

SOME APPLICATIONS OF THREE-DIMENSIONAL INPUT

by

Christopher Schmandt

B.S. (Computer Science) M.I.T. 1979

Submitted in Partial Fulfillment
of the Requirements for the
Degree of
Master of Science in Visual Studies
at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

January 1980

© M.I.T. 1980

Signature of the Author _____

Department of Architecture
January 18, 1980

Certified by _____

Nicholas Negroponte
Associate Professor of Computer Graphics
Thesis Supervisor

Accepted by _____

Professor Nicholas Negroponte
Departmental Committee for Graduate Students

Rotch
MASSACHUSETTS INSTITUTE
OF TECHNOLOGY

FEB 25 1980

LIBRARIES



Room 14-0551
77 Massachusetts Avenue
Cambridge, MA 02139
Ph: 617.253.2800
Email: docs@mit.edu
<http://libraries.mit.edu/docs>

DISCLAIMER NOTICE

The accompanying media item for this thesis is available in the MIT Libraries or Institute Archives.

Thank you.

Some Applications of Three-Dimensional Input*

by

Christopher Schmandt

Submitted to the Department of Architecture on January 18, 1980, in partial fulfillment of the requirements for the degree of Master of Science in Visual Studies.

ABSTRACT

Three-dimensional, six degree of freedom input is explored in an interactive computer graphics environment. A particular device, the ROPAMS of Polhemus Navigational Sciences, Inc. is an accurate, unencumbering device based on electromagnetics. It is evaluated as a three-dimensional input device, and such input itself is evaluated for appropriateness and interactivity in a graphics environment. Emphasis is placed on human factors (pointing, body position) as a mode of interactivity.

This thesis includes both text and a videotape.

Thesis Supervisor: Nicholas Negro Ponte
Title: Associate Professor of Computer Graphics

*Sponsored in full by the Defense Advanced Research Projects Agency, Contract No. MDA 903-77-C-0037.

Acknowledgements

A number of individuals at the Architecture Machine have been of assistance in various phases of this project, and I wish to thank them.

Professor Negro Ponte, my thesis advisor, contributed support, both of finances and hardware resources, as well as the innovative perspective which allowed the Laboratory to participate in these experiments.

Andrew Lippman, Research Associate, provided much hardware consultation, organized the original purchase of ROPAMS, and helped debug some of my ideas. Dr. Richard Bolt likewise has provided ideas and refinements through the entire project.

Eric Hulteen helped with most of the voice input programming of "Put-That-There." Paul Heckbert was invaluable in some of the work in three-dimensional perspective space. Blake Smith deserves much credit for the typing, without which I never could have met my deadline.

Finally, all the various members, from professor to undergraduates, of the Architecture Machine Group deserve praise for the attitude of research and intellectual cooperation they have fostered. That collective work spirit kindled my own enthusiasm for this research.

Table of Contents

	Page
INTRODUCTION	5
THE THREE DIMENSIONAL DEVICE	7
The Scope of Study	10
THE TESTING FACILITIES	14
The "Virtual Flashlight"	15
PAINT.	17
Scrolling Text by Head Attitude.	19
Parallax	22
Controlling Visual Objects in Motion	23
"Put That There"	25
"Talking Heads".	27
CONCLUSION	29
REFERENCES	30
APPENDIX I ROPAMS - Interdata Operation	31
APPENDIX II Videotape Description	36

INTRODUCTION

The desire to gain a higher degree of interactivity with computer generated images has led to a variety of approaches to three dimensional input devices. Early efforts have utilized sound [1], light [2,3], and mechanical linkages [4,5,6,7,8] to ascertain x,y,z positions of an input device. While increasing the range and intuitiveness of use of input parameters over tablets, joysticks, etc., all have proven either cumbersome or spatially limited due to problems of obstruction by the user's body.

A three dimensional digitizer free of many of these limitations, and hence amenable to a wide scope of applications in which it is imperative that the user feel natural and unencumbered, opens new doors to interaction with computers in human terms. Such a system, based on electromagnetics and originally designed for aircraft systems control applications, is manufactured by Polhemus Navigational Sciences, Inc. of Essex, Vermont. The ROPAMS (Remote Object Position and Attitude Measurement System) device acquired by the Architecture Machine Group, has been used in a variety of experiments into the nature of three dimensional graphical input and the feasibility of use of such input to control sophisticated processes through natural gestures.

The scope of this current work has included interfacing

this new device to an already existing computer graphics laboratory. A series of experiments or demonstrations has been conducted both to evaluate the ROPAMS as a particular device and also to explore possibilities of three dimensional input to the general problems of graphical interactivity.

THE THREE DIMENSIONAL INPUT DEVICE

The ROPAMS (also referred to as "the Polhemus" in this paper), is a full six degree of freedom measuring device, providing both x,y,z position as well as attitude described by the three Euler angles on a sensor body with respect to a reference radiator body. Working with electromagnetics, it eliminates both the burdens of linkages as well as transmission problems associated with light and sound.

The position and orientation measurement is based on near-field, quasi-static characteristics of an electromagnetic field. The system consists of two independent remote components, connected via electronics to a Nova minicomputer (see figure 1). One of the remote bodies, the radiator, consists of electronics for transmitting a signal and an antenna system mounted in a 4 cm. plastic cube. The sensor body consists of a set of receiving electronics, connected by a flexible cable to another antenna system housed in a smaller, 1.5 cm. plastic cube.

The transmitting component of the system produces a nutating dipole field. As the antenna system consists of three mutually orthogonal stub dipoles; varying signal strengths in each of them allows the axis of nutation to be pointed in an arbitrary direction. Measurement of the time variance in the signal induced in the receiver allows the

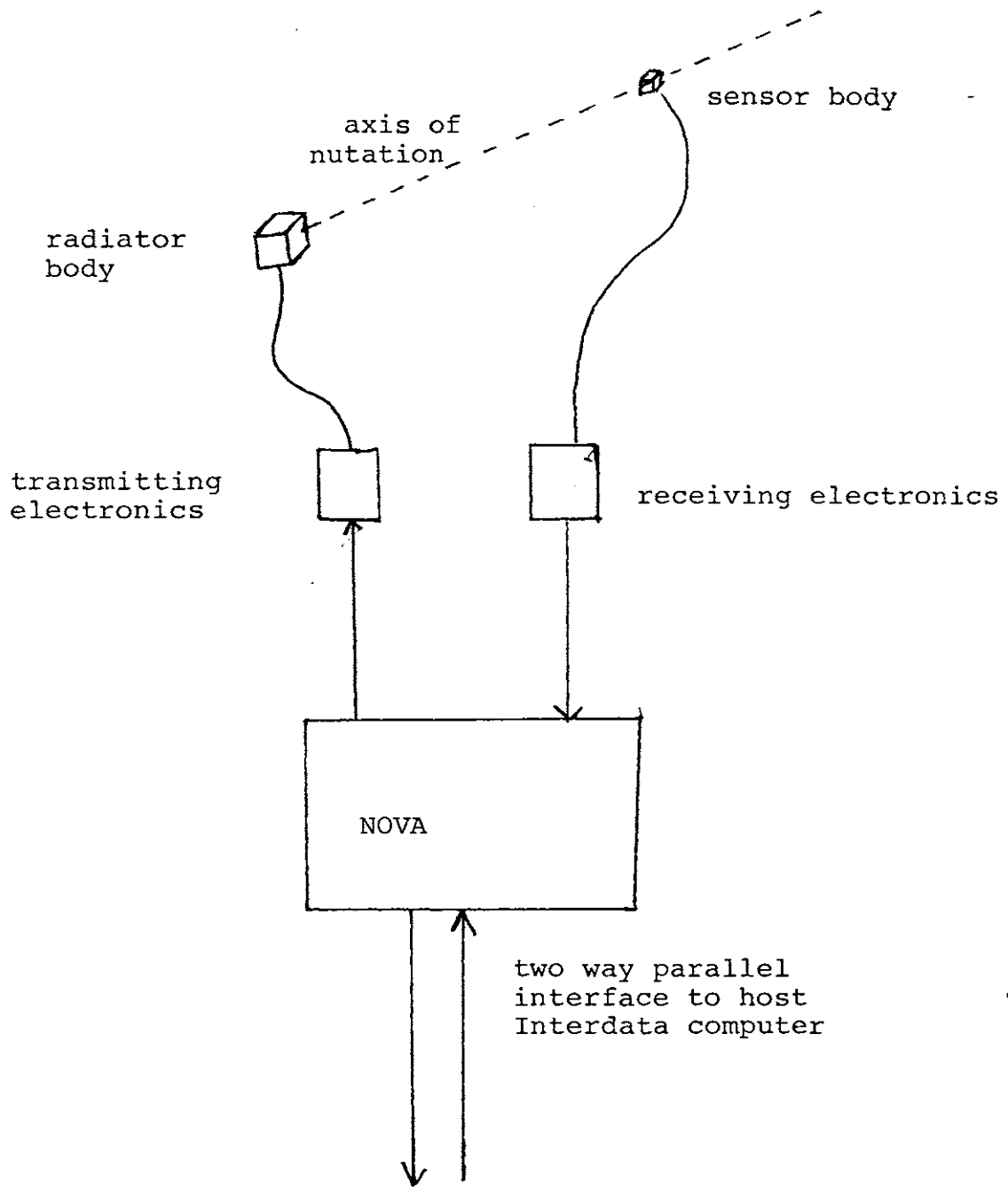


Figure 1. The ROPAMS configuration

axis of nutation to be pointed at the receiving sensor itself, thereby defining its position to be on a line of particular orientation with respect to the radiator. Distance along this line can be computed from induced signal strength, which is inversely proportional to radius cubed. The attitude of the sensor antenna can be derived from differential relationships of currents induced in each of its three mutually orthogonal receiving dipoles.

This system generates up to 40 points per second, with accuracies of better than 0.1 inch of position information and 0.1 degree of attitude angle. In the ROPAMS configuration, two independent sensor/radiator pairs are provided; either may operate alone, or both may operate simultaneously in time-multiplied mode. The range of the system is primarily limited by size of the transmitting body, and in these applications has been found to be approximately four feet; accuracy and response improve as distance between radiator and sensor decreases.

Computation is done in the Nova processor, which also maintains communication to the Laboratory's host computer, an Interdata 7/32, via a DMA parallel interface. The interface and software allow two way communication between these computers. An interrupt driven data path allows each newly computed point to be transmitted to the Interdata host by the

Nova. In addition, a separate interface allows simultaneous, status driven mode control of the Nova; modes include first, second, or multiplexed radiator pairs, insertion of software filters at the Nova end, etc. In addition, this interface is used to download the Nova with the ROPAMS core image as needed.

The Scope of Study

As mentioned earlier, two main areas were explored in an initial series of experiments. The first of these consisted of an evaluation of the ROPAMS device in particular based on a range of parameters which any graphical input device must be measured against. Linearity concerns the relationship between a change in output values versus a change in input position across the physical range of the device. Repeatability compares output values upon repeatedly returning to the same physical position. Responsiveness is computation time needed to calculate successive points, though tolerable levels of responsiveness will clearly be at least partially a function of the task being driven by the input device. Range in this case refers to the maximum and minimum (in this case, effectively contact) distance between the radiator and sensor.

The Polhemus device was found to be outstanding in almost

all the above mentioned areas, although on occasion forty points per second, or half that in multiplex mode, is less than comfortable. Of course, range is always a limitation; one would like to pick up a three dimensional device and walk around the room with it. Within the expected local range of three to four feet, however, accuracy remains remarkably high and predictable. A larger radiator antenna, if available, would increase the range. There has been considerable problems with noise produced by radiation from color CRT's in the vicinity of the radiator or sensor; it is hoped this will be eliminated by addition of a notch filter in the receiving electronics.

The other area of evaluation covers a broader issue of the appropriateness of a six degree of freedom input device to the world of graphical computer communication. This includes both its effectiveness as an input device, and imaginative applications which may be derived from the more obvious initial ones.

The first feature of six degrees of freedom is the removal of constraints; there need be no physical contact or even close proximity between radiator and sensor, as opposed to the proximity required, for example, with a tablet/pen combination. This begins to make ease or "naturalness" of interaction higher, more "obvious" to the naive user not

understanding principles of tablets, light pens, etc. This lends to the inclusion of a whole range of natural human gestures around pointing; pointing can be done from a variety of positions, the key being that the gesture indicates not a point in space, but rather a vector which intersects the object of interest.

Conscious gestures such as pointing at an object to control it raise issues of hand/eye coordination. Without going into an indepth series of experiments, practical experience by a number of users indicates that gestures may be easily performed on visual objects provided some form of visual feedback is included, e.g. a cursor on the display being pointed at or into. With feedback available, pointing out an object with a three dimensional input device is almost as natural as pointing out across the street to a friend.

Another whole field of applications is derived from less consciously directed three dimensional body position data. Sensor cubes may be mounted on the body in various ways, and used to compute continuous positional data above it. This is particularly possible as body mass doesn't interfere with the electromagnetics. This data may be used to update display processes via far more subtle "gestures" than pointing or hand waving - e.g. head attitude, head position, etc. [9]. The ability to use such inputs, which

greatly increases the range of human body factors which may be considered to be inputs, is possible mainly due to the relative physical independence of radiator and sensor bodies, and the unencumbering nature of the device.

THE TESTING FACILITIES

Experiments with the Polhemus 3-D digitizer were conducted using the facilities of the Architecture Machine Group's "Media Room" facility [10]. The Media Room is a room designed to be thought of as a walk-in computer terminal environment; as such it includes an easy-chair instrumented with touch-sensitive panels and joysticks, two touch-sensitive color television monitors, and an octophonic sound system embedded in the walls. One wall of the 16 feet deep by 10 feet wide by 8 feet high room is a wall-to-wall, floor-to-ceiling 8x10 feet video screen, rear projected from a General Electric Light Valve.

For these applications Polhemus radiators are placed to the immediate left and right of the chair in the center of the room, and sensors used in various combinations and mountings. The user sits in the chair, facing a "video wall" (not a "window," there are no edges around the screen). The size of the screen creates a large, immediately present graphical environment to manipulate via three dimensional input. Software on the Interdata processor reads data from the Nova/Polhemus, and drives a Ramtek 9300 frame buffer, which produces video displayed by the light valve on the large screen (see figure 2). In addition, for some applications a Nippon Electric DP-100 Continuous Speech Recognition System is employed, connected to the Interdata by a separate interface.

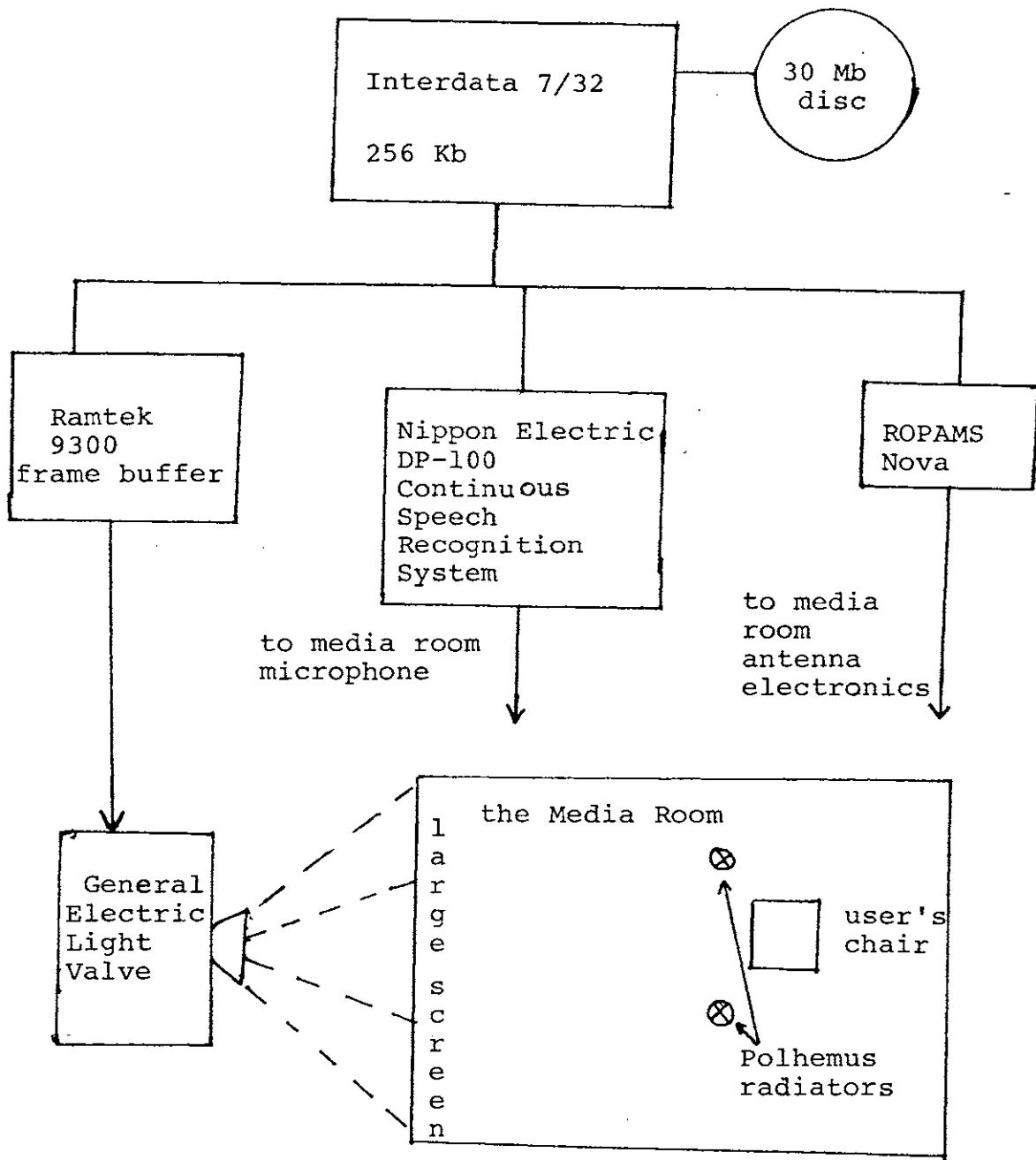


Figure 2. The testing facilities.

The "Virtual Flashlight"

The initial experiment was used primarily as a system debugging and alignment tool. A Polhemus sensor was attached

to a plastic flashlight; the flashlight bulb was connected by wires to a remote power supply (batteries in the flashlight would have interfered, because of their ferrous mass, with the digitizing system).

Software assumed the digitizing radiator was in a certain position in the room with faces parallel to the walls of the room. For a given piece of sensor locational data from the Nova, the following computations were performed. The x,y,z location of the sensor, relative to the radiator, combined with knowledge of the radiator's position in the room, and the physical dimension of sensor positional measurement units relative to physical size of a single pixel on the large screen, all lead to the position of the sensor in screen pixel x,y,z coordinates. This corresponds to knowledge of the sensor's absolute location in the room. The azimuth and elevation angles of the sensor project a line with the sensor as an endpoint. The intersection of this line with the plane of the screen give the screen coordinates of the spot pointed to by the sensor (and in this case, the flashlight also). This same projection routine continues to be used extensively in other applications.

Having obtained the point on the screen pointed to by the flashlight, a colored square is drawn there against a black background. As the point moves, the square follows it.

When all scale factors are correct, it is possible to turn on the flashlight in the darkened media room and have the square follow the moving area illuminated on the screen by the flashlight. Another version of this demonstration tracked two different color squares, one for each of the sensors, running in multiplexed mode.

In fact, the spatial correlation of the light beam and the calculated tracking square is quite high, typically within six inches of being on center. Some error was produced by the sensor being off axis of the flashlight; a solution utilizing the roll of the sensor about the flashlight was calculated but not implemented.

What was learned was that the overall system configuration worked well enough and fast enough to track a real point in space convincingly. Linearity was remarkably high, and repeatability excellent. Range proved to be the limiting factor; beyond three to four feet tracking errors grew, particularly in the direction of sensor azimuth angle. Even beyond ranges of four feet, however, tracking did respond to sensor movements to some degree. The issue of hand-eye coordination will be discussed in the next section.

PAINT

In this demonstration, one sensor is used as a paint brush; as the sensor is moved, pointing at the screen, colored

"paint" appears on the screen where the sensor is pointing. As noted with the flashlight experiment, five degrees of freedom (x,y,z, azimuth, elevation) are required to determine where the sensor is pointing.

Brush size was determined by distance from the screen; the nearer the hand-held sensor is to the screen, the larger the brush. Paint color is selected by the roll (or "twist") of the sensor; the roll angle selects a color from a rainbow wash of colors. In one hemisphere of roll a colored brush is selected; in the other hemisphere no paint is drawn, but a tracking cursor displayed, to allow drawing of discontinuous line segments. To aid comprehension of the color/cursor versus roll angle relation, the sensor was attached to a "magic wand" of wood, with colored stripes placed at appropriate angles. In addition, pointing at a red square in the upper left corner of the screen clears the screen.

The point of this demonstration is not to be the ultimate in computer paint programs. It has been, however, the first time one has been able to interact directly with the large screen display, you "draw where you point," literally. In addition, any painting program tests many parameters of an input device, as it is graphically responsive.

In this case, our previous conclusions about linearity, repeatability, and range were confirmed. Responsiveness

was found to be generally good, though slightly lagging input, enough to be noticeable to even a naive user.

Use of the software filter included in the ROPAMS Nova driver increased lag sufficiently to be rather annoying, so it was not used much after this point.

In attempting to draw figures, write words, etc., hand-eye coordination becomes both more of an issue and more measurable. It was found that, provided there is some visual feedback, e.g., a cursor or paint being drawn, the user can do quite well at pointing where he wants, even "painting from the hip." It was observed that attention is paid to the visual feedback on the large screen rather than hand position; as long as changes in hand position produce the same relative changes in visual position, a high degree of accuracy is felt. This suggests that linearity and repeatability are far more crucial than absolute position in the context of visual-manual 3-D coordination of input.

Scrolling Text by Head Attitude

Another set of experiments involves head mounted sensor applications. For these, a sensor is attached to the visor of an ordinary baseball hat; the cable from the sensor passes over the hat and down behind the user's back in some comfortable fashion.

The first such application controls the rate of scroll of text on the large screen. Text is ordinarily scrolling upwards - i.e. forward. As the reader, with head mounted sensor "looks" toward the bottom half of the screen, text continues to advance. (Here, "looking" implies maintaining a particular head elevation - i.e. the visor is tilted downward). Looking at the middle of the screen freezes the text, and looking up toward the disappearing top lines of text initiates a scroll backwards (i.e. downwards), to bring in lines of text which may have been missed. In addition, glancing to either side of the screen freezes the text motion, as it is assumed the reader is no longer paying attention and scrolling should not proceed unnoticed.

Two basic issues are touched on in this demonstration. The first is beginning to utilize unconscious, or perhaps more appropriately, reflex-level body actions to control a process. As opposed to painting, which requires a conscious "pointing" gesture, and hand-eye coordination, tilting the head to read text scrolling by on a large screen is a fairly "natural" reaction. Utilizing head attitude to control the rate of scrolling is a step toward designing a system responsive to ordinarily instinctive levels of body communication.

The second issue addressed is the relationship between

head attitude and the point of visual attention. If the text display was sufficiently small, very little head motion could be expected to be measured between reading the top and bottom of the screen, making head attitude only marginally useful. Clearly, the larger the display, the larger the excursions of head angle will be. We found that head attitude tended to stay within approximately five degrees of visual line of sight; in the case of our eight foot high display, this meant that reliable resolution of head elevation was better than necessary to make assumptions about area of visual focus.

The implication is clearly that head attitude is strongly correlated to the location of visual attention; in the case of interactive visual displays, head attitude is a valid indicator of what the user is looking at.

A relevant feature of the success of this demonstration is the use of relative head attitudes, i.e. changes in elevation and azimuth from some initial rest position. As the demonstration begins, the user is asked to sit back comfortably and pay attention to the middle of the screen. The head attitude in this position is taken as a reference, and relative offsets from this reference control the text scrolling. This compensates for different users' idiosyncracies - how they wear the hat, whether they sit rigidly or lean back somewhat in the chair, etc.

Parallax

In another head mounted application, head position is used to produce parallax effects on the large screen video display. In this case, a red rectangle partially obscures a piece of text underneath it. As the user, wearing a head mounted sensor, tries to "peek" above or around the obscuring rectangle, either it or the text moves appropriately to simulate parallax. The effect, quite convincing, is as if the rectangle were situated in space approximately half way between the user and the screen.

Two different modes of relative motion were used. In one, the text remained stationary and the rectangle moved in front of it. This would seem to correspond more closely with the actual visual parallax perception. In the other, the rectangle remained stationary and the text moved in and out behind it. Various users found one or the other more convincing.

In this application, gross body position (again, relative to some initial rest position) is used to drive the graphical output. Again, it is a reflex action once the user realizes he can control the display exactly as he would look around objects in real life. One can imagine applications in which it is desired to drive the graphical output. Again, it is a reflex action once the user realizes he can control the display exactly as he would look around objects in real life. One can imagine applications in which it is desired to display

a larger number of data items than could reasonably fit on screen; the user could select a particular item by peeking around stacked items and selecting a particular layer for expansion. In addition, parallax effects lend another level of reality, as well as "navigational aids," to 2 1/2 dimension "data spaces" [11].

Controlling Visual Objects in Motion

Before moving on to discussion of more sophisticated multi-media applications of 3-D input, several game "hacks" deserve mention. The point of these examples is interaction, the nature of any video game, in particular, interaction with moving objects, by pointing and other hand gestures, using multiple degrees of freedom.

The first, "catch the ducks," involves "ducks" flying somewhat randomly across the screen. A single player tries to catch a duck by pointing at it and stuff it in a "basket" before it has had time to escape off-screen. Up to six ducks may be on screen at a time. A duck is "caught" by pointing within its boundaries; once caught, as long as the 3-D pointer remains within a certain distance from the duck's center, the duck follows the pointer (quacking angrily of course!). If the pointer moves too fast, the duck flies off again.

With a number of ducks on screen, hurrying to escape and flapping around somewhat, players start to scramble, keeping track of several objects at once, and madly grabbing and pushing ducks around. When a cursor shows the position of the pointer, it is possible to catch nearly all the ducks; without the cursor players do well to catch any. Once again we see that with the proper visual feedback loop, a sophisticated level of hand-eye coordination is quickly obtained, even in dealing with moving objects with a "mind of their own;" without the feedback, the task is vastly more difficult.

The other example of interacting with motion is a version (called "phong" for idiosyncratic reasons) of the popular video game "pong." Phong uses both digitizing systems, running in multiplexed mode. Not only does a player's paddle appear on screen where the sensor is pointing, in addition the ball changes velocity as a function of x,y acceleration of the paddle it is hit by. Spin is imparted by the roll angle of the sensor, and the ball gains speed over the course of a volley.

Multiplexed digitizer mode slows response down somewhat more than is comfortable, but experienced players can maintain some fairly intense volleys. Given the speed of the ball and responsiveness of the paddles, however, most of the action consists of reacting to opponents shots, rather than

planning spin, etc., to give a tricky return. Trying to control such a fast paced environment utilizing all six degrees of freedom is too difficult without some period of training and concentration. Of course, using a more intuitive sensor mounting (e.g., attached to a ping-pong paddle) would aid immensely.

"Put That There"

The small, unencumbering Polhemus digitizer has opened new avenues of human interaction - gesturing, pointing, exploiting body attitudes, etc. Particularly exciting in this interaction is its sense of immediateness and natural ease of use; for example, literally "painting" a video wall by the simple art of pointing at it. In exploiting natural paths of human communication as the languages of computer interface, we are tempted to explore multi-modal human inputs, specifically the junctions of speech and gesture.

To this end, a Nippon Electric DP-100 Continuous Speech Recognition Unit is configured to communicate with the central Interdata processor. The DP-100 has a user defined vocabulary of up to 120 "utterances" (words or phrases). After being trained by speaking each word once, it will recognize and report back "sentences" of up to five such utterances in connected (i.e., "normal," unbroken) human speech.

"Put that there" [12] joins gesture and speech recognition to allow the user to build and modify a graphical data base by pointing and speaking. "Draw a small yellow triangle there" spoken while pointing at the screen causes the triangle to be drawn there, where the user is pointing. Objects may be created from a small repertoire of shapes, colors, and sizes, simultaneously building a data base in software. Shapes may also be created by relationship to other objects, e.g., "Create a blue circle to the right of the red square," or "make a circle below that square," while pointing at a particular square.

Objects may likewise be modified, e.g., "Move that... there" with two pointing gestures. Similarly, commands may take forms such as "make that a blue square," "move that... below that," (with two pointing gestures), "erase that," "change the blue square to red," etc. The data base is updated appropriately as well. Likewise, various calls to the operating system can be made by voice command, or the previously described painting program initiated.

The most striking feature of this demonstration is the sense of presence, immediacy and intuitive clarity of the interaction. A Polhemus sensor has been mounted on a leather watchband; strapped to the wrist, it allows the user freedom to wave his hands and point freely at the large data surface,

the video screen, immediately at hand. Command language is intuitive and simple; continuous speech and pointing. Although this particular demonstration involves a very simplistic graphical environment, one can imagine more complex symbols, drawn as non-destructive overlays on maps, diagrams, etc., and causing some higher level of computation or control functions.

In particular, it is the combination of hand gesture and speech which both is responsible for the enveloping sense of interaction and also allows a richness of command structure. Clearly a large range of manipulations require both a pointing (to disambiguate a reference or select a locality) as well as speaking component. In addition, knowing where the user is pointing may be used to correct ambiguities in the data returned by the speech recognizer, which is less than perfect in its recognition; a redundancy of control inputs provides an opportunity to resolve conflicting interpretations of their results. If the speech recognizer reports that a user is talking about "that diamond" while pointing to a square, it is probably incorrect in its voice interpretation, and the combination of pointing and knowledge of the data base allow for error correction.

"Talking Heads"

Another instance of the usefulness of real time three-

dimensional digitization of body position is a facet of a novel tele-conferencing scheme. Take an example of two (or more) remote users. A live conferee speaks to plastic faces of the remote participants, while a video camera immediately opposite is focused on his face. For each participant, both audio and video are transmitted. Each of the plastic masks at the remote ends are sculpted replicas of the particular remote participants (in the same relative location at each conference site), with the video of that person projected onto the mask; the result is astoundingly lifelike. In addition, each plastic head moves to mimic the head motion of the remote live person.

A demonstration of this effect was produced using a single mask and film projector. The moving head was achieved using hardware which originally used a film audio track to record two dimensions of camera attitude for later playback via mirrors as a "moving movie" [13]. In this case, during filming a Polhemus sensor was taped to the back of the subject's head. Two dimensions of head attitude were deduced from sensor coordinates and attitude, and values computed to drive two channels of D/A converters to produce appropriate voltages for the camera audio attitude recorders. In playback, the recorded attitude audio drives the head mask, while an image is projected on it to reflect the head attitude of the filmed subject.

CONCLUSION

Electromagnetics based six degree of freedom input is a new addition to the world of interactive computer graphics. Practical applications of ROPAMS have proven it to be a reliable and flexible input device. In particular, the relative unencumbering nature and absence of positional restrictions on the sensor body, make it ideally suited for a new range use of body position to control sophisticated processes.

The general theme of any input device is interaction. The work described in this thesis indicates that three-dimensional input is ideally suited to the world of computer graphics. In fact, we are still in a period of innovation with three dimensional input; every new application suggests yet another generation of ideas and experiments to further apply this input to humanized computer interaction.

REFERENCES

1. Roberts, L.G., "The Lincoln Wand," FJCC 1966, Spartan Books, Washington, D.C., p. 223.
2. Burton, R.P., "Real-Time Measurement of Multiple Three-Dimensional Positions," University of Utah Computer Science Department, UTEC-CSC-72-122, 1972., NTIS AD-762 028.
3. Burton, R.P. and Sutherland, I.E., "Twinkle Box: A Three-Dimensional Computer Input Device," NCC 1974, pp. 513-520.
4. Clark, J.H., "Designing Surfaces in 3-D," CACM, 19(8): 454, August 1976.
5. Geyer, K.E. and Wilson, K.R., "Computing With Feeling," Proc. IEEE Conf. on Computer Graphics, Pattern Recognition and Data Structure, May 1975, pp. 343-349.
6. Vickers, D.L., "Sorcerer's Apprentice: Head-Mounted Display and Wand," Ph.D. Dissertation, Dept. of Electrical Engineering, University of Utah (1973).
7. Sutherland, I.E., "A Head-Mounted Three Dimensional Display," FJCC 1968, Thompson Books, Washington, D.C., p. 757.
8. Noll, A.M., "Man-Machine Tactile Communication," Ph.D. Dissertation, Dept of Electrical Engineering, Polytechnic Institute of Brooklyn, (1971).
9. Negroponte, N. and Bolt, R.A., "Data Space," MIT Architecture Machine Group proposal to Cybernetics Technology Office, Defense Advanced Research Projects Agency.
10. Negroponte, N., "Media Room," MIT Architecture Machine Group report, 1979.
11. Bolt, R.A., Spatial Data Management, MIT Architecture Machine Group report, 1979.
12. Bolt, R.A., "Put-That-There:Voice and Gesture at the Graphics Interface," Paper submitted to SIGGRAPH '80.
13. Naimark, M., "Spatial Correspondance:A Study in Environmental Media," unpublished M.S. thesis, MIT, 1979.

Appendix I: ROPAMS - Interdata Operation

Loading the Nova

The Nova core image is downloaded from an Interdata, where it is stored on disc in the file system, via the parallel interface. Since ROPAMS uses only the lower half of the available Nova memory, downloading and other programs run in the upper half.

To download, the following code must be toggled into the Nova from the front panel, starting at location 40000.

(All Nova numbers are in octal.)

40000	062026	DOB, 0, 26
	060126	NIOS, 26
	060326	NIOP, 26
	063526	SKPBZ, 26
	777	JMP, -.1
	063077	HALT

ACO, (starting address to download to) must be set to contain zeros. Start the Nova at 40000, reset, and start again (this cleans out the Interdata to Nova interface).

Now, on the Interdata run load-nova. It will ask for a "rev number." In general, the highest rev number is the current version of ROPAMS; core images live in >u>cms>nova and are named nova_image.<rev number>. Load-nova will print a few lines telling how much has been down loaded; if it doesn't, either the interface isn't functioning or the Nova isn't executing the correct code.

Interfaces and Data Transfer

The Nova-Interdata interface consists of two independent devices. The Interdata to Nova side is device x'51' on the Interdata, the Nova to the Interdata side is device x'50'.

In normal operation, Polhemus points are transferred to the Interdata via an interrupt driven driver. The Interdata side of the interface receives an interrupt for each byte transferred as well as end of transmission; status bits indicate which case has occurred. The interrupt handler, rpol-inth, watches for an end of transmission interrupt to sync up on the data transfer, then continuously loads successive data bytes into the correct addresses of an input buffer. Higher level Interdata software translates this raw data into more conventional form (position and attitude are transmitted as signed twelve bit numbers).

Running the Nova

Assuming a correct core image, the following procedure will run ROPAMS. Put 2000 (the starting address) in the switches, and hit the "examine" switch. Then set the front panel for the desired operation mode, and hit the "continue" switch. The Nova should be running, and data will be transferred to the Interdata when its interrupts are enabled.

The front panel determines operation mode, as follows:

	<u>bit number</u>	<u>function</u>
MSB	0	read mode from Interdata
	1	not used
	2	not used
	3	insert software filter in sensor 2
	4	Boresight sensor 2
	5	test sensor 2
	6	standby sensor 2
	7	normal sensor 2
	8	not used
	9	not used
	10	not used
	11	filter in sensor 1
	12	Boresight sensor 1
	13	test sensor 1
	14	standby sensor 1
LSB	15	normal sensor 1

When reading the mode from the Interdata, the bit pattern in the Interdata to Nova interface is used instead of the front panel. The routines one, two and both run the respective sensors in normal mode from an Interdata.

Interdata Driver Software

Several levels of software at the Interdata end interpret the raw Polhemus data. Pol is the lowest level, and has the following entry points:

<u>set up</u>	loads the interrupt handler, enables interrupts from the Nova
<u>clear</u>	disables interrupts and clears the interrupt handler
<u>start</u>	enables interrupts only

stop disables interrupts only

get takes an argument, pol-ptr, a pointer.
It sets the pointer to a structure pol-data,
declared as below

get d same as get, with attitudes expressed in
degrees rather than radius

```
dcl 1 pol-data based (pol-ptr),
    2 psys(2),     /* one for each sensor */
    3 az float
    3 el float,
    3 roll float,
    3 x fix,
    3 y fix,
    3 z fix,
    3 valid fix, /* doesn't work, ROPAMS bug */
    2 packets fix /* serial Polhemus packet number */
    2 which fix, /* most recently received sensor number */
```

Another routine, rpolly, operates similarly to pol except that it returns x,y coordinates of the screen location being pointed to by the sensor body. To become oriented in the room, one must first call rpolly\$init (inits) where

```
dcl 1 inits (2)
    2 x, fix
    2 y, fix
    2 z, fix;
```

The x,y, and z are the approximate coordinates in the room of each radiator, in Ramtek coordinates at the same scale as on the Light Valve screen.

Now, calling rpolly (sys, screen_data) does the following. Sys is a bit array, each bit indicates whether to convert data for that sensor. Screen_data is the resulting structure of values.

```

dcl 1 sys (2) bit (1) ;

dcl 1 screen_data,
  2 polly (2)          /* one element per sensor */
  3 az float,         /* in radians */
  3 el float,
  3 roll float,
  3 x fix,            /* Ramtek coordinate */
  3 y fix,            /* Ramtek coordinate */
  3 z fix,            /* Polhemus units */
  3 valid bit (1),   /* doesn't work, ROPAMS bug */
  3 clipped bit (1), /* true if the screen coordinates
                       were clipped */
  2 packets fix,     /* most recent packet number */
  2 which fix,       /* most recent data sensor number */

```

Appendix II: Videotape Description

The videotape accompanying this text is meant to assist understanding of the various demonstrations. It runs as follows:

Log number	
10	the virtual flashlight, sensor and radiator illustration
45	PAINT
95	Parallax with head position
115	Scrolling text by head attitude
155	"Put that there"
240	Talking heads tele-conferencing