Traces: Embodied Immersive Interaction with Semi-Autonomous Avatars

Simon Penny
University of Portsmouth (UK) and Merz Akademie, Stuttgart, Germany,
Jeffrey Smith
Carnegie Mellon University Robotics Institute.
Phoebe Sengers
GMD - German National Research Center for Information Technology,
Andre Bernhardt
Karlsruhe University.
and **Jamieson Schulte**
Carnegie Mellon University, CALD

email: penny+@andrew.cmu.edu, jeffrey@cs.cmu.edu,
Phoebe.Sengers@gmd.de, ab@imdb.com, jscw+@andrew.cmu.edu

October 23, 2000

Keywords: Machine vision, CAVE, infrared video, wireless tracking, full-body interaction, autopedagogy, semi-autonomous agents

Abstract:

Traces is an artwork for the CAVE that uses a novel machine vision system to enable unencumbered full body interaction with a range of semi-autonomous agents without the imposition of any sort of textual, iconic or encoded-gestural interfaces and without

physically restrictive wiring, pointing devices, or headgear. Furthermore, Traces does not consist of a "world" which is "navigated"; instead, the movement of the user through the space leaves volumetric and spatial-acoustic residues of user movement which slowly decay. This project was motivated by a desire to explore and critique four central issues in contemporary HCI: (a) embodied interaction with computational systems; (b) rapid and transparent learning of interfaces by untrained users (the autopedagogic interface); (c) immersive bodily interaction with software agents, (d) extension and elaboration of the general conception of "interactivity" itself. To explore these issues, we built an infra-red multi-camera machine vision system which constructs a volumetric model of the whole of the users' body in real time. We have also developed custom 3D vision tools, graphical techniques and a range of techniques for generating and managing semi-autonomous agents in immersive environments.

Traces was developed between November 1998 and September 1999 and exhibited in the CAVE at the Ars Electronica Center for Ars Electronica 99. Development was supported by the Cyberstar98 award, by GMD Sankt Augustin Germany, and by Carnegie Mellon University.

1 Artistic, Theoretical and Technical Overview

Traces is an artwork motivated by the opportunities, limitations and inconsistencies in immersive technologies and the rhetoric surrounding them. In the late 80's and early 90's Penny engaged in a critical assessment of VR (then just emerging as a civilian paradigm) which contrasted with the futuristic rhetoric surrounding these systems and their ultimately retrogressive and anti-embodying qualities, metaphorised as dreams of transcendence and deliverance from the prison of the flesh. Two contradictory and untenable claims were made by these futurists: either the technology was an embodying technology, or it delivered the user from the body into a clean and fleshless world. These ideas were traced to the uncritical instrumentalisation of an essentially uninterrogated Cartesian value system, which privileges the abstract and disembodied over the embodied and concrete. This value system runs throughout digital practice, from the foundational hardware/software duality to the mythologies of cyberspace so celebrated by Gibson, Moravec,

et al., and has been popularized in the entire range of marketing rhetoric for computer technologies[10]. Traces is the most recent in a series of projects by Simon Penny which take a critical position with respect to the erasure of embodiment in computational systems[1].

In pursuit of this critical project, we decided to use the CAVE[2] which is an inherently more embodying technology than head-mounted-display (HMD) VR. In HMD VR, you cannot see your hand in front of your face — the body is erased, at least from the visual field. This erasure sets up a perceptually inconsistent and contradictory situation rather like phantom limb syndrome. In a CAVE, however, the user can see herself, and she shares her physical space with virtual objects. Limbs and body masses are "impacted by," intersect or interact, with virtual objects. Although this is a somewhat less embodiment-denying experience, in both traditional CAVE and HMD VR applications, the body is reduced to a single data-point in the computational system (the coordinates of the tracker) and the body, again, is erased.

A central goal of Traces was to enhance the embodying qualities of the CAVE experience by building an unencumbering sensing system which modelled the entire body of the user. To achieve this, we built an infra-red multi-camera machine-vision system which constructs a volumetric model of the whole of the users body in real time, and developed custom 3D vision tools, graphical techniques and an agent behavior environment. In Traces, the user is able to interact in a direct bodily way with the computational system without the imposition of any sort of textual, iconic or encoded-gestural interface: there is no pointer, wand, joystick or data-glove. Most importantly, the system recognizes and responds to the entirety of the users body, so, as in normal human-to-human communication, a gesture by a knee or the hips can be as significant and effecting as that of a finger. The user has unimpeded, kinesthetically intuitive full-body interaction with semi-autonomous entities, and is not encumbered by any wiring harness or headgear (except the unwired, lightweight shutter-glasses).

Artistically, the goal is to offer the user an experience which combines

[1] See, for example, `http://www.art.cfa.cmu.edu/Penny/works/fugitive/fugitive.html` and `http://www.art.cfa.cmu.edu/ Penny/works/petitmal/ petitcode.html`

[2] The CAVE, or "CAVE Automatic Virtual Environment" is a spatially immersive display consisting of a cubic room three meters on a side. Using multiple projectors, the CAVE can display stereo, rear-projected graphics on three walls and the floor. More information can be found at `http://www.evl.uic.edu/EVL/VR/systems.shtml`

the bodily immediacy of dancing with the spatial experience of sculpture. The user "dances" a "sculpture" into existence. These entities generated by bodily movement possess varying types of autonomous behavior, from percolating masses to flocking individuals. Emancipated by the lack of wiring harness, users enjoyed interacting with dynamic avatars, recognized the connection between their movements and their traces and generally interacted with their traces by engaging in enthusiastic physical activity of a type not usually seen in VR applications. People lay on the floor, jumped, danced, kicked and danced. It was amusing to see people emerge from the CAVE sweating, panting and red faced! They really had to do physical work to interact with the system. This in itself was proof that we had built a dramatically different type of CAVE experience.

The standard interactive paradigm for most immersive projects is constrained to virtual navigation through texture-mapped worlds. Within this paradigm are odd kinesthetic inconsistencies, for instance: the user can physically turn to face objects of interest, but cannot physically walk toward them. Rotational movement and translational movement are not compatible in the way that they are in the real world. That is to say, so called "immersion" is not intuitive and thus not truly immersive. The paradigm of bodily behavior in Traces does not suffer from this confusion. Instead, Traces stubbornly refuses to endorse the basic qualities of most Virtual Environments: there is no "world" and no "navigation." The bodily behavior of the user generates real-time graphics and (potentially) sound, in a limited and static virtual space: a virtual room about double the volume of the physical CAVE. The goal of Traces is precisely *not* to present a panoptic spectacle for the user, but to turn the attention of the user back onto their own sense of embodiment through time. For this reason, graphical representations are minimal, texture mapping and other gratuitous eye-candy was avoided. Instead, the movement of the user through the space leaves "traces": volumetric and spatial-acoustic residues of user movement; which slowly decay. As time progresses, the traces become more active, and in the last stage of the user experience, autonomous entities are spawned by the user, which have complex behaviors of their own.

The idea of the intuitive interface is not straightforward, just as intuition itself is a complex idea. Traces, like any artwork, is designed for an untrained general public. Yet interactive art-works (and other systems) almost always possess novel interfaces with eccentric behaviors, often unnoticed by the makers who are fully naturalized to the idiosyncrasies of their systems. Often, new users are utterly bewildered. In Traces, we continue research and

development of the notion of the "Autopedagogic Interface," the goal being to introduce interfacial complexities of the environment gradually and transparently ramp up, driven by the users desire and pleasure.

Another major goal of Traces is the development of a range of techniques for generating and managing semi-autonomous avatars in immersive spaces. In most virtual environments, avatars are thought of as a direct representation of the user. However, as the complexity of virtual environments increases and, with it, the scope and complexity of possible avatar behavior, it becomes more and more difficult to maintain a direct correlation between the user's wishes and the avatar's actions. Instead of this now inadequate notion of "avatar-equals-user," we propose that avatars can be fruitfully thought of as semi-autonomous agents, which have their own behaviors and intentionality, but are intimately tied to the user's actions. In Traces, user body movements spawn avatars which gradually become more and more autonomous. Because one is no longer tied to a model of avatar as direct, unmediated user representation, the formulation of avatars as semi-autonomous agents opens a new and rich conceptual space for design of avatar interfaces in virtual environments.

2 The Vision System

At the beginning of this project, we realized that we would have to build a multi-camera machine vision system specifically for the CAVE. Using a technique previously developed by Penny[3], the highly active visual environment of the CAVE was simplified by utilizing only the near infra-red range of the CCD video cameras, filtering out visible light from the cameras, and lighting the space with IR. Thus two separate optical "channels" were established, and video projection for the consumption of the user did not overlap with illumination of the space for the vision system. However, the physical structure of the CAVE and the projection path of the video present significant constraints on the placement of both cameras and IR lights. As every CAVE is different "backstage," each implementation of the camera and light system was different.

[3] `http://www.art.cfa.cmu.edu/Penny/works/fugitive/ fugitive.html`

2.1 Lighting

The goal for the lighting is to disambiguate the users body from the background. Without active lighting approaches, and assuming monochromatic video, this implies two possible solutions: to light the body against a dark background, or to silhouette the body against a bright background. After numerous experiments we chose the latter. To achieve this lighting scheme, we back-lit each video screen with 500-watt halogen flood lights fitted with near-infrared filters of our own design. In addition the "back" or open, side of the CAVE had to be fitted with a screen or curtain, in order to separate user motion from the noise of movement outside the CAVE space. This curtain was also back-lit. Depending on the context, up to 10 individually dim-able lamps were used, and were tuned during camera calibration procedures.

This lighting solution was inexpensive and it performed adequately, although not perfectly, due to environmental inconsistencies such as reflectivity of the floor. In addition (and somewhat alarmingly at the outset) we found that the hue of clothing in the visible range did not predict its "shade" in the IR range. For example, human hair, skin, and black clothing would often appear white under IR. This was undesirable, because if the body is white against a white background, no body data can be collected. Given the public nature of Traces and the relatively rapid throughput of users, a pragmatic solution was adopted: we supplied a reliably IR-black cotton over-shirt which users were asked to wear. This helped, but left some user body models disconcertingly headless and legless. A proposed black-cotton "head-bag" solution was deemed undesirable.

2.2 Cameras

Although up to eight independent cameras could be supported by our software, after numerous experiments with both the placement and number of cameras, we found that four cameras were sufficient for real-time model building. These cameras were mounted 210cm from the floor in each corner of the CAVE and were calibrated using a 1m grid on the floor. These inexpensive monochrome "board" video cameras were fitted with standard Kodak Wratten IR-pass filter material which eliminated visible light while passing the infrared. The combination of controlled IR lighting and blocking the visible light to the cameras filtered away the projected stereo video imagery allowing the system to derive unambiguous data about the user.

Using commercial video hardware and custom code running on a on a 366 MHz Pentium II Linux PC, the camera data was processed at better than 15 fps, with a 5cm voxel dimension, with about 1/30th second latency.

2.3 Code

In order to allow wireless full body interaction for Traces, a vision system had to be developed that was capable of building a three dimensional body model of a person in real time on inexpensive standard PC systems, limiting the mathematical complexity of the algorithm. At the outset, it was not clear that our project was possible at all, especially after researching several existing algorithms[4] for this task which take minutes on powerful machines to compute a single voxel model. Based on our experience of acceptable latency in kinesthetic/graphical interaction, we set a minimum acceptable frame rate of 15 fps. It became obvious that in order to achieve a good frame rate, there would be a necessary trade-off between temporal and spatial resolution. Our position was that for the project, temporal resolution was more important that spatial.

In order to get the required 15 fps, a voxel-space with the size of 60x60x45 voxels was chosen for Traces. With the fixed physical dimension of 3x3x2.5 meters for the CAVE, every voxel therefore represents a 5cm cube. On faster machines the resolution can be better than 15 fps, as long as the lookup tables fit into the memory. However, it is important to note that the memory access speed is more important than the CPU speed. A slower CPU with very fast memory access provides better results than a fast CPU with slow memory access.

Our algorithm begins by constructing silhouette images for each camera of the objects inside the CAVE. These silhouette images are created by a process of image differencing: a reference image is taken of the empty CAVE, lit by the infrared lights. Each new frame is then subtracted from this reference image making a person within the space appears dark in front of the bright screens.

[4]General methods for automatic reconstruction of 3D models are distinguished into "active" and "passive" types. Active methods use laser scanner or structured light (grids) projected on the object. Passive methods use the plain camera image either from different cameras or a series of images from a single camera (e.g. object on a turntable or camera on a track). While they require less equipment, passive methods are subject to certain limitations. They are, for example, restricted to convex objects. Given the limitations of context (the CAVE) and questions of cost, we chose to pursue the passive approach.

These silhouette images are then projected from the cameras' positions in a virtual model of the CAVE space, producing four frustri projecting into the space. Any voxel which is located in three of the four frustri is kept and becomes part of the body model point cloud. This intersection procedure carves away all voxels except those occupied by the body. The algorithm we ultimately implemented uses a equal-volume voxel-space representation rather than an octree representation for the 3D model. Thus, instead of transforming a voxel from world- to camera-coordinates and then performing the intersection tests with the silhouettes, our algorithm reduces every voxel to a single point that is transformed into camera-coordinates and then tested against the silhouette-images. This saves the substantial effort of transforming and intersecting lines.

To speed things up further, these transformation calculations (which are very time consuming matrix operations) are calculated in advance for every voxel and for every camera and are stored in large lookup-tables. These lookup tables contain pointers to the corresponding pixel in the silhouette picture memory. The storage order is optimized for sequential access by the CPU. Using four cameras, the CPU only has to get four pointers and look at the four values the pointers reference in order to determine whether or not a voxel belongs to an object within the space. This is determined by the following two criteria:

1. a voxel has to be visible in at least three out of four camera images to avoid ambiguity.

2. if a voxel transforms outside a silhouette in one silhouette image, the voxel does not belong to a person.

An advantage of our algorithm over oct-tree based methods is that the process of reconstruction always takes the same amount of time no matter if the space is empty, or occupied by one or more persons or objects. We also get noise filtration for free: because of the "three out of four" rule, pixel errors due to noise in the silhouette images seldom make it into the reconstructed model. However, this feature is double edged: a tiny spot with a bad contrast to the background creates a "hole" in the silhouette and thus a gap in the 3D reconstruction.

As with all passive construction algorithms, it is not possible to properly detect concave surfaces. For our purposes this is seldom a limitation: our tests have shown that there is practically no posture that a person can assume within the space that is not resolved properly as long as the person

is fully visible in all four camera images. Another potential source of error is the distance of a person to the camera; specifically, the closer a person is to a camera the larger the potential error. This error has two causes: first, the camera no longer able to capture the whole person. Second, and more importantly, the transformation of a voxel with a finite size that is close to a camera covers an area of more than one pixel in the silhouette image. However, the algorithm reduces it to one pixel no matter how far away the object is from the camera. One possible solution could be to define a minimum distance to the camera (which could be reflected in the lookup table). In our case it proved sufficient that the cameras were mounted diagonally opposite and perpendicular to each other to disambiguate the information and make up for this error.

3 Machine-Vision Driven Interaction

All the graphical events which happen inside the CAVE are derived from the three-dimensional model of the user's body that we construct with the vision system. However, our constructed body model is initially just a cloud of unsorted points which describe the volume of the user. Thus, our first task after constructing this cloud is to extract meaningful data about the user's position and configuration.

Ideally, we would like to be able to reconstruct all the relevant information about the user from this cloud: the global position and orientation or the user's whole body, the angles of her elbows, knees and hips, the direction in which she is looking, and so forth. And, given sufficient time, most if not all of this information could be calculated. But every millisecond spent analyzing data is another millisecond of latency in our system. It is a truism of human-computer interaction that latency is as important as frame rate for forming an impression of interactivity for a user.Thus, we have a real motivation for reducing the amount of time we spend reconstructing data about the user from our point cloud and we are forced to be selective about the types of data we do construct.

3.1 Position and Pose

Since we are using the model of the user's body to derive an interactive art experience, we are interested in certain types of information about her position inside the CAVE and the pose of her body. Specifically, we thought it most valuable to know the following properties of the user at any moment:

- Position within the CAVE
- Size ("mass")
- The location of the user's head, feet, hands and "center of mass"
- Approximations to the velocities and accelerations of these features

Luckily, the position and mass of the user are provided for free during the body model construction and no extra time has to be spent to extract this information. However, this still leaves us with the problem of identifying the user's head, hands and feet and tracking them through time.

We accomplish this identification and tracking by using a few simple heuristics. First and most generally, we can count on a certain amount of frame-to-frame consistency of the body model. That is, during the fifteenth of a second between consecutive frames, the user cannot have moved much, or changed the position of her hands and feet significantly. Secondly, for each body part to be tracked we can make an a priori guess at where it is likely to be. Specifically, we use the following assumptions:

- the head is usually highest point on the user's body
- the feet are usually the two lowest points
- the hands are those points furthest from the center of mass which are not the head or feet

Clearly, these heuristics are not foolproof and errors in tracking can occur. However, it is somewhat misleading to label the points we track "feet" and "hands" since we are not interested in the hands or feet per se. Rather, we are interested in those points that project from the torso and which are moving. Thus, whether we end up tracking an elbow instead of a hand for a few frames is not particularly significant. Furthermore, the user is not being told which points of her body are being tracked — there is no glowing "this is your hand" sign with which the user can find fault. The user simply sees graphical entities spawned by various parts of their body when in motion.

The one instance in which our inability to guarantee perfect feature tracking was a disadvantage was for head tracking. The head in particular is important to track because the stereo illusion is highly dependent on knowing the exact position and orientation of the user's eyes. Since, as

noted earlier, we wished to avoid wrapping the user in wires and hardware, we had no information on the location of the user's head apart from that which we could extract from the body model.

The X-Y-Z location of the user's head (and thus his or her eyes) was generated by using the heuristics we describe above, which were consistently accurate. In fact, the only time our heuristic would fail would be when the user was so contorted e.g. head between their legs, looking backwards, that the incorrect identification of the head would go unnoticed. However, the Cartesian location of the head is not the only important factor when projecting and drawing stereo images: the angular orientation of the head is equally important. Unfortunately we were not able at the time to produce as reliable a method for determining the orientation of the user's head as we were location. In the absence of this data, we adopted another heuristic: in general, the user would be looking in the direction perpendicular to the plane formed by the vertical axis of their body and the broadest horizontal axis of their body. This technique, while admittedly flawed, ending up performing adequately.

3.2 Velocity and Acceleration Data

During the first two phases of Traces — the "passive" and "active" trace — the user's body model was used as a three dimensional "brush" to fill in voxels as the user moved through space. The speed and acceleration of the user and her limbs affected the simulation only through what volumes they traveled through. The third phase, however, was critically dependent on these computed quantities.

In the final stage of Traces, the user's motions spawned autonomous creatures, which would fly around the space and interact with the user and each other. We did not wish for these agents to be created randomly, or merely based upon where the user was located in space. Instead, we wanted these entities to be "thrown off" by the user as she moved enthusiastically through the CAVE. Our tracking of the user's hands and feet provided us with rough approximations to the velocity and acceleration of those body parts which we used to control the creation of these agents.

The flying agents were created either when the hands or feet were moving faster than some threshold (about 1 meter a second) or were moving slower than some threshold, but had a high acceleration; i.e. had recently come to a stop or reversed direction. This simple set of rules gave a convincing impression that swift or violent motions of one's limbs threw off the flying

agents, like droplets of water being shaken from wet hands.

4 Generating and Displaying Graphics

The generation and display of graphics in Traces presented some unusual challenges. Our choice of what graphics were displayed was constrained by a number of factors, some mundane and some specific to the field of immersive electronic art. The most obvious constraint on the type and number of graphical entities that we could show in the CAVE was the limited processing power available to us. Although the multi-processor SGI Onyx which we had at our disposal can provide an enormous amount of number-crunching power, only one of our four processors could be devoted to graphics display. Furthermore, any graphical object must be drawn eight times per frame: once for both the right and left eyes, for each of the four walls of the CAVE [5]. Given that we required a display frame rate of at least ten frames per second to insure a feeling of real-time interactivity, our graphics could take no longer than 12 milliseconds to calculate and display. While the problem of limited computational resources is common to all CAVE applications, some of the more common workarounds were not available to us. For example, since we had no pre-existing "world" of texture mapped scenery, and all our graphics were being generated in real time from the body model, we could not perform time-saving preprocessing such as establishing display lists.

In addition to problems of limited resources, the choice of what graphics to use was constrained by artistic decisions as well. The original conception was that a user would create, by their movements, real time sculptural forms which would persist and fade, in a manner reminiscent of a 3D time-lapse photograph, and that these "traces" would take on increasingly autonomous behavior during the user's experience. The constraints of graphics computation mentioned above, along with a desire to avoid gratuitous eye candy, led to the graphical solutions presented at Ars Electronica, in which cubic voxels with procedural transparency formed the traces. Texture maps were avoided except for the virtual room itself.

4.1 Spatial Perception

A curious paradox of stereo illusion opened up during graphics development. We found that by utilizing these abstract volumes (cubic voxels), we jetti-

[5] The left, front and right walls in addition to the floor

soned a range of highly significant depth cues which are cultural, as opposed to physiological in origin. In a conventional VR representation, a world may contain an avenue of trees receding into the distance, or a road winding to the horizon. We contend that a substantial component of the persuasiveness of the stereo illusion is supplied by the user's real-world experience: a row of trees seldom is arranged by size; a road tends not to reduce in width. Hence much of the persuasiveness of VR spatial illusion is a result of users' cultural training, not of the optical stereo created by the technology. This realization became very clear when building the traces from abstract volumes. A sphere of a particular radius could appear to be a smaller sphere close to the user or a large sphere further away. Various time-varying graphical solutions were tried, from luminous point clouds to 3D crosses to spheres which expanded and became transparent, before we settled on the minimal cubic representation.

During the active and passive traces (the first two phases of the piece), an average of 2000 voxels are filled and displayed per time-step. Anything more complex than an OpenGL primitive slows the display down unacceptably. So plain cubes of standard size with procedural transparency, proved to be the only graphical representation which was adequately fast to draw, and which displayed the necessary qualities with minimal perceptual confusion. The age of each voxel was indicated by increasing transparency, and the non-varying size of the cubes allowed for some indication of distance based on perceived relative size. Another motivation for this solution was to avoid any pretense of organic form. There was a desire to be up-front about the fact that this was a computational environment, not some cinematic or hallucinatory pastoral scene.

The transition from off-line development using a 3rd person view (on a monitor) to the first person view in the CAVE produced amusing surprises. Solutions which were very satisfying in the simulation mode were useless in first person. For example, if a person is moving forwards, the trace is developing behind them and they can't see it! Worse, if they move even a voxel backwards, their head is "enclosed" in a form which is the volume of the head at the previous time step, so they can't see anything! Various workarounds were developed for these situations, the best being to establish a "clipping sphere" with a 50 cm radius, centered at the head, which would cull any entities drawn within that volume.

5 Traces Avatars

Generally, avatars, or "user embodiments," are thought of as soul-less bodies for which the user acts as mind. Correspondence between the user's wishes and the avatar's actions is maintained using low-level commands like "walk forward 3 steps" or "pick up the box," or by using hardware sensors on the user's body which are translated into the corresponding movements for the avatar. In this way of thinking; the avatar is what we might call a non-autonomous agent; the avatar, fundamentally, is the user.

But as the complexity of virtual environments increases and, with it, the scope and complexity of possible avatar behavior, it becomes more difficult for the user to directly control all aspects of the avatar using simple low-level commands. The currently dominant metaphor of "user = avatar" is no longer adequate to describe or innovatively solve problems that come up in designing avatars. In response, avatars have been built that allow the user to specify behavior at various levels, from "go north" to "find me an appropriate article" to "negotiate the release of hostages," while the avatar uses its own intelligence to fill in the details[1, 11, 4]. As these avatars become more independent, the idea that the avatar is just a simple extension of the user becomes problematic [6, 14]. As Bowers, O'Brien and Pycock argue, a great deal of technical and social effort is necessary to support the illusion that the avatar behaves non-autonomously, i.e. as a direct and accurate representative of the user [5].

Several researchers have done innovative work that, rather than attempting to get rid of unwanted autonomy, uses that autonomy as a resource to create new, useful forms of the avatar-user relationship. In Hannes Vilhjálmsson's and Justine Cassell's pioneering system BodyChat, avatars have autonomous body behavior [13]. That is, while the user is chatting with other people, their avatars autonomously display the kinds of physical signals humans unconsciously use to support communication, like using glances to show whether or not one is open to communication, raising eyebrows on emphasis words, and using gaze exchange to support turn-taking. Michael Mateas has developed "subjective avatars" for interactive fiction which behave non-autonomously, but have semi-autonomous sensing [7] [8]. These avatars are intended to help the user feel like a character in a story, by sensing the world in a way that reflects the character's perspective on events, drawing out details that matter to the character and describing them in terms of their impression on that character. In their work on Sympathetic Interfaces, Johnson et. al. [9] built a plush toy in the shape of the agent

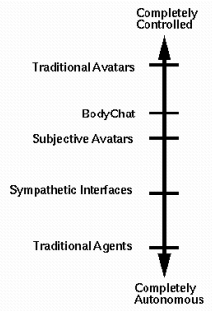

Figure 1: A range of semi-autonomous avatars.

the user is to influence, which the user can move in order to suggest behaviors to the agent; for example, moving the legs of the toy may cause the agent to run. The agent has a great deal of latitude in interpreting the user 'commands,' and engages in fully autonomous behavior when the user does nothing. In all these systems, avatars are no longer slaves to the user, but also engage in autonomous behavior that is connected to what the user has chosen to do.

We propose that in the context of these trends in avatar research, avatars can be more fruitfully thought of as semi-autonomous agents. That is, avatars are thought of as agents like any other with their own behavior and intentionality, but with a particularly intimate relationship with the human user. This notion of avatar not only describes current avatar work more accurately, but also widens the conceptual space of possible avatars.

In our work, we build on previous avatar work with different kinds of autonomy by suggesting they are not lone aberrations, but represent part of a continuum of kinds of avatars made possible by using the semi-autonomous avatars metaphor. Semi-autonomous avatars can be thought of as on a range of autonomy, from the traditional fully passive avatar to the traditional fully active agent (Figure 1).

In Traces, we explore and begin to fill out this range of autonomy levels. The user physically interacts with a series of avatars of increasing complexity and autonomy. The Traces avatars start out very much like traditional avatars, passively following the user's movements. Over time, they gradually become more complex, going through three stages: the Passive Trace, the Active Trace, and the Behaving Trace. At each stage, the trace-avatar adds

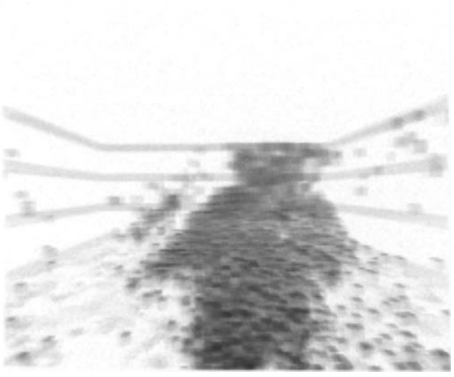

Figure 2: Third-person view of passive trace in action

new levels of autonomy and new complexity to the avatar-user relationship.

5.1 Passive Trace

During the first, Passive Trace phase, the avatar is conceived as a three-dimensional analog of time-lapse photography of the type first made by Eugene Marey. At every time-step, the current body volume is drawn into the space (represented as lilac cubes) which increase in transparency with age (disappearing in a few seconds) and which drift away from the user as if blown by a virtual breeze (Figure 3).

The Passive Trace is built up by the addition of the user's body model at each time step to the volume of voxels. Voxels fade and are removed as they age. All voxels drift away from the user into the virtual part of the room, beyond the front screen. The area currently around the user's head is left transparent, so that the user can see the volume they are creating. This avatar represents the user more or less directly by freezing her movements in the recent past.

The direct relationship between user movements and avatar behavior are easy to grasp because the user's bodily dynamics are analogically and more or less instantly represented in the voxel-space. In the production of persuasive kinesthetic interaction, minimal latency in response to the bodily behavior of the user is of crucial importance.This principle was key to previous projects Petit Mal and Fugitive. In all these systems, temporal resolution is seen to be more important than spatial resolution.

Figure 3: The Passive Trace of a user getting up from a backstand. The user model is displayed in black for reference; it is not displayed in the CAVE.

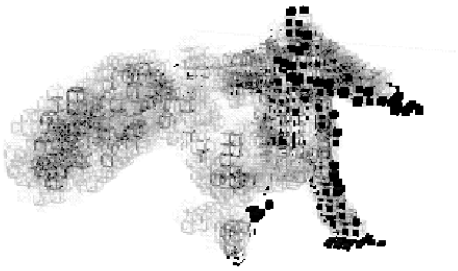

Figure 4: The Active Trace develops according to CA rules.

5.2 Active Trace

During the Active Trace, the shape of the avatar is no longer completely dependent on the user's movements. The voxels that make up the trace are still generated in the same way by the user's movements, but instead of simply fading passively, a 3D cellular automata algorithm[6] is employed. By this algorithm, the number of neighbors a voxel possesses determines whether it will persist into the next time-step, and also determines its color and level of transparency. (Figure 4).

ALife researchers have argued that cellular automata such as Conway's Game of Life provide a simple model of structures at the boundaries of life. Similarly, the cellular automata driving the Active Trace imparts on it the beginnings of livelihood. The trace, while still directly linked to user movements, no longer passively fades away, but generates structures of varying stability in places where the user has been. It changes shape, sparkles and percolates in unexpected ways.

5.3 Behaving Trace

During the Behaving Trace, the body movements of the user "throw off" agents, as though the user is shaking off water droplets (Figure 5). At first, these agents simply fly off the user. Then they exhibit their own behavior, flocking together and following the user or exploring the virtual space together (Figure 6). These agents have articulated bodies which consist of a

[6]Specifically, we implemented Conway's Life in three dimensions.

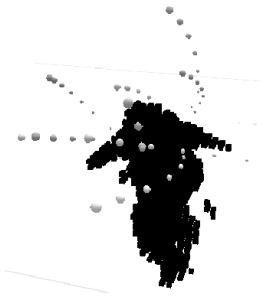

Figure 5: For the Behaving Trace, agents are spawned from the user's movements.

sequence of spheres, each of which follows the sphere before it. Agents can sense and react to the user's movements and the position of other agents. The agent's behaviors are written in a custom-made particle behavior language based in part on Craig Reynolds's steering behaviors [12].

At each frame, the system first updates the agents' sensors which are shared for efficiency. Then, each agent's behavior is run in turn. Behaviors compute the new direction and velocity of the agent, generally by calling functions similar to Reynolds' steering algorithms on data derived from sensors.

Agents' articulated bodies are implemented through the same behavior architecture. While the lead sphere of the agent runs the agent's overall behavior (e.g. "flock-behind-user"), each subsequent sphere runs a behavior to follow the particle before it. The result is an overall worm-like body which moves, bends, and turns smoothly and responsively with no more algorithmic effort than in the case of a non-articulated body.

At this stage, the avatar has become highly autonomous, engaging in autonomous behaviors and not necessarily following the user. At the same time, the avatar is not completely autonomous: it is still generated by and responsive to user movements. Because the agents flock together, they feel like a coherent entity in the environment, as a distributed Behaving Trace rather than as a bunch of unrelated creatures. Identification of the user with the Behaving Trace as a kind of half-alien self is enhanced by the gradual steps through which the user went to get to this stage; following Penny's theory of the auto-pedagogic interface, users gradually learn to understand

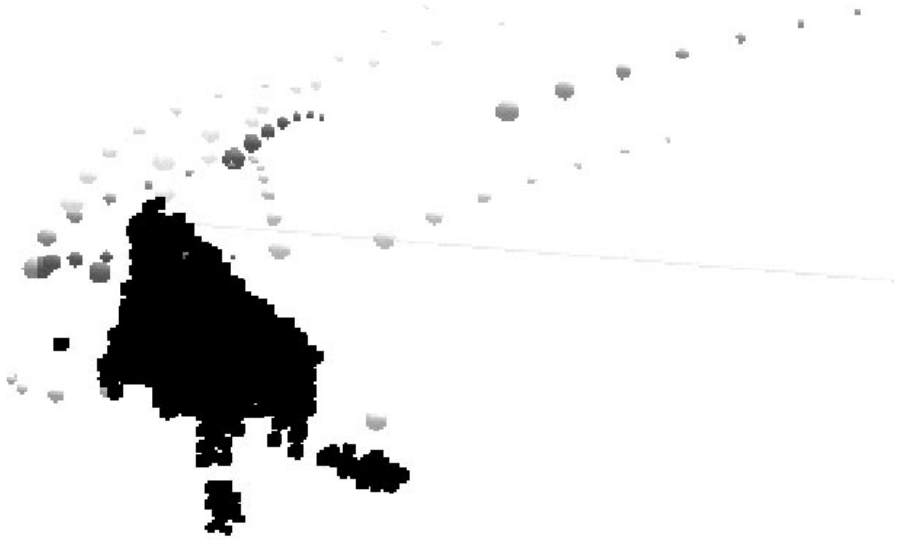

Figure 6: The agents that form the Behaving Trace have their own behaviors. Here they are following the user.

increasingly complex relationships with their avatar.

The Behaving Trace was particularly popular with users, due in large part to its animorphically mimetic quality. The experience of the Behaving Traces was not unlike being in an aviary or a chicken pen, or snorkeling amongst schools of fish. The creatures attended to the user and went about their business, by turns: cognizant of the user, but not beholden to her. The persuasive psychological effect of interacting with autonomous agents in the same space who can sense one's body and obviously react to it cannot be overstated. [7].

At the same time, we saw a number of problems with the Behaving Trace. First, users generally understood that more movement makes more agents appear, but not that they were generated by being flung off the user's limbs. This is because it is difficult to tune the starting velocities of agents and the length of time before they change to autonomous behaviors so that the user really gets a feeling of flicking them off. The second difficulty was that users wanted a deeper interaction with the agents. We wanted agents to be autonomous; but when agents are flocking around the CAVE, users can feel ignored and spend a lot of effort trying to attract their attention. Users want to dance with agents, to chase them and to be chased by them. We plan to put more development effort in this area.

6 Discussion: Ranges of Autonomy

Each of the avatars discussed here, the Passive Trace, the Active Trace, and the Behaving Trace, has a different level of autonomy, resulting in a different level of identification with the human user. These varying autonomy levels start to fill in the range of possible levels between complete nonautonomy and complete autonomy (Figure 7). At one extreme, semi-autonomous avatars become fully passive avatars in the traditional sense; at the other, they become completely autonomous like traditional agents. The dividing line between avatar and agent is thus blurred. Indeed, it must be blurred in order to understand the partially controlled agents which are gradually becoming a standard avatar practice.

In addition, we believe the concept of avatar as semi-autonomous of the user is important from a critical perspective. The idea of avatar as simple extension of the user has worried several critics [6] [2] [3], because,

[7] This effect has been noted by roboticists, but in our experience it is just as true in VR — when one's body is there.

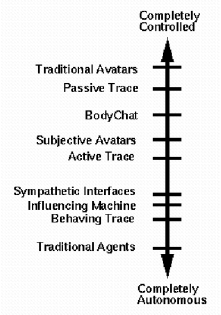

Figure 7: Traces begins to fill in the range of semi-autonomy.

as J. MacGregor Wise points out [14], the unrecognized disjunction between avatar and user makes it difficult for both researchers and users to develop a critical understanding of the possibilities and constraints imposed by the interface. While promising the user full engagement, avatars are frequently only able to do a small part of what the user wants, and, for more complex avatars such as information-gathering programs on the Web, may confound the user by acting on idiosyncratic, unstated interpretations of the user's commands. By presenting the agent as semi-autonomous, it is easier for users to develop critical distance and understand the limits of their avatar's ability to accurately represent them.

In the semi-autonomous avatar paradigm, rather than being identical to the user, an avatar must be thought of as the part of the system which is intimately connected to the user. In this way, the line between system, avatar, and interface also becomes blurred; the avatar becomes the interface, the point at which the computational system and the user make contact. In our experience, avatar design is interface design and must occur in concert with a host of design decisions about the entire system.

At the same time, there are limits to the usefulness of the range-of-autonomy concept. Autonomy is not monolithic or linear; rather, avatar designers must identify various axes of autonomy. We believe that the necessary fundamental progress in avatar research can be made, not simply by expanding the number of options along a single dimension, but by rethinking the metaphors underlying the avatar-user relationship. Viljhálmsson and Cassell work with the metaphor of avatar-as-body; Mateas speaks of his avatars as being a kind of "magic glasses" which alter perception; John-

son et.al. build on the metaphor of the voodoo doll; Traces is based on the idea of a user leaving behind traces of her movement. In each of these cases, a metaphor that makes explicit the non-identity of avatar and user becomes the basis for a new technology. We suggest that thinking of avatars as semi-autonomous opens a space for new metaphors for the avatar-user relationship that move beyond simple identity, metaphors which can lay the groundwork for a new generation of avatar technology.

7 Conclusion

Traces was run as an installation at the Ars Electronica Center in Linz, Austria during the 1999 Ars Electronica Festival (September 4-9, 1999). It was shown for a total of 3 days to about 100 users per day. The logistics of presenting a single user project to a mass public necessitates some consideration to those waiting. At Ars Electronica 99, those waiting watched video feeds from the vision system and the front wall of the CAVE. This served the useful function of aclimatising them to the Traces environment. In order to allow as many people as possible to see the piece in a limited amount of time, we had to manage a timetable in which each user had a 4 minute experience. We story-boarded an experience through the three different types of behavior described in the previous section, which ramp up in complexity. Applying the principle of Autopedagogy, each stage equipped the user with new skills for the following stage.

At Ars Electronica we publicly demonstrated a new sensor system for the CAVE, which captures the full extent of the users body as usable input data. Untrained users engaged in a physically involving and easily comprehended "bodily" interface to an immersive environment. Users enjoyed engaging directly with dynamic virtual entities. The vision system has proven to be an excellent complement to the CAVE environment.The vision system hardware is low tech and inexpensive and in some cases obviates the need for any other sensor system.

The Traces project is ongoing. Future development will include:

- Networking two or more CAVEs. This was part of the original project proposal but was not implemented at Linz due to network limitations.
- Development of the vision system for reliable gaze orientation tracking to allow dynamic stereo image construction.

- Implementation of complementary spatial sound. An eight channel spatial sound system was developed by Jamie Schulte, but it was not possible to install this at Ars Electronica due to technical idiosyncrasies of the Ars CAVE.

Research is also continuing in the embodied and spatial interaction with a wide variety of semi-autonomous agents and agent behaviors.

8 Acknowledgements

Traces was conceived, designed, directed and produced by Simon Penny. André Bernhardt built the Traces vision system. Jeffrey Smith built the graphics code and EVL interface. Jamie Schulte designed the spatialised sound system and consulted on engineering issues. Phoebe Sengers built the Agent Behavior system of which the Behaving Trace is one expression.

Traces is funded in part by the Cyberstar competition (WDR and GMD, Germany), by the CMU Robotics Institute and Studio for Creative Inquiry, by a Fulbright fellowship, and by the EU eRENA Project. VRCO donated EVL Cavelib. The GMD provided time on their CAVE for development of Traces. Traces was first shown in the CAVE at the Ars Electronica Center as part of the1999 Prix Ars Electronica Interactive Art Prize.

References

[1] Bruce Blumberg and Tinsley A. Galyean. Multi-level direction of autonomous creatures for real-time virtual environments. *Proceedings of SIGGRAPH*, 1995.

[2] Richard Doyle. *On Beyond Living: Rhetorical Transformations of the Life Sciences*. Stanford University Press, 1997.

[3] Paul Edwards. *The Closed World: Computers and the Politics of Discourse in Cold War America*. MIT Press, 1997.

[4] Barbara Hayes-Roth and Robert van Gent. Storymaking with improvisational puppets. In *Proceedings of the First International Conference on Autonomous Agents*, pages 1–7. ACM Press, 1997.

[5] Jon O'Brien John Bowers and James Pycock. Practically accomplishing immersion: Cooperation in and for virtual environments. *Proceedings of the ACM 1996 Conference on Computer Supported Cooperative Work*, pages 380–389, 1996.

[6] Jaron Lanier. My problem with agents. In *Wired Magazine*. November 1996.

[7] Michael Mateas. Computational subjectivity in virtual world avatars. In Kerstin Dautenhahn, editor, *Proceedings of AAAI-97 Workshop on Socially Intelligent Agents*, pages 87–92, 1997. Available from AAAI as Technical Report FS-97-02.

[8] Michael Mateas. Subjective avatars. In *Proceedings of the Second International Conference on Autonomous Agents*. ACM Press, May 1998.

[9] Bruce Blumberg Christopher Kline Michael Patrick Johnson, Andrew Wilson and Aaron Bobick. Sympathetic interfaces. In *Proceedings of the CHI 99 Conference on Human Factors in Computing Systems*, pages 152–158. ACM Press, 1999.

[10] Simon Penny. *Culture on the Brink: Ideologies of Technology*, chapter Virtual reality as the end of the enlightenment project. Bay Press, 1994.

[11] Ken Perlin and Athomas Goldberg. Improv: A system for scripting interactive actors in virtual worlds. *Computer Graphics*, 29(3), 1996.

[12] Craig Reynolds. Steering behaviors for autonomous characters. In *1999 Game Developers Conference*, March 1999.

[13] Hannes Vilhjalmsson and Justine Cassell. Bodychat: Autonomous communicative behaviors in avatars. In *Proceedings of the 1998 ACM Conference on Autonomous Agents*, pages 269–276, 1998.

[14] J. MacGregor Wise. Intelligent agency. *Cultural Studies*, 12(3), 1998.