Perceptive Spaces for Performance and Entertainment (Revised) †

Christopher R. Wren Flavia Sparacino Ali J. Azarbayejani Trevor J. Darrell
James W. Davis Thad E. Starner Akira Kotani Chloe M. Chao Michal Hlavac
Kenneth B. Russell Aaron Bobick Alex P. Pentland

Perceptual Computing Section, The MIT Media Laboratory ; 20 Ames St., Cambridge, MA 02139 USA
{cwren,flavia,ali,trevor,jdavis,thad,akira,cchao,hlavac, kbrussel,sandy}@media.mit.edu
http://www.media.mit.edu/vismod/

September 10, 1999

Abstract

Bulky head-mounted displays, data gloves, and severely limited movement have become synonymous with virtual environments. This is unfortunate since virtual environments have such great potential in applications such as entertainment, animation by example, design interface, information browsing, and even expressive performance. In this paper we describe an approach to unencumbered, natural interfaces called Perceptive Spaces. The spaces are unencumbered because they utilize passive sensors that don't require special clothing and large format displays that don't isolate the user from their environment. The spaces are natural because the open environment facilitates active participation. Several applications illustrate the expressive power of this approach, as well as the challenges associated with designing these interfaces.

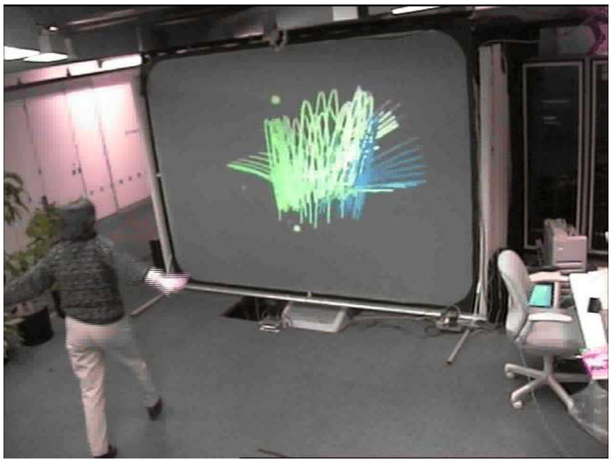


Figure 1: User dancing in a perceptive space and generating graphics.

1 Introduction

We live in 3-D spaces, and our most important experiences are interactions with other people. We are used to moving around rooms, working at desktops, and spatially organizing our environment. We've spent a lifetime learning to competently communicate with other people. Part of this competence undoubtedly involves assumptions about the perceptual abilities of the audience. This is the nature of people.

It follows that a natural and comfortable interface may be designed by taking advantage of these competences and expectations. Instead of strapping on alien devices and weighing ourselves down with cables and sensors, we should build remote sensing and perceptual intelligences into the environment. Instead of trying to recreate a sense of place by strapping video-phones and position/orientation sensors to our heads, we should strive to make as much of the real environment as possible responsive to our actions.

Very few remote-sensing technologies live up to these goals; humans have evolved to primarily use vision and audition as their sources of perceptual information. We have therefore chosen to build vision and audition systems to obtain the necessary detail of information about the user. We have specifically avoided solutions that require invasive methods: like special clothing, unnatural environments, or even radio microphones.

This paper describes a collection of technology and experiments aimed at investigating this domain of interactive

†Appears in: ATR Workshop on Virtual Communication Environments, April 1998, Kyoto, Japan. This is an updated version of the article that appears in Applied Artificial Intelligence, Vol. 11, No. 4, June 1997.

spaces. Section 2 describes some our solutions to the non-invasive interface problem. Section 3 discusses some of the design challenges involved in applying these solutions to specific application domains.

2 Interface Technology

While many advances have been made in creating interactive worlds, techniques for human interaction with these worlds lag behind. In order to allow a user to navigate a three dimensional space, most commercial systems encumber the user with head-mounted displays, electro-magnetic or sonic position sensors, gloves, and/or body suits [2]. While such systems can be extremely accurate, they limit the freedom of the user due to the tethers associated with the sensors and displays. Furthermore, the user must don or remove the equipment each time they want to enter or exit the environment. Some systems avoid this problem by passively or actively "watching" the user. These systems often modify the environment with specially colored or illuminated backdrops, require the user to wear special clothes, or involve special equipment like range finders or active floor tiles [16, 1, 26].

The ability to enter the virtual environment just by stepping into the sensing area is very important. The users do not have to spend time "suiting up," cleaning the apparatus, or untangling wires. Furthermore, social context is often important when using a virtual environment, whether it be for game playing or designing aircraft. In a head mounted display and glove environment, it is very difficult for a bystander to participate in the environment or offer advice on how to use the environment. With unencumbered interfaces, not only can the user see and hear a bystander, the bystander can easily take the user's place for a few seconds to illustrate functionality or refine the work that the original user was creating. This section describes the methods we use to create such systems.

2.1 The Interactive Space

Figure 2 demonstrates the basic components of an Interactive Space that occupies an entire room. We also refer to this kind of space as an Interactive Virtual Environment (IVE). The user interacts with the virtual environment in a room sized area (15'x17') whose only requirements are good, constant lighting and an unmoving background. A large projection screen (7'x10') allows the user to see the virtual environment, and a downward pointing wide-angle video camera mounted on top of the projection screen allows the system to track the user (see Section 2.2). A phased array microphone (see Section 2.4) is mounted above the display screen. A narrow-angle camera mounted on a pan-tilt head is also available for fine visual sensing. One or more

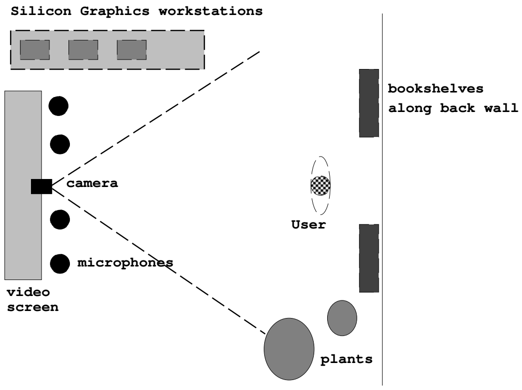


Figure 2: Interactive Virtual Environment hardware.

Silicon Graphics computers are used to monitor the input devices in real-time.[26].

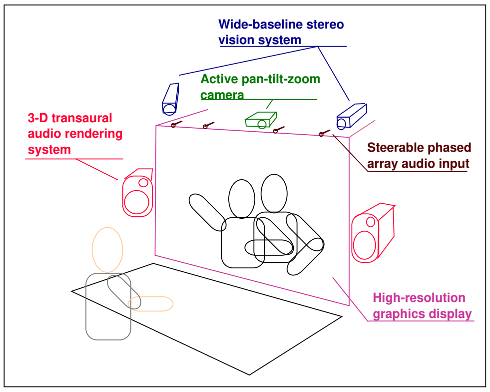


Figure 3: An Instrumented Desktop

Another kind of Interactive Space is the desktop. Our prototype desktop systems consist of a medium sized projection screen (4'x5') behind a small desk (2'x5'—See Figure 3). The space is instrumented with a wide-baseline stereo camera pair, an active camera, and a phased-array microphone. This configuration allows the user to view virtual environments while sitting and working at a desk. Gesture and manipulation occur in the workspace defined by the screen and desktop. This sort of interactive space is better suited for detailed work.

2.2 Vision-based Blob Tracking

Applications such as unencumbered virtual reality interfaces, performance spaces, and information browsers all have in common the need to track and interpret human action. The first step in this process is identifying and tracking key features of the user's body in a robust, real-time, and non-intrusive way. We have chosen computer vision as one tool capable of solving this problem across many situations and application domains.

We have developed a real-time system called Pfinder[32] ("person finder") that substantially solves the problem for arbitrarily complex but single-person, fixed-camera situations*(see Figure 4a). The system has been tested on thousands of people in several installations around the world, and has been found to perform quite reliably.[32]

Pfinder is descended from a variety of interesting experiments in human-computer interface and computer mediated communication. Initial exploration into this space of applications was by Krueger [16], who showed that even 2-D binary vision processing of the human form can be used as an interesting interface. More recently the Mandala group [1], has commercialized and improved this technology by using analog chromakey video processing to isolate colored garments worn by users. In both cases, most of the focus is on improving the graphics interaction, with the visual input processing being at most a secondary concern. Pfinder goes well beyond these systems by providing a detailed level of analysis impossible with primitive binary vision.[32]

Pfinder is also related to body-tracking projects like Rehg and Kanade [24], Rohr [25], and Gavrila and Davis [14] that use kinematic models, or Pentland and Horowitz [22] and Metaxas and Terzopolous [20] who use dynamic models. Such approaches require relatively massive computational resources and are therefore not appropriate for human interface applications.

Pfinder is perhaps most closely related to the work of Bichsel [7] and Baumberg and Hogg [5]. The limitation of these systems is that they do not analyze the person's shape or internal features, but only the silhouette of the person. Pfinder goes beyond these systems by also building a blob-based model of the person's clothing, head, hands, and feet. These blob regions are then tracked in real-time using only a standard Silicon Graphics Indy computer. This allows Pfinder to recognize even complex hand/arm gestures, and to classify body pose (see Figure 4b)[32].

Pfinder uses a stochastic approach to detection and tracking of the human body using simple $2\frac{1}{2}$-D models. It incorporates *a priori* knowledge about people primarily to bootstrap itself and to recover from errors. This approach allows Pfinder to robustly track the body in real-time, as required by the constraints of human interface.[32]

We find RMS errors in pfinder's tracking on the order of a few pixels, as shown in Table 1. Here, the term "hand" refers to the region from approximately the wrist to the fingers. An "arm" extends from the elbow to the fingers. For the translation tests, the user moves through the environment while holding onto a straight guide. Relative error is the ratio of the RMS error to the total path length.

For the rotation error test, the user moves an appendage through several cycles of approximately 90 degree rotation. There is no guide in this test, so neither the path of the rotation, nor even its absolute extent, can be used to directly measure error. We settle for measuring the noise in the data. The RMS distance to a low-pass filtered version of the data provides this measure.

test	hand	arm
translation (X,Y)	0.7 pixels (0.2% rel)	2.1 pixels (0.8% rel)
rotation (Θ)	4.8 degrees (5.2% rel)	3.0 degrees (3.1% rel)

Table 1: Pfinder Estimation Performance

Pfinder provides a modular interface to client applications. Several clients can be serviced in parallel, and clients can attach and detach without affecting the underlying vision routines. Pfinder performs some detection and classification of simple static hand and body poses. If Pfinder is given a camera model, it also back-projects the 2-D image information to produce 3-D position estimates using the assumption that a planar user is standing perpendicular to a planar floor (see Figure 4c)[32].

2.3 Stereo Vision

The monocular-Pfinder approach to vision generates the $2\frac{1}{2}$-D user model discussed above. That model is sufficient for many interactive tasks. However, some tasks do require more exact knowledge of body-part positions.

Our success at 2-D tracking motivated our investigation into recovering useful 3-D geometry from such qualitative, yet reliable, feature finders. We began by addressing the basic mathematical problem of estimating 3-D geometry from blob correspondences in displaced cameras. The relevant unknown 3-D geometry includes the shapes and motion of 3-D objects, and optionally the relative orientation of the cameras and the internal camera geometries. The observations consist of the corresponding 2-D blobs, which can in general be derived from any signal-based similarity metric.[4]

We use this mathematical machinery to reconstruct 3-D hand/head shape and motion in real-time (about 10 to 15 frames per second) on a pair of SGI Indy workstations without any special-purpose hardware. In tests similar to those used with pfinder (see Section 2.2), we find RMS errors on the order of a few centimeters or degrees, as shown in Table 2. The translation errors are larger than the corresponding translation errors in the 2-D case because estimation along the Z axis is a mathematically ill-conditioned

*Use of existing image-to-image registration techniques [3, 19] allow Pfinder to function in the presence of camera rotation and zoom, but real-time performance cannot be achieved without special-purpose hardware.

Figure 4: Analysis of a user in the interactive space. Frame **(left)** is the video input (n.b. color image possibly shown here in greyscale for printing purposes), frame **(center)** shows the segmentation of the user into blobs, and frame **(right)** shows a 3-D model reconstructed from blob statistics alone (with contour shape ignored).

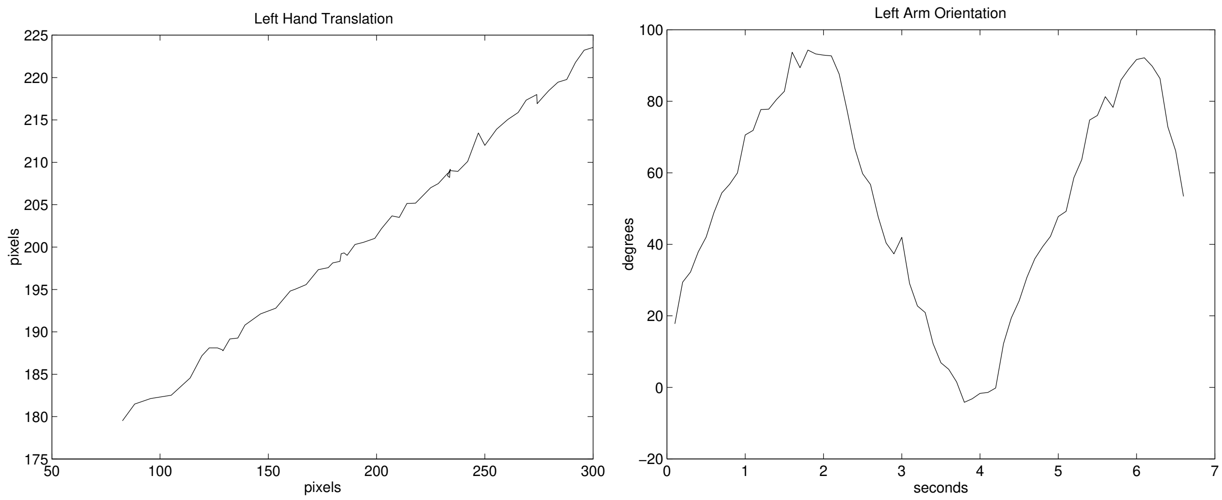
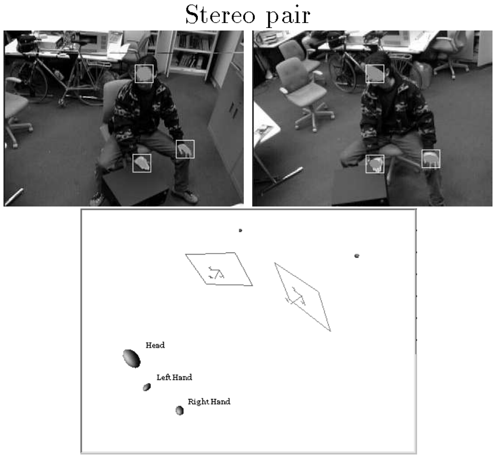


Figure 5: **(left)** shows data from hand tracking while the hand was slid along a straight guide. **(right)** shows a similar experiment for rotation

Stereo pair



3-D estimate

Figure 6: Real-time estimation of position, orientation, and shape of moving human head and hands.

test	hand
translation (X,Y,Z)	2.55 cm (1.8% rel)
rotation (Θ,Φ,Ψ)	1.98 degrees (2.2% rel)

Table 2: Stereo Estimation Performance

problem.

This stereo information is used by client applications much the same way the 2-D tracking is used: either as direct input to an interface application, or as input to a gesture recognition layer.[4]

2.4 Visually Guided Input Devices

Robust knowledge of body part position and body pose enables more than just gross gesture recognition. It provides boot-strapping information for other methods to determine more detailed information about the user. Electronically steer-able phased array microphones can use the head position information to reject environmental noise. This provides the signal-to-noise gain necessary for remote microphones to be useful for speech recognition techniques [9]. Active cameras can also take advantage of up-to-date information about body part position to make fine distinctions about facial expression, identity, or hand posture.[10]

3 Perceptive Spaces

Unencumbered interface technologies do not, by themselves, constitute an interface. A mapping must exist between the input technology and the system to be manipulated. This mapping must be carefully chosen, because it defines the metaphor that the user is forced use when they interact with the system. The desired level of abstraction, tolerance to interface accuracy and lag, even the prior expectations of the user must be taken into account when designing this mapping.

This section describes several systems that have been built in our lab, each with a distinct interface/system mapping. The focus will be on these interface mappings: how they work with the interface technology, and also how they affect the interactive experience.

3.1 SURVIVE

Figure 7: The user environment for SURVIVE.

The simplest mapping is, of course, the direct one: map interface device features directly (one-to-one) into the control space of some application. Usually a small amount of filtering will be required, and possibly it's desirable to use non-linear mappings, but otherwise interface outputs feed directly into application inputs.

SURVIVE (Simulated Urban Recreational Violence Interactive Virtual Environment) is an entertainment application that uses a direct mapping. SURVIVE allows the user to interact with a 3D game environment using the IVE space. Figure 7 shows a user in SURVIVE. The gestural interpretation provided by the vision system (Section 2.2) is mapped into the game controls for the popular id Software game Doom.

The user holds a large (two-handed) toy gun, and moves around the IVE stage. Position on the stage is fed into Doom's directional velocity controls. The hand position features are used to drive Doom's rotational velocity control. The results of a matched-filter on and audio input stream provide control over weapon changes and firing. This direct mapping, given the application, may be called "user-as-joystick".[26]

Although simplistic, this mapping has some very important features: low lag, intuitive control strategy, and a control abstraction well suited to the task. The mapping requires little post-processing of the interface features, so it adds very little lag to the interface. Since many games have velocity-control interfaces, people adapt quickly to the control strategy because it meshes with their expectations about the game.

Finally, it's insightful to contrast the SURVIVE interface with the standard keyboard Doom interface. The task in Doom is navigating through a virtual environment. This is usually accomplished by pressing keys on a keyboard. Changing the direction of travel is as easy as picking up one finger and pressing down another. Split-second decisions become split-second actions. The SURVIVE interface is much less forgiving. Movement of the virtual body is linked to the movement the real body. A change of virtual direction actually requires a movement in that direction, maybe several feet of movement. This leads to a much more engrossing, visceral experience of the game.

Interestingly, even when people use the keyboard interface, they tend to move their heads, and sometime their whole body, while playing the game. SURVIVE capitalizes on this natural link between visual and visceral experience to create a more immersive, if more physically demanding, experience.

3.2 Visually-Animated Characters

A literal mapping is one that treats the tracking features as exactly what they are: evidence about the physical configuration of the user in the real world. In this context the tracking information becomes useful for understanding simple pointing gestures. With quite a bit more work, systems can use this information to estimate a more complete picture of the user's configuration.

Figure 8: A synthetic character taking direction from a human user who is being tracked in 3-D with stereo vision

Complex 3-D characters can be built up and rendered using high-speed graphics rendering hardware, but they tend to lack natural coordinated movement because animators have to move joint angles individually. This problem is often solved using "motion-capture" systems in which a user is instrumented with accurate sensors to measure the locations and angles of joints whose dynamic trajectories are used to animate corresponding locations and angles of joints on the character (see Figure 8).

In a perceptual space instrumented with multiple cameras, the same procedure can be done passively with vision systems. We have implemented a system in which the stereo system described in Section 2.3, is combined with a literal mapping between user configuration and corresponding parts of an animated character.

The system allows the user to animate the 3-D head and hand movements of a virtual puppet by executing the corresponding motions in the perceptual space. The features from the vision system drive the endpoints of a kinematic engine inside the puppet.

3.3 City of News

A gesture-based interface mapping interposes a layer of pattern recognition between the input features and the application control. When an application has a discrete control space, this mapping allows patterns in feature space, better known as gestures, to be mapped to the discrete inputs. The set of patterns form a gesture-language that the user must learn. It is worth noting that this kind of rigid gesture-language tends to be sensitive to failures in tracking, classification, and user training. Systems that employ this kind of mapping must have very flexible, and quick, mechanisms for resolving misunderstandings. See Sections 3.5 and 3.6, for interesting answers to this problem. City of News is an example of an application that uses a gesture-based mapping.

City of News is an immersive, interactive web browser that makes use of people's strength remembering the surrounding 3D spatial layout. For instance, everyone can easily remember where most of the hundreds of objects in their house are located. We are also able to mentally reconstruct familiar places by use of landmarks, paths, and schematic overview mental maps. In comparison to our spatial memory, our ability to remember other sorts of information is greatly impoverished. City of News capitalizes on this ability by mapping the contents of URLs into a 3D graphical world projected on the large DESK screen. This virtual world is a dynamically growing urban landscape of information which anchors our perceptual flow of data to a cognitive map of a virtual place. Starting from a chosen "home page" - where home is finally associated with a physical space - our browser fetches and displays URLs so as to form skyscrapers and alleys of text and images through which the user can navigate. Following a link causes a new building to be raised in the district to which it belongs, conceptually, by the content it carries and content to be attached onto its "facade".

By mapping information to familiar places, which are virtually recreated, we stimulate association of content to geography. This spatial, urban-like, distribution of information facilitates navigation of large information databases, like the Internet, by providing the user with a cognitive spatial map of data distribution. This map is like an urban analogue to Yates' classical memory-palace information memorization technique.

To navigate this virtual 3D environment, users sit in front of the SMART DESK and use voice and hand gestures to explore or load new data. (Figure 9). Pointing to a link or saying "there" will load the new URL page. The user can scroll up and down a page by pointing up and down with either arm, or by saying "up/down". When a new build-

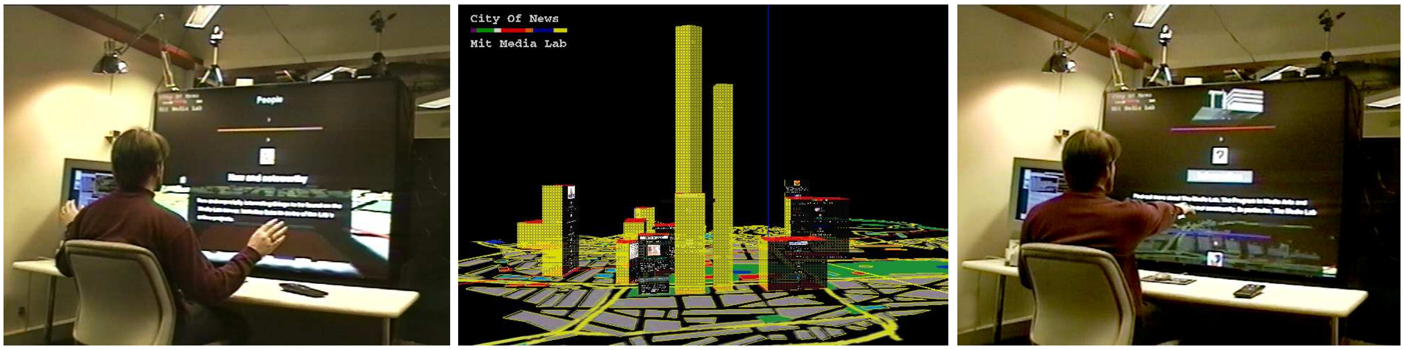


Figure 9: City of News.

ing is raised and the corresponding content is loaded, the virtual camera of the 3D graphics world will automatically move to a new position in space that constitutes an ideal viewpoint for the current page. Side-pointing gestures, or saying "previous/next", allows to navigate along an information path back and forth. Both arms stretched to the side will show a full frontal view of a building and its contents. Both arms up drive the virtual camera up above the City and give an overall color-coded view of the urban information distribution. All the virtual camera movements are smooth interpolations between "camera anchors" that are invisibly dynamically loaded in the space as it grows. These anchors are like rail tracks which provide optimal viewpoints and constrained navigation so that the user is never lost in the virtual world.

City of News follows NetSpace [31] and Hyperplex [30], our first experiment using IVE as an immersive browser for movies. The browser currently supports standard HTML with text, pictures and MPEG movies. City of News was successfully shown at the Ars Electronica 97 Festival as an invited presentation [28]. Future extensions include stereo browsing, with the use of Crystal Eyes glasses, and exploring more in depth the challenges and advantages of multimodal (speech and gesture) interaction techniques.

3.4 Virtual PAT and The KidsRoom

Two other very different examples of gesture-based mappings are the Personal Aerobic Trainer (PAT), and the KidsRoom.

We begin by describing a prototype system for PAT. In this system, the user is able to select which aerobic moves, which music, and which instructor they want for their workout session. The resulting program displays a virtual instructor which then guides the user through the workout while watching and commenting on the user's performance (See Figure 10).

The underlying motivation for building such a system is that many forms of media that *pretend* to be interactive are in fact deaf, dumb, and blind. For example, many of the aerobics workout videos that one can buy or rent present an instructor that blindly expels verbal re-enforcements (e.g. "Your doing great!") whether or not a person is doing the moves (or even is in the room!). There would be a substantial improvement if the TV just knew whether or not a person was moving in front of the TV. A feeling of awareness would then begin to be associated with the system.

To have the virtual instructor feel as if he/she is indeed watching the participant, a variety of perceptual mappings, or "levels of awareness" are needed. First, the system must know whether or not the person is in fact in the space. Upon startup of the system, it watches the space looking for a person. Only when someone enters the space and stands in front of the video screen for a few seconds does the program begin (passers-by do not trigger the system startup). Then if the person leaves the room prematurely (before the end of the workout session), the system can either pause waiting for the person to return, or terminate the program. Next, the system has been designed to look for movement of the person, and more importantly, whether the person is doing the correct exercise movement (the same as the instructor). This lets the instructor respond differently (and appropriately) to the user according to the current situation. Examples include when the user is not moving ("Get moving!"), moving but not performing the move correctly ("Follow me!"), or doing the move as expected ("Good job!"). In a future design of the system we wish to be able understand more of *how well* the move was performed (Similar to [6]).

All sensing for this environment is acquired through vision — wired body suits become quite encumbering when the person has to perform large scale aerobic movements around the space. The underlying representation used for all the vision tasks is the frontal silhouette of the person, extracted using a new process based on spectral selectivity [13]. By finding a large enough silhouette, we know that a person is inside the space (i.e. presence). Simple motion detection can be accomplished by image differencing these silhouettes in time (i.e. movement). Recent work has shown promising results in recognizing large-scale aerobic movements [11] from such input. This research uses temporally-collapsed motion templates to recognize various aerobic exercise (and other) movements in real-time. We therefore use the presence, movement, and recognition information from vision processes to drive the responses of the virtual instructor.

This system moves beyond the highly un-interactive media forms of video tapes and TV shows by having the system watch and respond to the user (instead of just the user watching the TV). Here, the user's actions are clearly reflected in the responses of the virtual instructor. A fuller description of the system is presented in [12].

A related environment, where the actions of the participant have consequence on the interactive experience, was the KidsRoom [8]. Designed in the spirit of Peter Pan, Bedknobs and Broomsticks, and Where the Wild Things Are, the KidsRoom was a fully-automated, interactive narrative playspace for children. Using images, lighting, sound, and computer vision action recognition technology, a child's bedroom was transformed into an unusual world for fantasy play (See Figure 11). Objects in the room became characters in an adventure, and the room itself actively participated in the story, guiding and reacting to the children's choices and actions. The children's positions and actions were tracked and recognized automatically by computer vision systems and used as input for the narration control. The vision techniques were tightly coupled to the narrative, exploiting the context of the story to determine both what needed to be seen and how to see it. Through voice, sound, and image the KidsRoom entertained and provoked the mind of the child.

Joint work with ATR, the SingSong system combines these ideas with an interval calculus that manages the progression of users through a predefined thematic script[23].

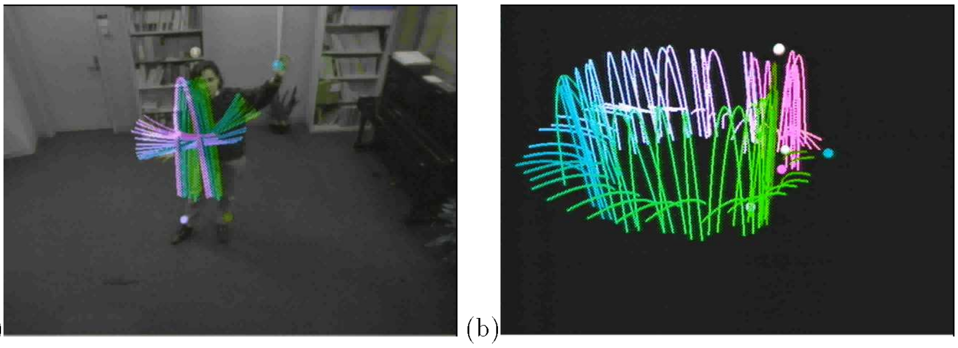
These systems are strong examples of interactive virtual environments which exploit the actions of the participant(s) to create a more naturally responsive and enjoyable experience.

3.5 DanceSpace

Closely related to the gesture-based interface mapping discussed in Section 3.3, the conductor-style interface mapping of DanceSpace also uses a form of predefined gesture language. The important difference lies in the design of that language. The gesture-language of City of News are rigid. Specific gesture sequences generate specific reactions, and conversely, failures in the tracking and classification of the user's actions can result in inappropriate actions by the system. The conductor mapping results in a much more fluid interface. The user can certainly try to explore the control space, learn it, and use it as a rigid language, but the system is designed to produce constructive, interesting results even when this doesn't happen. The interactions the user

Figure 10: Virtual PAT. A virtual personal aerobics trainer. Photo credit: Webb Chappell. Copyright: Webb Chappell 1998.

Figure 11: The KidsRoom: an interactive narrative playspace for children.



(a) (b)

Figure 12: **(a)** User dancing with her colored shadow in DanceSpace **(b)** Dancing shadow generated by the user in DanceSpace

has with this system are arguably more interesting when the user doesn't know the details of the mapping.

DanceSpace is an interactive performance space where both professional and non-professional dancers can generate music and graphics through their body movements [21] (See Figure 12).

The music begins with a richly-textured melodic base tune which plays in the background for the duration of the performance. As the dancer enters the space, a number of virtual musical instruments are invisibly attached to their body. The dancer then uses their body movements to magically generate an improvisational theme above the background track.

The dancer has a cello in their right hand, vibes on their left hand, and bells and drums attached to their feet. The dancer's head works as the volume knob, bringing down the sound as they move closer to the ground. The distance from the dancer's hands to the ground is mapped to the pitch of the note played by the musical instruments attached to the hands. Therefore a higher note will be played when the hands are above the performer's head and a lower note when they are near their waist. Both hands' musical instruments are played in a continuous mode (i.e., to get from a lower to a higher note the performer will have to play all the intermediate notes). The bells and the drums are on the contrary "one shot" musical instruments. When the performer raises their feet more than 15 inches off the ground then either of the bells/drums are triggered, according to which foot is raised.

The music that is generated varies widely among different users of the interactive space. Nevertheless all the music shares the same pleasant rhythm established by the underlying, ambient tune, and a style the ranges from "pentatonic" to "fusion" or "space" music.

As the dancer moves, their body leaves a multicolored trail across the large wall screen that comprises one side of the performance space.

The graphics is generated by drawing two bezier curves to abstractly represent the dancer's body. The first curve is drawn through coordinates representing their left foot, head, and right foot. The second curve is drawn through coordinates representing their left hand, center of their body, and right hand. Small 3-D spheres are also drawn to map onto hands, feet, head and center of the body of the performer, both for a reference for the dancer and to accentuate the stylized representation of the body on the screen. The multicolored trail is intended to represent the dancer's shadow that follows them around during the performance. The shadow has a variable memory of the number of trails left by the dancer's body. Hence if the shadow has a long memory of trails (more than thirty) the dancer can paint more complex abstract figures on the screen.

The choreography of the piece can then vary according to which one of the elements of the interactive space the choreographer decides to privilege. In one case the dancer might concentrate on generating the desired musical effect; in another case or in another moment of the performance, the dancer may want to concentrate on the graphics— i.e. painting with the body— or finally the dancer might just focus on the dance itself and let DanceSpace generate the accompanying graphics and music.

The philosophy underlying DanceSpace is inspired by Merce Cunningham's approach to dance and choreography [15]. The idea is that dance and movement should be designed independently of music and that music should be subordinate to movement and may be composed later for a piece as a musical score is done for film. When concentrating on music, more than dance, DanceSpace can be thought of as a "hyperinstrument"[17]. Hyperinstruments are musical instruments primarily invented for non-musical-educated people who nevertheless wish to express themselves through music. The computer that drives the instruments adds the basic layer of "musical knowledge" needed to generate a musical piece. Moreover we have thought of DanceSpace as a tool for a dancer/mime to act as a street performer who has a number of musical instruments attached to their body. The advantage of DanceSpace over the latter is that the user is unencumbered and wireless and can be more expressive in other media as well (its own body or computer graphics). The disadvantage is that DanceSpace is mainly a music improvising system and it is therefore difficult to use it to reproduce well known musical tunes.

Future improvements to DanceSpace include having a number of different background tunes and instruments available for the dancer to use within the same performance. Another important addition will also allow the user to adjust the music's rhythm to their rhythm of movement. We would also like the color of the dancer's graphical shadow to match an expressive or emotional pattern in the dance and become an active element in the choreography of the piece.

We see DanceSpace as a possible installation for indoor public places, as for example airports, where people usually spend long hours waiting, or interactive museums and galleries. DanceSpace could also become part of a performance space, allowing a dancer to play with their own shadow and generate customized music for every performance.

Our current work in building Performance Spaces includes also an **Improvisational Theater Space**. Improvisational Theater provides us with an ideal playground for an IVE-controlled stage in which embodied human actors and "Media Actors"[29] generate an emergent story through interaction among themselves and the public. Media Actors are semi-autonomous agent-based [18] text, images, movie clips, and audio. These are used to *augment* the play by expressing the actor's inner thoughts, memory, or imagery, or by playing other segments of the script. Among the wide variety of Theater styles and plays we have chosen to stage "Improv" (Improvisational Theater). This is an entertaining and engaging genre which allows the audience to drive part of the play. An experimental performance using the IVE setup was shown in February 1997 in the occasion of the Sixth Biennal Symposium on Arts and Technology, with improv actress Kristin Hall [27].

Figure 13: Improvisational Theater Space

In this series of very short plays we showed an actress in the process of interrogating herself in order to make an important decision. A Media Actor in the form of projected expressive text plays her "alter-ego"13 and leads her to making a decision. The Text Actor has sensing abilities— can follow the user around on stage, can sense a set of basic gestures, and understands simple words and sentences— and also expressive abilities— can show basic moods through typographic behaviors, like being happy, sad, angry, or excited. Ongoing progress is directed towards a more accurate IVE-based gesture recognition, exploring the challenges and advantages of multimodal interaction, and rehersing a variety of multi-branching improvisational plays according to the suggestions of the audience.

3.6 ALIVE

Figure 14: Chris Wren playing with Bruce Blumberg's virtual dog in the ALIVE space

The last of the gesture-language mappings is the most abstract. Again, it's related to the other gesture-languages discussed above, and the primary distinction lies in a subtle, but important, difference in the design of the interface. Best called "gesture in context" this mapping attempts to create an interface that is intuitive given the context. Ideally, the mapping is aligned so that failures in tracking or classification are transparent to the user. Clever mapping design can thus greatly reduce the need for sensor systems to perform flawlessly by playing off the expectations and socialization of the user. Because of that trait, this was the first system to be implemented in our lab, in the form of the Artificial Life Interactive Virtual Environment (ALIVE).

ALIVE combines autonomous agents with an interactive space. The user experiences the agents (including hamster-like creatures, a puppet, and a well-mannered dog—Figure 14) through a "magic-mirror" idiom. The interactive space mirrors the real space on the other side of the projection display, and augments that reflected reality with the graphical representation of the agents and their world (including a water dish, partitions, and even a fire hydrant).

The "magic-mirror" paradigm is attractive because it provides a set of domain constraints which are restrictive enough to allow simple vision routines to succeed, but is sufficiently unencumbered that is can be used by real people without training or a special apparatus.[18]

One agent the user can interact with in ALIVE is a puppet that tries to act like a small child. The user can interact with the agent using certain hand gestures, which are interpreted in the context of the particular situation. For example, when the user points away and thereby sends the puppet away, the puppet will go to a different place depending on where the user is standing. If the user waves or comes towards the puppet after it has been sent away, this gesture is interpreted to mean that the user no longer wants the puppet to go away, and so the puppet will smile and return to the user. In this manner, the gestures employed by the user can have rich meaning which varies on the previous history, the agents internal needs and the current situation. [18]

4 Conclusion

The preceding examples illustrate successful interfaces built for a wide range of application domains from animation to artistic expression to information browsing. They all differ in the mappings they employ between sensed features, and application control. However, they all have in common the use of remote sensing technology coupled with perceptual intelligence built into the environment. The common idea is the realization that state-of-the-art vision and audition systems are capable of providing enough information to drive interactive systems, and that they provide that information in a non-invasive way that is compatible with social, natural, and creative interaction.

By adding intelligence to the surrounding space to make it responsive to the user, Perceptive Spaces offer new venues for art and entertainment. They provide solutions to man-machine interface design problems that have historically been difficult or impractical to address with traditional technologies. We believe that the notion of a perceptual space will become central to future entertainment installations, and that this technology has the potential to enhance human expressive abilities.

References

[1] ACM. *Mandala: Virtual Village*, ACM SIGGraph, Computer Graphics Visual Proceedings, 1993.

[2] S. Aukstakalnis and D. Blatner. *Silicon Mirage*. Peachpit Press, 1992.

[3] A. Azarbayejani and A.P. Pentland. Recursive estimation of motion, structure, and focal length. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 17(6):562–575, June 1995.

[4] Ali Azarbayejani and Alex Pentland. Real-time self-calibrating stereo person tracking using 3-D shape estimation from blob features. In *Proceedings of 13th ICPR*, Vienna, Austria, August 1996. IEEE Computer Society Press.

[5] A. Baumberg and D. Hogg. An efficient method for contour tracking using active shape models. In *Proceeding of the Workshop on Motion of Nonrigid and Articulated Objects*. IEEE Computer Society, 1994.

[6] David A. Becker and Alex Pentland. Staying alive: A virtual reality visualization tool for cancer patients. In *Proc. of the AAAI Workshop on AI, Alife, and Entertainment*, Portland, Aug 1996.

[7] Martin Bichsel. Segmenting simply connected moving objects in a static scene. *Pattern Analysis and Machine Intelligence*, 16(11):1138–1142, Nov 1994.

[8] A. Bobick, S. Intille, J. Davis, F. Baird, L. Cambell, Y. Ivanov, C. Pinhanez, A. Schutte, and A. Wilson. The KidsRoom: Action recognition in an interactive story environment. MIT Media Lab Perceptual Computing Group Technical Report No. 398, MIT, Dec. 1996.

[9] Michael A. Casey, William G. Gardner, and Sumit Basu. Vision steered beam-forming and transaural rendering for the artificial life interactive video environment (alive). In *Proceedings of the 99th Convention of the Aud. Eng. Soc.* AES, 1995.

[10] T. Darrell, B. Moghaddam, and A. Pentland. Active face tracking and pose estimation in an interactive room. In *CVPR96*. IEEE Computer Society, 1996.

[11] Davis, J. and A. Bobick. The representation and recognition of human movement using temporal templates. In *Proc. Comp. Vis. and Pattern Rec.*, pages 928–934, June 1997.

[12] Davis, J. and A. Bobick. Virtual PAT: a virtual personal aerobics trainer. MIT Media Lab Perceptual Computing Group Technical Report No. 436, MIT, 1997.

[13] Davis, J. and A. Bobick. Sideshow: A silhouette-based interactive dual-screen environment. MIT Media Lab Perceptual Computing Group Technical Report No. 457, MIT, 1998.

[14] D. M. Gavrila and L. S. Davis. Towards 3-d model-based tracking and recognition of human movement: a multi-view approach. In *International Workshop on Automatic Face- and Gesture-Recognition*. IEEE Computer Society, 1995. Zurich.

[15] James Klosty. *Merce Cunningham: dancing in space and time*. Saturday Review Press, 1975.

[16] M. W. Krueger. *Artificial Reality II*. Addison Wesley, 1990.

[17] Tod Machover. *HyperInstruments: A Composer's Approach to the Evolution of Intelligent Musical Instruments*, pages 67–76. Miller Freeman, 1992.

[18] Pattie Maes, Bruce Blumberg, Trevor Darrell, and Alex Pentland. The alive system: Full-body interaction with animated autonomous agents. *ACM Multimedia Systems*, 5:105–112, 1997.

[19] S. Mann and R. W. Picard. Video orbits: characterizing the coordinate transformation between two images using the projective group. *IEEE T. Image Proc.*, 1997. To appear.

[20] D. Metaxas and D. Terzopoulos. Shape and nonrigid motion estimation through physics-based synthesis. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 15:580–591, 1993.

[21] J.A. Paradiso and F. Sparacino. Optical tracking for music and dance performance. In *Fourth Conference on Optical 3-D Measurement Techniques, Zurich, Switzerland*, September29-October 2 1997.

[22] A. Pentland and B. Horowitz. Recovery of nonrigid motion and structure. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13(7):730–742, July 1991.

[23] Claudio S. Pinhanez, Kenji Mase, and Aaron Bobick. Interval scripts: a design paradigm for story-based interactive systems. In *Proc. of CHI'97*. ACM, March 1997.

[24] J.M. Rehg and T. Kanade. Visual tracking of high dof articulated structures: An application to human hand tracking. In *European Conference on Computer Vision*, pages B:35–46, 1994.

[25] K. Rohr. Towards model-based recognition of human movements in image sequences. *CVGIP: Image Understanding*, 59(1):94–115, Jan 1994.

[26] Kenneth Russell, Thad Starner, and Alex Pentland. Unencumbered virtual environments. In *IJCAI-95 Workshop on Entertainment and AI/Alife*, 1995.

[27] F. Sparacino, K. Hall, C. Wren, G. Davenport, and A.P. Pentland. Improvisational theater space. In *The Sixth Biennal Symposium for Arts and Technology, Connecticut College, New London, CT*, February 27-March 2 1997.

[28] F. Sparacino, A.P. Pentland, G. Davenport, and et al. *City of News*. Ars Electronica Festival, Linz, Austria, 1997. 8-13 September.

[29] Flavia Sparacino. *DirectIVE: Choreographing Media for Interactive Virtual Environments*. MIT Media Lab, 1996. Master Thesis.

[30] Flavia Sparacino, Christopher Wren, Alex Pentland, and Glorianna Davenport. Hyperplex: a world of 3d interactive digital movies. In *IJCAI-95 Workshop on Entertainment and AI/Alife*, 1995.

[31] C. Wren, F. Sparacino, A. Azarbayejani, T. Darrell, T. Starner, Kotani A, C. Chao, M. Hlavac, K. Russell, and Pentland A. Perceptive spaces for peformance and entertainment: Untethered interaction using computer vision and audition. *Applied Artificial Intelligence*, 11(4):267–284, June 1997.

[32] Christopher Wren, Ali Azarbayejani, Trevor Darrell, and Alex Pentland. Pfinder: Real-time tracking of the human body. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(7):780–785, July 1997.