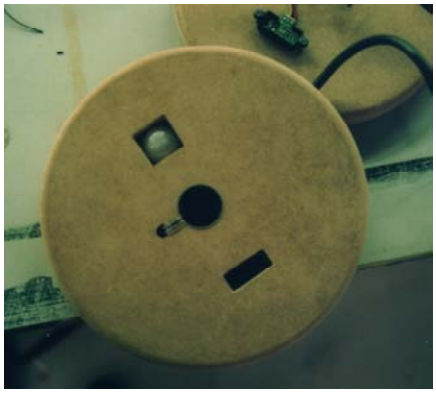
2.3 Trans-ports

Behind the ongoing project 'trans-ports' of Kas Oosterhuis lies the vision of a moving, interactive architecture, which displays information surrounding the inhabitants, and is connected to other systems in through networks. In 2000 a version was developed for the Architecture Biennale of Venice. For this three curved screens were hung in the space, enclosing a space where images were projected influenced by sensors. A system of three networked computers, one for each video projector, generated virtual worlds taking in the information from the sensors. The sensors were PIR motion detectors, hung from the ceiling above the audience, with the lenses partly covered with tape in order to limit the field of detection in the horizontal plane. The result is a detection grid which somewhat looks like the picture below.

2.4 Muscle

For the 'Non-Standard Architecture' exhibition in the Centre Pompidou in Paris (November 2003 – February 2004) the architectural office ONL of Kas Oosterhuis and Ilona Lenard devised a new structure called Muscle. It is a structure balanced by the pressure of an air balloon which is spanned by pneumatic 'muscles'. The pressure of the balloon volume is constant, while the tension of the muscles can be varied in real time under computer control by changing the air pressure of each individual muscle, resulting in a dynamic system of pushing and pulling forces. The size of the structure is 10 x 4 meters with a height of 2 meters. The controlling computer system and the valves that regulate the muscles are inside the balloon.

On the cross points of the pneumatic muscles eight sensor disks are mounted. Each sensor disk contains a proximity sensor, a motion detector, and a touch sensor.



Through these disks the audience is able to, explicitly or unconsciously, interact with and influence the behaviour of the Muscle 'body'.

2.5 ProtoSpace

At the architecture department of the Technical University of Delft a new interactive space has been set up in 2003 – 2004, ProtoSpace, by the

HyperBody research group of Professor Kas Oosterhuis. The aim of this space is that through multiple, full field of view and eventually 3D projections, teams of designers can work collaboratively on the creation of structures and environments. The parametric nature of these kind of architectural designs is particularly well suited for interactivating, that is, actively being interacted with by the users through sensor systems. For ProtoSpace we developed a system consisting of a combination of on-body and in-space sensing techniques, to control the virtual worlds and elements.
The In-Space elements are photocells, creating sensing paths across the room, PIR motion detectors, infrared proximity sensors, and a grid of switch mats on the floor. The On-Body sensors, to be held by the participants, are pressure sensors and tilt switches fixed on a little wooden cube and wireless game controllers with several DoF's sensed.

3 Sensor and systems

In this section the sensors and systems are described in more detail, which are commonly used in the architectural projects described above.
The technologies and techniques are often borrowed from the field of live electronic music, which has a longer tradition of real-time control of parameters of computer generated algorithms and models. Often however a translation has to be made from the intimate scale which is primarily the level of the musical instrument, to the architectural scale, including everything in between.

3.1 Proximity sensing

For sensing the proximity of a moving object, such as a hand approaching a surface, the Sharp 'Ranger' can be used, a small infrared reflection detector that is available in various ranges. We use the longest range available.
The unit transmits a beam of modulated infrared light, which can reflect off an object. This reflection is detected, and as the angle of reflection is proportional with the distance of the object, using a method called triangulation the angle (and therefore the distance) is deducted from the displacement. The receiving element is a linear CCD array, which senses the displacement of the reflected beam. It is quite insensitive to environmental light, and works with most objects including the human hand, it is mostly independently to colour, texture and type of material (it works best with a white sheet of paper). The output voltage first increases (although not linearly) when an object approaches the sensor, until at a distance of about 10 cm. Due to the way the optics work, the voltage will then decrease again. There is no way to tell what the object is doing, unless one adds another sensor (with a different range or placement) and work out the difference in signals for a larger range. The easier method is to place the sensor 10 cm behind the sensing surface if possible, or put a 10cm long tube around it.
These sensors are used in ProtoSpace, and particularly on the Muscle body sensing the close proximity of a person.

3.2 Touch sensing

To sense touching the Interlink FSR (Force Sensing Resistors) are used. They are thin laminates of plastic with conductive ink and contacts that vary electrical resistance proportionally with the force applied. When no force is applied the resistance is infinite, and from a pressure of a few grams (the slightest touch of the tip of the finger) the resistance starts to decrease until the full force of about 20 kilos is applied.
These sensors are used in ProtoSpace on an object to be picked up, and on Muscle (slightly bigger ones) for the audience to touch the system.
In the Water Pavilion these sensors were used in the foot and hand sensors, built in a contraption of wood and rubber to deflect the forces.

3.3 Motion sensing

To detect the movement of people, PIR (Passive InfraRed) sensors are used. This type of sensor is common as a burglar alarm, it detects the motion of infrared energy such as radiated by a human body. The sensing element is in fact a small CCD camera sensitive to infrared. It contains a circuit that infers motion of the infrared, and then closes an electric switch (relay). The light comes in through a fresnel lens, a type of (plastic) lens that fragments the light in such a way the various movements make certain patterns on the CCD, which then get detected. They typically have a very wide range, but with some black tape on the lens the angle of view can be adjusted, as we did in Trans-ports, in order to use them for position sensing. This type of burglar alarms often have an adjustable delay (switch on

time) which is unwanted because the minimum 'on time' is usually a few seconds. This type of behaviour is good for the security purposes but not for our application where it can be rather programmed in software if needed.
For ProtoSpace a smaller type is used, which is easier to use and more flexible. These react fast and the switch output follows the movement detected, that is, when the movement stops the signal changes immediately.

3.4 Position sensing (line)

For locating people in space, photocells can be used which produce a narrow infrared beam of light and detects the reflected light from a reflector. They are industrial types, insensitive to environmental light and only detecting the reflection of its own modulated light. When the beam is intercepted, by a moving person it will close an electric switch or relay. There are three types of photocells, for this application usually the reflective type is chosen because it has a long range (as opposed to the diffuse types that don't require a reflector) and is easier to install that the paired type (separate transmitter and receiver). This sensor was used for detecting people's position in ProtoSpace, and it was also used to locate the audience in an area of the Water Pavilion.

3.5 Position sensing (point)

To sense the presence of a person in a particular location, switch mats are used in ProtoSpace. These are simple devices, that contain electric contacts that close when a certain force is applied anywhere on the mat when stepped on.

3.6 Converters and systems

The sensor converter systems used in the projects described in this paper are also ones that are often developed for electronic musical purposes. In Transports, Muscle and ProtoSpace we use a Sonology MicroLab, a straightforward analog and switch sensor to MIDI converter. The MIDI protocol, also from the musical world, was used because of it being versatile and widespread. Part of software used, the real time modelling program VirTools, responds to MIDI commands (partly due to a collaboration with the company in the first project). In the Water Pavilion an Infusion iCube was used, which proved to be satisfactionairy, while another part of the system which was based on PCI cards and custom developed software brought in a latency of about 10 seconds....
In ProtoSpace experiments are also carried out with low cost solutions for part of the input, by applying standard USB keyboards and game devices to which the sensors and / or switches are connected.
Further developments are anticipated, enabling higher precision and speed.

4 Experiences and Conclusions

It takes time and effort to make a transition from the traditionally static media, such as graphic design, to dynamic media such as music and video, to the new interactive media forms. This can be seen in the developments in architecture too, from static, to time based, to interactive. One of the developments that accelerate this process is the thoroughly parametric nature of the architectural models, making them particularly well suited for real time control from the outside world. And once the gap is crossed by the sensors from real world actions into the computer systems, traditionally the most difficult step in some respect, the issue of *mapping* needs to be dealt with. In contrast to the musical disciplines, where there is more of a tradition of real-time and performative aspects of the whole trajectory including mapping, a lot needs to be invented and explored in the realm of interactive architecture. Also the scale is very different, or at least extended from the primarily intimate scale of the musical instrument to the larger scale of architecture, which has been discussed elsewhere (Harris & Bongers, 2002). The projects discussed in this paper attempt to deal with this, particularly the ProtoSpace project which is set up as a longer term research project.
Another issue to deal with is that in these situations, as opposed to most electronic musical instruments, is the multi-user (or multi-player) nature of the architectural situation. This is one of the main research issues addressed in the ProtoSpace project.
Also there are limitations of the current technologies, and future improvements have been identified. These include both the application of more advanced technologies from the real time disciplines such as music, as well as technologies from other research fields such as ubiquitous computing and HCI.
A tension was experienced between the extreme malleability of the virtual models and the limitations of the physical control elements. An early attempt to

interact with a model in VirTools, a virtual cube of which the dimensions and orientations were changed by manipulating a real world object that consisted of a wooden cube with pressure and tilt sensors, showed the strength and at the same time the limitation of such an approach.

The strength is the direct connection between the real world object through which the virtual object is manipulated, which turns into a weakness when one realises that the real world object can't change as much as the virtual object, thus limiting the malleability. On the other hand however is the more traditional way of manipulating the virtual objects and movement of a character or viewpoint ('camera') in virtual space, by pressing the arrow keys.
Somewhere on the line between these two extremes may be an optimum, but perhaps it can only be found on a line in a completely different plane. This is what we search for, which is very fascinating and hopefully it will drive us towards solutions.

References

Bert Bongers, Interactivating Spaces, *Proceedings of the Systems Research in the Arts conference*, August 2002, Germany.

Bert Bongers, Interactivating Spaces, L'Arca Edizione, June 2003

Bert Bongers, *Interaction with our electronic environment – an e-cological approach to physical interface design*, Cahier Book series no 34, Utrecht, March 2004

Kari Jarmakka, *Flying Dutchmen, Motion in Architecture*. Birkhäuser, 2002

Yolande Harris, Architecture and Motion: Ideas on Fluidity in Soundm Image and Space, *Proceedings of the Systems Research in the Arts conference*, August 2002, Germany.

Yolande Harris and Bert Bongers, Approaches to Creating Interactivated Spaces, from Intimate to Inhabited Interfaces. *Journal of Organised Sound*, Cambridge University Press, Special issue on Interactivity 7/3, December 2002

Marcos Novak, Liquid Architectures in Cyberspace. In: Michael Benedikt (Ed.), *Cyberspace: First Steps*. MIT Press, 1991.

Kas Oosterhuis, Liquid Architecture. Archis Magazine, 11, 1995. Also in: *Architecture Goes Wild*. 010 Publishers, 2002

Ineke Schwartz, A Testing Ground for Interactivity, Archis Magazine, 9, 1997.

Lars Spuybroek, *Deep Surface*, Rotterdam 1999.

Ignasi de Solà-Morales, Liquid Architecture. In: Cynthia C. Davidson (Ed.), *Proceedings of the Anyhow Conference*, Rotterdam 1997. Anyone Corp. NY / MIT Press, 1998.

Valerio Travi, *Advanced Technologies; Building in the Computer Age*. Birkhäuser, 2001.

Peter Zellner, *Hybrid Space - New Forms in Digital Architecture*, Thames & Hudson London, 1999.

Improving Gestural Articulation through Active Tactual Feedback in Musical Instruments

Bert Bongers, Gerrit van der Veer and Mireia Simon

HCI, Multimedia and Culture Research Group
Department of Computer Science
Vrije Universiteit
Amsterdam, The Netherlands
bertbon@cs.vu.nl

Abstract

In this paper an experimental setup is described, which will be demonstrated at the symposium. Various movement sensors are used in combination with tactile actuators, which provide feedback on the movements made. The aim is to investigate the application of active haptic feedback to improve gestural control of electronic musical instruments. The project is part of ongoing research which aims to improve human-computer interaction at the physical level by applying tactual feedback. The paper describes the background and some of the theory, rather then presenting results. The purpose of this paper is to introduce the ideas, set-up and approach.

1. Background

Musicians rely strongly on their sense of touch when playing traditional instruments, which helps them to control and articulate the sounds produced. In these cases, there are three sources of information for the player:

- kineasthetic feedback: the internal sense of the players own movement (proprioception)
- passive tactual feedback, the shape of the instrument and the elements touched (strings, keys)
- active tactual feedback, through the vibrations or other changing properties of the instrument

As with other electronic systems in general, players of electronic musical instruments such as synthesizers lack the information channel of active tactual feeback, unless it is explicitely built into the system. Due to the decoupling between control surface and sound source through the MIDI protocol, players are not inherently in touch with the means of sound production. The third feedback modality of a traditional instrument as mentioned above is missing. However, this decoupling can also be used as an opportunity because of the two-way nature of the link between interface and sound source, by designing and applying the active tactual feedback.

Ever since the invention of the famous Thereminvox around 1920, an instrument played by moving one's hands in the air in two planes near a pitch and a volume antenna, gestural controllers have been popular in electronic music. However, from the three feedback modalities above now only one remains, the proprioception. It is therefore more difficult to play accurately.

1.1 Tactual Perception

The human sense of touch gathers its information through various channels, together called *tactual perception* (Loomis & Leederman, 1986). These channels and their sub-channels can be functionally distinguished, although in practice they often interrelate.

Our sense of touch has three sources: the signals from the mechanoreceptors in the skin (our cutaneous sensitivity) informing our *tactile* sense, the mechanoreceptors in the muscles and joints (our proprioceptors) inform our *kinaesthetic* awareness of the location, orientation and movement of body parts, and the efferent copy signal that occurs when a person is actively moving by sending signals from the brain to the muscles (Gibson 1962). *Haptic* perception involves all three channels, which is usually the case when a person manipulates an object or interacts with a physical interface.

The feedback discussed in this paper mainly involves the tactile sense, particularly addressing the fast adapting and diffuse mechanoreceptors in the skin. This is often called the Pacinian system

(named after the Pacinian corpuscles that are the relevant mechanoreceptors), and is important for perceiving textures but also vibrations – its sensitivity overlaps with the audible range (Verrillo, 1992).

1.2 Tactual Feedback Modalities

In addition to the player's internal feedback, the instrument has to be designed to supply feedback information about the musical process manipulated. Reflecting the tactile and kinaesthetic sensory perception modalities, the system can address these modalities with (vibro-)tactile feedback and force feedback, respectively. The current research set up focuses on applying various forms of tactile feedback to display information to the player.

2. Gestures and Feedback

The movements of the player can be detected by various motion sensors, which are used as input by the system. This then generates both the sounds and the feedback information displayed through tactual actuators. The elements of this interaction loop are described in this section.

2.1 Gestures

A gesture can be defined as a multiple degree-of-freedom meaningful movement. In order to investigate the effect of the feedback a restricted gesture may be choosen. In its simplest form, the gesture has one degree-of-freedom and a certain development over time.

2.2 Motion Sensing

In the last decades several gestural music controllers have been developed. Sensing techniques can be those that are worn by the user (On-Body) or placed in the room 'looking' at the performer (In-Space). Several sensing technologies are available for movement tracking:

- *ultrasound* as used in Michel Waisvisz "Hands" and Laetita Sonami's "Lady's Glove".
- *infrared light* such as the Dimension Beam instrument, photocells, and the Sharp Ranger as used in our research.
- *laser beams* as used in instruments like the Termanova (Hasan et al, 2002) and the LaserBass (Bongers, 1998).
- *radiowaves* as used by the Theremin instruments and replicas, some of the MIT MediaLab instruments and the Solo piece by Joel Chadabe (1997).
- *camera tracking* with systems such as STEIM's BigEye, the Cyclops object in Max, and the EyesWeb system (Camurri et al, 2000).
- *inertial sensors* such as accelerometers and gyroscopes which measure acceleration and rotational speed respectively, from which position can be inferred.
- *magnetic field sensors* are used for small range movement sensing.
- *tilt switches* can be used to deterimine the inclination of an object or body part.

An inclusive overview of sensing techniques is beyond the scope of this paper, and is covered elsewhere in more detail (Bongers, 2000).

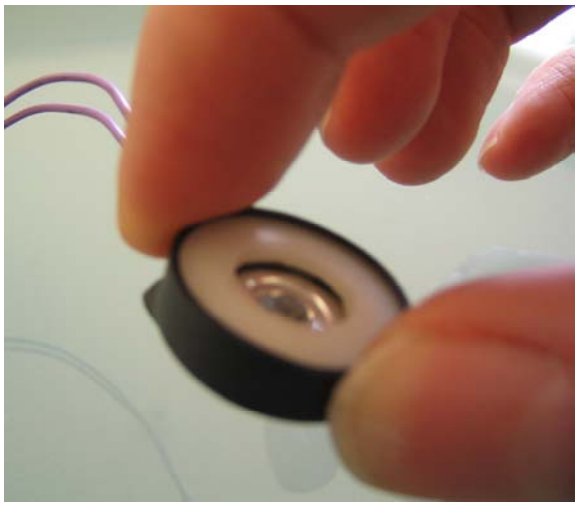
2.3 Tactual Feedback

As mentioned above, there are several forms of tactual feedback that can occur through the tactual modes as described, discriminating between cutaneous and proprioceptive, active and passive. Various technologies are available for tactual display, from 2D input devices with force-feedback (such as game joysticks) to multiple degree-of-freedom apparatus such as the Phantom, and custom built systems, which all have been used to investigate the effects of added tactual information feedback with positive results (Hardwick et al, 1996), (Oakley et al, 2001), (Chafe and O'Modrain, 1996), (Akamatsu and MacKenzie, 1996). For (vibro-)tactile feedback electromagnetic actuators (Chafe, 1993), in combination with little loudspeakers can be used. (Bongers, 1998, 2004).

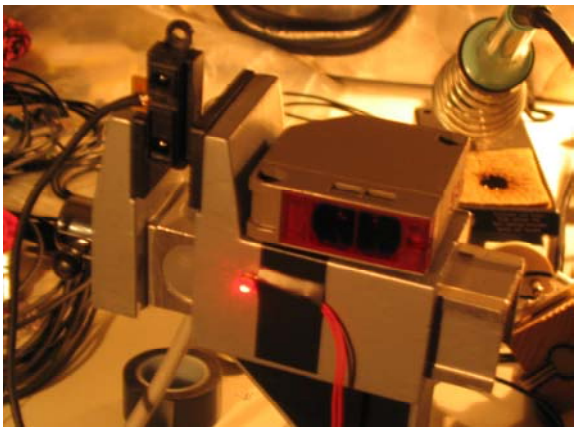
3 The experimental set-up

The goal is to feel something in a certain position or area when moving one's hand in space. A ring is worn with two tactile actuators. One actuator is a small electromagnetic device which can be used to produce the 'attack' of touching the virtual string or object.

The other actuator is a small loudspeaker acting as a vibrotactile element, producing further articulatory feedback from sound parameters such as pitch and envelope. This method has been used for earlier research in which it was proved that performance of pointing tasks with a mouse can be improved by supplying this kind of feedback (Bongers and Van der Veer, 2004).



The vibrotactile actuator consists of a small (Ø 20mm) loudspeaker, covered by a ring with a hole in it through which the vibrations can be felt with the fingertip without it being dampened by the finger tip pressure.
Currently we are exploring a combination of sensing technologies. To create the experience of playing and touching a virtual string we use a combination of a laser beam (and light sensor) to detect the hand of the player being in a certain line in space, and an infrared proximity sensor (Sharp Ranger) to detect the position in the line. (The original LaserBass, developed by the first author at Sonology about ten years ago for the Dutch composer Florentijn Boddendijk who still performs with it, is using ultrasound for the position sensing.)



The signals from the sensors are measured through Teleo Making Things hardware, read through the USB port in a Max/MSP patch running on a Apple PowerBook G4 computer, generating the responses. This results in the audible sound and the tangible sounds (virbrotactile feedback) using the sound outputs of the computer, and using the same Teleo hardware the attack can be made tangible with the tactile ring.

4 Future research and conclusion

Latency and speed are important issues in a musical context. For our purpose of generating tactual feedback this is even more relevant, in order to create a convincing experience of touch. Our first experiments so far therefore involve measurements of these issues, and informal results are promising.
For the main experiment, subjects will be given a musical task (such as playing a phrase) under two conditions: one with and one without tactual feedback. Performance parameters will be measured both qualitiative as well as quantitative.
The hypothesis is that under certain circumstances it will be easier to play a phrase accurately. In the first phase we will concentrate on one or two degrees of freedom, but in the near future it is hoped to explore the possibilities for applying articulatory feedback in a threedimensional space. Linking to one of our other projects, we will also investigate the application of tactual feedback in real-time video performance.

References

M. Akamatsu and I. S. MacKenzie, Movement characteristics using a mouse with tactile and force feedback. *International Journal of Human-Computer Studies*, 45, 483-493.

A. J. Bongers, Physical Interaction in the Electronic Arts: Interaction Theory and Interfacing Technology, in *Trends in Gestural Control in Music,* IRCAM 2000.

A. J. Bongers and G. C. van der Veer, Tactual Articulatory Feedback and Gestural Input, Journal of Human-Computer Studies, 2004 [submitted]

A. J. Bongers, Palpable Pixels: A Method for the Development of Virtual Textures, Chapter in S. Ballesteros & M. Heller, *Touch, Blindness and Neuroscience*, Madrid 2004

A. J. Bongers, Tactual Display of Sound Properties in Electronic Musical Instruments, *Displays journal*, 18(3), special issue on Tactile Displays. 1998

A. Camurri, S. Hashimoto, M. Ricchetti, R. Trocca, K. Suzuki, G. Volpe EyesWeb – Toward Gesture and Affect Recognition in Interactive Dance and Music Systems, *Computer Music Journal*, 24:1, pp. 57-69, MIT Press, Spring 2000.

J. Chadabe, *Electric Sound – the Past and Promise of Electronic Music.* Prentice-Hall, 1997

C. Chafe (1993). Tactile Audio Feedback. *Proceedings of the ICMC International Computer Music Conference*, pp. 76 – 79.

C. Chafe and S. O'Modhrain, Musical Muscle Memory and the Haptic Display of Performance Nuance, *Proceedings of the ICMC International Computer Music Conference*. pp. 428 – 431.

J. J. Gibson, Observations on active touch. *Psychological Review,* 69/6, 477-491.

A. Hardwick, J. Rush, S. Furner and J. Seton, Feeling it as well as seeing it – haptic display within gestural HCI for multimedia telematic services. *Progress in Gestural Interaction*, Springer-Verlag, pp. 105-116.

L. Hasan, N. Yu and J. A. Paradiso, The Termenova: A Hybrid Free-Gesture Interface. *Proceedings of the NIME New Instruments for Musical Expression conference*, May 2002, Dublin Ireland.

J. M. Loomis and S. J. Leederman, Tactual Perception. In: *Handbook of Perception and Human Performance*, Chpt. 31.

I. Oakley, S. A. Brewster, P.. D. and Gray, Solving multi-target haptic problems in menu interaction. In: *Proceedings of the CHI 2001*, ACM Press, pp. 357-358

R. T. Verrillo, Vibration sensing in humans. *Music Perception*, 9/3, pp. 281-302.

Expressive gesture and multimodal interactive systems

Antonio Camurri, Barbara Mazzarino, Stefania Menocci,
Elisa Rocca, Ilaria Vallone, Gualtiero Volpe

InfoMus Lab – Laboratorio di Informatica Musicale
DIST – University of Genova
Viale Causa 13, I-16145 Genova, Italy
www.infomus.dist.unige.it

Abstract

This paper introduces multimodal interactive systems as user-centred systems able to interpret the high-level information conveyed by users through their non-verbal expressive gesture, and to establish an effective dialog with users taking into account emotional, affective content. The paper covers two crucial research issues in the design of multimodal interactive systems: (i) the multimodal analysis, i.e., approaches and techniques for extracting high-level non-verbal information from expressive gesture performed by users, and (ii) the interaction strategies that such systems should apply in the dialog with users in order to produce a suitable multimedia output, given the information provided by the analysis and the current context. The two issues are discussed with reference to recent research projects at the DIST – InfoMus Lab and to examples of concrete applications based on the EyesWeb open platform (www.eyesweb.org).

1 Introduction

Multimodal interactive systems originate from the convergence of the need for designing user-centred systems allowing natural interaction from one side, and of the wide range of possible solutions that multimedia techniques provide for this aim on the other side. Multimodal interactive systems employ information coming from several channels to build an application designed with a very special focus on the user, and in which interaction with the user is the main aspect and the main way through which the objectives of the application are reached.

The design of multimodal interactive systems can highly benefit of cross-fertilization among scientific and technical knowledge on the one side, and art and humanities on the other side. The shift of attention from machine to human-machine interaction puts in evidence the need of a deep investigation of the mechanisms of human-human communication in order to employ this knowledge in the interaction design. This need of cross-fertilization opens novel frontiers to research in both fields: if from the one hand scientific and technological research can benefit of models and theories borrowed from psychology, social science, art, and humanities, on the other hand these disciplines can take advantage of the tools technology is able to provide for their own research, i.e., for investigating the hidden subtleties of human behaviour at a depth that was never reached before.

The shift of the focus on natural interaction with users has another important consequence: the increasing importance of the information related to the emotional, affective sphere. If for many years research was devoted to the investigation of more cognitive aspects, in the last ten years lot of studies emerged on emotional processes and social interaction. Consider for example the research on Affective Computing at MIT (Picard, 1997) and research on KANSEI Information Processing in Japan (Hashimoto, 1997). At the same time a growing interest can be observed on physicality: from studying human beings as "brains", the focus moved to the study of human beings as subjects having a body interacting with the environment. Thus, the relevance of movement and gesture as a main channel of non-verbal communication becomes evident, and a growing number of research is in this direction (e.g., see the Gesture Workshop series of conferences started in 1996).

In this framework, our research is centered on *expressive gesture*, i.e., on the high-level emotional, affective content gesture conveys, on how to analyse and process this content, on how to use it in the development of innovative multimodal interactive systems able to provide users with natural expressive interfaces (Camurri. Mazzarino, and Volpe 2003).

Expressive gesture is thus a key concept in our research (Camurri, Mazzarino, Ricchetti, Timmers, and Volpe, 2004). Kurtenbach and Hulteen (1990) define gesture as "a movement of the body that contains information". Gesture is not only intended to denote things or to support speech as in the traditional framework of natural gesture, but the information it contains and conveys is often related to the affective, emotional domain. From this point of view, gesture can be considered "expressive" depending on the kind of

information it conveys: expressive gesture carries what Cowie and colleagues (2001) call "implicit messages", and what Hashimoto (1997) calls KANSEI. That is, expressive gesture is responsible of the communication of information that we call *expressive content*.

Expressive content is different and in most cases independent from, even if often superimposed to, possible denotative meaning. Expressive content concerns aspects related to feeling, mood, affect, intensity of emotional experience. For example, the same action can be performed in several ways, by stressing different qualities of movement: it is possible to recognize a person from the way he/she walks, but it is also possible to get information about the emotional state of a person by looking at his/her gait, e.g., if he/she is angry, sad, happy. In the case of gait analysis, we can therefore distinguish among several objectives and layers of analysis: a first one aiming at describing the physical features of movement, for example in order to classify it, a second one aiming at extracting the expressive content gait coveys, e.g., in terms of information about the emotional state the walker communicates through his/her way of walking. From this point of view, walking can be considered as an expressive gesture: even if no denotative meaning is associated with it, it still communicates information about the emotional state of the walker, i.e., it conveys a specific expressive content. In fact, from this perspective, the walking action fully satisfies the conditions stated in the definition of gesture by Kurtenbach and Hulteen (1990): walking is "a movement of the body that contains information". Other studies aim at analysing the expressive intentions conveyed through everyday actions (like walking): for example, Pollick (2004) investigated the expressive content of actions like knocking or drinking.

This paper will explore two main issues in research on expressive gesture and in particular on their role in the development of multimodal interactive systems: (i) how a multimodal interactive system can analyse the high-level expressive content conveyed by its users through expressive gesture, and (ii) which strategies a multimodal interactive system can employ to produce suitable expressive gesture in response to users' gesture, in order to interact with users.

2 Multimodal analysis of expressive gesture: approaches and techniques

Several problems have to be faced when analysing expressive gesture. Firstly, there is the need of identifying a collection of cues for describing and representing expressive gesture. Secondly, algorithms have to be defined and implemented to extract measures for such descriptors. Finally, data analysis has to be performed on these measures in order to obtain high-level information. Given this very rough summarization of the analysis process, this section will give a quick review of our research on these issues.

From a cross-disciplinary perspective, research on expressive gesture descriptors can build on several bases, ranging from biomechanics, to psychology, to theories coming from performing arts. For example, in our work we considered theories from choreography like Rudolf Laban's Theory of Effort (Laban, 1947, 1963), theories from music and composition like Pierre Shaeffer's Sound Morphology (Shaeffer, 1977), works by psychologists on non-verbal communication in general (e.g., Argyle, 1980), on expressive cues in human full-body movement (e.g., Boone and Cunningham, 1998; Wallbott, 1980), on components involved in emotional responses to music (e.g., Scherer, 2003).

Two approaches have been employed to proceed in identifying descriptors for expressive gesture, the first one moving in a bottom-up perspective, the second using a top-down, subtractive method. In order to be effective, the approaches have to start from a quite constrained framework where expressiveness can be exploited to its maximum extent. One such scenario is dance and it is also a good example for describing the two methodologies.

The bottom-up approach requires a dancer performing a series of dance movements (short choreographies) that are distinguished by their expressive content. We call "microdance" a short fragment of choreography having a typical duration in the range 15-90 s. A microdance is conceived as a potential carrier of expressive information, and it is not strongly related to a given emotion (i.e., the choreography has no explicit gestures denoting emotional states). Therefore, different performances of the same microdance can convey different expressive or emotional content to spectators: e.g., light/heavy, fluent/rigid, happy/sad, emotional engagement or evoked emotional strength. Human testers/spectators judge each performance of the microdance. Spectator ratings are used to isolate features related to expressive content of gesture and to help in providing experimental evidence with respect to the cues that choreographers and psychologists identified. This is obtained by the analysis of differences and invariants in the same microdance performed with different expressive intentions. Notice that the same approach can be applied to music by asking a performer to perform the same piece with different expressive intentions. In fact, lot of research on expressiveness in music performance and on expressive gesture of performers employ this method (e.g., see De Poli et al., 2004; Gabrielsson and Juslin, 1995; Wanderley, 2001).

The top-down, subtractive approach starts from the live observation of genuinely artistic performances, and their corresponding audiovisual recordings. A reference archive of artistic performances has to be carefully defined for this method, chosen after a strict interaction

with composers and performers. Image (audio) processing techniques are employed to gradually subtract information from the recordings. For example, parts of the dancer's body could be progressively hidden until only a set of moving points remains, deforming filters could be applied (e.g., blur), the frame rate could be slowed down, etc. Each time information is reduced, spectators are asked to rate the intensity of their emotional engagement in a scale ranging from negative to positive values (a negative value meaning that the video fragment would rise some feeling in the spectator but such feeling is a negative one). The transitions between positive and negatives rates and a rate of zero (i.e., no expressiveness was found by the spectator in the analysed video sequence) would help to identify what are the movement (music) features carrying expressive information. A deep interaction is needed between the image processing phase (i.e., the decisions on what information has to be subtracted) and the rating phase. This subtractive approach is currently under investigation. In a recent pilot study (McAleer et al., 2004), for example, it has been employed to investigate animacy perception of spectators exposed to stimuli obtained with progressive elimination of information from two microdances (unprocessed microdance, silhouette only condition, geometrical shapes related to motion parameters, geometrical shapes related to dancers' positions only).

Once some expressive cues are identified, they need to be measured (possibly in real-time) on the expressive gestures the user performs. Expressive cues are likely to be structured on several layers of complexity. Going on with the dance example, some cues can be directly measured on the video frames coming from one or more videocameras observing the dancer. Other cues need for more elaborate processing: for example, it may be needed to identify and separate expressive gestures in a movement sequence in order to compute features that are strictly related to single gestures (e.g., duration, directness, fluency).

For this reason, in the framework of the EU-IST project MEGA (Multisensory Expressive Gesture Applications, www.megaproject,org) a conceptual framework for expressive gesture processing has been defined, structured on four layers (Camurri, Mazzarino, Ricchetti, Timmers, and Volpe, 2004).

Layer 1 (*Physical Signals*) includes algorithms for gathering data captured by sensors such as videocameras, microphones, on-body sensors (e.g., accelerometers), sensors of a robotic system, environmental sensors.

Layer 2 (*Low-level features*) extracts from the sensors data a collection of low-level cues describing the gesture being performed. Cues often comes from the cross-disciplinary sources listed above and are identified through the discussed methodologies. In case of dance, for example, cues include kinematical measures (speed, acceleration of body parts), detected amount of motion, amount of body contraction/expansion, etc. In figure 1 four of them are showed in the case of one videocamera.

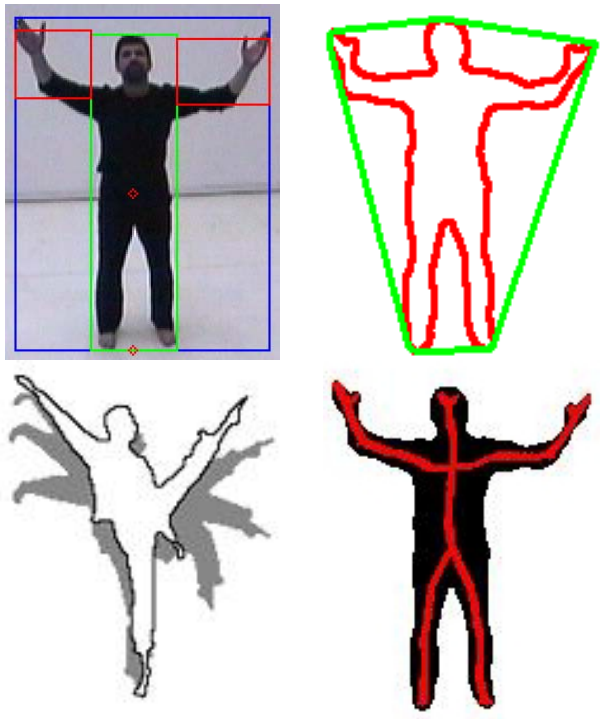


Figure 1: Examples of low-level motion cues extracted from one videocamera.

The cues have been extracted from a microdance using the EyesWeb open platform (www.eyesweb.org) and, in particular, the EyesWeb Expressive Gesture Processing Library (Camurri, Mazzarino, and Volpe, 2004). In the top-left figure some sub-regions of the body are individuated together with the body barycentre. The temporal evolution of both sub-regions and the barycentre can be analysed. The top-right figure shows the contour of the body silhouette and the minimum region surrounding the body. In the bottom-left figure the temporal evolution of movement in the last few frames is represented. The grey area is proportional to the amount of detected motion: what we call Quantity of Motion (Camurri, Lagerlöf, and Volpe, 2003). Finally, the bottom-right figure shows ongoing experiments on extraction of body skeleton. Some of these cues are currently subject of experiments aiming at (i) validating the algorithms employed for measuring them, and (ii) understanding how much these cues are really important in motion perception and in expressive content communication. For example, in a recent experiment (Camurri, Krumhansl, Mazzarino, and Volpe, 2004) we studied the relevance of the movement of the barycentre for motion perception and in particular for expectation in dance. Another experiment focuses on the perceptual relevance of Quantity of Motion.

Layer 3 (*Mid-level features and maps*) deals with two main issues: segmentation of the input stream (movement, music) in its composing gestures, and representation of such gestures in suitable spaces. Thus, the first problem here is to identify relevant segments in the input stream and associate them with the cues deemed important for expressive communication. For example, a fragment of a dance performance might be segmented into a sequence of gestures where gesture boundaries are detected by studying variations in velocity and direction. Measurements performed on a gesture are translated to a vector that identifies it in a semantic space representing categories of semantic features related to emotion and expression. Sequences of gestures in space and time are therefore transformed in trajectories in such a semantic space. Trajectories can then be analysed e.g., in order to find similarities among them and to group them in clusters. In Figure 2 an example of such process is shown: gestures are represented in a 2D space whose X axis represents Quantity of Motion while Y axis is Fluency. The analysis of the trajectories in the space was used for the real-time dynamic interpretation of two pieces of classical music: a neutral music score was dynamically interpreted and played (in a heavy, light, hard, soft way) depending if the same expressive intention was detected in the input gestures (demo developed in collaboration with DEI-CSC University of Padova at IBC2001, Amsterdam).

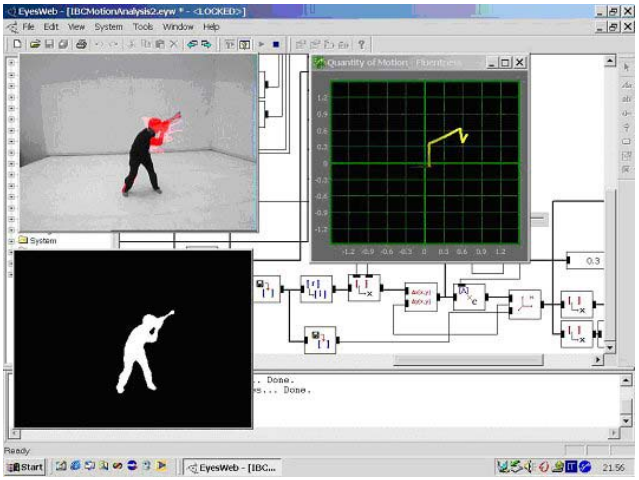


Figure 2: gesture represented as trajectory in a 2D space (X axis: Quantity of Motion, Y axis: Fluency)

Layer 4 (*Concepts and structures*) is directly involved in data analysis and in extraction of high-level expressive information. In principle, it can be conceived as a conceptual network mapping the extracted features and gestures into (verbal) conceptual structures. For example, a dance performance can be analysed in term of the performer's conveyed emotional intentions, e.g., the basic emotions anger, fear grief, and joy. However, other outputs are also possible: for example, a structure can be envisaged describing the Laban's conceptual framework of gesture Effort, i.e., Laban's types of Effort such as "pushing", "gliding", etc. (see Laban, 1947, 1963). Experiments can also be carried out aiming at modelling spectators' engagement or intense emotional experiences, i.e. not related to emotional labels such as basic emotions. Machine learning techniques are here employed, ranging from statistical techniques (e.g., multiple regression and generalized linear techniques), to fuzzy logics and probabilistic reasoning systems (e.g., Bayesian networks), to various kinds of neural networks (e.g., classical back-propagation networks, Kohonen networks), support vector machines, decision trees. In a recent experiment described in (Camurri, Mazzarino, Ricchetti, Timmers, and Volpe, 2004) we tried to classify expressive gesture in dance performance in term of the four basic emotions anger, fear, grief, and joy. Results showed a rate of correct classification for the automatic system (five decision tree models) in between chance level and spectators' rate of correct classification. In another experiment, discussed in the same paper, we measured the engagement of listeners of a music performance (a Skriabin's Etude) and analysed correlations with extracted audio cues and with cues obtained from the movement of the performer (a pianist).

3 Strategies for expressive interaction

The main task of a multimodal interactive system is to interact with the user, i.e., to establish a dialog with him/her. Multimodal interactive systems able to process high-level expressive information can benefit of this ability to make the dialog more effective. In other words, once extracted the high-level information from the incoming users' gestures, the system should be able to produce a response containing information suitable with respect to the context and as much high-level as the users' inputs. In order to perform this task, the multimodal interactive system should be endowed with strategies (sometimes called mapping strategies) allowing the selection of a suitable response. Such strategies are very critical, since they are often responsible of the success (and of the failure) of a multimodal interactive system.

A first example of strategy is based on *expressive semantic spaces*. An expressive gesture is analysed and is represented as a trajectory in a space (as in Figure 2). Then, such trajectory can be used for generating outputs (e.g., audio and visual content). Different outputs can be associated to different regions in the expressive space, and different metaphors can be applied. For example, an output can be produced having the same expressive features of the input gesture possibly in another modality (e.g., an expressive content movement mapped on

auditive/musical output as in the case of the "interactive HiFi system" described in the previous section), or it is possible to generate an output having opposite expressive content with respect to the input, thus trying to induce a feeling of perceptual contrast (paradox). This is a broadening of artistic research.

A more complex model takes into account the layered structure of the input expressive cues. The model is sketched in Figure 3. It can be illustrated with an example: consider an artistic performance where a multimodal interactive system is employed on stage. Several people (dancers, musicians, actors) are on the stage. Moreover, the stage contains fixed and mobile (e.g., robot) scenery, and virtual elements (e.g., information associated to particular paths). The system observes the environment and analyses the expressive gestures of the performers. The expressive information extracted by the four layers of analysis is fed as input to the interaction (mapping) strategies. Low-level cues will produce immediate responses: e.g., an increased energy in the motion of a dancer could produce an increased rhythm in percussions; a crescendo of a pianist can produce more vivid colours in projected abstract shapes. High-level cues will produce slower but continuous changes in the context of the performance: a slow transition between a rigid and angry movement toward a smooth and joyful one will produce a gradual and continuous change in the association of dancer movements to sound and musical instruments, e.g., from percussions to strings.

Similarly to cue extraction, interaction strategies are thus structured on more layers as well. In particular, in this model strategies are grouped in two layers:

- Simple and direct expressive strategies
- Complex and indirect expressive strategies

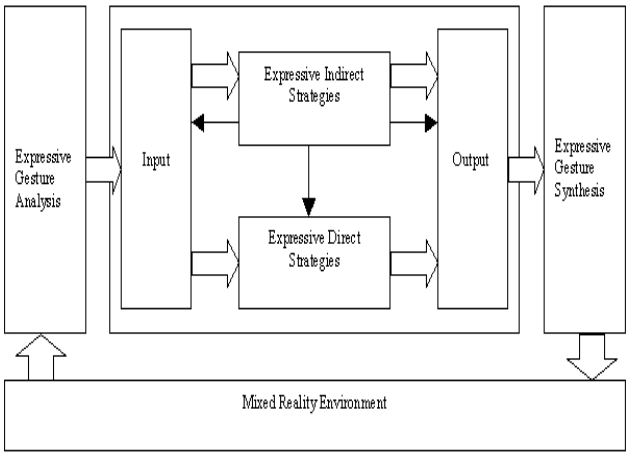


Figure 3: a model for expressive interaction strategies structured on more processing layers. Two basic kinds of expressive interaction strategies are considered: simple and direct strategies allowing the implementation of reactive behaviours and complex and indirect strategies more related to rational and cognitive processes.

With *expressive direct strategies* we mean an association without any dynamics of expressive cues of analysed expressive gestures with parameters of synthesised expressive gestures. For example, the actual position of a dancer on the stage can be mapped onto the reproduction of a given sound. Expressive direct strategies are often associated with the lower levels of the conceptual framework discussed in the previous section: for example, parameters extracted in Layer 2 (e.g., amount of motion – loudness) can be used to control particular features in the real-time generation of audio and visual content. Expressive direct strategies allow obtaining simple reactive behaviours of the multimodal interactive system. Several possible implementations are available for these strategies. One of them consists of collections of pre-defined condition-action rules, i.e., sets of rules associating given configurations of parameters coming from the analysis side with given configurations of synthesis parameters. Another one employs collections of algebraic functions computing values of synthesis parameters depending on values of analysed expressive cues. It should be noticed that while the complexity of an algebraic function can be freely increased according to any possible need, it anyway remains a static function, i.e., the mapping it induces does not change anymore once the function is defined and put at work.

Expressive high-level indirect strategies can be associated with explicit use of reasoning techniques, and are therefore related to rational and cognitive processes. They are characterized by:

- A state evolving over time (that is, they are dynamic processes): such a state can be updated for example by applying some kind of reasoning technique to the available information.
- Decisional processes, i.e., the system can make decisions based on the incoming information from analysis and the acquired knowledge. Such decisions can concern the kind of expressive content to produce and how to convey it, and can be related for example to the narrative structure of a performance.

Production systems and decision-making algorithms can be employed to implement this kind of strategies. Indirect strategies can also intervene on direct strategies. For example, a decision-making algorithm can be employed to decide which of K rules (direct strategies) that can be applied in a given situation (i.e., whose conditions are matched) should be applied.

A further layer of processing could be envisaged influencing both direct and indirect strategies. Such layer would concern the evaluation of the effectiveness of the currently employed strategies whether they are direct or indirect. Effectiveness could be considered under several aspects: for example, in artistic performances it could be related to the audience's engagement; in a museum scenario it could be associated to visitors' quality of the fruition of the museum exhibit. Such a measure could be

the result of a direct evaluation by spectators, in case it is not possible to calculate it automatically. Once a measure of effectiveness is available, it could be used to make decisions aiming at improving the overall performances of the multimodal interactive system by modifying and adapting its behaviour (i.e., its direct and indirect strategies) in order to maximize effectiveness.

4 Conclusions

Multimodal interactive systems based on expressive gesture processing proved their effectiveness in a number of different application fields. In the framework of the EU-IST project MEGA prototypes have been developed for applications in multimedia exhibits and performing arts. Performing arts demonstrated to be a very important testbed for multimodal interactive systems. Scientific and technological results, in fact, can interact with art at the level of the language art employs to convey content and to provide the audience with an aesthetical experience. Interaction at this level requires technology to be able to deal with the artistic content, i.e., what the artist wants to communicate and with the communication mechanisms enabling the experience in the audience. In this perspective, research on expressive gesture as a main conveyor of information related to the emotional sphere allows a redefinition of the relationship between art and technology: from a condition in which art uses technology for accomplishing specific tasks that only technology can afford (or that computers can do better than humans) to a novel condition in which technology and art share the same expressive language and in which technology allows the artist to directly intervene on the artistic content and in the expressive communication process.

Art, however, is not the only application field that can benefit of research on multimodal interactive systems. Another domain of interest is therapy and rehabilitation: we carried out some pilot experiments in the framework of the EU-IST project CARE HERE. For example, we developed prototypes of multimodal interactive systems to analyse body movements of different kinds of patients (Parkinson's patients, severely handicapped children, people with disabilities in the learning processes) and to map the analysed parameters onto automatic real-time generation of audio and visual outputs, attempting to create "aesthetic resonance". The underlying idea of aesthetic resonance is to give patients a visual and acoustic feedback depending on a qualitative analysis of their (full-body) movement, in order to raise engagement in patients (and consequently introduce emotional-motivational elements) without the need of neither the rigid standardisation required for typical motion analysis, nor of invasive techniques: subjects can freely move without on body sensors/markers in their environment. A pilot experiment carried out in order to test the developed techniques on Parkinson's patients is described in (Camurri, Mazzarino, Volpe, Morasso, Priano, Re, 2003).

A particular focus of our current research is on Tangible Acoustic Interfaces (TAI) that employ physical objects and space as media to bridge the gap between virtual and physical worlds and to make information accessible through touchable objects as well as through ambient media. TAI are addressed in the framework of the European Project TAI-CHI (2004-2006, 6FP, IST), whose primary objective is the development of acoustic-based remote sensing technologies which can be adapted to virtually any physical objects to create tangible interfaces, as a component of multimodal interfaces allowing the user to communicate freely and naturally with a computer, an interactive system, or the cyber-world.

Acknowledgements

We thank our colleagues at the InfoMus Lab who contributed to this research with useful discussions. In particular we thank Matteo Ricchetti, Paolo Coletta, Massimiliano Peri, Andrea Ricci.

References

Argyle M. Bodily Communication. Methuen & Co Ltd, London, 1980.

Boone R. T., Cunningham J. G. Children's decoding of emotion in expressive body movement: The development of cue attunement. *Developmental Psychology*, 34, 1007-1016, 1998.

Camurri A., Mazzarino B., Ricchetti M., Timmers R., Volpe G. Multimodal analysis of expressive gesture in music and dance performances. In A. Camurri, G. Volpe (Eds.), *Gesture-based Communication in Human-Computer Interaction*, LNAI 2915, Springer Verlag, 2004.

Camurri A., Mazzarino B., Volpe G. Analysis of Expressive Gesture: The EyesWeb Expressive Gesture Processing Library. In A. Camurri, G. Volpe (Eds.), *Gesture-based Communication in Human-Computer Interaction*, LNAI 2915, Springer Verlag, 2004.

Camurri A., Krumhansl C. L., Mazzarino B, Volpe G. An Exploratory Study of Anticipating Human Movement in Dance. In Proc. *2nd International Symposium on Measurement, Analysis and Modeling of Human Functions*, Genova, Italy, June 2004.

Camurri A., Mazzarino B., Volpe G. Expressive interfaces. *Cognition, Technology & Work*, Springer-Verlag, December 2003.

Camurri A., Mazzarino B., Volpe G., Morasso P., Priano F., Re C. Application of multimedia techniques in the physical rehabilitation of Parkinson's patients, *Journal of Visualization and Computer Animation*, 14(5), 269-278, Wiley, December 2003.

Camurri A., Lagerlöf I., Volpe G., "Emotions and cue extraction from dance movements", *International Journal of Human Computer Studies*, 59(1-2), pp. 213-225, Elsevier Science, July 2003.

Cowie R., Douglas-Cowie E., Tsapatsoulis N., Votsis G., Kollias S., Fellenz W., and Taylor J. Emotion Recognition in Human-Computer Interaction. *IEEE Signal Processing Magazine*, no. 1, 2001.

De Poli G., Mion L., Vidolin A., Zanon P. Analysis of expressive musical gestures in known pieces and in improvisations. In A. Camurri, G. Volpe (Eds.), *Gesture-based Communication in Human-Computer Interaction*, LNAI 2915, Springer Verlag, 2004.

Gabrielsson A., Juslin P. Emotional expression in music performance: between the performer's intention and the listener's experience, *Psychology of Music*, 24:68-91, 1996.

Hashimoto S., "KANSEI as the Third Target of Information Processing and Related Topics in Japan", in Camurri A. (Ed.) *Proc. Intl. Workshop on KANSEI: The technology of emotion*, AIMI (Italian Computer Music Association) and DIST-University of Genova, pp101-104, 1997.

Kurtenbach G., Hulteen E. Gestures in Human Computer Communication. In Brenda Laurel (Ed.) *The Art and Science of Interface Design*, Addison-Wesley, 309-317, 1990.

Laban R., Lawrence F.C. Effort. Macdonald&Evans Ltd., London, 1947.

Laban R., Modern Educational Dance. Macdonald & Evans Ltd., London, 1963.

McAleer P., Mazzarino B., Volpe G., Camurri A., Paterson H., Smith K., Pollick F.E. Perceiving Animacy and Arousal in Transformed Displays of Human Interaction. In *Proc. 2nd International Symposium on Measurement, Analysis and Modeling of Human Functions*, Genova, Italy, June 2004.

Picard R., "Affective Computing", Cambridge, MA, MIT Press, 1997.

Pollick F.E. The Features People Use to Recognize Human Movement Style. In A. Camurri, G. Volpe (Eds.), *Gesture-based Communication in Human-Computer Interaction*, LNAI 2915, Springer Verlag, 2004.

Schaeffer P. Traité des Objets Musicaux. Second Edition, Paris, Editions du Seuil, 1977.

Scherer, K.R. Why music does not produce basic emotions: pleading for a new approach to measuring the emotional effects of music. In *Proc. Stockholm Music Acoustics Conference SMAC-03*, pp.25-28, KTH, Stockholm, Sweden.

Wallbott H.G. The measurement of Human Expressions. In Walbunga von Rallfer-Engel (Ed.) *Aspects of communications*, pp. 203-228, 1980.

Wanderley M. Quantitative Analysis of Performer Non-Obvious Gestures. *IV Intl. Gesture Workshop*, London, UK, 2001.

Toward real-time multimodal processing: EyesWeb 4.0

Antonio Camurri[1], Paolo Coletta[1,2], Alberto Massari[1], Barbara Mazzarino[1],
Massimiliano Peri[1,2], Matteo Ricchetti[1,3], Andrea Ricci[1], Gualtiero Volpe[1]

[(1)] Infomus Lab – Laboratorio di Informatica Musicale, DIST – University of Genoa, V.le Causa 13, 16145 Genova, Italy
[(2)] NumenSoft s.n.c. di M. Peri & C., V.le Brigate Partigiane 10/4, 16129 Genova, Italy
[(3)] Eidomedia s.a.s., Via Assab 4/2, 16131 Genova, Italy

Abstract

The EyesWeb open platform (www.eyesweb.org) has been originally conceived at the DIST-InfoMus Lab for supporting research on multimodal expressive interfaces and multimedia interactive systems. EyesWeb has also been widely employed for designing and developing real-time dance, music, and multimedia applications. It supports the user in experimenting computational models of non-verbal expressive communication and in mapping gestures from different modalities (e.g., human full-body movement, music) onto multimedia output (e.g., sound, music, visual media). It allows fast development and experiment cycles of interactive performance set-ups by including a visual programming language enabling mapping, at different levels, of movement and audio into integrated music, visual, and mobile scenery.

EyesWeb has been designed with a special focus on the analysis and processing of expressive gesture in movement, midi, audio, and music signals. It was the basic platform of the EU-IST Project MEGA (www.megaproject.org) and it has been employed in many artistic performances and interactive installations. However, the use of EyesWeb is not limited to performing arts. Museum installations, entertainment, edutainment, therapy and rehabilitation are just some of a wide number of different application domains where the system has been successfully applied. For example, EyesWeb has been adopted as standard in other EU IST projects such as MEDIATE and CARE HERE in the therapy and rehabilitation field, and EU TMR MOSART. Currently, it is employed in the framework of the EU-IST project TAI-CHI and in the 6FP Networks of Excellence ENACTIVE and HUMAINE. EyesWeb users include universities, public and private research centers, companies, and private users.

1 Introduction

Our paper presents EyesWeb 4, a contribute to the area on music and interaction (Rowe 1993, 2001; Chadabe 1996), on expressive content communication in active spaces where integrated human movement (e.g., of a music performer, or a dancer), visual, and music languages concur as a whole perceived entity. EyesWeb contributes to research, experiment, and build applications in multimodal scenarios where different communication channels are used in human-computer interaction.

In the area of multimodal applications, a number of systems are available that let users to work on audio or video streams, such as PureData (Puckette, 1996), Max/MSP (www.cycling74.com), Isadora (www.troikatronix.com), vvvv toolkit (vvvv.meso.net) and others. However, such systems are often limited in that they are particularly oriented toward a modality of interaction, i.e., they might perform well only when working with video or audio. In other cases, the limitations are in the number of physical devices that can be managed by the system. Furthermore, designing multimodal interface is not only a matter of working with streams of different types, but mainly concerns the ability to work at different abstraction levels: in the framework of the EU-IST project MEGA (Multisensory Expressive Gesture Applications, www.megaproject,org) a conceptual framework for expressive gesture processing has been defined, structured on four layers (Camurri, Mazzarino, Ricchetti, Timmers, and Volpe, 2004). EyesWeb is a temptative to design an open platform that can be used at the different levels.

In recent years, EyesWeb has been satisfactorily used by our lab both for research purposes and for several types of applications, in museum exhibits or in the field of performing arts. Moreover, the platform has been made freely available on the Internet and the number of users has rapidly grown. This has enlarged the field of applications of the software platform, which has brought us to redesign the software in order to support the new requirements.

This paper will explore the new requirements in Section 2, and will explain the new characteristics of the software in Section 3, with an in-depth analysis of the added features and concepts. The new graphical user interface is briefly introduces in Section 4. Finally, Section 5 will give some concluding remarks.

2 From EyesWeb 3 to EyesWeb 4

Intensive use of the EyesWeb platform up to version 3.x has brought to evidence a number of new requirements. Many of these requirements have been faced by releasing updates to the existing EyesWeb version. However, some new requirements implied a deep revision of the software at different levels. This led us to the decision of redesigning the system from scratch, keeping into account original and new requisites, and trying to forecast possible future requirements.

Users of the EyesWeb platform are spread over a number of different fields, from education to performing arts, from industry to research, and more. This widespread use of the system has the consequence that user requirements have been collected and summarized through different means. One main channel has been the availability of public newsgroups (see www.eyesweb.org) where users can post support requests, comments, and suggestions. Other channels have been a direct contact with final users (e.g., research centers), and, of course, our direct use of the software both for research purposes or for several types of applications, from artistic and museum installations to therapy and rehabilitation.

Such different user requirements have been collected and discussed in depth, before becoming the software requirements for the new EyesWeb version described in this paper. In the following, we'll focus on such new requirements, trying to put in evidence the difference with previous releases.

In brief, main requests concern conceptual issues on multimodality, and issues concerning usability, performance, robustness, interoperability with other systems, optimizations for some common operations.

U*sability*, concerns the availability of a *subpatch* mechanism and some deep modifications of the scheduling algorithm. Providing subpatches means to provide a mechanism to hierarchically group a subset of a patch (up to a complete patch) to form a single component that can be managed as a single block. This has been one of the first requisites emerged from the use of EyesWeb 3.x, as the complexity of patches grew up as EyesWeb was used in scenarios of increasing complexity. More importantly, subpatches have been implemented with the perspective of supporting future meta-levels in which a supervisor software can activate and control different (sub)patches dynamically. This is particularly useful for implementing the indirect interaction strategies (see Camurri et al, 2004 in these proceedings).

Concerning modifications of the scheduling mechanism, they are oriented to hide some inner details to the user (e.g., the difference between active and passive blocks in version 3), and to manage inside the kernel some synchronizations issues which, in the previous version, had to be faced by the final user. Thus, synchronization of audio and video is now supported and managed in the EyesWeb language. From the implementation point, EyesWeb tries to use a single clock to schedule the patch execution.

Performance concerns the optimization of the kernel when managing audio and video streams, as well as the exploitation of the characteristics of the actual processors. In particular, a main focus is on multiprocessors systems. Nowadays, motherboards with dual processors are available at reasonable prices; the new version of the EyesWeb kernel is built in such a way that dual (or multi) processors computers can be exploited at best.

Robustness is obtained mainly by completely separating the graphical user interface from the kernel. As a consequence, kernel execution will not suffer of possible bugs of the user interface. Standalone applications which do not need user interface and have to operate for days in unsupervised environments (e.g. a museum) will take advantage of the versions with a minimal (or without) GUI. Moreover, the interaction paradigm between the interface and the kernel has been greatly simplified if compared with previous versions. Two main methods are used to communicate from the interface to the kernel or to notify events from the kernel to the interface. Such methods will be explained in detail in the next section.

Interoperability with other systems concerns the capability of the platform to embed plugins from existing systems. The previous version of EyesWeb already supported standard plugins, such as the VST plugins. However, the implementation of the adapter between EyesWeb and the external plugins was not simple, as EyesWeb blocks and EyesWeb GUI were strictly coupled. The simplification of the communication paradigm between the EyesWeb kernel and the EyesWeb interface implies that their coupling has been reduced. Thus, the implementation of the adapter for existing plugins will be greatly simplified.

Finally, *optimizations* for some common operations implies that some modules and datatypes are implemented natively inside the kernel. This implies that optimizations can be performed by the execution engine for such objects, as it can make more assumptions on their implementations. Among such modules we may enumerate flow control blocks (i.e., the blocks that were previously included in the Generic library), operations on basic datatypes (strings, integer, booleans, etc.), and more.

3 EyesWeb 4.0

The new version of EyesWeb introduces a number of features and concepts which were not available in the previous version. This section is devoted to the explanation of such novel characteristics and to give the motivation for their existence. In particular, the focus of this Section will be more on the new kernel features than on the user interface features (however, the interface has

been renewed and improved with a comparable number of novel characteristics).

The first feature is the distinction between the kernel engine and the patch editor. The kernel engine is contained in a separate dynamic-link library (dll) on which the user interface relies for some services. On the opposite, the kernel does not rely on the user interface for its proper working. Consequently, different interfaces might be provided to edit as well as to execute patches. As a matter of fact, besides the main editor which we commonly refer to as the user interface, two more execution interfaces will be provided. The first is a simple command line interface which runs in Windows console mode; the second is a Win32 application which runs hidden and just displays an icon in the tray area (the small area were the Windows clock is usually displayed). Both these interfaces do not provide editing capabilities; they only let users execute patches with no further overhead.

In the kernel, new concepts have been introduced: clocks, devices, catalogs, kernel objects, subpatches, pins, and collections.

3.1 Clocks

Clocks are the objects which are responsible of providing the current reference time, and to generate proper triggers and alarms when needed. Although more than one clock might be used to schedule the patch execution, the default behaviour of EyesWeb will be to use a single clock. Multiclock behaviour must be forced by the user choices, or are adopted in rare cases. EyesWeb can provide an internal clock, if no one of the objects in the patch can provide its own clock; such *default* clock is based on the Windows multimedia timer. Objects in the patch can provide their clock and ask the system to use that one. However, during the initialization of the patch for its execution, the kernel elects a preferred clock and, if some rare conditions are not verified, that specific clock is used to schedule the patch. The criteria to choose which clock to use when more that one is available is to give priority to renderer blocks (e.g., the sound playout blocks), than to source blocks, and finally to the other types of blocks. If this criterion is not sufficient to establish the winner clock (this may happen if more than one clock is available in the class with the highest priority), a weighted priority is computed based on the values assigned to the outputs of the blocks by the blocks developers. If this further criterion is not sufficient, the patch is executed using more that one clock, and the possibility of jitters is signalled to the user.

The new concept of clocks, besides reducing synchronization issues for the final users, is mainly useful to support multimodal interaction. As a matter of fact, interaction of different streams is simplified by handling the various streams with a common clock. Moreover, when a common clock cannot be used as the streams come from intrinsically unsynchronized sources, a synchronization mechanism can be provided natively by the kernel.

3.2 Devices

Devices are internal objects which manage the interface with hardware. They cannot be used directly by the user, but they are used by blocks developers to interact with hardware. This new layer of abstraction, which was not available in the previous EyesWeb version, adds the possibility to map the available physical hardware resource to the virtual devices used by blocks. This makes patches more portable among different computers, even if the hardware configurations of the systems are not equivalent

3.3 Catalogs

Catalogs are responsible to enumerate the available blocks, subpatches, clocks, devices, and datatypes, and to instantiate and deallocate such objects. They also have the responsibility to provide authorship information, i.e., name and description of the authors and company that developed the module, as well as information about the licence of that block. In brief, catalogs carry meta-information about the blocks, and provide a factory for their instantiation.

The delegation of the enumeration and factory responsibilities to an object which is not the kernel itself, provides a powerful mechanism to simplify the support of plugins from other platforms. In facts, the methods to enumerate or to instantiate plugins from other platform can vary considerably from one system to another; e.g., the methods to enumerate and instantiate plugins from DirectShow architecture is completely different from the method to enumerate VST plugins. Delegating these responsibilities to an object which is not the kernel itself loosen the coupling between the kernel and the objects (blocks, datatatypes, etc.). A further advantage is given by the fact that catalogs can be implemented both inside or outside the kernel, thus, it will be possible to provide the support for new types of plugins without the need to modify the EyesWeb core.

3.4 Kernel objects

The concept of kernel objects in another main difference with the previous EyesWeb version. With kernel objects we refer to some EyesWeb objects (i.e., blocks, datatypes, clocks, subpatches, or devices) which are embedded in the kernel and which have a priviledged access to the kernel. This is a main difference with the approach of the previous architecture: in EyesWeb versions up to 3.x all blocks and datatypes were treated as external plugins. EyesWeb did not rely on the exhistence of any of these objects for its proper working. At the first glance this new approach might strengthen the dependencies between EyesWeb and some objects, hence limiting expansibility of the system. However, you must consider that the use of kernel objects is limited to the components which really need access to the kernel

internals. Among such components we may enumerate control flow blocks (switch, for or while loops, conditional operations, etc.), basic datatypes (integers, doubles, booleans, strings, etc.), basic clocks (multimedia timer) or devices (keyboard, serial, or mouse). Moreover, this approach has some more advantages: one of them is the homogenization between datatypes and parameters. Previously, data which flowed from outputs to inputs of blocks was completely different from data which flowed to parameters. The former was implemented through external plugins, which we called datatypes, whereas the latter was implemented through a limited set of standard types (int, double, bool, char *). With the new approach, the basic standard types are implemented as kernel objects. Thus they have both the advantages that the kernel can safely rely on their existence, and that they have the same standard interface of all other datatypes.

3.5 Subpatches

Complexity of patches can increase as long as EyesWeb is used to operate in real scenarios. Subpatches offer a way to handle this complexity by managing a set of interconnected blocks as a single object. Different modalities to use subpatches will be provided in the new version of EyesWeb. They all share the concept to manage a complex set of blocks as a single one, but they differ in how multiple subpatches instances are managed and in the visibility scope. The first model is based on the assumption that different subpatches of the same type are disjoint; thus, modifying one subpatch does not alter the other instances of subpatches of the same type. A second model is based on the assumption that different instances of subpatches of the same type do share the subpatch class: modifying a subpatch acts on all instances of such subpatch. Making a comparison with a programming language, the first model is comparable with a macro, which causes the source code to be duplicated wherever the macro is used; the second mode is similar to a function call, where the function code is shared among all function calls.

Another difference is related to the visibility of subpatches: subpatches might be built in the scope of a patch and not exported outside the patch. Such subpatches will be visible only when the owner patch is loaded, and will not be usable outside that patch. Another modality is to export the subpatch in order to make it visible to the whole system. In such a case, all patch shall see such subpatches and referring to them will not result in an error.

3.6 Pins

Pins enable blocks to interconnect one another, or to connect with subpatches. The new version of EyesWeb has brought pins to the level of kernel objects, instead of beings just graphical objects as it happened up to EyesWeb 3.x. Thus, pins can be instantiated in a patch not only when attached to a block (i.e., as inputs, outputs, or parameters pins) but also as standalone objects, or as a facility to specify exported inputs, outputs, and parameters of subpatches.

Standalone pins can be places in the patch and have incoming and outgoing links. Besides providing a facility to avoid redrawing links when a source or destination block is removed, they allow the logical connection of graphically unconnected pins by simply assigning them the same name. Thus, remote parts of the same patch can be connected through pins with the same name without having to draw an actual link.

Another possible use of pins is to specify the exported parameters of subpatches. When a set of blocks is grouped to form a subpatch, not all pins are exported and visible by the users of such subpatch; the pins to be exported can be specified by selecting a subset of the available one; in such subset is it also possible to include pins which are placed in the patch as standalone pins.

3.7 Collections

Collections of different types are included among the objects that kernel can manage natively. The kernel itself is built upon collection facilities. Patches, for instance, contains collections of blocks, datatypes, links, devices, clocks, etc. Besides being used by the kernel, collections can be used by developers e.g. to build complex datatypes basing on the available ones. This simplifies developing higher level datatypes based on the ones available at the lowers levels, as well as a mechanism to link together different types of data; thus, in a way, they provide a basic support to multimodality.

4 Graphical User Interface

The EyesWeb Graphical User Interface has been deeply redesigned, in order to adapt to the new features introduced by the current EyesWeb version, and to provide a more modern look and feel. A first difference with the previous GUI is visible in the Catalog View (the treeview which is by default placed on the left side, and which shows the list of available blocks). Besides proposing the enumeration with the same characteristics as EyesWeb 3.x, it adds a new mode (called Catalog mode) which let users see the distinction of the blocks in disjoint catalogs. Another feature added by the Catalog View is the possibility to filter out some blocks from the list, thus simplifying the process to find out the desired module among all the available ones.

The Patch View, which is the main panel visible in Figure 1, adds zooming grid alignment capability. Moreover, it supports settings the parameters of multiple selected blocks in a single operations. For this purpose, the Properties View, which replaces the previous param dialog, is single instance. This means that only one instance of such view is active at a given time. However, such unique instance is able to show multiple values: when multiple blocks are selected, homogeneous parameters can be managed together, i.e., it is possible to set multiple values at the same time.

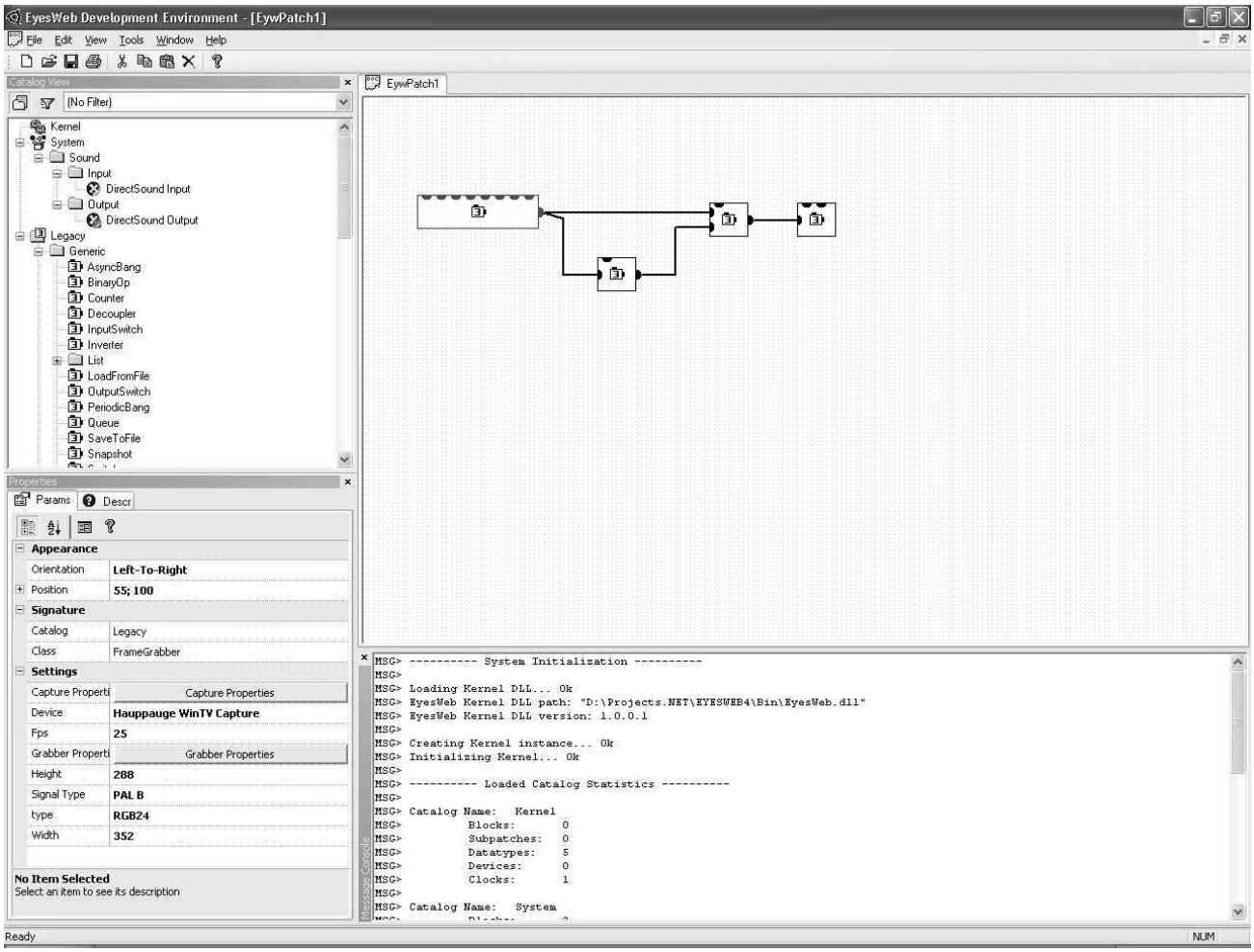


Figure 1. A screenshot of EyesWeb 4.0 at work.

5 Conclusions and future works

The paper has analyzed the upcoming version of EyesWeb, with a particular focus on the requirements that have emerged by the extensive use of the previous EyesWeb versions. EyesWeb was originally conceived to support multimodal human-computer interaction: the upcoming version of EyesWeb tries to enforce this characteristic, while improving the performance in the different application scenarios, and with the perspective of supporting different levels of abstraction in multimodal processing.

The upcoming EyesWeb version is still under development at the time of writing this paper. A first public demonstration will be done at the AISB 2004 convention.

6 References

R. Rowe, *Interactive Music Systems*, MIT Press, 1993.

R. Rowe, *Machine musichianship*, MIT Press, 2001.

J. Chadabe, Electric Sound. *The past and promise of electronic music*. Prentice Hall, 1996.

M. Puckette, Pure Data. Proceedings, International Computer Music Conference. San Francisco: International Computer Music Association, pp. 269-272, 1996.

Cycling74, http://www.cycling74.com

TroikaTronix http://www.troikatronix.com

Meso http://vvvv.meso.net/

A. Camurri, B. Mazzarino, M. Ricchetti, R. Timmers, G. Volpe, Multimodal analysis of expressive gesture in music and dance performances. In A. Camurri, G. Volpe (Eds.), *Gesture-based Communication in Human-Computer Interaction*, LNAI 2915, Springer Verlag, 2004.

A Movement Recognition Engine for the Development of Interactive Multimedia Works

Todd Ingalls Thanassis Rikakis Jodi James Gang Qian Loren Olson Feng Guo Siew Wong

Arts, Media, and Engineering Program
Arizona State University
PO Box 873302, Tempe, AZ, USA 85287-3302
480.965.9438

todd.ingalls@asu.edu, thanassis.rikakis@asu.edu, jodi.james@asu.edu, loren.olson@asu.edu, gang.qian@asu.edu, feng.guo@asu.edu, siew.wong@asu.edu

Abstract

This paper presents three interconnected activities being realized under the motion^e project at the Arts, Media, and Engineering Program at Arizona State University: the development of an motion-capture based, gesture recognition engine, the creation of queries based on this engine that extract structural information from dance performance, and the use of this information to create interactive sound and 3D animation that is tied to movement at a structural level. Results from testing of the gesture recognition engine are described and descriptions of a multimedia demonstration using the engine are given. Links to videos of demonstration of the system are also included.

1. Introduction

The work described in this paper is part of the motion^e project being developed at the Arts, Media, and Engineering Program at Arizona State University. This research project has several goals: 1) The creation of tools for the semi-automated extraction, of lexica and syntax for specific modern dance/movement styles from motion capture data; 2) The development of systems for automated dance documentation and score creation; 3) Research into algorithms and hardware for real-time multi-modal feedback (e.g. aural, visual and tactile) driven by motion capture data; and 4) fusion of all these elements for the creation of real-time interactive multi-media works where movement, sound and image can be correlated at different levels (from surface levels — the physical manifestation, to the level of the form) and achieve integrated contributions to meaning.

This paper presents three interconnected activities being realized under this project: a) creation of a motion-capture based, gesture recognition engine for a specific style of movement/choreography; b) formulation of real time queries that use this gesture recognition engine to extract structural information important in the communication of meaning; c) correlation of the extracted structure to the structure of sound and 3D animation, creating a integrated multimodal vehicle for the enhanced communication of meaning.

2. Background

Without a tool for the real-time analysis of movement at the level of structure, it is difficult to create interactive art works where movement can be related to other media at levels beyond the surface. If only surface level data is being considered it is not possible to connect, in real-time, movement motive, phrase and style manipulation to corresponding sound or image motive, phrase and style characteristics.

Different cues related to human moment have been used to drive interactive systems. For example, Camurri et al. (2000) have used the Laban movement qualities (Moore, 1988) to guide music synthesis. In addition, Dobrian and Bevilacqua (2003) have described in detail a system for mapping motion capture data to musical control data.

Our project is creating the tools and processes that allow for real time extraction of structural movement elements of semantic importance with the goal of establishing structural correlations between movement, sound and image. The project is realized by an interdisciplinary team that includes dancers, composers, visual artists and engineers. All stages and activities of the project (from the structuring of the experimental material, to the creation of the extraction engine, the creation of queries, the choice of feedback and the semantic correlations) require participation of all team members.

The team forums provide the full context for each aspect or problem. Specialized subgroups are formed from this

larger team to allow for more context aware decisions that are necessary for any type of meaningful analysis or feature extraction. For example, we will see later in this paper that decisions on the algorithms for gesture matching or query formation were based on information about the structure of the piece given by the choreographer and developed further by the whole team.

3. Choice and Creation of Movement Material

Recognizing that real-time extraction of structural elements of movement is a large and complex challenge, our team has chosen to take a very gradual approach with the initial developmental stages thereby allowing us to closely monitor the success of each choice in our methodology.

With this in mind, the first movement composition we chose to work with was the piece *21* by Bill T Jones. The piece has a number of important characteristics that reduced the feature extraction challenges. The piece is based on a fixed, finite, unambiguous vocabulary of 21 gestures which the choreographer describes as a "gesture tone row." Each gesture consists of a transition into a pose, a holding of that pose followed by a departure from the pose and a transition into the next pose. He numbers each gesture out loud as he performs the first exposition. This is followed by a second exposition where he repeats the gestures giving each one a related, connotational word or phrase that also indicates the famous cultural image that inspired the pose that is the mid-point of the gesture. He then goes into the development section where he mixes the gestures and their connotational words or phrases with improvisatory movement and stories from his life.

The structure of the piece and the ability of Bill T Jones to perform the gestures consistently offer the following feature extraction benefits. The gesture vocabulary being extracted is clearly defined and presented. The core of each gesture is a pose allowing for robust, transient-free recognition. The original ordering of the gestures is clearly defined and presented. The improvisatory movement combined with the 21 gestures in the development section introduces additional movement vocabulary that is of a very different nature than the pure 21 gestures thus making the gestures more discernable.

A second set of movement material was chosen for the extension of the single subject recognition engine. This second set of material was to help us investigate two challenges: capture and analysis in real time of multiple subjects and recognition of different movement vocabularies when performed simultaneously. This material was from the repertory of Trisha Brown and was taught to four ASU Department of Dance graduate students by Brown's choreographic assistance, Carolyn Lucas, during a residency in early January 2004.

To serve our goals we chose two phrases containing clearly distinct movement characteristics in combination with some similarities (to test for robustness). Phrase A had fairly static lower body movement with elaborate arm gestures and could be performed either separated in space or together in interlocking patterns. Phrase B involved travelling in space and also contained quite elaborate arm gestures.

4. Methods of Capture

Our current motion capture setup consists of a Vicon 8i DataStation with 8 infrared MCam2 cameras. In our setup the DataStation synchronizes digitized data from each camera at 120Hz.

For capturing the performance of Bill T Jones, we used a standard marker set provided by Vicon, HumanRTkm, which contains 41 markers. This marker set and associated kinematic model was adequate for these single subject captures.

In capturing the movement material from Trisha Brown, two problems had to be resolved. First, since we were attempting to capture four dancers at once, the standard 41 marker model on each dancer caused significant lag in the motion capture system as it had to track more markers. Secondly, because the dancers remained in close proximity to each other while performing interlocking arm movements, the markers were often mislabelled through crossover and therefore experiencing great difficulty in getting accurate data.

To alleviate these problems we developed a simple 17 marker-set customized for our goals. This kinematic model only had two markers per limb segment therefore rotations had to be eliminated and joint definitions had to be simplified.

5. Single Subject Gesture Recognition Engine

In this section we describe the gesture recognition engine, the various steps required in developing the system, and the various queries derived from the recognition. An overview of the gesture recognition engine can be seen in Figure 1.

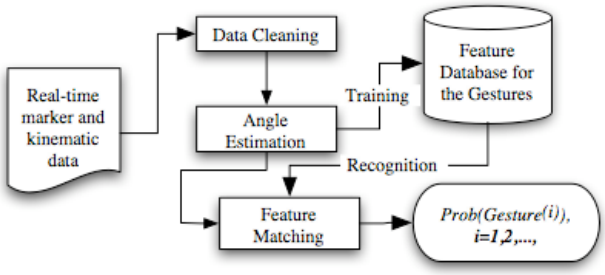


Figure 1: Gesture Recognition Engine

5.1 Body Segmentation

The whole human body is segmented into 10 body parts, including head, torso, upper arms, forearms, upper legs and lower legs. Feet and hands are ignored in this implementation. Each of these 10 body parts is regarded approximately as a rigid object.

5.2 A Real-time Data Cleaning Algorithm

Due to inherent issues of marker occlusions and marker mislabelling as described in Section 3, the first step in developing a reliable gesture recognition engine is to ensure that accurate data is being obtained from the motion capture system. To tackle the occluded marker problem, we have developed the real-time marker cleaning algorithm shown in Figure 2. To fill in missing markers we are using two different methods which varied depending on context. In the simplest case we use a static-body-part (SBP) method. This method explores the temporal correlation of the marker position in a frame in which the body part has paused. To determine whether a body part is static, we look at the marker movements on the same body part within recent frames and determine whether it is below a pre-chosen threshold. If this is the case, we can fill in a missing marker by copying the marker position from the previous frame into the current frame.

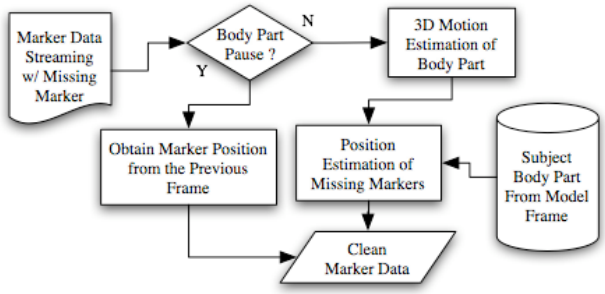


Figure 2: Data Cleaning Algorithm

In the more complex case, when the related body part is in motion, the missing marker is filled using a rigid-body (RB) based method. RB estimates the 3D translation and rotation of the body part between the current frame and the model frame, which is obtained from the subject calibration database of the motion capture system. The model frame has the global positions of all markers on each body part. For example, assume that the body parts related to the missing marker have at least 4 markers and that $m_c^{(4)}$ is the missing marker in the current frame. Let $m_c^{(1)}$, $m_c^{(2)}$ and $m_c^{(3)}$ be 3 non-collinear markers and O_c be their centroid in the current frame. Let $m_m^{(1)}$, $m_m^{(2)}$, $m_m^{(3)}$ and O_m be the positions of the same set of marker and there centroid in the model frame. Due to the rigidity of the body part,

$$m_c^{(i)} - O_c = R(m_m^{(i)} - O_m)$$

where R is the rotation matrix between the model and current frame and it can be computed using the visible markers.

The coordinate of the missing marker in the current frame can be computed using

$$m_c^{(4)} = R(m_m^{(4)} - O_m) + O_c$$

Figure 3 shows the data cleaning results for a missing marker for a hundred frames using the second approach. The large gap in the upper figure was caused by marker occlusion and it the lower image shows the cleaned data.

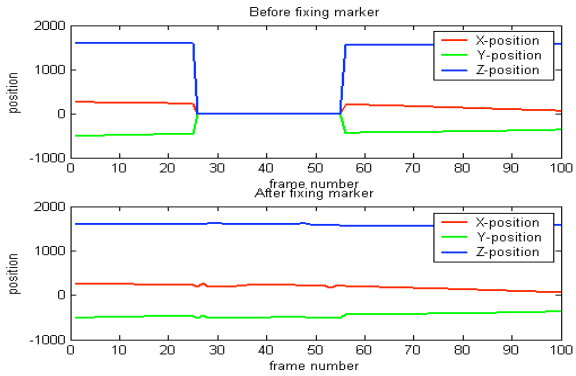


Figure 3: Data Cleaning Results

5.3 Gesture Feature Extraction

Once the data is cleaned, the joint angles between adjacent body parts and the torso orientation at the static part of the gesture (the core pose of the gesture) are extracted as features to represent the different gestures. The local coordinate system with respect to each body part is constructed and the joint angles between adjacent body parts are computed.

The rotation about the body's longitudinal axis is ignored since the gestures with only different facing direction are considered the same. Given head and torso markers, it is straightforward to estimate the rotation angles between the two related local coordinate systems. The joints between the upper arms and torso have 3 DOF. However, we have found that the estimation of the 3 joint angles is sensitive to marker position noise. To tackle this problem, we use 4 angles to represent this 3

DOF joint. For example, for the left forearm, let V_{au} be the vector from left shoulder marker to left elbow marker and V_{al} from left elbow marker to left wrist marker. The directions of V_{au} and V_{al} in torso local system can be represented by 2 angles, respectively. The use of an additional angle increases the stability of the estimation. The same method is used for the hip angle computation. Hence $X=\{X_1, X_2,\ldots,X_{10}\}$ is constructed as the feature vector to represent each pose, and X_i's are joint angles vectors with different dimensions of 2, 3 or 4, depending on the particular joint.

5.4 Training the system

For the needs of this project, Bill T Jones performed the beginning of the piece a number of times while being motion captured.

The data for training the system was generated through eight captures of the sequence of 21 gestures. Four of those capture sessions were performed by the creator of the work, Bill T Jones, during one of his residencies at AME for the motion^e project. The other 4 were performed by AME faculty Jodi James. We used 3 trials of each to train the system.

The same gesture will be executed differently by different people or the same person during different trials. To capture this execution variability, a multivariate Gaussian random vector is used to model the statistical distributions of the joint angles for each gesture, which is given by

$$p(X \mid C_i) = \prod_{k=1}^{10} \frac{1}{\left|\Sigma_i^{(k)}\right|^{\frac{1}{2}} (2\pi)^{\frac{d_k}{2}}} e^{-\frac{1}{2}(X_k-\mu_i^{(k)})^T \Sigma_i^{(k)-1} (X_k-\mu_i^{(k)})}$$

where i=1,2,…,C is the gesture index and *C* is the total number of gestures and d_k is the dimensional of feature vector X_k. *k=1,2,…,10* is the body part index. We assume that the joint angle distributions of different body parts are independent. In the training procedure, $(\mu_i^{(k)}, \Sigma_i^{(k)})$ is related to each body part of all the gestures to be computed.

5.5 Gesture Matching

Gesture matching is based on the pose portion of each gesture and is cast as a multiple hypothesis testing problem. We calculate the joint angle feature vector of each frame as each frame is treated as a possible unknown pose. This decision was made because, as mentioned earlier, the development section includes moments where the dancer moves very fast through the pose section of a gesture without actually stopping. However, in the formation of the queries we allowed for dynamic setting of the threshold for gesture recognition (see section 5.6). Given the joint angle feature vector of an unknown pose (or of each frame) there are two steps to decide whether the pose represented in the frame might be part of the core pose of one of the gestures: 1) determine whether this pose is inside the gesture space 2) if it is, the likelihoods of the feature vector given different gestures are computed using the mean and variance variables from the training process. To accomplish the first step, a body part distance matrix ***D*** is constructed. ***D*** is a *C*-by-*10* matrix. *C* being the number of gestures and *10* the number of considered body parts. Its component $D_{i,k}$ is the *Mahalanobis* distance computed using the related parameters by

$$D_{i,k} = (X_k - \mu_i^{(k)})^T \Sigma_i^{(k)-1} (X_k - \mu_i^{(k)})$$

Then, the validity matrix ***V*** is computed using ***D*** and $d^i_{\max,k}$, which are the maximum distances, pre-computed using the training data. $V_{i,k}$ is computed by

$$V_{i,k} = \begin{cases} 1 & D_{i,k} \le d^i_{\max,k} \\ 0 & D_{i,k} > d^i_{\max,k} \end{cases}$$

$$I_i = \sum_{k=1}^{10} w_k V_{i,k}$$

I_i indicates the validity of this unknown pose: if any of the I_i*'s* is larger than a threshold, this pose is considered inside the gesture space. w_k's are the weights to emphasize body parts differently. In this case, these weights were determined by the limb predominance in the work *21*. If more than one I_i is larger than the threshold, the corresponding likelihood can be computed. Assuming uniform prior distribution of the gestures and common cost function, it is well known that choosing the hypothesis with the largest likelihood gives the Bayesian detector.

5.6 Queries Derived from Single Subject Gesture Recognition

The interdisciplinary team developed five queries that used the gesture recognition engine to extract structural movement features of semantic significance. Since Bill T Jones had already described to the team some of the structural and semantic procedures he used to create the piece it was easier for the team to make context aware decisions regarding what significant features needed to be extracted and how best to structure the queries that could extract them.

Here is a short description of each of the queries. As we discussed in the previous section, results are streamed for each individual frame.

1) *Gesture recognition*: a number between 1 and 21 is given based on the gesture each particular frame matches best. If a frame does not go above the set threshold (7

angles) a zero is given. If the same gesture number is given for X continuous frames (X being a dynamically controlled threshold) then the system assumes that the gesture corresponding to the number showing is being recognized. The dynamic threshold being used is typically between 10-35 frames (83-291 ms.). The reason for the dynamic threshold is to allow the system to make context aware decisions based on the amount of activity or the results of other queries. For instance, in an improvised section the gestures may be done more quickly without stopping in a pose. Setting the threshold lower for these sections can aid in recognizing when one of the 21 gestures is being referred to.

2) *Ambiguity/entropy*: An array of 21 floats indicates the probability that the frame in question represents each of the 21 gestures. The more the probabilities are evenly spread across gestures the bigger the ambiguity and entropy.

3) *Gesture mixing*: For instance, it may be possible that the upper body is related to one gesture and the lower body is related to another. So for the current frame we find the 10 angles corresponding to each body part. These are compared to the maximum angle distances for each gesture in the training data. If 4 to 6 angles belong to one gesture and the others belong to another gesture, we will say it is a mixture of the two gestures, also indicating what gestures contribute to this mixture and what body parts are contributing to this determination.

4) *Surprise probability*: Through this query we are attempting to see whether one gesture is approached and then a quick jump to another gesture occurs. Once a gesture $\boldsymbol{x}$ is recognized we assume it takes time $\boldsymbol{t}$ to approach that gesture. $\boldsymbol{t}$ is set dynamically depending on the speed of the section (i.e. in a faster it will take less time to approach a gesture). If within $\boldsymbol{t}$ we find candidate frames for a gesture other than the gesture just recognized (gesture $\boldsymbol{x}$) we assume that the gesture of the candidate frame was being prepared with a quick jump to the recognized gesture **x**. The definition of the candidate frame is that there are 6 or 7 angles in this frame contributing to a pose. The probability of surprise is relative to the number of continuous candidate frames found within $\boldsymbol{t}$.

5) I*mprovisatory/non-established vocabulary*: This query is a determination of whether the performer has moved outside of the established vocabulary. If the engine determines that the current frame (z) is not a specific gesture from the vocabulary, it will search back through a time $\boldsymbol{t}$. In this time $\boldsymbol{t}$, we will find the last frame which belongs to a gesture ($\boldsymbol{y}$), which means at least 7 angles contribute to a gesture in this frame. The probability is equal to $(time_z - time_y)/t$. If no such frame is found, the probability is 1.0 meaning the performer has moved outside of the vocabulary

5.7 Experimental Results

In testing our system we achieved high results of accuracy for identification of gestures (query 1) with false positives occurring less than one percent of the time. Figure 4 gives the associated recognition rates, which is the ratio of the correctly recognized frame number and the total frame numbers.

Test Sequence Num	1	2	3	4
Dancer 1	94.7	100	100	100
Dancer 2	100	100	100	99.7

Figure 4: Gesture Recognition Rate (%)

A video recording of the gesture recognition being performed in real-time can be found at:
http://ame2.asu.edu/projects/motione/gesture_rec.mov

This demonstration was performed using a gesture recognition threshold (see Section 5.6) of 10 frames.

Since all five queries are based on accuracy of gesture recognition the robust functioning of the gesture recognition engine guarantees the validity of the results of the other queries and allows us to move on to using the results of the queries to create a semantically coherent multimedia work with structural correlations between movement, sound and image.

6. Extensions of Gesture Recognition Engine for Multiple Subjects

These gesture recognition algorithms were extended so they could be applied to recognizing and classifying the two different but simultaneously performed vocabularies of the two phrases (Phrase A and Phrase B) by Trisha Brown as described in Section 3. The phrases were simultaneously performed, in different combinations, by multiple dancers (up to four dancers in our experiments) and the system was asked to determine whether each participating dancer was using vocabulary from Phrase A, Phrase B or neither.

6.1 Differences from Single Subject Engine

As described in Section 3, when capturing four dancers we decided to use a smaller marker set to receive better performance from the motion capture system. However, this had the adverse effect of no longer allowing us to use the same data cleaning algorithm (Section 5.2) since there were not enough markers on each limb segment to do the requisite calculations.

Figure 5 shows the modifications we made to the engine to work in this new environment. As can be seen, we are not using any data cleaning algorithms (although some are in development) and decided to test the system with just the data being sent from the Vicon system. Also, the end result of this engine is not the identification of specific gestures but rather whether the material being performed by each dancer is from vocabulary from Phrase A, B, or is outside of both.

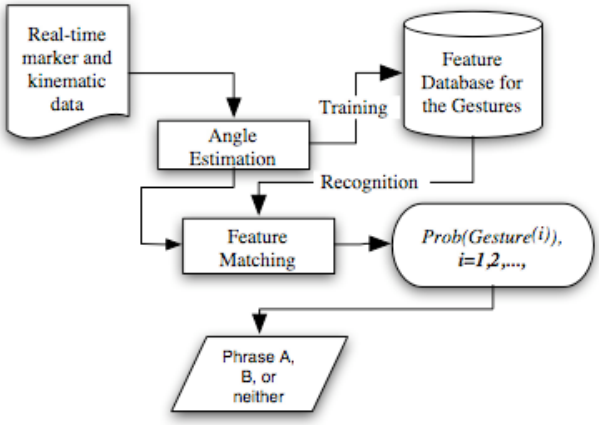


Figure 5:Multiple Subject Engine

6.2 Training the system

To solve this vocabulary classification problem, we first manually selected two different sets of feature poses from both vocabularies. The joint angles related to feature poses are also used as a feature vector to represent each feature pose. During the training session, a number of frames corresponding to each feature pose were used to extract the related joint angles. Furthermore, a multivariate Gaussian random vector was used to capture the variation of execution of poses among different dancers, or the same dancer at different time. The mean and covariance matrices for each pose were also estimated during the training session.

To combat false recognition error, a smoother is applied to remove sudden inconsistent changes in vocabulary recognition. Only consistent changes between vocabularies over a certain period of time will be accepted as true vocabulary transitions.

6.3 Experimental Results

We were able to maintain quite high accuracy while capturing as many as four dancers in real-time when each individual dancer was shifting between vocabulary A and B. However, when it came to identifying when a dancer was performing movement outside of the two vocabularies our results were much less accurate. We encountered errors in which a dancer who was performing outside either movement vocabulary was found by the system to still be in one of the two predefined vocabularies. Through further investigation we believe that the problem is threefold: 1) We found that the improvised sections sometimes contained features from the two main vocabularies and this triggered false positives 2) we found that having a static analysis window for determining vocabulary classification was too rigid and we are looking at ways to make this window size context dependent and dynamic 3) because our initial gesture recognition engine is based on poses, when dealing with a transient vocabulary this engine is not appropriate.

7. Real-time Architecture

The gesture recognition engine is implemented using Visual C++ in the .net environment on a Pentium 4 PC with 2.4GHz CPU. Every frame of motion capture data is analyzed with very low latency. The results of the gesture recognition engine are then streamed using a UDP multicast protocol.

The visual feedback system was written in C using OpenGL, in the Xcode development environment, running on a Apple PowerMac G5. The visual feedback system has access to both the real-time stream from Vicon's Tarsus server over TCP/IP, as well as receiving the movement analysis stream and communication from the audio engine containing audio features.

For the audio engine, two new objects were written for the Max/MSP environment. The first receives the real-time stream from the motion capture system and the second can send and receive data over UDP multicast to both parse the analysis stream for control of interactive audio and to communicate with the visual feedback engine (Figure 6).

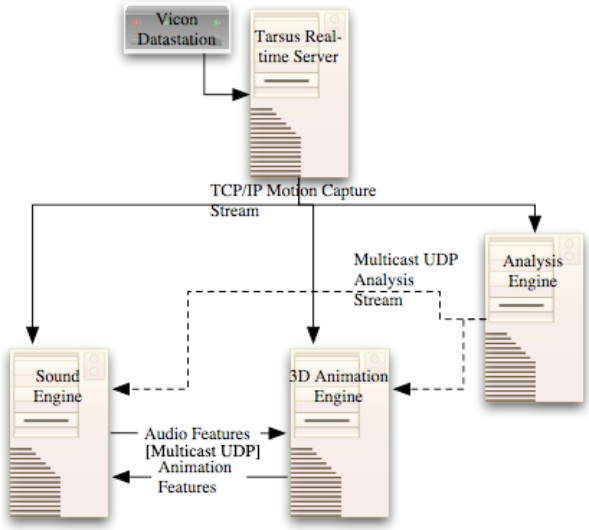


Figure 6: Multimedia Architecture

8. Creation of Structurally Coherent, Interactive Multimedia Works

The authors and the motion^e guest visual artists (Paul Kaiser, Shelley Eskhar and Marc Downie), and guest composer (Roger Reynolds) are using the gesture recognition system described above to develop a new multimedia piece based on new choreography of Bill T Jones. We are also developing the necessary interfaces, procedures, forms and correlations to achieve structural interactivity and integration of elements. However, since this is a work in progress, scheduled to be premiered in April 2005 in Galvin Playhouse at Arizona State University along with the other pieces resulting from the motion^e project, it cannot be presented here. However, a proof of concept demonstration, based on ideas of the entire team, was developed by the authors and AME animation faculty Loren Olsen for the 2nd annual meeting of the National Art and Technology Network (NATN) hosted by AME in November 2003.

Although the 21 gestures by Bill T Jones are used in the demo piece to drive (and demonstrate) the gesture recognition system and although some of the ideas of the demo piece are related to discussions about *21* and the new piece it must be mentioned that the demo piece is not related to the piece *21* nor to the piece being developed by Bill T Jones and the other project participants for 2005. The demo piece simply aims to show some of the feature extraction and structural interactivity possibilities being explored by the project.

The complete interactive, multimedia demonstration is recorded in:
http://ame2.asu.edu/faculty/todd/video/proof_of_concept.mov.

In the next section, we will summarize the interactivities between movement and sound included in the demo piece.

8.1 Multimedia Interactivity

For the purposes of our proof of concept work, we decided on a simple A-B-A' pattern that could demonstrate the detection of different features by the five queries described in section 5.6 and some different possibilities of structural correlations between movement, sound and animation that could be driven by the results of these queries. The demo piece begins with the exposition of the 21 gestures (section A). Section B of the piece was mostly improvisatory movement with brief punctuations created through short returns of one of the 21 gestures. In section A' the dancer returned to predominantly using the recognizable 21 gesture vocabulary.

Sound interactivity: In the first section of the work, only the recognition of the 21 gestures had any affect on the sound. Two different granulated sound sources where mixed through and FFT based spectral attenuation. Each recognized gesture revealed a specific frequency range of the second sound source while at the same time decreasing the spectral energy in the first sound source thereby producing an unmasking of the second source. Some of the symbolism of this choice was driven by one of the ideas underlying Bill T Jones 21. Specifically, self-exploration in the wider context of the exploration of one's surrounding culture (like the exploration of personal movement in the context of poses/gestures that have played a significant role in our cultural tradition).

In section B, the sound was produced by various additive synthesis methods with pitch material being selected by a simple genetic algorithm. As time increased between a return of a recognized gesture the GA evolved through a fitness function that selected for increasing dissonance based on Parncutt's roughness model (1989). Also, as the time interval since a previous recognized gesture increased an expansion in movement of sound sources also resulted though changing the avoidance radius of a simple flocking algorithm that was mapped to the 5.1 surround setup.

The start of section A' was signalled by a return to the recognized gesture vocabulary. The audio functioned in much the same manner as in the first section A except that this time the frequency ranges of the first sound source was unmasked by each recognized gesture.

Visual interactivity: The visual feedback system plays animations from previously recorded motion captures, driven by the recognition system. Multiple animation sequences were associated with each gesture. The choreographer also characterized each of the dancer's gestures for the purpose of associating different color schemes for the animations triggered by the gesture.

The system achieves good real time performance, with frame rates varying widely, depending on how many animations are playing at once, and how much screen space is covered. With several of the gesture animations playing, the frame rate exceeds 120 frames per second. When several of the large out of vocabulary animations fill the entire screen, the frame rate can drop to 30 fps. Typical frame rates during the performance are 70-80 fps. These timings are on an Apple PowerMac dual processor, 2Ghz G5, ATI 9800 Pro graphics card. The display screen running at 1024x768, double buffered, with multisample anti-aliasing turned on.

9. Future Work

The work presented in this paper is just the first stages of the creation of a fuller gesture recognition system and multimodal architecture for the development of multimedia performance works.

In 2005, the faculty of AME will be premiering a piece choreographed by Jodi James, with real-time music by Todd Ingalls and live 3D animation by Loren Olson. This piece will be an outgrowth of the discoveries we make in gesture recognition and connecting movement, sound and image at various structural levels. The concept for the piece is based on the idea of weightlessness and flight.

Acknowledgments

We would like to acknowledge all our colleagues and students that are participating in the motion^{e} project and making this work possible: Bill T Jones, Bebe Miller, Trisha Brown, Roger Reynolds, Paul Kaiser, Shelley Eskhar, Marc Downie, Janet Wong, Carolyn Lucas, AME Faculty Loren Olson and Frances McMahon Ward, AME graduate student Yurika Abe, Department of Dance Faculty Karen Schupp, and AME Manager of Projects and Programming Sheilah Britton.

The motion^{e} project is supported in part by The National Endowment for the Arts and ASU Public Events

References

A. Camurri, S. Hashimoto, M. Ricchetti, A. Ricci, K. Suzuki, R. Trocca, and G. Volpe. EyesWeb: Toward Gesture and Affect Recognition in Interactive Dance and Music Systems. *Computer Music Journal*. 24(1), 57-69. 2000.

C-L Moore, K. Yamamoto. *Beyond Words: Movement Observation and Analysis*. Gordon and Breach Science Publishers, New York. 1988.

C. Dobrian, F. Bevilacqua. Gestural Control of Music Using the Vicon 8 Motion Capture System. P*roceedings of the 2003 Conference on New Interfaces for Musical Expression*, Montreal, Canada. 2003.

R. Parncutt. *Harmony: A Psychoacoustical Approach*. Berlin: Springer-Verlag. 1989

Object Design Considerations for Tangible Musical Interfaces

Martin Kaltenbrunner
[*]Music Technology Group
Audiovisual Institute - Universitat Pompeu Fabra
Ocata 1, 08003 Barcelona, Spain
mkalten@iua.upf.es

Sile O'Modhrain
[†]Palpable Machines Group
Media Lab Europe
Sugar House Lane, Dublin 8, Ireland
sile@media.mit.edu

Enrico Costanza
[‡]Liminal Devices Group
Media Lab Europe
Sugar House Lane, Dublin 8, Ireland
enrico@mle.ie

Abstract

In this paper we describe object design considerations for the reacTable* project, a novel tangible musical instrument, developed at the Audiovisual Institute at the Universiat Pompeu Fabra. The work presented in this paper is the result of a collaboration with the Palpable Machines Group at Media Lab Europe, which focussed on haptic design aspects of the reacTable* instrument. We present a simple haptic encoding scheme for the mapping of abstract sound synthesis objects onto tangible proxy objects.

1 Introduction

The reacTable* is an electro-acoustic musical instrument in the tradition of Jordà's FMOL synthesizer (Jordà, 2002). The aim is to create a tangible electronic musical instrument that allows expressive collaborative live performances for professional musicians without the limits of many screen-based interfaces for electronic music. Many of these interfaces have very limited control possibilities and provide little feedback on the creative process for both the performer and the audience. As suggested by its name, the reacTable* is a table-based instrument, allowing direct manipulation of any object in the synthesis chain. By arranging a set of objects that are available on the table surface, the performer constructs and plays the instrument at the same time. Each of the objects has a dedicated function for the generation, modification or control of sound flow, and reacts with compatible objects near it. While the table itself is equipped with sensors for the identification and tracking of the objects' position and state, the performers do not need to wear any controller devices or sensors. In addition to the sound which is obviously produced while playing, the reacTable* also provides visual feedback by projecting a graphical representation of the sound and control flow onto the table surface. In order to create a truly multi-modal interface experience particular effort has been spent on the haptic design of the object and table properties. This paper reflects the current state of the instrument, which still differs in various aspects from the final design; especially its size will be significantly bigger than the current prototype. For a more detailed description of the original reacTable* concept see (Jordà, 2003).

2 Instrument Components

During the initial project phase we have been developing the basic reacTable* concepts within a software prototype only, simulating the tangible user interface component with a graphical interface. This approach allowed the rapid prototyping and the introduction of new synthesizer and interaction elements without worrying about sensor and hardware problems. At a second stage we added the d-touch computer vision framework (Costanza et al., 2003a), which allowed the construction of a first stage tangible prototype including a set of reacTable* objects.

The current system was implemented in a completely modular way, allowing the easy reuse or replacement of the five basic functional components. A sensor module tracks the state, position, and orientation of any object that is present on the table. These raw sensor parameters are passed to the central management component, which interprets the user gestures based on the incoming data, generating a dynamic patch network that drives the two actual synthesis components for the sonic and graphical feedback. The synthesis engine is implemented using the open-source PD language (Puckette, 1996). We are currently also integrating a graphics projection system into the prototype, and informal tests show that visual feedback will be crucial for the usability of the instrument. All these components are completely independent and are communicating via a simple proprietary network protocol, which we are considering upgrading to OpenSound Control (Wright et al., 2003) compatibility if necessary. This separation allows execution on various hardware platforms avoiding possible performance bot-

tlenecks since each of these modules requires significant computational resources. In this paper though, we will focus on the tangible controller, which is comprised of a transparent Perspex panel and a set of hand crafted objects, which will be discussed in detail below.

3 Synthesis Object Types

The reacTable* objects can be generally categorized into seven different functional groups: Generators, Audio Filters, Controllers, Control Filters, Mixers, Clock synchronizers and Containers. There are also some exceptions that do not fit within any of these categories.

- *Generators* are sound sources that can produce various types of synthesized or sample based sound. They have an audio output and various control inputs. We are currently considering adding a sound input port to generator objects as well in order to allow FM synthesis.
- *Audio Filters* can modify incoming sound based on their internal algorithms, which can range from a simple band-pass filter to any possible sound effect. Filters have generally one or two sound inputs and a sound output as well as several inputs for control.

Control inputs permit the constant modification of the object parameters that can be controlled either by changing the spatial object properties (e.g. position, orientation, distance to the next object, angle to the next object, distance to the center, angle to the center, etc.), in some cases even its morphological properties (e.g. bending, shape), or by connecting control data flows to their control inputs. These data flows are generated by a third object type, the Controllers.

- *Controllers* generally produce their control data by algorithmic generation which can include from simple low frequency oscillators to complex chaotic or fractal generators. Like in any other object, their respective parameters (e.g. frequency and range in a low frequency oscillator) depend also on the spatial properties of the object, and can be permanently modified. Controllers do not yet have inputs but we plan to implement this feature soon. With some exceptions, controller output is generally adimensional, which means that the effect of a controller depends on the control input it connects to. *Control Filters* process control data. They have a control input and a control output, and unlike regular controllers, their output can sometimes be dimensional; the output values of a harmonizer or a chord generator, for example, are always mapped to pitch.
- The *Mixer* object can take various sound streams as an input and produces a single output stream. Inverted Mixers (*Splitters*) can split a single sound into multiple output streams.
- *Clock synchronisers* introduce a higher hierarchy; they can influence several objects in their proximity at once and in several ways, like sending them synchronised triggers or correcting their low frequencies in order to match a given pulsation. Clock synchronisers have one fundamental parameter, tempo (they also have tempo subdivision), which can be modified by repeatedly hitting the object several times.
- High-level *Container Objects* can virtually contain any pre-built set of sub-patches, allowing the construction of more complex sound structures.

The objects do not need to be connected explicitly: a set of basic connection rules automatically connects compatible objects in respect to their activation, distance or availability. This of course does not exclude the possibility of an explicit connection gesture. See (Kaltenbrunner et al., 2004) for a more detailed description of the Dynamic Patching concept.

4 Object Handling

The objects available on the table can be manipulated by the players in various ways, when placed on the table, an object is identified and activated, moving it on the table surface, its position is tracked as well as its rotation angle. Based on this position and orientation data, inter-object relations such as relative distance and angles are calculated.

Most reacTable* objects are *plain* and *passive*, meaning that they do not come with any cables, switches, buttons whatsoever. The user also does not have to wear special sensors or controller equipment for the object handling: plain hands are the only necessary controller. This, of course, does not rule out the possibility of smart objects that incorporate additional internal electronics in order to retrieve some additional sensor data coming from squeezing, bending or bouncing them, like in the case of the Squeezables (Weinberg and Gan, 2002). In any case, this has to be achieved in a completely transparent way, using wireless technology for example, so that the performer can treat all objects in an equal way. A simple rubber hose is an example suggesting some of these additional control possibilities, whose state could be either determined by the computer vision or by using some bending sensors like in the Sonic Banana (Singer, 2003), can serve as a bending controller producing multi-dimensional control data.

More than manipulating the table objects, the hands can be considered to be reacTable* objects themselves, acting as a kind of meta-controller. Tracking of the hands' position and state allows the recognition of various natural hand gestures, such as pointing, painting, waving, etc. Wavetable objects, for example, allow the painting of a waveform next to them, while a simple karate style gesture on a sound flow will result in muting this connection.

5 Tangible Object Types

As already stated above, the reacTable* objects are *plain* and *passive* objects, meaning that they generally do not come with any embedded electronics. This implies that we do not have access to any *active* or computer controlled haptic feedback (vibration, force feedback, etc.), and therefore we can only provide *passive* haptic feedback as defined by the physical object properties only.

The reacTable* objects act as physical and tangible representation of the various virtual synthesis components. They are proxy objects, or *phycons* (Ishii and Ullmer, 1997), which allow the direct manipulation of any of these synthesizer components as required by the performer. Since most synthesizer objects are of rather abstract nature, we decided to reflect this in a more abstract object design as well. Complex everyday objects are used, but have some special functions as discussed below.

We have considered the various haptic dimensions such as shape, size, and material (including texture, weight, density, temperature) to create a suitable haptic encoding scheme for the various abstract object types and their variations, in order to allow rapid and accurate object identification by simply grasping them with the hand. We have been especially concentrating on haptic properties of the object's top surface, but this was mainly due to the lack of available material variations.

5.1 Haptic Encoding

Shape defines the various generic object types, such as generators (square), processors (circle), controllers (star) and mixers (triangle). Simple shapes are easily accessible both visually and haptically, and provide a suitable encoding for the abstract object types. Color would meet similar requirements but is only accessible in the visual domain. Simple geometric shapes can be identified quite easily with a grasp or hand enclosure. More complex shapes would require time consuming contour following with the hand (Lederman et al., 1996) and cannot always be identified completely. Therefore we only defined a small set of easily distinguishable geometric shapes.

Size was not chosen as an encoding dimension, because, in traditional instruments, size often correlates to pitch (tuba – trumpet). Nevertheless we evaluated three different sizes: 4,6, and 9 cm diameter, which can be held and manipulated with three, four, or five fingers, at least by an average adult player. Both 2D (flat) and 3D (cubic) objects were constructed, although this feature is not used for encoding. We are using a wooden cube as a sample player; for example, where each of the six sides represents a different sound sample.

Surface texture was chosen for encoding of the object subsets. We are using two methods to create haptic surface cues. The first is laser engraving onto plastic surfaces to encode abstract haptic feedback, while attaching various materials such as felt or sanding paper onto the objects top surface can represent certain timbral properties of the sounding object. A simple clean sine wave, for example, can be represented with a clean surface; whereas a saw-tooth generator would come with a rough surface. Noise generators have a completely irregular texture and different types of sanding paper can represent a granular synthesizer. Further formal testing will evaluate the correct mapping between the perceived surface and the sonic experience provided by the corresponding synthesis object.

Material We are using both natural and synthetic material with different weight, density, thermal, and texture properties. For each functional object, we are trying to choose a material which haptically represents a close match to the sonic properties of the virtual sounding object. For synthetic sounds, for example, we choose synthetic materials, such as plastic. A sound sampler therefore, is best represented using organic materials such as wood. This early symbolic mapping needs to be evualated in later testing.

Some examples: A sine-wave oscillator is a synthetic sound source with a smooth sonic appearance. According to our haptic encoding scheme, this can be represented by a plastic square with a smooth surface. A simple band-pass filter therefore results in a round plastic disk with a deep engraving through its centre. One of the sound effect filters was constructed by attaching felt on top of a round plastic disk. Furthermore, a wooden cube would be a sample source, while a cube made of a synthetic grainy material represents a granular synthesizer. This scheme was used to encode current reacTable* objects in the most meaningful way to the authors. In the future informal subject test will refine these mappings. Figure 1 shows the first set of reacTable* objects as used in the current tangible prototype.

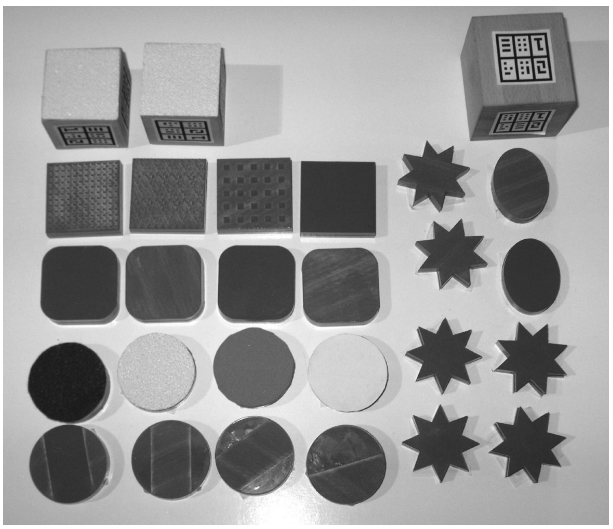


Figure 1: Some reacTable* objects

5.2 Everyday Objects

Ready made *everyday objects* are considered rather for the symbolic meaning and mechanical properties rather than matching them into the haptic encoding scheme. Within the reacTable*, these objects basically have three different functions:

- *Containers* are known and tagged objects that are part of the provided object set, and can be used as sub-patch containers. Due to their highly symbolic meaning, sub-patch containers should be easily identified and remembered by the player. They can include any possible everyday object such as coffee mugs, chocolate bars or rubber ducks.

- *Super-Controllers* Ready made toys such as a (flexible) wooden snake, can be introduced as a multi-dimensional super controller. This of course requires the previous programming of the behavior of such an object, as well as the mapping of the various control parameters. The object state should be tracked completely by computer vision without any changes to the object itself. Only in special cases invisible and wireless sensor technology should be added.

- *Visitor objects* In the context of a public installation one can expect visitors will place their own objects onto the table surface expecting them as well to interact with the intrument. Since one can anticipate somehow what visitors will carry (mobile phones, keys, glasses,) these objects should be identified and integrated into the table: e.g. a mobile phone starts to play an annoying melody, or keys a rattling sound.

5.3 Further haptic design considerations

Haptic orientation cues The table edges are marked with a simple tape, which provides a haptic cue for the table dimensions, because, moving an object over this edge can be felt easily. The same principle was used to mark the table centre by applying a symbol made of transparent tape. This is both haptically and visually accessable, but does not interfere with the computer vision sensor. The localization of the table centre is important for the overall dynamic patch system.

Magnetic objects We also have been experimenting with magnets in order to provide a simple connection or compatibility cue. This idea produces a nice haptic effect, but is unfortunately not very flexible. Ordinary magnets can produce three object classes: positive, negative and neutral. This problem could be overcome with electromagnets that can be switched on or off, and even can change their strength, but this would require significant electrical power, which is not likely to meet our requirements for *plain object* design.

6 Design Constraints

The current prototype is based on computer vision. This has the advantage of simplicity and low cost, requiring only an off-the-shelf USB web-cam. The d-touch framework is based on the localization and recognition of fiducial markers, namely black & white graphical symbols that can be printed on labels and simply attached onto the objects. This system is quite robust thanks to the concurrent design of the markers and detection algorithm (Costanza and Robinson, 2003). The obvious downside of this approach is the need for tagging the objects with visible labels, which is partly overcome by attaching the markers onto the object's bottom side. The choice to place the camera below the transparent table also prevents occlusion of the objects by the player's hand and body during the performance.

The label size, and thus the object size is constrained by the system resolution. This depends not only on the optical resolution of the camera, but more significantly, on the available processing power. The lower the image resolution, the bigger the objects have to be for correct recognition. However, the image processing algorithm's computational cost has been observed to be approximately linear with the number of pixels. In fact, increasing the image resolution over 640 by 480 pixels would result in an unacceptable temporal resolution, which is around 7Hz on our test system based on a 1 GHz Intel Pentium III processor. As discussed in (Costanza and Robinson, 2003), the marker size is also related to the maximum number of different objects supported by the system. We are using a marker set of 120 different symbols, which is currently sufficient, but could be easily exceeded by a larger collection of objects, although it is unlikely that these would be used within a single session.

Currently the label size is around 3 by 3 cm on an interaction surface of around A3. This is acceptable compared to the desired object's size, based on our ergonomic considerations. Additionally, the topological approach used for the recognition (Costanza and Robinson, 2003) allows the design of labels of different shapes allowing more object and symbol variations, such as circles.

Computer vision generally has some considerable performance limitations, such as visual and temporal resolution, as well as some side effects that are degrading recognition performance, such as poor lighting or motion blur. Even scratches on the table surface or dirt on the symbol markers affect the performance significantly.

We are considering the option of employing a hybrid system for a later version of our instrument. RFID tags could be used for the identification and tracking of the reacTable* objects, while computer vision would be utilized for hand gesture recognition and for tracking objects introduced by the player without previous tagging. This should allow faster, more robust, and computationally more efficient object tracking, at a much higher system cost of course.

7 Observations on related tangible musical interfaces

Audio d-touch (Costanza et al., 2003b) is a collection of three tangible interfaces for music composition and performance: the *Augmented Musical Stave*, the *Tangible Drum Machine* and the *Physical Sequencer*. Like the current implementation of reacTable, it is based on the d-touch framework. In the layout chosen for audio d-touch a web cam observes the interactive surface from above so the fiducial labels are clearly visible to the user. This approach suffers from the occlusion problems mentioned above, but permits a simpler system setup. Audio d-touch was conceived as a desktop instrument that can be used on any table; for example, in a house or a school. By arranging the interactive objects on the interactive surface the user can play notes and understand the musical score notation, create drum beats, or record and arrange audio samples in a loop. The interactive area is covered with a printed piece of paper where visual cues give hints about the mapping between the block position and the sound generation parameters.

The design of the interactive objects' shapes has been mainly driven by ease of construction, leading to the use of simple rectangular blocks. These blocks are marked with machine-readable fiducial symbols as well as human-readable cues related to the object function. The musical notes used in the augmented stave have obvious meaning. The tangible drum machine blocks are as small as the system resolution allows them to be: in this case there are only two types of blocks (loud and quiet), so they are differentiated by the color of the sides.

Clearly, the cues currently used are merely visual. Several possibilities to improve the simple block design and make them distinguishable by touch are under consideration. For example, the block's physical size can be related to the note length or to the drum sample volume. In the sequencer application, different block types can be associated to different geometrical shapes. Functional areas on the interactive surface can be carved with different tactile textures.

The Audiopad (Patten et al., 2002) was primarily designed as a tangible instrument controller, the physical objects are mainly used to control a projected graphical user interface. Therefore, the objects, in this case mainly circular pucks, have the basic function of knobs like in a standard MIDI interface, mimicking their tactile and visual appearance. An additional object, the *Selector*, is shaped in a different functional way, which adds directional cues to make it easier to point to the desired selection areas. Both object types have a simple push button on their top side, which allows the triggering of certain actions associated to each object. The Audiopad is using two RF tags for each object to track position and rotation. Due to physical limitations the current system can only track up to nine different objects.

The Music Table (Berry et al., 2003) uses the AR Toolkit (Kato and Billinghurst, 1999) computer vision engine, and reduces the tangible object design to a minimum by attaching the necessary symbols for the vision system onto simple cards. These card symbols are readable both by the user and the computer vision system. Rather than crafting physical objects, the Music Table places virtual 3D objects onto the card surface; a common augmented reality technique. While the physical table contains the set of tangible proxy objects, the player is actually controlling a screen based instrument representation. The system defines an interesting set of musical objects, and also tries to overcome the object-container problem as discussed in (Kaltenbrunner et al., 2004), by defining a manipulation card for virtual objects.

The Musical Trinkets (Paradiso and Hsiao, 1999) are a collection of tiny plastic toys equipped with wireless magnetic ID tags. These objects are pre-loaded (by mapping sounds to their ID) with a certain musical behavior, "such as bird calls, shakers and percussive things" which is activated when an object is placed or moved towards a reader device. Distance to the sensor and speed of movement control the object's sound. Other objects are modifiers, such as pitch-shifters or sound effects including vibrato. The Musical Trinkets also generate visual feedback, which is projected onto the instrument's surface.

BlockJam (Newton-Dunn et al., 2003) uses, unlike the previously listed instruments, a set of sophisticated synthesis objects, which in this case aren't simply proxies for virtual processing elements, but do actually carry the necessary circuits for sound processing within. Basically, they are square boxes with simple plugs on the edges, which allow the assembling of physical sound processing patches. The boxes also come with a small LED display array to provide visual feedback on the object's state, and a touch-sensitive controller to program the object's behavior using a dial gesture.

Figure 2: the reacTable* prototype

8 Future Work

In continuation of this work, we are planning to adapt the tangible reacTable* interface as a test platform for a formal evaluation of strategies for object-to-sound mappings in tangible musical instrument interfaces. We are planning to use this platform for the further development and evaluation of our haptic encoding scheme; especially focussing on the tactile surface and material properties and their mapping to sound timbre.

In the near future though we will continue to work on the completion of the sound synthesizer functionality as well as on the integration and refinement of the visual feedback. The final prototype will then also be subject to informal user tests and will be explored within first experimental musical performances. We are also planning to focus on the various aspects of collaborative musical performance.

9 Acknowledgments

The authors would like to thank the following people who have contributed to this work: The reacTable* team: S. Jordà, G. Geiger, and I. Casasnovas at the Pompeu Fabra University. D-touch framework: S. B. Shelley and J. Robinson at the University of York. Media Lab Europe for providing the facilities and materials for object manufacturing, and Sam Inverso for his proof reading. This work was partially supported by a European Commission Cost287-ConGAS action grant for a short term scientific mission.

References

R. Berry, M. Makino, N. Hikawa, and M. Suzuki. The augmented composer project: The music table. In *.Proceedings of the 2003 International Symposium on Mixed and Augmented Reality*, Tokyo, Japan, 2003.

E. Costanza and J. Robinson. A region adjacency tree approach to the detection and design of fiducials. In *Proceedings of Vision, Video and Graphics*, Bath, UK, 2003.

E. Costanza, S. B. Shelley, and J. Robinson. D-touch: A consumer-grade tangible interface module and musical applications. In *Proceedings of Conference on Human-Computer Interaction (HCI03)*, Bath, UK, 2003a.

E. Costanza, S. B. Shelley, and J. Robinson. Introducing audio d-touch: A tangible user interface for music composition and performance. In *Proceedings of the 2003 International Conference on Digital Audio Effects*, London, UK, September 8-11 2003b.

H. Ishii and B. Ullmer. Tangible bits: Towards seamless interfaces between people, bits and atoms. In *Proceedings of CHI 97 Conference on Human Factors in Computing systems*, Atlanta, USA, 22-27 March 1997.

S. Jordà. Fmol: Toward user-friendly, sophisticated new musical instrument. *Computer Music Journal*, 26(3): 23–39, 2002.

S. Jordà. Sonigraphical instruments: From fmol to the reactable*. In *Proceedings of the 3rd Conference on New Instruments for Musical Expression (NIME 03)*, Montreal, Canada, 2003.

M. Kaltenbrunner, G. Geiger, and S. Jordà. Dynamic patches for live musical performance. (submitted to NIME04), 2004.

H. Kato and M. Billinghurst. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *Proceedings of the 2nd International Workshop on Augmented Reality (IWAR 99)*, San Francisco, USA, October 1999.

S. J. Lederman, C. Summers, and R.L. Klatzky. Cognitive salience of haptic object properties: Role of modality encoding bias. *Perception*, 25:983–998, 1996.

H. Newton-Dunn, H. Nakao, and J. Gibson. Block jam: A tangible interface for interactive music. In *Proceedings of the 2003 International Conference on New Interfaces for Musical Expression*, Montreal, Canada, May 22-24 2003.

J. Paradiso and K. Hsiao. A new continuous multimodal musical controller using wireless magnetic tags. In *Proceedings of the 1999 International Computer Music Conference*, pages 24–27, Beijing, China, October 22-28 1999.

J. Patten, B. Recht, and H. Ishii. Audiopad: A tag-based interface for musical performance. In *Proceedings of the 2002 International Conference on New Interfaces for Musical Expression*, Dublin, Ireland, May 24-26 2002.

M. Puckette. Pure data. In *Proceedings of the International Computer Music Conference*, San Francisco, USA, 1996. International Computer Music Association.

E. Singer. Sonic banana: A novel bend-sensor-based midi controller. In *Proceedings of the 3rd Conference on New Instruments for Musical Expression (NIME 03)*, Montreal, Canada, 2003.

G. Weinberg and S. Gan. The squeezables: Toward an expressive and interdependent multi-player musical instrument. *Computer Music Journal*, 25(2):37–45, 2002.

M. Wright, A. Freed, and Momeni A. Opensound control: State of the art 2003. In *Proceedings of the 3rd Conference on New Instruments for Musical Expression (NIME 03)*, Montreal, Canada, 2003.

The Scangloves: a video-music instrument based on Scanned Synthesis

Loic Kessous*, Daniel Arfib*
*CNRS-LMA
31 Ch Joseph Aiguier, 13402 Marseille, France
kessous@lma.cnrs-mrs.fr, arfib@lma.cnrs-mrs.fr

Abstract

This paper describes the design and the use of a glove-based musical and visual instrument. Sensors are used to measure pressure, flexion, and orientation of the hands. This controller is mapped to a scanned synthesis model. The visualisation is based on the waveform display and the envelop following but is also configured with control data. the goal of the mapping strategy is to give an instrument which allow both melodic control and spectral manipulation.

1 Introduction

The possibilities brought by sound effects and synthesis have already changed our way of conceiving musical composition. Today, the composer in many different music currents is accustomed not only to write melodies, rhythms and harmonies[1], but also timbres and spectral evolutions. The performer can play, interpret, and possibly improvise the sound itself. Computer technologies have powerfully encouraged mixing between different arts by providing easiness of use. There are different possibilities to consider the interaction between dance, video and music. Choices are generally made by preference due to background of artist. From video to sound, from sound to video, from gesture or dance to sound and video; many ways can be explored. Alternative controllers find here a place to convey their potentials. They give to the instrument designer the whole freedom needed to fit to a desired expressiveness[2]. Data gloves have yet been used in digital arts in several different ways, their potentials in new multimedia forms of arts are probably still partially unexploited.

2 Description

The "Scangloves" is a Two-handed instrument designed by Loic Kessous. It consist of two different gloves equipped with sensors. Gesture data are used to control the parameters of a scanned synthesis model, by the way of an adequate mapping. The sound produced is used to generate a visual part. This visualization is based on the waveform display and the envelope following, but data from the gloves are also mapped more directly to some parameters of the visualization. Finally, this instrument is both a visual and a musical instrument. Explicit and implicit mapping (Artificial Neural Network) are used to provide both dynamic play and use of symbolic gesture.

[1] or notes, nuances and instrumental techniques like in instrumental contemporary music

[2] here, expressiveness must be understood as a dynamic identity

3 Motivation

3.1 The power of gloves

Gloves can provide a many gesture degrees of freedom. They can also add a semiotic (Cadoz and Wanderley, 2000) potential for gestural control by using symbolic gesture. Scenic and choreographic aspects can also be enhanced due to the movement degrees of freedom induced. Generally, Alternate Instruments can provide the power to express an artistic message to the audience through several sensory channels. An extension of this basic concept could lead to a system where the performer and the audience would be immersed into the kernel of the artistic purpose.

3.2 Previous works

The first version of the Scangloves was only a musical instrument. the software interface was displaying the values of sensor data on sliders, and the names of signs recognised by an Artificial Neural Networks. I started to design the visual part of the Scangloves after a first experiment with visualisation, which is at the origin of this work. It was a collaboration with Serge Ortega who is a visual artist, a programmer and also a musician. He has developed a software for real-time visualisation of sound. The project was to analyze the sound produced with the Scangloves with his software and to find a set of parameters to have a personalised and adapted visualisation. His software provides high level of analysis, formal representation and colour configuration. During test I felled sensa-

Figure 1: The 5DT glove

tions that I have probably never felt before as a performer. In this version, I particularly enjoy the possibility to explain things I want to Serge in a figurative language (the notion of funnel, horizon and vortex was used a lot) and the relative facility to adapt the software, even by code entry. However, the system lacks by his heaviness, It uses one Macintosh for sound and two Windows PC to run. Unfortunately, a regular and intensive collaboration is not possible due our others activities. After this first experience, I began addicted to this concept. Now, I can difficulty imagine this instrument without a visual part. I decided to carry on my experiments with a reduced system running on a Macintosh Ibook G3 800 MHz, by replacing some feature extractions of the sound analysis by control data mapping.

4 Controllers

4.1 Data acquisition

The non-preferred hand uses a 5DT Data glove (FifthDimensionTechnologies, 2004). This glove measures 5 flexions by the way of optic fiber sensors and 2 orientations thanks to inclinometers. The preferred hand uses a home made glove. This glove measures 2 pressures thanks to FSR sensors, and 2 flexions. The Data acquisition is made with an IcubeX (InfusionSystemsLtd., 2004) or with a modified USB gamepad.

4.2 Position of sensors for the preferred hand

Two pressure sensors are positioned on first and second phalanx of the index finger. The thumb is used to act on it. These sensors are used to trig notes and to continuously excite the system but also to define the octave; the upper sensor corresponds to the higher octave and the

Table 1: Gesture Information Processing

Information	Preferred Hand	Non-Preferred
Mechanical (or Physical)	2 pressures, 2 flexions	5 flexions 2 inclinaisons
technology	FSR, 2 piezo-resistors	5 optic fiber inclinometers
functional	excitation, modulation	symbolic, selection, moduation

lower sensor corresponds on the octave below. Two others flex sensors are used to control continuous parameters. The first one is placed on the middle finger and the second one on the little finger. Movement of the 4th finger is too dependent of the others fingers to be used independently. Middle finger and little finger are better candidate because they are relatively more independent.

5 Sound Synthesis

I use the Scansynth$\sim$ Max/MSP externals developed by Jean-Michel Couturier (Couturier, 2002). In scanned synthesis (Verplank et al., 2000), the shape of a simulated mechanic system is scanned at audio frequencies to produce sound. The Scansynth external object generates a circular string (boundary conditions at the end of the string are transferred to the beginning of the string) modeled with finite differences. We can act on the initial shape and on the forces we apply. The synthesis meta-parameters that we have used are global damping, force gain and force extra-parameter [3].

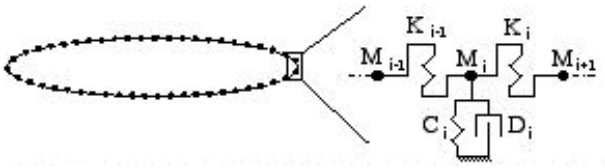


Figure 2: Circular string for scanned synthesis

6 Visual Synthesis

The visualization is based on the waveform display. This kind of display is currently used in visualizer of music player software like iTunes on Macintosh or Media Player on Window. I use a RGB display of the waveform, and

[3] suggested by Max Mathews and implemented by Jean-Michel Couturier in the scansynth$\sim$ external object, this extra-parameter generate higher harmonics by modifying the waveform

effects like zoom and rotation function on this display[4]. The x and y ranges can be configured. The decision to use such a visualisation has been influenced by the level of abstraction that it can convey. I focus my attention on the pertinence of the link between sound and video. An important consideration is also to give way to imagination of the audience, as it would probably not be with a more figurative representation of sound. One must consider this visualisation more as an extended stage lighting effects than as a narrative video part. It could also be considered as a visual feedback for the performer even if it is not the primary goal.

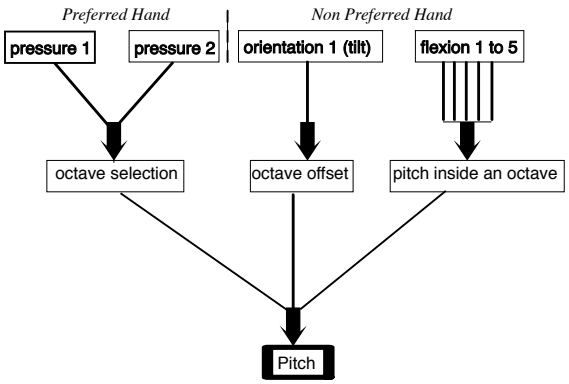


Figure 3: Pitch Mapping

7 Mapping strategies

7.1 Mapping chain

As Hunt et al. (2003) explain it, connecting the input device to the sound source, traditionally inseparable in an acoustic instrument is not trivial. An approach can be to separate the mapping in different layers. In (Arfib et al., 2002) and in (Hunt and Wanderley, 2002), the authors suggest to use three layers in the mapping chain. In (Arfib et al., 2002) we explain this choice by using the concept of perceptual spaces. In this case, using three layers in the mapping chain means: from gesture data to gesture perceptual space, from sound perceptual space to synthesis model parameters, and between the two perceptual spaces. To get a simple mapping between the gesture perceptual subspace and the sound perceptual subspace, we need to focus our attention on the two other mappings.

7.2 Mapping complexity

In Hunt Wanderley experiments, sems to show that a complex mapping can be more effective than a simple one. In the case of a many-to-one, the cooperation of the two hands in a common task according to Guiard's kinematic bimanual model could explain it. Concerning pitch selection and control one can also consider the importance of the octave in pitch perception

7.3 Mapping strategies for sound

7.3.1 Used mapping

For the Non-Preferred hand (5dt Data glove) we use pattern recognition based on Artificial Neural Network to do selection gesture. The 5 flexions from the 5dt data glove are mapped to symbolic signs. A Multi Layer Perceptron (a Max external object [5] (CNMAT, 2004) is used to map flex data from optic fibers of the 5dt glove to symbolic signs. The MLP external is trained to recognize patterns of Mimophony. The mimophony is a gestural code of an empty-hand symbolic sign representing a pitch note. From long time ago polyphonic singers from Corsica (a nice Mediterranean island near the south of France) used it to communicate with each other while improvising. A Contemporary Orchestra named Allegro Barbaro (I was playing guitar in this orchestra) has used it to conduct improvisation-based pieces. Selection of sign is interpreted as selection among the 12 semitones of the chromatic scale and selection of upper or lower pressure sensors in the Preferred hand (home made glove) is interpreted as choosing the octave. The third component of the pitch is the tilt orientation of the 5DT glove; it gives the possibility to play in the two higher octaves. Finally, pitch is mapped to fundamental frequency.

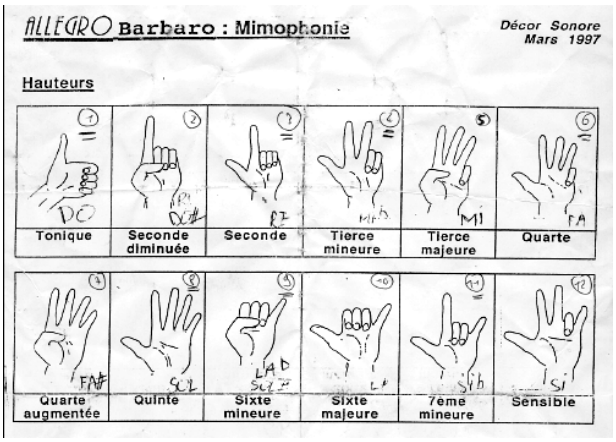


Figure 4: Mimophony; $simplicity = effectiveness$

[4] a patches named max-tunes concerning related works is part of the Max/MSP distribution

[5] This MLP can be trained in a real-time (Max/MSP) context, and patterns can be stored in the Max/MSP context; this can be very useful for other experiments

To detect the velocity of an instantaneous excitation with the pressure sensors, I tried two different explicit mappings. The first one is to measure the time interval between a low threshold and a high threshold. The second one is to extract local maximum of the pressure. These two mappings are globally equivalent but give a strong difference in expressive identity of the instrument. This instantaneous excitation velocity is mapped to a interpolation parameter; this parameter give initial shape which is the result of an interpolation between two different initial shapes, a noisy one and a smooth one. Pressure is also mapped to a continuous excitation equivalent to an AFTERTOUCH message for MIDI keyboard; the pressure is mapped to the force gain meta-parameter. This meta-parameter apply a force profile to the virtual scanned string. For the home made glove, the flexion of the middle finger is mapped to the damping meta-parameter and the flexion of the little finger is mapped to the force extra meta-parameter to increase the brightness using a waveform-based harmonic enhancement. Finally, instantaneous excitation velocity is mapped to initial shape properties and force gain and extra meta-parameters are mapped to low-level force parameters.

7.3.2 Alternative concerning pitch

Pitch can change only when a note is triggered with pressure sensors on the preferred hand or can be modified continuously at each time a new sign is recognized by the Artificial Neural Network. Data obtained can be smoothed or filtered in different ways. One can use a low pass filter to avoid rapid changes that may be undesired. One can do an interpolation between discrete pitch to avoid discontinued change of pitch but one can also use it to induce a rhythmic behaviour. One can also only take account of precise sign to allow the performer to play on a defined mode or write a algorithm to replace a note not included in this mode to another one included in it. This musical filtering can be useful for the beginner but of course add limitations to the play. All this possibilities have been experimented, most of them gives the feeling to have a instrument less dynamic.

7.4 Visual mapping

I use a unidimensional navigation in a colorspace to map pitch to color. The pitch is mapped to the parameter named Hue (corresponding to the x-axis of the figure). The pitch range does not correspond to the whole range of Hue vale available but is limited between violet and green. This choice is first aesthetic but has probably sensory integration implication. This color range seems to be a continuous and well bounded domain.

Zoom in and zoom out, rotation are the basic transformations used in this Vizualizer. The RMS energy is mapped to the luminance (y-axis on figure) values and to the rotation.

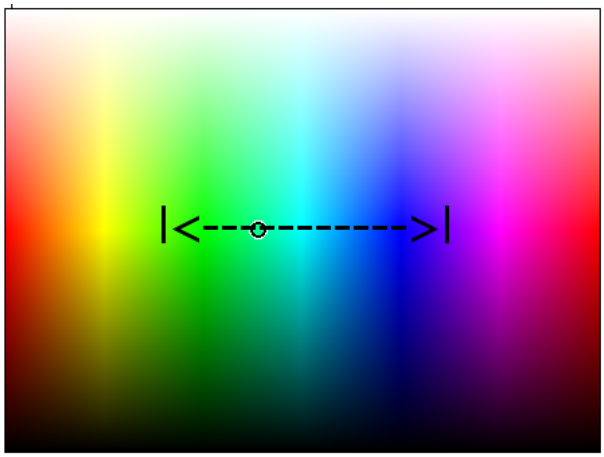


Figure 5: pitch to color mapping, from E1 to E4 (right to left)

$RMS = \sqrt{\sum_{k=1}^{n} S_k^2}$, where S_k is the k^{th} sample of the signal vector.

CPU limitation has initially influenced my choice to not use pitch tracking and spectral analysis but using control data instead of them also allow more flexibility.

8 Sensory and motor interaction

8.1 Visual and sound sensory interaction

If we consider the visualisation as a visual feedback, sensory information affluence and coherence should be considered. The degree of abstraction of the visualisation, the range and dimension of colour used could be parameters to take in account. Visualisation has changed my way to play because now the message passes to the audience and gives me feedback both through the visual and the auditory sensory channels.

8.2 Two-Handed interaction

Guiard (1987) has described the kinematic chain model, a general model of skilled bimanual action (i.e a serial linkage of abstract motors).The KC model hypothesizes that the left and right hands make up a functional kinematic chain. This leads to three general rules:
(1) Preferred-to-non-preferred reference: The preferred hand performs its motion relative to the frame of reference set by the non-preferred hand.
(2) Asymetric scales: The preferred and non-preferred hands are involved in asymetric temporal-spatial scales of motion. The movements of the non-preferred are low frequency compared to the detailed work done by the right hand. The preferred hand acts efficiently at microscopic scales and the non-preferred hand at macroscopic scales.
(3) Preferred hand precedence: The non-preferred hand precedes the preferred hand; for example left hand posi-

tions the paper, then the right hand begins to write (for a right-hander).
This model was described in a context where the two hands cooperate for the same goal (writing, using a microscope, playing golf or using a gun). Observation on conventional instruments may confirm some of the rules in a task independent context. If one observes the basic mode guitar playing[6] one can see that, as in the rule (3), the Non Preferred Hand precedes the Preferred Hand; it first selects the pitch before the Preferred Hand plays the note.

9 Musical evaluation on stage

this instrument has been used in different occasion on stage. A musical piece named Voodoo Gloves is based on musical gestures and themes from Jimi Hendrix. Musical examples are available online, three of them will be commented. These concerts allowed me to evaluate the instrument, but training was also very instructive. The velocity of execution needed to reproduce a musical phrasing is primordial, of course training will provide it, but the system must be robust and precise and synchronised in time to allow it . Dynamic indeed by force feedback can also give better result. Reverb, flanger, phaser and others effect were also used along the pieces. A Wah-Wah foot pedal was added at certain moments.

9.1 Introduction of Voodoo Gloves

During this introduction, I first expose the range of damping and the range of excitation velocity. The pitch doesn't change with the non-preferred hand but I use the possibility to play in different octave by choosing between the two pressure sensors located on the index of the preferred hand. Then the play becomes rhythmic and is still using the same two notes. Finally, I use a continuous excitation and play with the spectrum, then without playing new notes i change the pitch by doing new signs.

9.2 Voodoo Chile theme

this example show me playing the theme of Voodoo Chile (lightly revisited) first at low speed, then at more high speed. At the same time, I also develop the continuous spectral possibilities of the instrument by playing on damping, force gain, force extra parameter. I also use a Wah-Wah pedal which is essential to play this Jimi Hendrix's theme.

9.3 Chorus improvisation

This chorus is inspired by Star Spangled Banner mimics. The sound is sustained by applying a continuous force with the aftertouch pressure on the preferred hand index, and modulated with the little finger which acts on the force extra parameter and modifies the spectrum. Changing the damping has also an influence on the spectral evolutions. The pitch is changed occasionally, but this gesture is not so melodic but principally contribute to the atmosphere of this part of the piece.

10 Discussion

10.1 Using symbolic gesture

Using symbolism could be very helpful to drive complex harmonic structures while playing with the spectrum. For this purpose ones should used gesture and not sign recognition like it is currently done, and then analyze style of these gesture. This suppose to be able to segment gestures and to be able to qualify them.

10.2 Choice of sensor technology

There are other possibilities for sensors like Hall effect sensors, infrared sensors, optic sensors that can be used instead of FSR sensors, it could be interesting to compare them for measuring the same instrumental gesture (i.e.: playing a note [7] and using aftertouch [8] with a finger in the context of glove-based instrument. Measuring flexion can be done by different technologies (using piezo resistance, optic fibers and others) some are better robust than other, some are more ergonomic than others. Choice of sensor must be considered seriously but is not so evident.

Acknowledgements

Thanks to Jean-Michel couturier who has implemented the scanned synthesis Max/MSP External, and to Matt Wright to have port the MLP externals during the last years.

References

Daniel Arfib, Jean-Michel Couturier, and Loic Kessous. Strategies of mapping between gesture data and synthesis parameters using perceptual spaces. *Organised Sound*, 7(2):127–144, 2002.

Claude Cadoz and Marcelo Wanderley. Gesture-music. *Trends in Gestural Control of Music, CD-ROM, diteurs M. Wanderley and M. Battier, publication Ircam*, 2000.

CNMAT. Max/msp multi layer perceptron external. *http://cnmat.cnmat.berkeley.edu/MAX/neuralnet.html*, 2004.

[6] and not tapping, hammer-on, pulling-off techniques which are more expert level techniques

[7] instantaneous excitation

[8] continuous excitation

Jean-Michel Couturier. A scanned synthesis virtual instrument. *New Interface for Musical Expression*, pages 176–178, 2002.

FifthDimensionTechnologies. 5dt data glove 5. *http://www.5dt.com/products/pdataglove5.html*, 2004.

Yves Guiard. Asymmetric division of labor in human skilled bimanual action: The kinematic chain as a model. *Journal of Motor Behavior*, 19(4):486–517, 1987.

Andy Hunt and M.Marcelo Wanderley. Mapping performance parameters to synthesis engine. *Organised Sound*, 7(2):103–114, 2002.

Andy Hunt, M.Marcelo Wanderley, and Matthew Paradis. The importance of parameter mapping in electronic instrument design. *Journal of New Music Research*, 32 (4):429–440, 2003.

InfusionSystemsLtd. I-cubex. *http://www.infusionsystems.com*, 2004.

Bill Verplank, Max Mathews, and Robert Shaw. Scanned synthesis. *Proceedings of the 2000 International Computer Music Conference, Berlin, Zannos editor, ICMA*, pages 368–371, 2000.

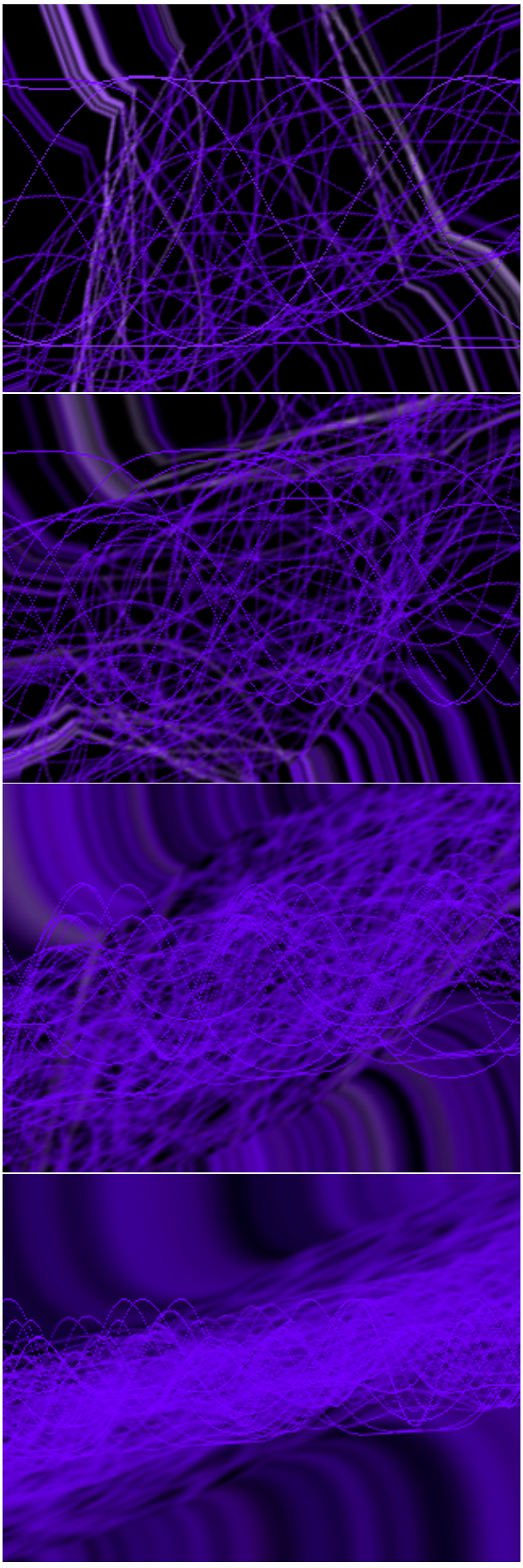


Figure 6: Sound visualization used in the scangloves, t=1, 2, 3, 4

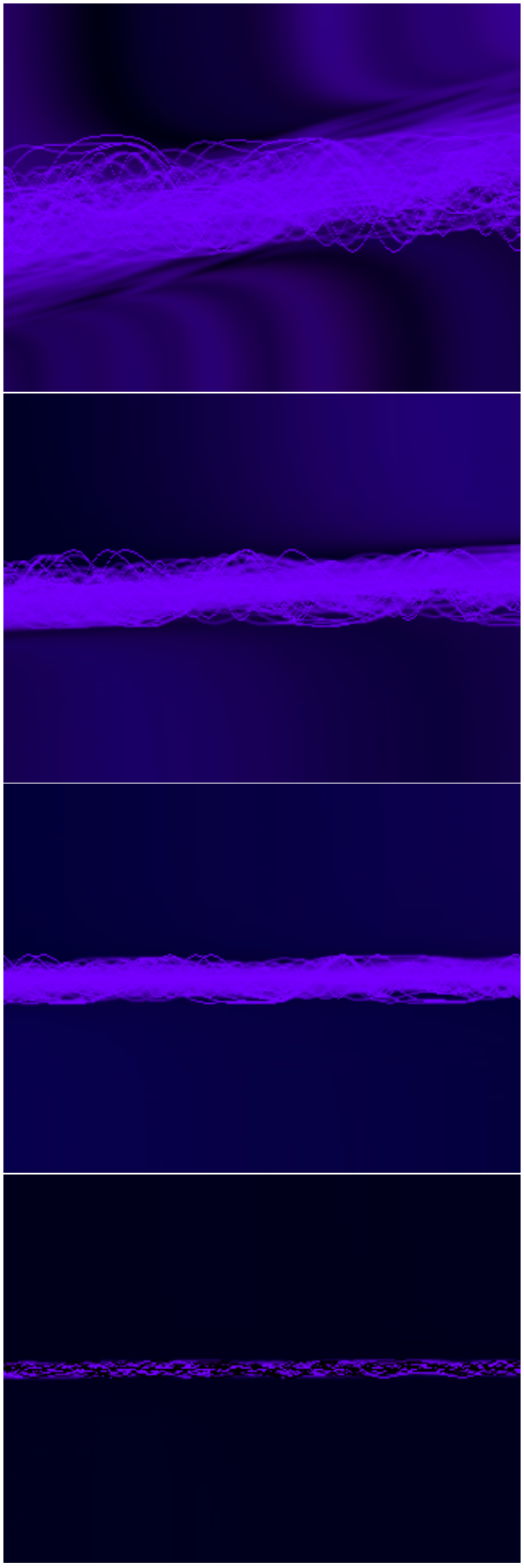


Figure 7: Sound visualization used in the scangloves, t=5,6,7,8

Using Optical Motion Capture for Gesture Recognition

D. Lowe, N. Murray, J. Y. Goulermas and T. Fernando
Vicon Motion Systems Ltd.14 Minns Business Park, West Way. Oxford. OX2 0JB
Centre for Virtual Environments, Business House, University of Salford, Salford, M5 4WT, U.K.
David.lowe@vicon.com n.murray@salford.ac.uk j.y.goulermas@salford.ac.uk t.fernando@salford.ac.uk

Abstract

The paper will discuss the use of an optical motion tracking system and specifically it's use for tracking the movements of the hand and recognising hand gestures as an input into a virtual environment.

1 Introduction

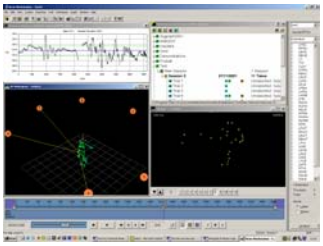
Tracking is the process of obtaining the location (position and orientation) of a moving object in real-time. Tracking is the probably the most important component within a virtual environment system and especially if it is to be used for recognizing human gestures and interpreting those gestures into commands or actions. With inaccurate tracking the process of interacting within a virtual environment can prove cumbersome and detract from the experience of using the environment and lead to simulator sickness. This can be caused by proprioceptive conflicts, such as static limb location conflicts, dynamic visual delay (lag) and limb jitter or oscillation. The following are some of the criteria defining the requirements of a tracking system:

- Accuracy – the error between the real location and the measured location. (Ribo et al., 2001) state that position should be within 1mm, and orientation errors should be less than 0.1 degrees.
- Degrees of freedom – the capability of the system to capture up to 6 degrees of freedom (DOF)consisting of position and orientation (roll, pitch and yaw). VR systems typically need 6 DOF capabilities.
- Range – the maximum working area within which the system can or needs to operate. This will be dependent on the size of the workbench or cave volume. These range from 1.5 metre to 3metre cubic volumes.
- Update rate – maximum operating frequency of reporting positional values. VR systems need to maintain a frame rate of at least 25Hz. Therefore a tracking system should be able to track objects at a minimum of 25Hz.
- User requirements – it is preferable that the system should not restrict the mobility of the user so it would be an advantageous for the tracking components to be wireless, furthermore they should be light and easy to hold/wear.

None of the current commercially available VR tracking systems fulfils all of the above criteria (see (Rolland *et al.*, 2000) for a recent survey). The following section will describe how the Vicon system operates. With this integration complete, the extra advantages offered by the system over the acoustic tracking were explored by the creation of alternate input devices. This involved the initial creation of a tracked hand and led to the development of a wireless glove based gesture recognition system. Finally, conclusions and future work are outlined.

2 System Operation

Current areas of use of the Vicon motion capture system have been life sciences, with applications such as clinical gait analysis, biomechanics research and sports science and within the area of visual arts for broadcast, post production and game development. Vicon track reflective balls known as markers. Vicon track the position of these markers with software developed by their engineers.



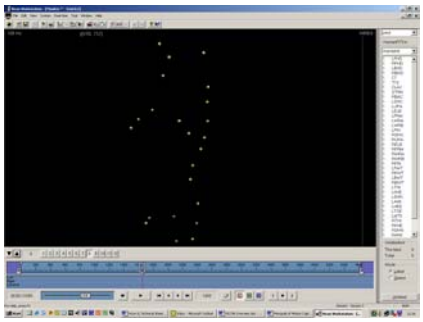
To track markers Vicon must first have a means of capturing the information / images that are going to tracked. Vicon capture the images of the markers with high resolution cameras. The reason that the markers are reflective is so that Vicon can highlight the markers clearly and obtain a high contrast image of the markers from the background image.

Vicon get a very bright reflection from the markers by shining a bright light that is digitally controlled to run at different speeds and power that will give the best reflection from the markers, Vicon call this the strobe light – as seen from the image above.

Here is an example of a live image that is seen by a Vicon camera. The camera sends this entire image down to the Vicon 8i datastation. The datastation controls the cameras and strobe lights. The datastation

also powers, synchronizes and processes the video images seen by up to 24 high resolution cameras.

1. Without camera flash (top) 2. With camera flash (middle) 3. With strobe light (bottom)



To be able to track the 3d position of each individual marker the software needs to be able to triangulate the position of the marker in every frame. To triangulate the position of a marker you need 2 or more cameras to be able to see the same marker.

Vicon get a very bright reflection from the markers by shining a bright light that is digitally controlled to run at different speeds and power that will give the best reflection from the markers, Vicon call this the strobe light – as seen from the image above.

Because the markers are always in motion Vicon must also know the exact 3d position and orientation of each camera as well as information about the shape of the lens in use. To calculate this (position, orientation and lens information) Vicon "calibrate" and "linearise" all the cameras together in a single process. This process takes no more than 2 minutes to complete.

What you are trying to capture will determine where you position the cameras. Vicon can track any reflective marker of practically any size. 2mm to 300mm is the known range that anyone has ever wanted to capture. This means Vicon can track the subtle movements of the face or the movement of a basketball and the players on a court. These two extremes represent completely different camera positions and choice of lens. Just like a normal camera Vicon can change the lens to suit the performance area you are trying to capture.

The one thing that is consistent with all camera setups is that the subject or area that you want to capture is sufficiently surrounded and seen by all the cameras. A good rule of thumb is to have at least 5 camera views overlapping a single area. This will account for the subject obstructing markers attached to their front or back.

The first defining factor is what type of movement is to be captured. If it is a dance sequence with no heavy person to person contact then you will need fewer cameras. This is because the overlapping camera views will be able to get enough shots of each actor from every side to get a complete set of 3d reconstructions from the markers they are wearing. Also in this instance you can have a fairly large volume so each camera can be set further back to cover a larger field of view and up to it's maximum depth of field.

Definition: [Field of View] – This is the angle of view that the camera can see an image from.

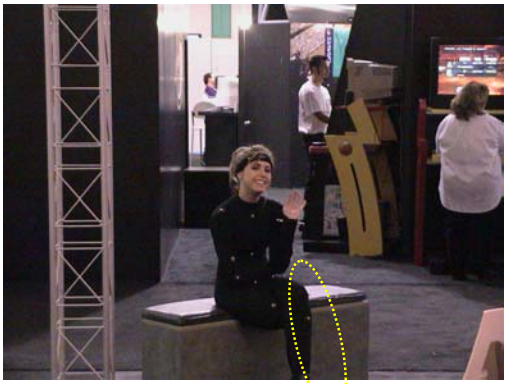
Definition: [Depth of Field] – Depth of field is the range that is in focus at a particular aperture.

If you wanted to capture 6 people dancing together you would find the absolute minimum size or space that the performers would require to do their movements. You then mark this area out and get as many cameras covering that area from different angles as possible. The most experienced and professional motion capture studios will rehearse the most demanding moves before they set the cameras up as these moves will define the performance area that you are working with for the remainder of the shoot. Sometime you will have several different demanding shoots that require different setups. In this case it is best to assign the rest of the moves to the most appropriate setup and then make a shoot list based on the best performance area. Getting this right will drastically improve the data quality that you get out of the system.

Vicon provide special suits with our systems that enable you to stick the hook side of Velcro directly to

the suit. The positioning of the markers is entirely up to you. But before you start going crazy sticking markers everywhere, here are a couple of helpful ideas to get the BEST actor setup.

You need 3 markers do define translation and rotation of an object. If you think of person as a series of objects connected together then you can break down where markers should be positioned. For example the arm has three objects (or segments as Vicon call them). These are upper arm, lower arm and hand. To be able to capture the rotational and translational motion from these segments Vicon need 3 markers per segment. Because all 3 segments remain connected they can also share markers between segments.



This method can be applied for arms and legs or other animal limbs depending on what you are trying to capture. The next 2 important things are the head and waist each of these segments have 4 markers, this means that if 1 marker is out of view you can still get all the motion from the other 3. Sometimes Vicon place 5 markers on these segments, especially if the motion will hide markers or risk markers being torn off during the motion

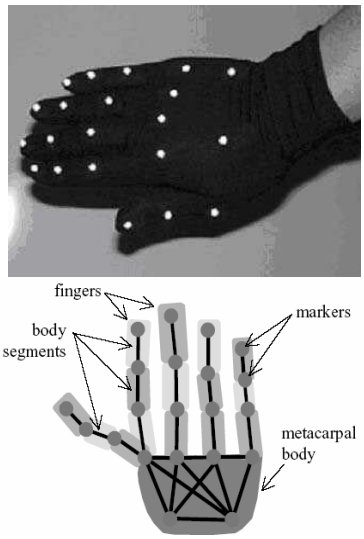
The first thing you always do is teach the system where each marker is for your actor or subject and what the name of each marker is. This means that the software can do all the hard work for you and apply a kinematic model to the motion you have captured. This will make your job far easier when you come to export your motion to your desired 3D software package.

3 Gesture Recognition

To implement gesture recognition, Vicon attach a number of 4mm markers on a fabric glove. As shown in Figure 2 (left), there is a small number of markers placed on the metacarpal and some on each individual finger. The latter markers are placed at the finger joints, so that the distance betViconen immediately adjacent ones remains fixed; this allows the formation of rigid body segments as seen in Figure 2 (right). The metacarpal is considered a single almost rigid body, while on average there are three bodies per finger.

The VICON system is used in two stages. In the former, the user repeatedly performs a series of gesticulations which span the entire human kinematic range at all possible finger positions and angles. Subsequently, manual labelling of the markers takes place and the system calculates the kinematic information relating the trajectories of all markers of the entire motion trial to a predefined kinematic structure which correspond to the hand marker and body model. This is an off-line procedure and it only needs to take place once, unless the glove size and/or markers position change. The second stage is the real-time operation of the system which provides the spatio-temporal data, while the user is performing a series of gestures. During this phase, the captured marker trajectories are best-fit to the stored model information.

- metacarpal
- body
- fingers
- body
- segments
- markers



Marker attachments on a glove (top). Schematic placement of markers and formation of body segments in a typical glove marker set configuration (bottom).

We target recognition of two types of gestures: static gestures (postures) and dynamic (timevarying) ones. For both cases, we make use of Hidden Markov Models (HMM) (Rabiner, 1989), which have been previously used successfully in gesture recognition but utilising different data sensors; see for instance (Yoon, Soh, Bae & Yang, 2001), (Nam & Wohn, 1997). A HMM is a type of probabilistic state model whose states cannot be observed directly, but only through a sequence of observations. First-order HMMs follow the property that the current state only depends on the immediately preceding one. A HMM consists of the following elements: a set of hidden states *S*={*s1, ¼sN*}, a set of observation symbols *V*={*v1, ¼vM*} a state transition matrix *A*=(*aij*), with *aij* being the probability of moving from state *si* to *sj*, an observation probability matrix *B=(bik)*, where *bik* is the probability of emitting symbol *vk* at state *si* and p an initial probability distribution for each state *si*. In this way, a HMM is fully defined as l=*(A,B,* p*)*.

There are three basic issues involved with a given model l. The first, the *evaluation*, relates to the calculation of the probability of observing a given sequence of symbols *o=(o1, ¼,oT)* for *T* discrete time events, i.e. *P*(*o*| l). If *s* defines some state sequence of length *T*, then Vicon have:

$$P(o/\lambda) = \sum_{s} P(o/s,\lambda) \cdot P(s/\lambda)$$

In Equation 1, the probability of *o* given a sequence *s* is $\prod_{t=1}^{T} b_{s_t o_t}$, while the probability of having generated *s* is $\pi_{s_1} \cdot \prod_{t=1}^{T-1} a_{s_t s_{t+1}}$. For the actual evaluation however, we use the Forward-Backward algorithm, which effectively alleviates the exponential complexity of Equation 1. The other two important issues related to HMMs are: *decoding*, that is how we can estimate the optimal state sequence *s* given *o* and l, and, *learning*, that is how to estimate the three parameters of l=*(* p,*A,B)*, given some sequence *o*, such that P(*o*| l) is maximised. These two questions are normally handled with the Viterbi and the Baum-Welch algorithms, respectively; see (Rabiner, 1989).

All data was preprocessed so that discrete observation sequences were obtained for training and testing the HMM. In order to achieve positional and rotational recognition invariance, Vicon used normalised marker distances between selected pairs of markers for static gestures. For dynamic ones, a type of chain coding of the trajectory of the averaged marker positions (single point) was used. For a total of *G* gestures, a set of l*1* , ¼, l*G* models was stored, and each captured test frame was compared with all models to find the one yielding the highest value probability ( ) *i* | *o P* l . As mentioned in (Nam & Wohn, 1997), the recognised gesture can be used as symbolic commands for object description and/or action indication. Navigation, manipulation and environmental control are easily and more naturally achievable by using gesture recognition. The motion based tracking system was integrated with a virtual environment for assembly and maintenance simulation and training. The environment allows for the assembly and disassembly of components via geometric constraints and supports the simulation of the mechanisms (allowable movements) of the constructed components.

4 Conclusions and Future Work

The Vicon system was successfully integrated and found to be more accurate than other tracking systems. The system offers further advantages in that it is possible to place the markers on any device for calibration and so allows the easy addition of extra input devices for minimal extra cost. These were used for tracking the wrist and then a simple pinch hand to a gesture recognition system utilising markers placed on a glove. For gesture recognition, the provided marker and body data were rich enough to allow off-line training and online recognition. Although accuracy depends on data pre-processing as well as the pattern recognition algorithm used, the utilisation of the provided 3D positional data was straightforward.

The system also satisfies the criteria that it should be minimally invasive as the markers are light weight and do not require wires. The disadvantages of the system are that it suffers from line of sight problems although this was not found to be a problem with the relatively small operating area around the workbench as the user is always facing the screen and so is facing 4 of the 6 cameras. Also, certain problems were caused in the real-time mode, where a number of frame sequences had missing marker information from the data stream. This problem could be improved by adding extra markers on the wrist for stability of recognition.

References

[1] Ribo, M., Pinz, A., and Fuhrmann, A. L. (2001). A new optical tracking system for virtual and augmented reality applications. In IEEE Instrumentation and Measurement Technology Conference.

[2] Rolland, J. P., Baillot, Y., and Goon, A. A. (2000). A survey of tracking technology for virtual environments. In Barfield, W. and Caudell, T., editors, Fundamentals of Viconarable Computers and Augmented Reality.

[3] Rabiner, L. R. (1989). A tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proc. IEEE*, 77 (2), 257-286.

[4] Yoon, H. S., Soh, J., Bae, Y. J., and Yang H. S. (2001). Hand gesture recognition using combined features of location, angle and velocity. *Pattern Recognition*, 34, 1491-1501.

[5] Nam, Y. and Wohn K. (1997). Recognition of hand gestures with 3D, nonlinear arm movement. *Pattern Recognition Letters*, 18, 105-113.

Novagen: A Combination of Eyesweb and an Elaboration-Network Representation for the Generation of Melodies under Gestural Control

Alan Marsden
Music Department, Lancaster University
Lancaster, LA1 4YW, UK
A.Marsden@lancaster.ac.uk

Abstract

A system which generates melodies influenced by the movements of a dancer is described. Underlying the melody-generation is a representation based on the theory of the early 20th-century German musicologist, Heinrich Schenker: a melody is derived from a simple background by layers of elaboration. Overall, the theory has similarities to a generative grammar. Generation of melodies is achieved by repeatedly applying elaborations to the background to achieve the desired number of notes. Elaborations are selected by a weighted random process which can take into account the pattern of elaborations used earlier in the melody. A number of parameters control this process, both by setting the relative weights for different elaborations and by controlling the number of notes generated, their distribution throughout the bar, and the degree of similarity of the generated pattern to previous sections of the melody. These parameters are adjusted via MIDI messages from an Eyesweb application which tracks a dancer via video to categorise the pose or movement observed into one of four categories, and to determine the degree of 'activity' in the movement. The result is real-time generation of a novel melodic stream which appears meaningfully related to the dancer's movements.

1 Introduction

In existing systems where gesture controls some form of musical output, it is not common for the music to be actually generated in response to the gestural input; more commonly either the gesture triggers, controls or otherwise modulates pre-composed music, so the metaphor for gestural control is conducting a musical ensemble, or the gesture triggers individual notes or sounds, and the metaphor is of playing a musical instrument. In this project, not only is the generation of a melody under gestural control, but the melody is intended to conform to the stylistic traits of common-practice tonal music, such as found in music from Bach to Brahms. This is achieved by basing the melody-generation on an established theory of the music of that period, i.e., the theory of Heinrich Schenker (1868-1935). The primary objective of the project is to test the underlying theoretical model as a formalisation of musical structure, a model which has potential applications in a number of musical spheres. If the objective had been principally creative or principally to develop a gestural interface, the path of research would have been different. Thus the approach taken towards the creation process does not have a high degree of artistic sophistication. On the other hand, in the area of gestural interface, a high degree of sophistication was found readily to hand in the form of Eyesweb (Camurri et al., 2000; www.eyesweb.org), which allowed the easy extraction of useful information from the movements of a dancer by the simple means of a video camera.

Overall, the project can be reported to have been successful, in that the essential concept is proven to be able to produce melodies whose characteristics recognisably change in relation to the movements of the dancer, and which remain stylistically 'correct': the music never sounds incoherent or 'wrong'. On the other hand, the melodies produced do not sound particularly musically appealing, and in particular they lack division into meaningful phrases. To date, the system has only been tested with a set of video

films of a dancer rather than with a live video feed, but in principle there is no reason why it should not work in this situation also, where there would also be the added advantage of feedback from the melody-generation to the dancer, allowing the possibility of creative interaction between the dancer and the system.

2 Underlying Musical Structure

2.1 Schenkerian Theory

A common theme of music theory from the eighteenth century has been that underlying the sequence of notes which forms the 'surface' of a melody is a less elaborate framework. The idea finds its fullest exploitation and culminating exposition in the work of Heinrich Schenker, whose seminar work *Der freie Satz* (1935). Computational implementations of the theory are found in the work of Kassler (1967), Frankel, Rosenschein & Smoliar (1976), and a number of more recent authors. Pursuing the common parallel between music and language, the theory has been compared to generative grammar, and a number of computational implementations of musical grammars have been reported also, some more closely related to Schenkerian theory (e.g., Baroni, 1983, and Baroni, Dalmonte & Jacoboni, 1992), and others of a very different nature (e.g., Kippen & Bel, 1992). The two ideas have come together also in the influential theory of Lerdahl and Jackendoff (1983), which has itself been subject to attempts at computer implementation (e.g., Baker 1989).

2.2 Representation System

The system used here was first reported in Marsden (2001) as a means of representing musical pattern. It differs fundamentally from the adaptation of Schenkerian theory by Lerdahl and Jackendoff in that elaborations are taken to apply to intervals between notes rather than to the individual notes of background and middleground structures. Indeed, the foundation of the system is a set of elaborations which could be expressed as rewrite rules whose left-hand sides are almost always a pair of notes and whose right-hand sides are three notes ('almost always' because in a few cases, such as suspensions, a context wider than just two notes must be taken into account). Thus each elaboration generates a new note. In some cases this note occurs between the two original notes. Thus an 'upper neighbour note' is a note one step higher than the original second note, and placed in time between the original first and second notes. (This is American/German terminology commonly used in Schenkerian writing; the traditional British terminology for the same thing is an 'upper auxiliary note'.) Some elaborations (referred to as 'accented elaborations') displace the first note so that it occurs later and put a new note in its place. Thus an appoggiatura (which can also be described as an accented upper neighbour note) is a note one step higher than the original first note which occurs at the original time of that note and is followed by a note of the same pitch as the original first note but placed in time between the original first and second notes. Manipulation of the representation is simplified by insisting that each elaboration produce no more than one new note, but this does mean that passing notes and other elaborations which produce more than one note can require a series of interdependent elaborations in the representation.

Figure 1, which is screenshot showing a fragment of melody generated by the system, demonstrates the principles of the system. The pair of notes on the top line are part of the underlying background. Each lower line shows a layer of elaboration on the way to the final melody shown in the bottom line. The boxes in between contain codes for the elaborations used in generating the melody.

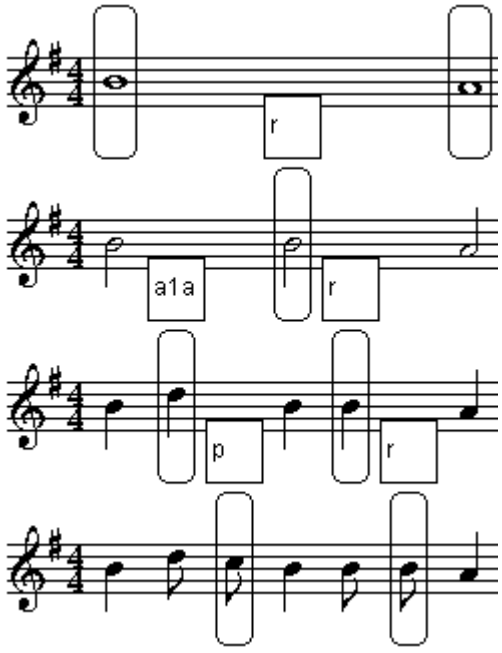


Figure 1. Example of generated melody

For each note in the melody, there is a prevailing key, harmony and metre. These influence the generation of new notes, so that, for example, what is meant by 'one step' depends on the pitch of the original note and the prevailing key: it might be a whole tone or a semitone. The placement of notes in time is determined by the metre and by a property of each elaboration which determines whether it is 'even', 'short-long' or 'long-short'. The last, for example, would produce dotted rhythms in a duple metre.

2.3 Benefits for Automatic Generation

This system of representation has three benefits for automatic generation. Firstly, just as a grammar ensures that any sequence generated by the rules of the grammar is 'grammatical', this representation system ensures that any melody generated by a process of elaboration from a simple background is 'musical', at least to the degree that the sequence lacks notes which sound 'wrong'.

Secondly, the derivation from an underlying framework ensures that the melody follows a logical harmonic pattern (e.g., a conclusion on the tonic can be guaranteed). Furthermore, it increases the likelihood of some sense of purposeful goal-directed motion in the melody, in contrast to the impression of aimlessness which can result from generation mechanisms such as stochastic processes which essentially append notes repeatedly to an existing sequence.

Thirdly, that the system explicitly represents musical pattern (as described in Marsden, 2001) allows explicit operations on musical patterns. Thus the pattern of an earlier part of the melody can be replicated by repeating its component elaborations in the new context. This will result in a different sequence of notes and a different sequence of intervals according to the interval and prevailing harmony of the new context: what is an interval of a third in one context might become an interval of a fourth in another, etc. (This kind of adjustment of intervals to preserve a pattern in a different context is a common characteristic of actual pieces of music.) Furthermore, a pattern need not be simply replicated, but it can be made more or less elaborate by the addition or deletion of elaborations. Thus the common musical procedures of variation can be implemented.

3 Generation Procedure

Generation begins from a given background, which may be specified by the user in advance. This background consists of a single note per bar, including a specification of the key and metre, and the harmony for each bar. The generation procedure repeatedly passes over this background, adding elaborations in real time, and playing the resulting melody. The selection of elaborations, including the decision of whether to elaborate or not, is made at random, using weights governed by a number of time-varying parameters

The first of these parameters is a target number of notes per bar. As elaboration proceeds to deeper levels, the target number of notes to be generated obviously decreases (since notes have already been generated at higher levels), so elaboration inevitably stops when the target number reaches zero. At higher levels, a second parameter of 'evenness' determines how evenly the targets are divided between 'left' and 'right'. Thus, for example, the target number for a whole bar might be eight notes. There is one note already given by the background framework at the beginning of the bar, and at the first layer of elaboration, a new note will be generated, let us say in the middle of the bar. Thus six more notes are to be generated at lower levels. If the 'evenness' is high, these will be distributed three in the first half of the bar and three in the second half. If, on the other hand, it is low, these might be distributed one in the first half of the bar and five in the second half. The 'evenness' parameter can vary with the level of elaboration in addition to time. Thus it is possible that at higher levels notes might be distributed evenly while at lower levels they are distributed unevenly.

Another set of parameters provides a likelihood profile of elaborations. This can also vary with level in addition to time. Thus a melody might be generated with a high likelihood of arpeggiations (leaps within the prevailing harmony) at higher levels but a high likelihood of passing notes at middle levels, and repeated notes most likely at the lowest levels.

A final set of parameters controls 'regularity', which refers to the degree to which elaborations in one part of the melody follow the pattern of those in a preceding part. Different parameters determine the degree of regularity with relation to different time intervals, so at one time the melody might copy the same pattern of elaborations from one bar to the

next while with a different setting of parameters the melody might copy the same pattern of elaborations four times within the same bar. This is implemented by determining a degree of likelihood that the chosen elaboration will be a copy of the elaboration found previously in the generated melody at each of the appropriate time intervals (quarter bar, half bar, full bar, etc.), and a residual degree of likelihood that the elaboration will be chosen at random according to the likelihood profile determined by the parameters described above.

This generation procedure has been implemented as a Java application, building on previous work to implement the representational system derived from Marsden (2001).

4 Gestural Control

4.1 Eyesweb

Eyesweb was chosen as the mechanism for capturing gesture for the following reasons. Firstly, it requires no special hardware and operates with standard video and computer equipment. Thus the movements of the dancer are not impeded in any way, and the system can be readily used in many circumstances. Secondly, it is a platform which is efficient and yet remarkably easy to use, following a paradigm of interconnecting processing units which is familiar to musicians who have used such software as MAX and Pd. Thirdly, it has built-in facilities for the extraction of information about the position and movement of a human figure. Finally, it has facilities for MIDI output, which makes real-time intercommunication with musical software very simple.

4.2 Gesture Capture

The Eyesweb application developed for this project assumes that the input is video of a dancer moving in an otherwise unchanging scene. For each frame, Eyesweb extracts the outline of the figure by subtraction from the background and can compute various pieces of information about the figure in space, such as the position of the limbs. For this application, however, the information used is the bounding rectangle, the area occupied by the figure within that rectangle, and its 'centre of gravity'. On the basis of this, two fundamental pieces of information are determined.

Firstly, the gait or posture of the dancer is placed into one of four categories. When 'walking' the figure occupies a large proportion of the bounding rectangle, whose sides are large in relation to its top and bottom, and the 'centre of gravity' moves at a moderate speed. When 'dancing', by contrast, the area occupied by the figure is smaller in relation to the bounding rectangle, because the dancer will often have arms or legs extended, and the rectangle may be more square, but the centre of gravity is once again moving. A third gait described as 'posed' is similar, but the centre of gravity moves little (the dancer might be moving the limbs, and so might not be posed in the strict sense of the word). Finally a 'crouching' gait is recognised by once again little movement of the centre of gravity but a large proportion of the bounding rectangle taken up by the figure. These four gaits were selected on the basis of the movements of the dancer observed in the video films used as experimental material in the course of this project.

The second piece of information extracted in the Eyesweb application is the degree of activity in the dance, related to the speed of movement of the dancer. This might be movement of the whole body from one place to another or movement of the limbs without moving the body. Thus the measure is based on the speed of the fastest-moving edge of the bounding rectangle. Movements of the whole body are likely to cause both the sides of the bounding rectangle to move in the same direction. Movements of the limbs are likely to cause just one side or the top of the rectangle to move, or perhaps both sides in opposite directions. Only movements towards or away from the camera, which are rare in isolation, cause no movement of the edges of the bounding rectangle. In order to compensate for the distance of the dancer from the camera, the speed of movement is scaled according to the size of the bounding rectangle.

The Eyesweb application therefore captures not so much individual gestures as global information about the characteristics of gestures at any one moment. This information is transmitted via MIDI control messages which indicate (1) the nature of the dancer's gait at that time, and (2) the speed of movement.

4.3 Control of Melody-Generation

The melody-generation application receives MIDI messages and responds to the control messages used

to transmit the gestural information by varying the parameters described above in section 3. The speed of movement of the dancer is related directly to the number of target notes: the faster the dancer moves, the more notes in the melody.

Gait, on the other hand is related to sets of other parameters so that generated melodies with different characteristics are associated with each gait, following metaphors of music and movement which appeared viable to the author. Thus, a crouching gait is associated with a high degree of regularity—so that the unchanging position of the dancer is reflected in the unchanging pattern of the music—and with small intervals—the compact shape of the dancer is related to the small degree of movement in pitch.

5 Conclusions

The overall musical results are, as predicted, melodies which sound 'musical' to the degree that they follow the kind of harmonic and intervalic patterns found in real music. The association with the movements of the dancer also seem credible in that changes in the dancer's pattern of movement are accompanied by meaningfully related changes in the melody. One does not have an impression of the dancer controlling the music, though, in part probably because there is not a sufficiently rapid and tight connection between the dancer's movements and events in the music. There is nothing, for example, which suggests that a particular note or set of notes has been generated specifically because of a particular gesture by the dancer—there is no sense of the dancer directly making things happen. The impression, rather, is of an independent musician who is responsive to the movements of the dancer and who varies the melody played according to the character of those movements.

The project demonstrates Eyesweb to be an effective and useable gesture-input system for this kind of application. A higher degree of control, along the lines mentioned in the previous paragraph, would have required a finer analysis of the position and movement of the dancer. Eyesweb does include such facilities, but, as described in section 4.2 above, for this project these facilities have not been used.

The major objective of the project was to test the underlying representation system as a basis for melody-generation, and in this it has been successful only to a degree. The melodies generated have some musical credibility, but the expectation that the generation of melodies by elaboration of a coherent framework would ensure a sense of goal-directedness has not been realised. The melodies also lack any sense of phrase (any sense of starting and stopping at certain points, or of division of the stream of notes into coherent melodic units). These two deficiencies are probably related. They might be rectified by a revision of the melody-generation process so that melodies are explicitly generated in phrases which in turn are made up of smaller melodic units, and so on. A disadvantage of this is that it would introduce a coarser level of granularity into the generation process and create difficulties for a design which aimed to see changes in the gestural input reflected quickly in the melodic output.

A second possible approach to overcoming the lack of goal-directedness and phrase structure is a more reflective melody-generation system. One pattern of notes can generally be generated as a result of a number of different patterns of elaboration, especially if different possible background frameworks are considered also. Thus while a melody might have been generated as a result of one pattern of elaborations, it might be perceived as a different pattern of elaborations. This is particularly important where a melody aims to show some degree of regularity by copying the pattern of elaborations used earlier: if the pattern of elaborations perceived by a listener is different from the pattern used by the generation process, then the regularity might not be perceived. Thus the melody-generation process needs to analyse its own output to consider how a sequence of notes might be perceived. There is a considerable quantity of research to be done here, most fruitfully probably in analysis of real melodies.

Overall, the general concept is proven effective, and the paradigm of a system of music-generation responding to a dancer's movements is shown to be aesthetically viable. One can imagine a more sophisticated system in future with which a dancer becomes familiar so that he or she can produce music and movement in a single unified yet improvised art work.

Acknowledgements

This research has been supported by a grant from the Arts and Humanities Research Board in 2001 under their scheme of Small Grants in the Creative and Performing Arts which paid for a visit to the University of Genoa. I am grateful to Antonio Camurri for his assistance and support, particularly in allowing me to work for a period in his laboratory at the University of Genoa, for guidance from his co-researchers in the use of Eyesweb, and for the use of video recordings of a dancer taken in Genoa.

References

Michael Baker. A Computational Approach to Modeling Musical Grouping. Contemporary Music Review, 4: 311-325, 1989.

Mario Baroni. The Concept of Musical Grammar. Music Analysis, 2: 175-208, 1983

Mario Baroni, Rossana Dalmonte & Carlo Jacoboni. Theory and Analysis of European Melody. In Alan Marsden & Anthony Pople (eds.) Computer Representations and Models in Music, London: Academic Press, 1992: 187-205.

A. Camurri, S. Hashimoto, M. Ricchetti, R. Trocca, K. Suzuki, G. Volpe. EyesWeb – Toward Gesture and Affect Recognition in Interactive Dance and Music Systems. Computer Music Journal, 24(1): 57-69, 2000.

R.E. Frankel, S.J. Rosenschein & S.W. Smoliar. A LISP-Based System for the Study of Schenkerian Analysis. Computers and the Humanities, 10: 21-32, 1976.

Michael Kassler. A Trinity of Essays. PhD thesis, Princeton University, 1967.

Jim Kippen & Bernard Bel. Modelling Music with Grammars: Formal Language Representation in the Bol Processor. In Alan Marsden & Anthony Pople (eds.) Computer Representations and Models in Music, London: Academic Press, 1992: 207-238.

Fred Lerdahl & Ray Jackendoff. A Generative Theory of Tonal Music. Cambridge, Mass.: MIT Press, 1983

Alan Marsden. Representing Melodic Patterns as Networks of Elaborations. Computers and the Humanities, 35:37-54, 2001.

Heinrich Schenker. *Der Freie Satz*. Vienna: Universal Edition, 1935. Published in English as *Free Composition*, translated and edited by Ernst Oster. New York: Longman, 1979.

Virtual Sculpture - Gesture-Controlled System for Artistic Expression

Mark Marshall*
*Interaction Design Centre
Computer Science and Information Systems Dept.
University of Limerick
Ireland
mark.t.marshall@ul.ie

Abstract

This paper describes the development of a gesture-based interface system for interactive multimedia applications. The system makes use of two-handed gestures, in the form of both free and direct gestures. This paper begins by describing the physical interface device that has been developed for the system, and then goes on to describe the first of the applications which have been developed to make use of the system in evaluating the use of different forms of gesture as a primary form of interaction for an interactive multimedia system.

1 Introduction

With the advent of ubiquitous computing the focus on mouse- and keyboard-based input for computing systems has begun to change towards a focus on less traditional forms of interaction which will allow user's to interact naturally and meaningfully with computer systems. As a result, there have been a number of systems to make use of various forms of gesture-based interaction. The natural, meaningful and rich form of control which gesture provides us with makes it especially useful for interactive multimedia systems.

As part of ongoing work in the area of gesture control, a system has been developed which allows for two-handed input in the form of both linguistic gesture, where the user's interaction is in the form of description of the action to be performed, and physically-based manipulation gestures, where the user directly grasps and manipulates the virtual objects.

Much of the work presented here builds upon earlier work performed as part of the Sounding Object (SOb) project. The Sounding Object Project[1] has pioneered recent attempts to couple physical simulations to efficient sound synthesis techniques. The European Commission funded this project to study new auditory interfaces for the Disappearing Computer initiative[2]. The work on this project included the investigation of gesture-based control of interactive sounding models, and the development of a virtual musical instrument, called the Virtual Bodhran (or Vodhran) (Marshall et al., 2002), which used gesture to interact with a sound model to simulate the playing of the traditional Irish instrument, the Bodhran.

The gesture control system which was created for the Vodhran, was limited in a number of ways. The system only tracked the position and orientation of the beater object held in the user's dominant hand, and the position of the user's secondary hand. From this data it extrapolated a secondary layer of data, such as direction of movement, speed of movement, and certain important events, such as sudden changes of direction. No provision was made for more detailed gesture capture, such as the detection of patterns of movement, or the detection of the positioning of the fingers of the hand. As a result of these deficiencies, it was decided to develop a more general and detailed gesture recognition system, the result of which is described here.

[1] http://www.soundobject.org
[2] http://www.disappearing-computer.net

2 The Gesture-Interaction System

In order to allow for a broad range of gestures to be used as interaction to the system, it was decided to develop a system that could detect the movements of the hands, the orientation of the hands, and the position of the fingers. This would allow hand posture detection to be used to indicate commands, and hand rotation and movement to indicate the parameters of a command. The use of two-handed gesture was decided on for a number of reasons.

Firstly, the use of the second hand offers an intra-modal increase in interaction over a single-handed system (Bolt and Herranz, 1992). Also, the use of two hands for object manipulation tasks is more natural to the user. In his previous works on gesture, Hauptmann (1990) has found that for certain tasks, the use of two hands is more common than the use of one. More specifically he found that:

- For a translation task, users on average use 1.1 hands
- For a rotation task, users on average use 1.2 hands
- For a scaling task, users on average use 1.5 hands

Taking this in to account, we would expect a two-handed system to feel more natural for the user when ma-

nipulating the objects, especially for the scaling manipulation.

2.1 The Gloves

In order to detect the posture of the hand, it is necessary to detect the position of the fingers. Many systems achieve this through the use of a number of bend-sensors placed along each finger. This allows for measuring of the bending at each joint in the finger and so can give a very accurate representation of the posture of the finger. However, for this system, this was deemed to be overly complex. By simply measuring whether or not a finger is bent, each finger can function as a simple on/off switch style of input. This gives us a total of 32 postures for each hand which, couple with information on the movement and rotation of the hand, should allow for a rich enough interaction with the system for many interactive applications.

This also allows for configuration of the system to match specific user's, including those with limited mobility of the fingers. Each bend sensor gives a range of output dependant on the amount of bending. While the system only uses the output from the sensors to detect whether a finger is bent or not, it is possible within the system to modify the threshold at which a finger is considered to be bent. Thus, while for a user with full mobility of the fingers, the sensor may have to indicate a bend of over 45° before the finger is considered bent, for a person of limited mobility of the fingers, this threshold could be set to a lower figure such as 10°.

Therefor, two gloves were built by attaching a single bend-sensor to the inside of each of the fingers of two gloves. This allowed for the measurement of the bending of a single joint in each finger. The output from each of the bend sensors was measured using a 10 channel, 10 bit analog-to-digital converter at a sampling rate of 10kHz, and sent to the PC. Tracking of the position and orientation of the hands was performed using a Polhemus Fastrak[3] position tracking device, with each glove having a single Fastrak sensor attached to the back of the hand. The Fastrak sensors allow us to track each hand with 6 degrees-of-freedom, and an update rate of 120Hz.

2.2 Gesture Recognition Software

There are two main forms of gesture interaction which the system is required to recognise. These forms are physically-based manipulation, where gesture is used to provide a multi-dimensional direct control over objects, and linguistic gestures, where gesture is used as a symbolic language. Weimer and Ganapathy (1987) viewed these two forms as being at opposite ends of the direct manipulation spectrum, however for the system described here it was decided that both forms of input must be recognisable by the system, although actual applications need not make use of both forms.

[3] www.polhemus.com

Each of these forms of gesture is based around the use of postures. A posture is a specific configuration of the fingers of the hand. For physically-based manipulation gestures the user places their hand (or hands) into a specific posture to indicate the operation being performed, and then moves the hand(s) to indicate the parameters of the operation. For linguistic gestures, the user makes a series of postures, to indicate a command to be issued to the system. For instance, an open hand posture, followed by a closed hand posture, followed once more by an open hand posture might indicate selection of an object under the hand.

In order to detect postures, the software system compares the input data from each of the bend sensors to the threshold for that sensor, to determine which fingers are bent, and which are not. Once this is determined, the system compares the data to the list of stored postures to determine which posture is currently being maintained.

To determine which gesture is then being made involves the use of a *feature tree*, such as the partial one shown in Figure 1. This method is based upon that of Jones et al. (1993).

For instance, if the user makes and maintains a grasping posture with one hand, the traversal takes the leftmost branch. Then, if while maintaining this posture, the user rotates their hand (i.e. a rotation movement), then the traversal takes the rightmost branch which indicates a rotate gesture. This shows interaction with the system through a physically-based manipulation. However if the user were to perform the same action in a linguistic gesture based fashion, the user would first make a grasping posture, again causing traversal to the left, then a rotate posture, which would again cause traversal to the left, finally leading to a rotate gesture. Therefor the same application can activate a gesture in a number of ways, using either form of interaction. While the list of postures available is part of the gesture recognition system itself, the feature trees must be created by the specific applications, to indicate which gestures they will respond to.

3 The Virtual Sculpture Application

In order to demonstrate and test the gesture control system an application had to be developed which made use of gesture as its primary method of interaction, and which allowed for both of forms of gesture which the system recognises to be used either separately or together. As gesture is generally used as a form of interaction for the manipulation of physical objects in the real world it was decided to create an application which allowed the user to create and manipulate virtual objects using gesture.

A number of systems have previously been created which allow the user to create and interact with virtual objects using gesture, and these systems have used a number of different methods for the creation of the 3-dimensional

Figure 1: A sample Feature Tree, linking postures and movements to form gestures

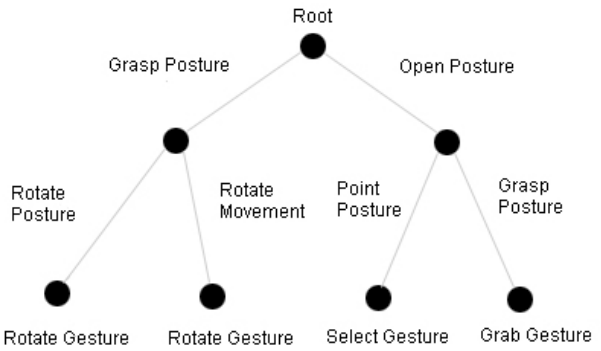


Figure 2: Two methods of forming the rotation gesture

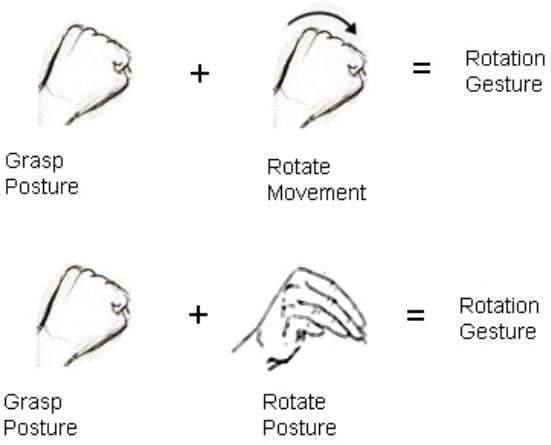
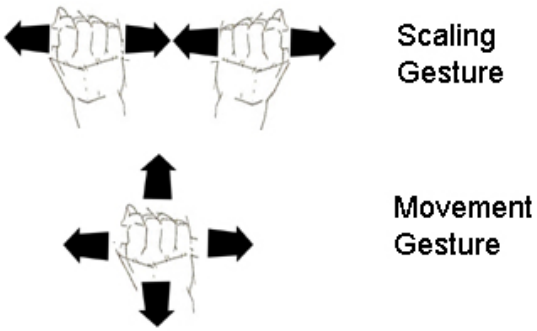


Figure 3: Movement and scaling gestures



objects by the user. The modelling of 3-dimensional objects using two-handed interaction was initially pioneered by Kruger (1993), whose VIDEODESK application allowed the user to control the shape of the objects being created using their own hands. The system made use of computer vision technologies to detect certain features, such as the positioning of the user's fingers and thumb.

A method to create freeform polygonal surfaces was shown by Shaw and Green (1997) which used two-handed interaction to create and manipulate control points on the shape through direct manipulation. A number of approaches to creating objects using *superquadrics* have also been proposed, such as Yoshida et al. (1996) which used a statistical method to calculate deformations of the shape from hand gestures, and Nishino et al. (1998) which allowed the user to create complex objects using two-handed gestures to perform blending and axial deformation of superquadrics.

Our Virtual Sculpture application makes use of a simpler form of object creation and manipulation. Rather than the use of superquadrics, or of meshes and control points, the system makes use of only a small number of geometric primitives, such as the sphere, cylinder, cone, torus and rectangular box. These objects are manipulated by simple axial transformations, which include scaling, rotation, and shearing. Objects can also be joined together to create a composite objects.

Objects are created by moving the hand over an icon representing the object at the top left of the display, creating a grasping posture, and "dragging" onto the display in the appropriate position.

Once an object is on the screen, it can be manipulated through a combination of physically-based manipulation and linguistic gestures. The system can be configured to use only linguistic gesture based controls, only physically-based manipulation controls or a mixture of both. This allows us to examine the use of the different forms of gesture to perform the same tasks.

When making use of the physically-based manipulation controls for the system, the user appears to act directly on the object. For instance, to rotate the object the user places a hand over the object and makes a grasping posture to indicate "grabbing" the object. The user then rotates their hand which causes the object to rotate through the same angles. When operating in a linguistic gesture based mode, the user indicates selection of an object by "grabbing" it as previously described, but then creates a number of different rotation postures, to rotate the object to the desired position. Figure 2 illustrates the two different approaches to rotating an object in the system. Some examples of other control gestures can also be seen in Figure 3.

3.1 Virtual Sculpture Installation

The installation for the Virtual Sculpture is a large, public space based installation. It makes use of a 100cm by 75cm display, which is built into a wall. The size of the display was chosen to allow for most people to comfortable reach the majority of the display area. The Virtual Sculpture application is then back projected onto this display. The gloves are placed on a small table in front of the display and connected to the wall of the installation. The user interacts with the system from a position standing in front

of the display.

The decision to use this form of installation was made for a number of reasons. The size of the display allows most users to be able to reach the majority of the display, and also allows for the objects created to be fairly large, making it easier for the user to interact with the objects. It also makes it easier to observe the user during the testing process. Finally, it allows for others to observe the objects being created by the user, perhaps allowing for collaboration between the user and those watching. While the user interacts with the system from a standing position in this installation, it would also be possible to lower the screen and allow the user to work from a sitting position. This was not deemed necessary for this particular installation, as it was felt that interaction with the system would not be for so long a time that the user would begin to feel tired.

4 System Testing

The main aim of creating the Virtual Sculpture application and installation was to test the gesture interface system which has been developed, and to evaluate the use of different types of gesture in an interactive application.

In order to achieve this, testing is being performed with a number of users, in order to evaluate the ease of use of the gesture control system, and to examine which forms of gesture are suitable for which tasks. The testing involves the performance of certain object manipulation tasks, first using only linguistic gesture based controls, and then using only physically-based manipulation controls. Finally, the user's are asked to configure the controls with the gestures which the find most suit each manipulation, and to perform the tasks a final time.

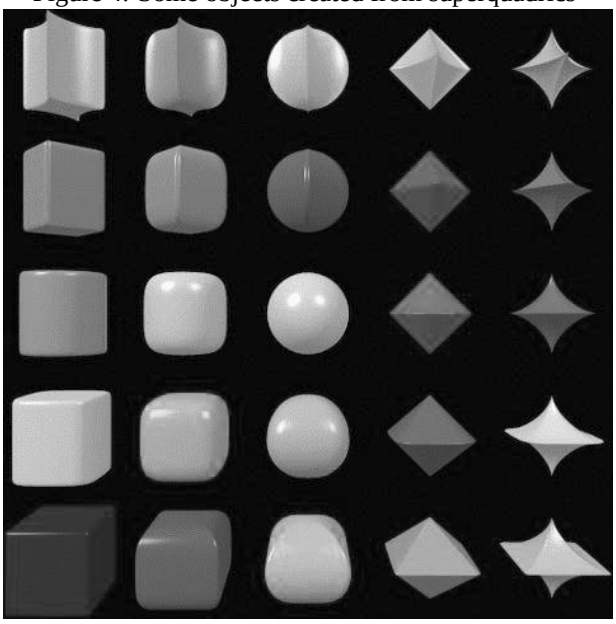
The task provided range from simple manipulations, such as creating an object and moving an object, to more complex tasks such as rotation only in one dimension by a set distance, to the most complex tasks which involve the creation of certain compound objects from the simple primitives provided.

By evaluating the results of this testing it will be possible to examine the suitability of a gesture interface to this form of interactive application, and also to determine whether user's find it most intuitive to use a physically-based manipulation gesture system, a linguistic gesture based system or a combination of each.

One strength of the system which has become particularly clear throughout its development and testing is that of the two-handed input. Performing manipulation of virtual objects using two-handed gesture is a very natural method of interaction for the user. This has been shown clearly from the comments of users as they interact with the application.

One manipulation which shows this in particular is that of scaling. Scaling an object using a two-handed gesture, where the hands are held in the grasping posture, and moved along the direction in which we would like the object scaled has proven to be extremely natural and no single-handed gesture has been found which feels as natural to the user. This bears out our initial reasoning behind the use of two-handed input, and also confirms the findings of Hauptmann (1990).

Figure 4: Some objects created from superquadrics



5 Further Development

The use of simple objects and transformations in the Virtual Sculpture system means that it is not particularly suited to the creation of intricate or detailed 3-dimensional objects, but rather to the creation of more simplistic objects. This decision was initially made to allow for concentration on the interaction with the system and on the use of gesture-based controls in particular.

In order to make the application more useful for the creation of 3-dimensional objects, and especially more useful as a form of artistic expression, a more complex set of objects will be required, along with a larger set of transformations and manipulations.

The use of superquadrics rather than the current simple geometric primitives would be one way of allowing the user to create more complex objects. Objects created in this way would allow for a greater control over the shape of the object, and would also respond to a larger set of transformations, including bending and twisting transformations which are not available in our system at present. See Figure 4 for examples of some shapes created from superquadrics.

However, to allow for the creation of really complex objects, an even greater degree of control might be required. In this case it may be necessary to develop the system to create objects from 3-dimensional meshes, so that particular points on an object can be manipulated, al-

lowing for the distortion of the surface of an object as well as transformation of the object as a whole.

For further developments in the area of the gesture control interface itself, it may become necessary to treat the input from the bend sensors on the gloves as an analogue value rather than as a binary switch value. While the decision to use the inputs as switches was made for the reasons given earlier, namely that of ease of use for those with limited mobility, and as it gave a rich enough vocabulary of gestures for many applications, it still provides us with a limit to the number of different postures which the system can detect.

Another improvement which has been suggested for the system would be the ability of the user to train the system to certain gestures, rather than have the gestures set only by the application. This enhancement to the system would allow a user to set their own control gestures, which would be useful in the case of gestures which were not felt to be natural enough for the user, and might also be useful for users of limited mobility, which was one of the initial ideas behind the development of the system. By allowing users with limited mobility to set their own gestures, we eliminate the possibility of the system making use of a gesture which the user is physically unable to perform, or which might in any way be uncomfortable for the user.

As well as reducing any discomfort for the user, the tailoring of the system to the users mobility may allow for easier interaction with the system by the users. Tailoring the motion input of a system to the user has proven in the past to produce an increase in the success rate of the users at manipulating the system, to the point of producing success rates for disable users which clearly overlapped those of non-disabled users for a large portion of their range (Pausch et al., 1992).

Finally, the addition of some form of haptic feedback to the system might result in some advantage. Some users commented on the lack of any "real feeling" when using the system, which might be accomplished by the addition of some haptic feedback, to indicate when the hand is over an object, and also to give some impression of grasping objects. This would increase the realism and therefor would be hoped to increase the naturalness of the interface for the user (Burdea, 1996). Haptic feedback has been shown to be a major integral part of many tasks, and Dai (1998) has shown that for the process of object prototyping, which is a very similar process to that used in our system, it is a necessity.

6 Conclusion

This paper presented a recently developed gesture interaction system which can be used as a main form of input to an interactive multimedia application. It also presented an application, called Virtual Sculpture, which demonstrates the use of the system as the main input into an interactive application, and which can be used to evaluate the suitability of the system as an input device, and also to examine the forms of gesture most suitable for use in these interactions. It discussed the use of this system for such an evaluation, and detailed an installation of the system in a public environment for use as a tool for artistic expression. Finally, some possible enhancements to the system were presented, which may be incorporated into a future version of the system.

References

R. Bolt and E. Herranz. Two-handed gesture in multimodal natural dialog. In *Proceedings of the 5th Annual ACM Symposium on User Interface Software and Technology*, pages 7–14, Monteray, California, USA, November 1992.

G.C. Burdea. *Force and Touch Feedback for Virtual Reality*. John Wiley and Sons, Inc., New York, USA, 1996.

F. Dai. *Virtual Reality for Industrial Applications*. Springer, New York, USA, 1998.

A. Hauptmann. Speech and gestures for graphic image manipulation. In *Proc. Of CHI89 Human Factors in Computing Systems*, pages 241–245, New York, USA, 1990. ACM Press.

M. Jones, R. Doyle, and P. O'Neill. The gesture interface module, GLAD-IN-ART deliverable 3.3.2. Technical report, Trinity College, Dublin, Ireland, January 1993.

M.W. Kruger. Environmental technology: Making the real world virtual. *Communications of the ACM*, 36 (7):36–37, 1993.

M. Marshall, B. Moynihan, and M. Rath. The vodhran - design and development of a virtual instrument. In *Proceedings of ICMC 2002*, Goteborg, Sweden, 2002.

H. Nishino, K. Utsumiya, and K. Korida. 3d object modelling using spatial and pictographic gestures. In *Proc. Of ACM Symposium on Virtual Reality Software and Technology*, pages 51–58, New York, NY, USA, 1998. ACM Press.

R. Pausch, L. Vogtle, and M. Conway. One dimensional motion tailoring for the disabled: A user study. In *Proc. Of SIGCHI Conference on Human Factors in Computing Systems*, pages 405–411, Monterey, California, USA, 1992. ACM Press.

C. Shaw and M. Green. THREAD: A two-handed design system. *ACM Multimedia Systems*, 5:126–139, March 1997.

D. Weimer and S.K. Ganapathy. Interaction techniques using hand tracking and speech recognition. In M. Blattner and R. Dannenberg, editors, *Multimedia Interface Design*, Frontier Series, pages 109–126. Addison-Wesley Publishing Company, 1987.

M. Yoshida, Y. Tijerino, T. Miyasato, and F. Kishino. An interface system based on hand gestures and verbal expressions that generate shapes for 3d virtual objects. *Tech. Report of IEICE, MVE95-64*, pages 33–40, 1996.

Oscillation
Work for saxophonist, 3D animation and real-time sound processing controlled by motion capture data

Franziska Schroeder[⋆†‡]
[⋆]School of Arts, Culture and Environment
The University of Edinburgh
franziska@lautnet.net

Pedro Rebelo[†]
[†]Sonic Arts Research Centre
Queens University Belfast
P.Rebelo@qub.ac.uk

Peter Nelson[‡]
[‡]School of Arts, Culture and Environment
The University of Edinburgh
P.Nelson@ed.ac.uk

Abstract

We address issues raised in designing a new work for saxophonist and digital media involving 3D motion capture technology. A saxophonist's kinaesthetics are recorded with a 3D motion capture system, and used as control data in the real-time processing of sound and 3D visuals. We address current gesture research and propose an extension to current gesture taxonomies. Linguistics has been taken as a basis for analysis of communication in musical processes, and serve as a fundamental reference in our discussion. The writings of Julia Kristeva, in particular her concept of dialectic oscillation between the semiotic and symbolic modalities, inform our approach to designing a system which articulates gestural intention with technological imprint. Notions of physicality and effort in the context of sound based performance prove useful in this research as the sophistication in processing that is associated with real-time audio-visual systems is often not complemented by gestural intention. By focusing on short performance segments we are able to test various mapping strategies that address the issue of relating control-spaces of different dimensions and qualities. The resulting connections between the performers gesture (as recorded by the motion capture system) and the sonic output are a product of the character of each individual segment and its context within the work. The work is designed using the MAX/MSP/Jitter graphic-programming environment (Cycling74, 2004).

1 Introduction

There is currently a high level of research activity being carried out in areas such as responsive computer systems and real-time interactive data processing. The artist-user is thus challenged to engage with processes and tools suggested by technological development[1]. Performance practice has always relied heavily on technologies that promote communication and spectacle. Technologies such as musical instruments are often responsible for interfacing artists and audiences. Further, these technologies, articulate the way in which the body performs; they modulate, resist, and stimulate body gesture by establishing two-way non-hierarchical systems. The notion of "the body as inscriber, and not just transmitter; simple receiver" (Barthes, 1977) the body that inscribes while being affected by the process of inscription itself, informs our practice. The issue of the performers physicality and effort in playing a musical instrument, and the transformation of that activity into data which in turn can be employed for performance interaction, are at the centre of this investigation. Intrinsic layers of expression are literally deconstructed in this project as body gesture and performative intentions are "reduced" to digital data - lists of Cartesian coordinates that correspond to markers tracking kinaesthetic relationships. This digital translation of the performer herself retains an inherent bodily aspect. This data becomes a significant element in the performance situation, as it is not only reflected in visual elements (3D form manipulation), but also controls/modulates/interferes with human-computer interaction. In this research project we employ existing systems and software packages such as an industry standard 3D motion capture system and the MAX/MSP/Jitter graphic-programming environment. These technologies are used as a platform for creating a new work that explores gestural control of digital media content.

[1]see artists such as Stelarc, Sensorband, Michael Waisvisz, and Laetitia Sonami.

2 Oscillating the Semiotic and the Symbolic

A vast array of definitions regarding musical gesture, as well as manifold discussions on the meaning of such gesture exist. While it is beyond the scope of this paper to go into a detailed overview of gesture classification, a detailed discussion of such classification can be found in (Wanderley, 2001). We have taken as a point of departure P. Feyereisens and J.-D. de Lannoys definition which states that any movement or change in position of a body segment may be considered a gesture and that gestures are mainly actions before becoming means of communication (Wanderley, 2001). We are interested in looking at, and utilising a particular performers gestural vocabulary. This discussion arises out of the performers own practice as a saxophonist. Most performers would agree that producing a sound on any instrument involves at least two types of gestural activity: one that is integral for the production of sound and another that is in no direct relationship to the sonic output. Various people have discussed this idea (Wanderley, 2001), and these two main categories have been referred to in various ways. Wanderley speaks of effective movements: gestures that modify the instrumental properties; and of ancillary gestures: those that are not related to sound production. These two groups have also been entitled ergotic, haptic gestures: those that involve physical contact; and free, semiotic, naked gestures: those in which no physical contact is present (Wanderley, 2001). Orio, studying the performance gestures of a guitarist, speaks of the basic gesture that produces sound and of gesture nuances, those gestures that convey timbre information . On examining ones performance movements more closely it becomes evident that these two groups of communicative actions must be further extended. Delalande, studying the performance gesture of Glenn Gould, attempted such extension by differentiating three types of gesture, namely effective gestures: those involved in sound production, accompanist gestures, such as head movement and figurative gestures: those perceived by the listener; and gestures that have no direct relation to a body movement (Wanderley, 2001). However, a further extension in order to allow for micro-gestural information to be included in the communication process is a necessity. We want to look at a performer producing a single sound event in order to clarify states in our performers communication channel. Those states that consequently turn into means of communication are subtle actions controlled by the performer. The way in which she emits, directly (visibly) or indirectly (invisibly) information is of interest to the viewer/listener, and it is this dichotomy that challenges the listener/viewer as he is required to translate the performers actions through his sense of hearing. In producing a single sound on the saxophone, we can clearly identify a preparatory phase in which the performer has to prepare mentally as well as physically. Mental preparation consists of readying the body for the type and time-span of the energy to come. While the mind knows what is to come, it needs to be assured of the actuality of an appropriate bodily attitude. In playing the saxophone, the physical preparation would be the inhalation of air in order to fill the lungs, the readying of the fingers and the forming of the embouchure. This anticipatory period precedes what has been entitled the effective or basic gesture, and we shall refer to this moment in the communication process as ”cosmetics”; cosmetics in the sense of the Greek word ”kosmein” - to arrange. The word cosmetic (”kosmos”) also refers to concepts such as ”people, universe, world” [www.kypros.org/cgi-bin/lexicon]. It is this stage of arranging oneself, the positioning of the fingers, the state of transferring oneself into a certain performance situation, the forming of an idea of how to address the listener, and communicating ones sound to the ”world”, an intention that up to the moment of the actual sound production solely exists in the performers mind and body. This is also a state of ”outside”, as the performer has not yet transferred her energy into the instrument; this is the moment that precedes the inscription of the body onto the instrument. This next level or stage we shall entitle ”ergotic” or ”muscular” [2].

In this phase ”the body controls, conducts, coordinates, having itself to transcribe what it reads, making sound and meaning, the body as inscriber and not just transmitter, simple receiver” (Barthes, 1977). Therefore this phase is about muscular activity and energy. It occurs when force, in the case of the saxophonist, breath and finger energy, is applied. Although the listener clearly perceives the motion of the performer in the ”cosmetic” phase and is therefore actively engaged, in the ergotic state, the listener will only perceive the strength of the force by the sound output that follows; hence the listener engages only on a passive level. In this state ”outside” merges into ”inside”, the breath travels through the instruments, the embouchure tightens and the fingers have to be on the ”right” keys for the particular note to sound. This state is followed by what we shall refer to as an ”epistemic” level. Cadoz states that this function is performed by the capacity of touch. The saxophonists touch refers to fingers on the key, breath control and the intricate positioning of the tongue. This level is exclusive to the performer herself. The key pressure of the fingers will depend on what sound is to be produced. Although it might seem that differing key pressure will not inform or modify the sonic output, such varying pressure is an inevitable action as the body mass of the performer inscribes itself onto the saxophone. It is therefore perceivable that a strong attack will be preceded by stronger pressure on the keys. Continuing in the gestural channel the resulting state in which the sound can be heard we shall refer to as the ”semiotic” state. Semiotic, as it is

[2] For the study of hand gestures, Cadoz proposes three functions in which ergotic is the first. He refers to it as a state in which no communication of information, but solely energy between hands and object is conveyed (Wanderley, 2001)

the state in which the intended communication, the sound, takes place and is finally perceived by the listener. The communication channel however does not end here. We propose that in order for signification to take place, the semiotic should be followed by the ”symbolic” modality. The usage of the terms semiotic and symbolic is informed by Julia Kristeva’s writings in ”Revolution in Poetic Language” (Kristeva, 1984). Linguistics have been taken as a basis for the analysis of communication in musical processes, and the works of Efron (Efron, 1942) and Kendon (Kendon, 1981) for example serve as a fundamental reference in current gesture discussion. The term semiotic is understood as in its Greek meaning of ”distinctive mark, trace index, precursory sign, proof, engraved or written sign, imprint, trace, figuration” (Kristeva, 1984). Kristeva denotes the semiotic as the bodily drive associated with rhythms, tones and movements of the signifying practice; the element of meaning within signification that does not signify. The symbolic is associated with the grammar and structure of signification. Signification requires both, the semiotic and the symbolic; the semiotic giving rise to, and challenging the symbolic. The relationship between the two elements she calls dialectic oscillation. Kristeva points out that, although music belongs to a nonverbal signifying system that is constructed exclusively on the basis of the semiotic, no signifying system can be either exclusively semiotic or exclusively symbolic (Kristeva, 1984). It is this concept of oscillation between the two modalities that we consider of great importance in designing our work. In creating the work we adhere to the idea of the semiotic and symbolic informing each other. It is therefore that we abstain from interpreting the semiotic modality. In order to inform the symbolic, we abstract the semiotic content into 3D motion data, thereby imposing a certain grammar onto our work. We argue that in the current discussion of gesture analysis and gesture application to sonic process, the need for the link and oscillation between these modalities has been overlooked. It seems that a variety of works rely on the interpretation of the semiotic, inferring from it in order to derive the symbolic element. Thus, one might encounter works in which an upward arm movement is interpreted and mapped as a rise in pitch. It is through such linearity that the complexity of performed sound has not been adequately addressed.

3 Physicality and Effort

> . . . it is through the physical that time is integrated with other musical components. That is: effort binds time to the measure of control (Ryan, 1996).

Whereas the performer has a relative certainty about the musical gesture which will result from her chosen physical gesture in parallel, the listener must perceive these gestures and relate them to a sonic event - the lack of physicality in computer-based sound processing systems makes necessary the development of some performance-based gesture input. Hence the potential of encoding a performers motion through 3D motion capture, and then digitally decoding and mapping this data onto other platforms, provides a powerful tool for generating creative and musically significant human computer interaction. Mapping physical performance gesture onto exterior processes (e.g. sound generation/manipulation parameters) implies an analysis of the performers playing effort, which is transferred to 3D motion capture data, processed and then converted into control parameters for the sound processing system. Further, the mapping of these gestures onto a real-time sound and visual processing environment can provide links between the instrumental performer, computer technology and the listener/viewer. While computer-based sound processing tools offer extremely sophisticated ways of dealing with real-time situations from the point of view of sound generation and manipulation, we lack appropriate control interfaces. We constantly struggle with trying to create musical gestures, limited by basic human-computer interaction interfaces such as the mouse and keyboard. In this project, the performers gestures are deconstructed into digital data which in turn is fed back into a visual/sonic control system that itself interferes with/modulates the performers sound; hence, strongly linking the human-computer interaction process into the following cycle: physical input data capture - data reconstruction data mapping data output modulation physical/data output. Haptic sensation in playing an instrument and its intrinsic motor control are tightly coupled. Transferring the entity of the highly personalised gestural language of the saxophonist onto a digital platform simultaneously imbues the sonic/visual output with a distinct bodily and gestural connection, while acting as an interactive feedback system for the performers sonic output. The listener is subsequently invited to participate in reconstructing the connections between the physical context, the sound and the visual components of the performance.

4 Music and Gesture

The role of the visual in musical performance is at the centre of this project, and it is an issue whose problematic nature extends and intensifies when new technologies are involved. The traditional field of musical performance configures the relationship between the performer and the instrument in a way which renders the spectacle in a way we might call sufficient but not necessary. That is to say, in a sensory structure where the visual dominates over the auditory, it is easy to swamp the senses with visual information and to render the auditory input subliminal. This is clearly what happens in film when the soundtrack prompts but does not dominate the visual imagery. In musical performance it is possible to close your eyes without losing track of the narrative thread. But even with the

eyes open, the visual domain always only prompts the auditory; the auditory remains at the perceptual focus. There are clearly thresholds here which alter the field when they are crossed, thus convention judges a showy performer, who pushes the visual imagery beyond the threshold, in a certain way, as it judges a film soundtrack which is insistent. This suggests a more sophisticated discussion, but the outline is clear for the present project. New technologies problematise this field in two ways: in the first, the semiotic system relating the interaction between performer, instrument and sonic output is not only not always conventionally understood; it is not always apparent. In the second, the intrusion of additional visual elements, video let us say, can threaten to swamp the visual sensory domain and render the auditory domain peripheral. These two difficulties become compounded in electroacoustic music, where the lack of any visual field seems to impoverish the communicative nexus, yet its presence is compromised by the lack of any necessary semiotic link between the sounds and any visual representation whatever. As a performance, "Oscillation" seeks to shed some light on these issues, by linking instrumental performance, generated audio and generated visual images, in a certain relationship. The musical text is composed of a succession of materials, of different sorts, which bear certain significant relationships to one another: thus there are musical gestures whose performance is recorded in motion capture but whose original sound is never heard, there are musical gestures which are performed live but where the bodily movement of the performer is directed to be like a recorded gesture, there are musical gestures which sound like the result of a recorded motion gesture but which are directed to be performed differently (e.g. fast notes played with as little player movement as possible) and so on. Whatever the relationship between the live performer and the sonic output, there is always a clear gestural relationship between the processed sound and the motion-capture images, a play of causality between live and processed sound.

5 Motion Analysis and Visualisation

The use of a studio-based system such as an industry standard facility for 3D motion capture[3] (typically used for recording human movement, that is consequently mapped to the motion of 3D character animated entities) raises several issues in the context of instrumental performance. Once performative action is fragmented by virtue of the studio environment, which by definition involves segmentation, repetition, and an audience-less performance condition, the result is a range of self-contained events. The performer plays a sonic event; consequently the captured data requires to be re-constructed into what a human eye sees as a plausible movement. What results is an assemblage of objectified tracking modulated by the subjectivity of the human eye. As a set of marker data reaches the editing environment, an operator needs to manually identify the position and relative structure of each marker in relationship to the standard human skeleton. The resulting gesture data, far from being an objective representation of performative action, is a careful reconstruction of both machine and human-based knowledge systems. The vulnerability that is present in this process informs subsequent visual and sonic processes.

5.1 Nurb Surface

Starting from the derivation of a set of control markers present in the motion capture data, we define the control matrix for a nurbsurface[4]. This consists of a polygon in which each marker defines a vertex. The standard use of motion capture data relies on structured polygon definitions which connect nodes to bones, bones to limbs etc., creating a character representation of the marker set (biped). In our case, the intention to visualise gestural movement without specifically referring to a humanoid figure was realised with a system that, while referring to the trajectory of each independent marker, uses the 3D data to control the shape of an arbitrary surface, rather than the individual bone-based structure of a biped[5]. If the starting point of a nurb-surface is a 2-dimensional grid, then the movement of the marker-based polygon acts as a force vector, which applies contraction and expansion to that grid. A complex smooth object might be defined in such a way that the movement of one of the control points might either drastically change the shape of the entire object or subtly mould a small portion of the surface. Iterative application of modifiers in a curve's control points often produces results which are analogous to organic growth, gradual deformation or fluid forms. This process has some similarities to those described in the work of biologist D'Arcy Thompson, who applied linear and non-linear functions to pictures of living organisms on a grid. His method allowed for the transformation of pictures of baboon skulls into the skulls of other primates or humans (Thompson, 1917). The rendering of a human-derived polygon movement as a dynamic, smooth three-dimensional shape allows for the visualisation of gestural events without necessarily referring to the encoding of the

[3] A Motion Capture System such as that used at EdVEC (Edinburgh Virtual Environment Centre) relies on a circle of 8 infra-red cameras which track a set of reflective markers attached to a human body. These 8 video signals are then processed to create a three-dimensional mapping of the movement of each marker (x, y, z values) from which a skeleton can be derived.

[4] Non-Uniform Rational B-splines are a mathematical model for representing arbitrary curves and surfaces. The shape of a NURB surface is determined by the position of a set of points (control points). Some control points can affect a larger region of a curve than others (hence Non-Uniform) and some points can affect the curve more strongly than others.

[5] For works that make use of a biped derived from 3D Motion Capture technology, see example such as Stelarcs Movatar (Stelarc, 2004) and Merce Cunninghams dance work Biped (Cunningham, 2004).

human body in such a gesture. The result is a high-level abstraction, which transmits parameters such as tension and release, movement stasis, balance and body weight. Once the control polygon for the nurb surface is populated with three-dimensional motion capture data, it can be used to render nurbs of various degrees; this variation in degree results in a range of objects in 3D space that reach from a simple line to a complex meshed surface. The level in which the nurb degree is defined relates to the resolution with which the control polygon is interpreted. The notion of resolution and sampling in the visual rendering of the surface offers a useful parameter in which a higher resolution suggests a more direct recognition of human movement. This is however challenged when a high resolution/high degree nurb surface is reduced to a simple polygon consisting of three or four lines, and, by being able to discern parameters clearly derived from human body movement, still remains recognisable as human.

6 Motion Analysis and Synthesis Mapping

Motion capture data is used to define and control parameters for Resonator Bank synthesis (CNMAT resonators object for MSP). This consists of parallel banks of two-pole resonators mapping frequency, gain and decay rate into filter coefficients (Jehan and Dudas, 1999). Motion capture data is analysed in order to deduce a small number of parameters, which correspond to perceived change in the visualisation of the data. This analysis was done by observing video recordings of the performance together with the movement of the nurbs controlled surface. In the particular case of the performer playing a soprano saxophone and with the given gestural content, certain kinetic aspects proved to be more relevant than others from the point of view of relationship between a gesture and a sonic result. The key relationships identified are:

1. Relative distance between the two elbows on one axis. The distance between the Left Elbow and Right Elbow markers represents an expansion and contraction of body volume; something highly perceivable in performance, particularly in relation to preparation, anticipation and negotiation between states of breathing and states of blowing. Comparing the pattern of this data with amplitude values in the audio recording of a corresponding segment, one can detect a relationship in which the distance is indirectly proportionate to the amplitude. A common case is the increase of the distance in preparing an attack or loud note; as the note actually sounds, the distance decreases.

2. Relative distance between the saxophone bell and the pelvis marker on three axes. Video recordings of the performance revealed that the space between the instrument and the performers body represents a sophisticated relationship with clear and immediate correspondence in the sonic output. Observing this particular parameter suggested that at some level a relationship between the bell-pelvis-distance and finger placement on the saxophone exists.

3. The overall perception of how much the body (and the instrument) actually move represents a general level of performance activity/effort, which is important in its relation to the sonic output. As with the distance between the two elbows, one can draw a parallel between general body activity (measured by means of displacement of all markers) and density in the sonic output. Observing our particular performer one can perceive a pattern, whereby ”large-scale” body movement often takes place when no actual sound result occurs; for example, preparing a long note implies a large breath intake and preparation of body tension that force the performer to such extensive body movement. The intention of this project was not to emulate the sound of the actual performance but to explore relationships which problematise connections between our learned expectation in body gesture and a sonic result. We have mapped the above parameters to a relatively simple synthesis environment based on the MSP object resonators (Jehan and Dudas, 1999). The motion capture analysis parameters are used to control a bank of resonators, which derive their spectra from analysis of the saxophone itself. In the work there is a combination between pre-stored spectra of particular notes, multiphonics, timbral events, and spectra, which are captured live from the saxophonists output. One of the possible mapping models attempts at inverting some of the perceived relationships explained above. Hence general values of body and saxophone displacement are mapped to general amplitude envelope (the original displacement values are scaled and smoothed). The peaks in the displacement value provide attacks, which take their amplitude from the elbow distance, such as the larger the distance the louder the attack. The distance between the saxophone bell and the pelvis is used to control frequency relationships, i.e. values along the axis that point away from the body (x) are scaled and mapped to a frequency shift of the resonators spectrum (although this data is continuous, values for frequency shift are taken only at each attack). The distance on the y- and z-axes are scaled and mapped to the spectral corner and slope of a shelving EQ spectral envelope.

7 Conclusion

We have developed a work entitled ”Oscillation” that maps the gestural vocabulary of a saxophonist onto sound and 3D animation processes. By using a 3D motion capture system we were able to recognise and analyse the movement activity of a specific performer playing the saxophone. The resulting 3D data is used to control the shape of a complex visual surface. While referring to the trajectory of each independent motion capture marker, it proved that the resulting polygon appeared as a dynamic,

smooth three-dimensional shape closely reminiscent of a human body in motion. Although the resulting visuals are a high-level abstraction of the performers movements, we were able to successfully transmit parameters such as tension and release, movement stasis, balance and body weight without having to refer to the individual bone-based structure of a biped. The 3D motion capture data is employed for the definition and control of parameters for Resonator Bank synthesis. For the purpose of our work, the relationship between the visual gesture and a generated sonic event was a focal point for exploration. Through analysis of the movement data in relation to the sound produced by the saxophonist, patterns of reversal seem to emerge (i.e high levels of movement equate with low levels of sound generation and vice versa). We took this reversal further in developing a mapping model that generates sonic events directly triggered by gestural activity. In that way, a high level of performer energy, usually not resulting in sound output, was mapped to a high level of sonic events. A review of current gesture literature led us to propose an extension to existing taxonomies of gesture. We suggest Julia Kristeva's notion of oscillation between the semiotic and symbolic modality as central to an understanding of gesture in the context of performance, as well as pivotal for the design of new works that deal with mapping of gestural data.

References

R. Barthes. Image music text. *Fontana Press, London*, 1977.

M. Cunningham. Biped. *http://www.merce.org*, 2004.

Cycling74. Max/msp/jitter. *www.cycling74.com*, 2004.

D. Efron. Gesture, race and culture. *The Hague: Mouton*, 1942.

A. Jehan, T. Freed and R. Dudas. Musical applications of new filter extensions to max/msp. *ICMC Beijing China ICMA*, 1999.

A. (ed.) Kendon. Nonverbal communication, interaction and gesture. *Mouton Editor, New York, The Hague. Paris*, 1981.

J. Kristeva. Revolution in poetic language. *New York: Columbia UP*, 1984.

J. Ryan. Some remarks on musical instrument design at steim. *Live Electronics. Contemporary Music Review. Harwood.*, 6(1), 1996.

Stelarc. Movatar. *http://www.stelarc.va.com.au (as of March 2004)*, 2004.

D'Arcy Thompson. On growth and form. *Cambridge University Press*, 1917.

M Wanderley. Performer-instrument interaction: Applications to gestural control of sound synthesis. *Doctoral Thesis - University Paris 6*, 2001.

Acknowledgments

This project was supported by the AHRB (Arts and Humanities Research Board) Small Grant in the Creative and Performing Arts. We would like to acknowledge the Edinburgh Virtual Environment Centre (EdVEC), and particularly thank Wendy for supporting us with the post-processing of the motion capture data.

On Webcams and Ultrasonic Anemometers:

applications of touchless sensors in the white cube context

Malte Steiner
Germany
steiner@block4.com www.block4.com

Abstract

This paper introduces some practical applications of two different input devices which I have used in exhibitions. Both have in common that they enable touchless motion detection. Interactivity is an essential part of my installation work: its absence would make hard to justify the use of computers. In cases where only video and/or sound is featured, it is easier, more reliable, and cheaper to install a DVD player. Even generative software art could be simulated so long as it doesn‘t react on any outside input.

My intention is to create virtual playgrounds that employ artistic processes. The viewer can interact with the exhibition in a relaxed and easy way. Key is the accessibility or, let me borrow the term from the software engineering, usability of the installation. Although my work can be cryptic, it should still remain easy to enter without being introduced in any way. The goal is to hold the visitor and let him/her understand immediately that he or she is a part of the installation. I saw a lot of computer art gems which are not discovered because the interface is too twisted or, in other cases, too mundane (i.e. a mouse) -- which is not accepted either because of people’s associations with office work. So there is a need for alternative input devices, and a lot can be done.

One concern is that audiences tend to abuse these controllers. They ‚‘work the interfaces as hard and fast as possible (Ulyate and Bianciardi 2002). That is another argument in favour of touchless sensors that can‘t be damaged. Of course, they also produce another artistic meaning, introducing an ethereal aspect that should be considered in the artistic development process.

1 Webcam

1.1 Introduction

The webcam is widely available in two flavours, USB or Firewire connectors. These low-cost webcams are not only for internet use. In fact, the input is pretty standard and can used in different applications. I used them to create interactive installations, using Max/MSP/Jitter and Pure-Data/Gem multimedia programming environments. Both competing packages are rather similar and can process video data with the Jitter or Gem add-ons. Even motion-detection is possible with both. While Jitter processes all data in a matrix, Gem translates the OpenGL library into the visual programming space of PD.

1.2 Captured‘: first study in security techniques

The first installation I did, and have shown in several places since 2002 called ‘Captured’ and sports a big screen with the unaltered picture of the webcam -- nothing particular so far. The Maxpatch grabs moving parts from the big picture and displays them on the sidebar which holds 4 of these snapshots. These are replaced when new parts are captured.

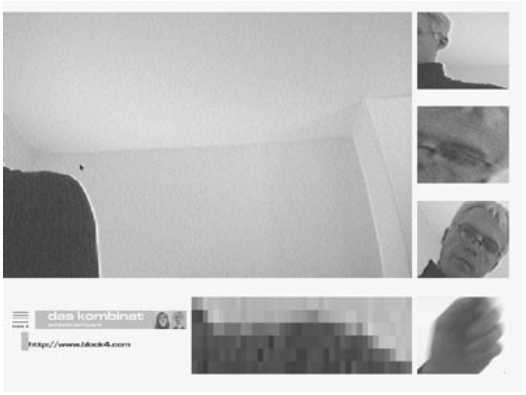


Figure 1 Captured

I decided that the detection process is interesting enough to be exposed below the big image. It shows how algorithms try to find enough motion to pass a certain threshold. This threshold is necessary because these cheap webcams introduce, due the CMOS technology, a lot of noise. This can irritate the system and lead to unsatisfying artistic results. I recommend CCD cams which do, however, raise the budget a bit.

This first installation deals a lot with the security concerns in that year, and how similar technology was discussed in several antiterrorism campaigns. The

snapshots give unexpected details and, once the audience caught onto the system's operation, they started to play with it. This patch totally lacks audio output but watching people playing with it gave me the further idea of incorporating sound into it, creating a virtual touchless instrument.

1.3 ‚Zeitfragment', a virtual instrument

First I worked over the visual output and discarded the various displays. One big picture is fragmented into several pieces, mirroring detected motion.

Figure 2 Screenshot of Zeitfragment

The next step was to create an audio engine. Having the idea of an harp in mind, I chose a Karplus-Strong synthesis approach, where the pluck is modeled with an FM-algorithm. It doesn't necessarily mimic a harp correctly because I never was a supporter of the idea of imitative synthesis. The use of 2 basic synthesis principles allows great control of the sound. My motion-capturing patch gives me 3 parameters: x,y and the amount of movement. I routed y to frequency of the fm burst. The amount influences the fm operators´ frequency ratio, and thereby the tone colour. Small motions create a sharp sound while more generous strokes give deeper ones. In fact, I created 2 waveguides which are positioned Left and Right in the Stereo field. I routed x to a panning algorithm so that the FM pluck is distributed around the delay lines. The pluck is always triggered when a motion is detected and resonate the guides, generating a polyphonic effect.

This was first shown at the end 2003 at the Ausklangfestival in Hamburg. The installation was projected while the cam detected the picture. An on-location problem was the balance between getting a good picture from the projector and getting one from the camera: the projector prefers a darker room while the cam loves light so it was a question of clever lightning. I am going to continue the research with another solution by having infra-red lights enhancing the scenery, an idea which was proposed on the Pure Data mailing list recently. It might be good for the motion-detection process, but it has been tested relative to its effects on the quality of the image. It may also produce new and interesting artefacts.

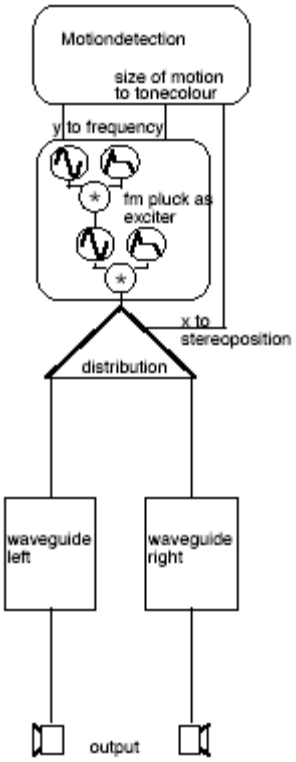


Figure 3 Audioengine of Zeitfragment

1.4 ‚Gespenster', ectoplasmatic artifacts in the digital domain

While I was programming the application for Zeitfragment, I incorporated the subtraction of the current image from the preceding one to let just the difference remain. For the developing and debugging process, I let Max show me the results instead keeping them internal, and was impressed by the artistic quality of the artefact. It somehow creates a ghosting effect, some ectoplasm following of the audience's motions. That leads to my latest webcam-installation called Gespenster, German for Ghosts. Created with PD and Gem, it shows a haunting black and white picture --black is dominant in it particularly when nothing is moving. When there is something moving, a white shadow follows it. I further enhanced the result by creating more ghosting pictures with the use of a visual delay.

The legend has it that ghost phenomenon are a memory of past fates which are connected to certain places. I saw a thematic connection with computer memories. Sonically, I wanted to create a sound-memory algorithm, which stores the whispers and noises of the audience when there is something occurring. I did something similar in my 2002 installation Maschinenraum. The

audio engine of Gespenster detects if there is an input and samples it when it supersedes a certain threshold. The stored sound is replayed through a phase vocoder, so I have control of the frequency, position in the recording, and duration independently. Once again, I mapped x to the stereo position, which is easy to recognize by the audience. I keep the frequency unaltered (which is not possible with standard sample-replay approaches) and mapped the y value of the center of motion to the part in the sample which can be heard. Output is triggered when movements occur, and is processed by a reverb, giving additional supernatural room information. So the audience can sweep through the recording or stay put at a certain point of the recording.

Figure 4 Screenshot of Gespenster

2. Anemometer, it's in the air

Anemometers are devices to measure wind. Mechanical ones incorporate the windmill principle but there are further approaches. One is the super sonic anemometer which lacks any moving parts. That makes them fairly robust, for instance the one I got for research and development courtesy of the Metek company has passed a rough time on Mount Washington in New England.

Let me quote the product description:

> "Short pulses of ultrasonic sound are exchanged in three different directions by couples of sound probes which are used alternately as transmitting and receiving units. The sound probes are mounted in aerodynamically shaped housings providing a significant reduction of flow distortion.
>
> The sound velocity derived by travelling time of the ultrasonic pulses is composed of the sound propagation of the motionless air itself, i. e. the wind speed parallel to the trajectories of the ultrasonic pulses. Combining the sound velocities of different propagation directions, the 3-dimensional wind vector can be determined. Furthermore, the sound velocity in a motionless atmosphere is derived which corresponds to a measurement of the virtual temperature.
>
> The sound velocity depends not only on the temperature, but also on humidity. Therefore, the measured temperature represents the virtual temperature which is needed for most investigations of atmospheric stratification." (Metek 2003)

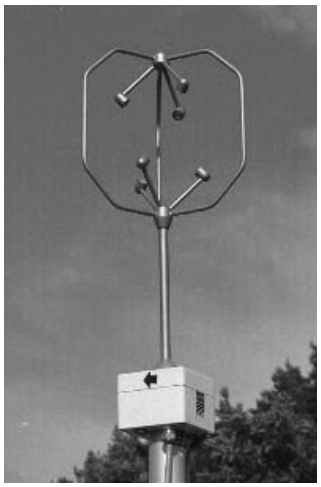


Figure 5 ultrasonic anemometer

This device is to be attached to the computer through a serial interface. It delivers signed x,y,z values where the sign gives the direction. In addition, the temperature is transmitted too. The sampling frequency can be altered up to 25 Hz, so it can transmit 25 value bundles in a second. Of course, it measures constantly, so between each delivery it averages the measured data. This data, even the numbers, are delivered in ASCII code. Adjustments can be done by sending AT commands to the device through the serial interface.

In collaboration with Ulrich Raatz and the meteorologist Dr. Andreas Pflitsch I created an interactive sound/video installation which we exhibited for the first time in Hamburg in February 2004. My task was to develop and create the sonic part of the installation. Our discussions resulted in machinery with four sound engines, each randomly selecting a sample out of a pool of acoustic sounds and replaying it with random frequency, duration and stereo position. The pitches and the durations are influenced by one anemometer. In the exhibition there are several cabins, each equipped with computer, speakers and a sensor. Additionally, videos by Raatz where shown and in later version should be also altered by the measurement data.

It took us a week to record and cut loads of samples from acoustic instruments. The choice was driven by the idea to keep the sound somehow organic and not distant. We recorded steel drums, blown organ pipes, Orff

Instruments, etc. and organised them into four folders for our voices, one with low pitched sounds, one with middle-low, one with middle-high, and finally one of the high tuned sounds. Each voice got access only to one of these folders. To create these voices, I used granular synthesis to again detach pitch from duration.

The nastiest part, in my opinion, was creating the reception and parsing of the incoming data. First I needed to have a serial to USB converter because even my not-so-current Mac doesn't bear legacy serial interfaces, all are replaced by USB. After obtaining a working product, the first real pitfall was to parse the information. The anemometer delivers ASCII code bytes which needed to be transformed to at least a string. These strings where not aligned so the data couldn't be determined based on its position in the string. I was close to launching my c compiler and creating a custom external, but I resolved it with a regular expression for which there is a module in Max/MSP Jitter.

The first hearing of that patch reveals that there was too much random and not enough interactivity. So I changed it by removing the random generators from the pitch and duration. With a correct offset, the anemometer satisfyingly controlled the soundscape, even for people not involved in that project. Although we did not intend to create a virtual instrument like Zeitgfragment, we wanted to provide certain interactivity in the constant flowing soundscape.

Mapping is the key in this installation as I learned when we brought the stuff for a first test in the exhibition room. Raatz arranged a first show in a gallery in an abandoned building in Hamburg which currently houses several ateliers and a exhibition room in February 2004. That place was rather windy in December and the soundscape went into the red immediately, unlike at my place. It became necessary to change the effect of the sensor and I was lucky that I had implemented an adjustable multiplier and an adder for providing an offset for each parameter. Because the multiplier can also work with fractional factors, it can turn into a divider to lower the impact. I received x,y,z and temperature. X and z were mapped to the frequency, y to the duration, temperature to both. This mapping has proven to be the most effective one in the field test.

Figure 6. An offset/multiplier pair in Max

References

Ryan Ulyate and David Bianciardi (2002). "The Interactive Dance Club: Avoiding Chaos in a Multi-Participant Environment", Computer Music Journal, 26(3):40–49.

Product description on Metek's website, http://www.metek.de

Das Kombinat, an installations, http://www.block4.com/kombinat.htm

Geteilter Blick, the anemometer work: http://www.zierbaustelle.de/

on Anemometers inclusive DIY, http://thiesclima.com/usanemo_e.htm

PD, an open source media programming environment: http://www.pure-data.org

EyeCon – a motion sensing tool for creating interactive dance, music and video projections

Robert Wechsler
Doncaster College School of Intermedia and Performance Arts
Doncaster, England
robert@palindrome.de

Frieder Weiß
Palindrome Inter-media Performance Group
Nürnberg, Germany
frieder@palindrome.de

Peter Dowling
Independent Composer
Glasgow University
Glasgow, England
pdowling@fish.co.uk

Abstract

EyeCon is a video-based motion sensing system which allows performers to generate or control music and projected images through their movements and gestures in space. It was developed Frieder Weiß as an application-driven project of the Palindrome inter-media performance group and is thus more a tool for artists than for researchers. EyeCon does, however, allow motion sensing according to a variety of movement parameters and lends itself to experimentation with media mapping.

Unique and largely unexplored problems face composers, graphic artists and choreographers as they collaborate on interactive performance works, not the least of which is settling on schemes for mapping the various parameters of human movement to those within the world of sound and image. Among the myriad mapping alternatives, good design choices are paramount to creating effective interactive performance works. The authors have made progress in understanding the special issues involved and are devising strategies for these choices.

1 Introduction

The authors' work is an investigation into the perceptual relationships of human motion to sound, and, to a lesser extent, those of human motion to video projection art in which the motion of the performer determines or influences the secondary medium. Through nine years of collaborative work creating interactive performances, the authors are able to draw certain conclusions concerning the nature of interactive performing, i.e. what makes it artistically tenable and what does not. Some of these conclusions relate to the choices of media, some to the process of collaboration and still others to the choices of mapping. These are summarized in the form of tips for performers wanting to work with interactive systems.

Mapping is the process of connecting one data port to another, somewhat like the early telephone operator patch bays. In our case mapping has a very specific connotation—it means the applying of a given gestural data, obtained via a sensor system, to the control of a given sound or video synthesis parameter. The dramatic effectiveness of a dance, however, invariably depends on myriad factors—movement dynamics of body parts and torso, movement in space, location on stage, direction of focus, use of weight, muscle tension, and so on. And although sensors may be available to detect all of these parameters, the question remains: which ones to apply in a given setting, and then to which of the equally numerous musical or visual parameters they should be linked.

2 Motion Sensing and Analysis

The primary motion-sensing system the authors use is the EyeCon software. It is a camera-based motion sensing system developed by the two principle researchers for this project, Robert Wechsler and Frieder Weiss, especially for stage and installation art work as an on-going project of the Palindrome Inter-Media Performance. The first version of EyeCon appeared in 1995.

EyeCon permits movement to control or generate sounds, music, text, stage lighting or projectable art. It is adaptable to an enormous number of applications, lending itself to experimentation in genuinely new directions in performing and installation art. It offers a way to create interactive video environments without the need to get into graphical or script-based programming. This means, that people without special computer skills are able to use it.

Although EyeCon has been used by a number of dance and theater companies, singers and performance artists, Palindrome is its dominant user. For a overview of some of the work Palindrome has created with EyeCon (including video samples) visit Palindrome's homepage at www.palindrome.de.

2.1 Terminology

The terms motion tracking, motion capture, motion recognition, motion analysis and motion sensing are used variously and with overlapping applications. Motion capture is a technique developed by the film industry for creating more realistic movement of animated characters and while the technology has found numerous other uses, it has generally not been possible or practical to use it in live performance settings. The data collection process involves cameras completely surrounding the performance area and the performers must wear highly noticeable reflective markers. Even *if* a performer were willing to perform with these accoutrements, the fact remains that motion capture systems have, until very recently, not been capable of rendering the data into graphic images in real time. The frist major are, however, now underway to do this by Paul Kaiser, Trisha Brown, Bebe Miller and others at Arizona State University's "Intelligent Stage" lab, as a

part of the "motione" project[1].

The term *motion tracking* also fails to hit the mark for the authors' work. Or, better said, it would seem to imply only limited aspects of human motion, namely where a person is located on the stage and the speed and direction they are traveling. This information is not only limited in its expressive potential -- audiences are much more concerned with what a performer is doing than where he or she is located -- but it is also a particularly poor choice of parameter for reasons of transparency. Because a performer cannot jump instantly from one part of the stage to another, the transitions involved are slow. This makes the interactivity far less convincing since it could easily be simulated (faked, as in a technician following the performers motion with mouse movements). Indeed, it may simply appear that the performer is following the media in instead of the other way around!

2.2 How the Motion sensing Takes Place

A video signal is fed into the computer and the video image appears on the computer screen. Using the mouse, lines or fields in different colors are superimposed onto the video image. When a performer, through their video image, touches one of these lines, or moves within a field, a media event is triggered or modulated, for example, a certain sound might be heard. Eyecon works by comparing the individual pixels of two different video frames and analyzing them for differences in brightness or color. The difference between these two frames is time. That is, they are the same scene, but one is the present time, and one in the past. In some cases, there may be only 0.04 seconds between the two, but when applied to a body part in motion, this is easily sufficient time to allow the determination of motion. Generally speaking, not all of the pixels in the two images are compared, but rather only those marked off by EyeCon's elements.

2.3 EyeCon's Sensing Elements

EyeCon provides an assortment of different ways to sense motion. To understand the need for this diversity, let us look for a moment to the expressive potential of movement. Human movement in general, and dance in particular, is perceived as a collection of movement features -- parameters if you will. That is, human movement is not one quantity, but rather a great many, which are weighted very differently in our perception depending on their artistic intent. In one case, it may be overall speed of the mover which speaks to us, while in another, we are concerned with the precise positioning of the limbs. Thus if a motion sensing system is to be of use to artists, it must be capable of sensing a variety of different movement parameters. Without this palate, and the skill to use it effectively, the dance becomes a slave to the system. To be effective, the movement must be choreographed to *fit* the technology, instead of the other way around.

The systems by which EyeCon senses movement are represented through graphic structures called Elements. The Elements are superimposed onto the live video image in the Video Window where they can be scaled and manipulated. The Elements are then assigned their individual properties such as the volume control of a particular sound or changes in a projected image.

TOUCHLINES. Touchlines are lines drawn on the video image in the computer. these act as triggers for the presence or absence of body parts or objects, but can also be scaled, so that different places along the line can have different effects. In this sense they provide an easy way to track position along a line.

DYNAMIC FIELDS. Dynamic fields are boxes which represent fields and respond to movement dynamic. They can be used to trigger sounds and images or to control the volume and pitch of sound files so that, for example, the faster the dancer moves, the louder the sound or the higher the pitch.

FEATURE FIELDS. Feature fields begin with the same boxes as Dynamic Fields, but provide formal analysis of objects within the field. For example, a Feature field might measure the dancer's overall size (how expanded or contracted they are), or how close two dancers are to each other. They can also analyze shape (width compared to height), height or degree of left-to-right symmetry. Finally, there are a set of controls for sensing the direction of movement, so that a step to the left, for example, will sound different than a step to the right, reaching up different than reaching down, etc.

POSITION TRACKERS. Position trackers track the mean position of one or multiple persons as they move around the video image. This means, if you have an overhead camera, you can track the location of persons as they move within a room and thus the environment can be made to respond to each person differently. Touchlines and Dynamic fields can be attached to the tracker, so that a given array of controllers can move with the performer as they move around the space. Finally, this feature permits color-specific tracking, presenting the possibility of distinguishing between dancers by the color of their costumes.

EyeCon is often used to control secondary programs either running on the same machine as EyeCon, in another computer via data link or network. The additional computer does not, of course, need to be a PC. External software and hardware which can be controlled by EyeCon include software and hardware synthesizers, as well as programs like Director, MAX/msp, Reactor, and Isadora. The data link can be via Ethernet and use MIDI or OSC protocols.

3 The Eyecon User Interface

Figure 1 shows the user interface of EyeCon 1.60.

[1] http://ame2.asu.edu/projects/motione/.

Figure 1.

In contrast to many interaction-oriented digital media systems, EyeCon is not freely programmable, i.e. does not work like a programming language. Rather it is based on the fixed architectural elements described above. Rather than creating entire environments with highly accurate and specialized properties, EyeCon is conceived more in terms of offering dynamic control -- both spatially (movable shapeable elements) and temporally (sequencer functions). The result is a relatively intuitive system for performers and by its nature, relatively easy to operate. Artists with little or no computer experience can use it.

The EyeCon software can be divided in two completely different parts: motion sensing and multimedia action(s). The concept of EyeCon is that you create motion sensing elements which are in many cases graphically represented as lines and rectangular fields. The Element Editor is the main mapping panel where you assign how movement controls media.

EyeCon has two operating modes, *Simulation* and *RUN* mode. In RUN mode that actual video analysis is performed and media is played. In simulation mode you can use tiny boxes and move them with the mouse to emulate moving objects in the video image. The Control window allows you to control the main parameters of the video processing like camera selection, threshold for movement detection, basic timing etc.

The Video window allows you to watch the live video and see how the motion sensing elements react.

To allow the creation of complex interactive setups, EyeCon offers a scene management function. Each sensing element is functional in a selectable level. The Sequencer Window allows you to arrange these interactive setup levels.

4 Interaction as an Artistic Phenomenon

Regarding interactive performance works, one could speak of four levels of interactivity: One is the purely technical level, that is, if the system successfully accomplishes the conversion of one media to another (for example, if a dancer's movement does indeed generate a sound). The second level is whether the performers are aware that their movements are affecting their environment. The third level comes when a system functions in such a way that the audience is aware of the interaction (without having it pointed out). The fourth level is when the audience members themselves cause changes to occur in media, for example, through their movements in their seats.

Some have argued that a successful interactive performance work requires only that the second level interaction be reached. The reasoning is that since the performer is aware of the effect they are having on their environment, then this will invariably affect the way they perform. And thus the audience, even with no clear understanding of what is happening, nevertheless feels the art work in a different way. While believing that this point has some merit, the authors of this paper rather see this as a cop out. If an art work and its technology are prepared carefully enough, then the interactivity can be accessible to an audience directly and this the greatest artistic potential.

4.1 The Psychology of Interaction

There is another aspect to interaction which, in a sense, supercedes the question of who, from a technical or perceptual standpoint, interacts with whom. Like wolves and other primates, humans are an extremely interactive species. They tend to clump together in groups, spending inordinate amounts of time speaking, gesturing, touching and otherwise communicating with one another. Its what we are *not* doing now. If you were to write us back a letter, we may touch on this quality, but we don't really start interacting until we sit down together and hash it out. That we do with such relish belongs to the most primitive of human instincts.

Human beings have been dancing and making music for 10,000 years. During most of this long history, performances were highly interactive -- much more so than they are today. The distinctions of "performer" and "audience", and even those of "musician", "dancer", etc. were far less clear. Everyone was part of the event. There are still today examples in Africa of traditions for which the same word is used for both dance and music[2]. Participants fed off of each other's energy in a way which is seen today only in such settings as night clubs (the good ones). Jazz music provides perhaps a last bastion of interactive performing in the West.

Beginning with the predominance of the bourgeoisie, theater in Europe saw a closing off of interaction between performer and audience. With bright lights on one side of a proscenium, and a darken area with seats on the other, the audience's role was pretty much reduced to sitting quietly and then clapping before going home. This has not changed much in the last two hundred years.

Ironically perhaps, modern technology is a major culprit. Recording and sampling techniques have meant

[2] The *Awa* of the Dogon tribes (Guinea Coast) offers one example of many.

that musicians, for example, often work separately. Pop music relies heavily on sampling ("stealing with respect", as it is known among DJs) rather than creating sounds from scratch or playing musical instruments.

Dancers and musicians, meanwhile, rarely work directly with one another anymore. Only a tiny fraction of dances performed today directly involve musicians in any part of the actual stage production: creation, rehearsal or performance.

But the biggest interactivity-buster of all is surely the projection screen. Not only did video further reduce the need for dancers and composers to work together creatively, but of course one doesn't even need to be part of an audience today to watch a performance. Just turn on the television.

Palindrome has a piece called *Publikumsstück* ("audience piece") in which 10 audience members are brought backstage during the intermission. There, they are taught ingredients for a structured improvisation -- essentially given tasks to accomplish with one anther -- as well a crash-course in interactive performing. After the intermission the piece is performed within an interactive stage environment so that different audience members control different sounds with their movements. A woman approached us after the show and commented that she liked the part that "we were involved in". She was not one of the ten. She meant *we the audience*.

This woman's reaction points out a fundamental principle of how interaction works. It is very much a "feeling thing" -- a subjective, rather than objective phenomenon. In some cases, small amounts of participation by an audience can utterly change their experience of a performance event. On the other side, giving the audience many things to do may have little effect on their sense of interactivity. I.e. their sense that they were *part of* the event depends on special factors.

Thus, whether we are speaking of interaction between artists, between artist and audience or between a person and a computer system, the same basic principles apply. They share psychological roots and in practice function in a similar way. In all cases, it is dependent on the performer being relaxed enough in their role to be able to respond genuinely, in a sense innocently, to what they are experiencing. Note, this may and may not involve improvisation on the part of the performer, at least not in the sense that the word is generally used by dancers and musicians. There must however be some degree of *play* in the performance. Each time must be different. *If a performance is utterly fixed, there can be no interaction.* It is the paradox of all good performing; it should look and feel spontaneous, even when it is carefully prepared.

5 Two Current Motion Tracking Based Projects Of The Authors

Two current and very different examples of applications are presented (with live demonstration). The first is an excerpt of a motion-to-sound piece, and the second, an excerpt from a motion-to-projected video image. These and similar examples are available at www.palindrome.de.

5.1 "ICE 9" (2003)

The real-time sound synthesis environment was designed in MAX/MSP. A PC running EyeCon is linked to a Macintosh PowerBook running MAX/MSP, sending the gestural data gathered by EyeCon to the real-time sound synthesis parameters.

The MAX/MSP program for "ICE 9", is a musical synthesis environment that provides many control parameters, addressing a number of custom-built DSP modules that include granular sampling/synthesis, additive synthesis, spectral filtering, etc. All mapping is accomplished within the MAX/MSP environment, and changes throughout the work.

Control of the musical score to "Ice 9" is accomplished through a cue list that enables/disables various EyeCon movement sensing parameters, mapping and DSP modules to be implemented centrally. Both EyeCon and MAX/MSP software components are organized as a series of "scenes", each describing a unique configuration of video tracking, mapping, and DSP. Scene changes for both computers are synchronized and can be initiated by a single keystroke from either station.

"Ice 9" is a music-dance performance/research work that applies choreographic evaluation methods and motion sensing technology to music composition issues. Of particular interest are certain qualities which may be applied to both music and dance and the application of these to multi-modal expression. For example, we have developed a motion sensing (motion tracking) system which responds to the direction of the dancers' motion. That is, whether the impulse, or overall movement tendency is to the left (vs. right), upwards (vs. downwards), and downstage (vs. upstage). In this way, 3 fundamental bi-polar tendencies can be identified and, importantly, easily perceived by the observer. Each of these *directional parameters* are mapped to a different bi-modal acoustic model which is applied to the real-time generation of sound.

The application represents the essential paradigm for us. Needless to say, the music will be influenced and inspired by the process and style in both gesture and in the transparent mapping of movement to music. However, the composer's role must not only be seen as concentrating on the *composition* of sound sources and the live processing of triggered materials (and of course their creative implementation with relevance to the choreographic situation), but also as determining ways in which the *form* of implementation – how we go about doing this -- can be creative in its own right. The selection of mappings, for example, is in itself a complex issue; choices must be made between multifarious possibilities with the criteria for selection being of a cross-disciplinary, and largely unexplored nature, i.e. involving both parameters and expression of human movement (dance) as well as those of sound (music). It must be understood that these choices are neither cosmetic nor trivial, but are crucial to the

transparency, ergo the effectiveness of the complete sound/movement experience.

In short, the interactive (and technological) process is on a par with the composing of the sounds and the creation of the choreography when thinking conceptually and artistically about the invention of the piece. Our performing experiences have shown that, generally speaking, when only *part* of a piece is transparent and convincing in its interactive relationships, then audiences tend to accept additional more complex relationships. They become 'attuned', as it were, to the functionality of the piece.

This instills in us a wish to negate the obvious and the transitory notion of mimetic dialogue. If we take electroacoustic musical discourse as being constituent of *aural* discourse – abstract musical content – and *mimetic* discourse – a complex of auditory, visual and emotional stimuli, we have a basis for multi-layering perceptions.

For example, *timbral* mimesis is the direct imitation of the timbre of a sound, whereas *syntactic* mimesis is the imitation of relationships between natural events, like the orchestration of speech rhythms (which *Palindrome* have approached creatively on numerous occasions in the past). In composition (and to an extent in choreography), both [aural] discourse and mimesis are always present in some form – a continuum exists between the two. So, in aural discourse we can always extract 'pure' musical elements, even from directly recorded natural sounds. In mimetic discourse we can always perceive (or imagine) some source of or – importantly in this collaborative case – *cause* of the sound(s).

To cross reference these concepts with the interdisciplinary issues at stake in mixing the dance (the visual stimulus of movement) with the music is essential. They are refracted through the interface of the technology and the processes of artist and audience interaction. In this way, there is generated a fascination with the creative *technique* as well the more obvious potential for multi-layering of perception generated from work of this kind.

5.2 "Ich, mich und mir" (2004)

The work applies the age old theater technique of a shadow play, only here it is combined with digital media -- specifically, an infrared light source, an infrared camera, motion sensing and real time image and audio signal processing.
What would happen if we could release our shadow image, follow it like in a dream, multiply it or face it?

Since the beginning of human consciousness, people have known their "virtual" companion: the body shadow. It follows us quietly. You can neither catch it, nor step across it. And while belonging to our body, it is in no way a part of it. It is always darker than ourselves. Some cultures believe it belongs to an emotional world. The shadow was thought to be the home of the soul. Who ever does not have a shadow is regarded as dead.

After many years of a tremendous hype about *the virtual experience,* most of us are left with disillusionment. We are learning that we can't live in a world that is disconnected from our physical and emotional realities.

Figure 2. "Ich, mich und mir.

Our performance piece is meant as a reminder of the organic connection between body-image and body-reality. Our theme is the shifting border between body and mediated virtual body image. Figure 2 shows a moment in the dance.

5.2.1 The Technology

Our shadows gain freedom from their source in the following way: In one corner of the space we put a light source which is throwing the shadow image on the large projection screen. This shadow, however, is of a type which the human eye cannot see. Because all of the visible light has been filtered from the light, only the infrared light is reaching the screen. A special infrared camera picks up the shadow image from the screen and feeds the digitized image into the computer where it is processed. A connected video projector makes the processed images visible, in fact they are projected on to the exact surface where the invisible shadow is located. (Video projector light, being low in infrared, does not interfere with the shadow we are filming).

The digital processing includes continuously variable delays, multiplication, transposition, coloring, reversing, accelerating/decelerating, freezing and dissolving.

Brain research has found that what humans experience as 'now' is actually a time band of up to three seconds. Our brain is constantly trying to sync our different senses, making predictions about what is most likely to happen, and then integrate the whole into a perception we call 'now'. How much can a shadow be delayed before we loose the sense that it is still "connected" to its owner? The computer shadows seamlessly shift between what we are used to and the unexpected. It is this play along the borderline between the *known* and the *surprising* makes the piece fascinating to watch.

6 Conclusion: Some Specific Suggestions for Interactive Performers

- Performers must be relaxed and open enough on stage so that their performance can be informed by the influence they are having on their environment.

- Map to multiple outputs. For example, you may wish to link a movement to a particular sound *as well as* to a visual element such as a stage lighting change or video projection element. Although this may seem obscure the correlation, because the mappings are parallel, it will have the have effect of reinforcing the connection.
- Think about camera angles. Choose one which helps the movements to be accurate and repeatable.
- The dancers (or moving performers) should tell the choreographers or technicians what they feel they need, instead of the other way around.
- Link media events to particular gestures. The way a movement becomes marked in the mind of the audience. I.e. use memorable, movements, those with character, even though technically there may be no advantage to doing it that way.
- Trigger or control the same events from the same stage positions or from the same body posture even though, again, this may be irrelevant to the system you are using.
- Look for intuitive mappings (higher body level-to-higher pitch, faster-to-louder, busier movement-to-busier sound, heavier movement-to-heavier sound, etc.). Artists tend to have great reluctance to do this for reasons which are not entirely clear to us.
- Near the beginning of the piece, or at least *during* the piece, use the system in a clear and transparent way. In this way, it can explain itself to the viewer. Having done this, the audience becomes sensitized to the interactive experience. They will then be attuned to and accepting of later, subtler mappings.
- Either before, or after the piece, explain to the audience what the technology is and how it works. There are as many good reasons to do this as there are not to, but it is an option. Some pieces don't need it, some don't want it, and others simply love it. Either way, whether you do it or not, I will guarantee you one thing: someone will come to you after the show and thank you from the bottom their heart for doing it, just as the next person in line will castigate you for the same thing.
- Program notes are not nearly as communicative as an announcement, yet may be far less intrusive to a work of art.

Acknowledgements

The authors would like to thank the dancers and choreographers with whom the works described in this paper were created: Emily Fernandez, Helena Zwiauer and Larry Hahn. Important background work concerning DSP was conducted together with Butch Rovan.

References

Wechsler, R., Weiss, F., "Motion Sensing for Interactive Dance", *IEEE-Pervasive Computing, Mobile and Ubiquitous Systems*, IEEE Computer Society publications, Jan-March 2004.

Rovan, B., Weiss, F. and Wechsler, R., "Seine hohle Form: a project report. Artistic Collaboration in an Interactive Dance and Music Performance Environment", *COSIGN 2001 -- 1st International Conference on Computational Semiotics In Games And New Media*, Amsterdam.

Quniz, E., Lovell, R., "Digital Performance", *Anomalie digirtal_arts 2002,* ATI Press, Rome, pps. 124-130.

Michel H., "Afrikanishce Tänze", DuMont Buchverlag, Cologne, 1979.

Dinkly S., Leeker, M., "Tanz und Technologie - auf dem Weg Zur Medialen Inszenierungen". Alexander Verlag, Berlin, 2002, pps. 196-201.