

Browsing 3-D spaces with 3-D vision: body-driven navigation through the Internet city

Flavia Sparacino
MIT Media Lab
flavia@media.mit.edu

Christopher Wren
MERL
wren@merl.com

Ali Azarbayejani
MIT Media Lab
ali@media.mit.edu

Alex Pentland
MIT Media Lab
sandy@media.mit.edu

Abstract.

This paper presents a computer vision stereo based interface to navigate inside a 3-D Internet city, using body gestures. A wide-baseline stereo pair of cameras is used to obtain 3-D body models of the user's hands and head in a small desk-area environment. The interface feeds this information to an HMM gesture classifier to reliably recognize the user's browsing commands. To illustrate the features of this interface we describe its application to our 3-D Internet browser which facilitates the recollection of information by organizing and embedding it inside a virtual city through which the user navigates.

1. Introduction

Recent technological progress allows today most home users to be able to afford powerful graphics hardware and computer processors. With this equipment people can navigate in sophisticated 3D graphical environments and play engaging computer games in highly realistic and fascinating 3-D landscapes [<http://gamespot.com>]. Such progress has not been paralleled by equivalent advances in man-machine interfaces to facilitate access and displacement in virtual worlds. People still use quite primitive and limiting interfaces: the joystick, button-activated game consoles, or the computer keyboard itself. Full immersion and skillful exploration of 3-D graphical environments is limited by the user's ability to use these interfaces, and the consequences of repetitive use often involve undesired and painful medical consequences to the user's wrists, fingers, arms, or shoulders [1]. New, more natural interfaces are needed to navigate inside virtual worlds.

On the other hand people spend today an increasingly large amount of time exploring the Internet: a bi-dimensional environment, which is quite unsophisticated and simple compared to the previously mentioned popular computer games. While the Internet allows designers to author and display animated web pages, with moving text and images,

the browsers we have available today are still quite primitive. Internet browsers are flat: they are based on the old multimedia metaphor of the book, with 2-D pages filled with links to other pages, and bookmarks as a memory aid, to represent and organize the information available on the net. The only advantage of such information-interface is its non-linearity and rapid, visible access to interconnected data. The main disadvantage is that it is easy to get disoriented while navigating the Internet, as we rapidly lose track of what we've seen before the current page, and do not have perspective of what is accessible from the current page. The Internet could benefit from the same 3-D graphical progress which has determined the surge of the computer games industry, and provide the public with 3D graphical browsers to help us better visualize, organize, and remember information.

This paper presents two connected contributions: an Internet 3-D web browser and a natural interface to browse it. Our browser is based on the architectural metaphor of the city and organizes information by embedding it inside a virtual urban space. Providing a natural interface to navigate in our 3-D web browser is similar to designing a new interface for a 3-D computer game.

2. The Interface

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. It follows that a natural and comfortable interface may be designed by taking advantage of these competences and expectations. Instead of using cumbersome and primitive game consoles, we should be able to interact with computers in a natural way.

Instead of strapping on alien devices and weighing ourselves down with cables and sensors, we should build remote sensing and perceptual intelligence 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.

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 ultrasound 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 [3]. 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.

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.

Our targeted interactive space is the desktop. Our prototype desktop system consists of a medium sized projection screen (4'x5') behind a small desk (2'x5') [figure 1]. The space is instrumented with a wide-baseline stereo camera pair, an active camera, and a phased-array microphone. The wide-baseline stereo is used for visually tracking the macroscopic movements of the user. The foveating (pan-tilt-zoom) camera is used to obtain high-resolution images of an area of interest. The phased-array microphone is used to pick up audio from a direction of interest, usually from the user's head. 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 [figure 2]. This type of interactive space is suited for detailed work. The research described in this paper makes use exclusively of the stereo camera pair, while the other available input and output devices are used for other research and will be used in the future to improve the user's experience in the described application.

3. Background

Pavlovic [4] and Wu [5] have explored use of hand gestures in human computer interaction, with an emphasis on 3-D tracking and multi-modal approaches for gesture recognition. Starner [6] has shown one of the first examples of effective gesture recognition using HMMs. Brand, Oliver, and Pentland [7] have demonstrated the higher performance of coupled HMMs for tasks which require gesturing with both hands at the same time. Campbell and others [8] have studied the effects of the appropriate feature choice for a gesture recognition task, using stereo vision. Jojic, Brumitt, and Meyers [9] use stereo cameras to detect people pointing and estimate the direction of their pointing. As opposed to the blob tracking approach, they use disparity maps which are less sensitive to lighting changes. In our blob-based approach, light invariance can be achieved using adaptation, or implementing color invariant classification [10].

Our vision system is related to body-tracking research by Rehg and Kanade [11], and Gavrilu and Davis [12] that use kinematic models, or Pentland and Horowitz [13], and Metaxas and Terzopoulos [14] who use dynamic models. Such approaches require relatively massive computational resources and are therefore not appropriate for human interface applications. These systems all require accurate initialization and cannot deal with occlusions. Functionally our system is most closely related to the work of Bichsel [15], and Baumberg and Hogg [16]. 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. Our interface goes beyond these systems by also building a blob-based model of the person's head and hands in 3-D.

Dodge and Kitchin have shown that 3-D web modeling is an active field of research in the 3D visualization and computer graphics communities [17]. The authors have pioneered research in 3-D web visualization [18] and have explored 3D web browsing with pointing gestures [19]. Waterworth [20] proposes criteria for the construction of three

dimensional personalized web spaces to help users organize web information. Modjeska [21] conducted user testing to prove that people prefer to explore the web as a structured 3-D virtual world rather than a

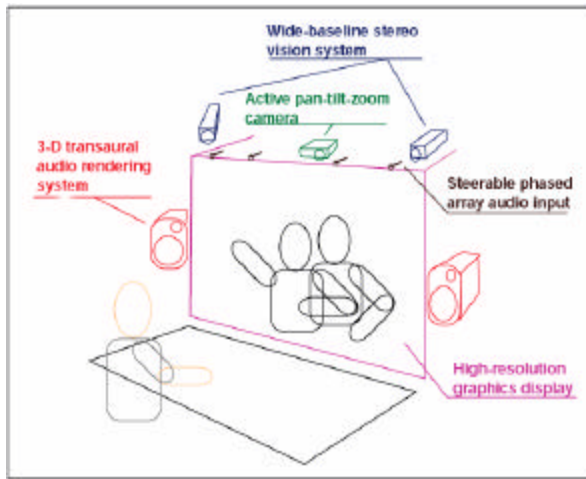


Figure 1. The Interactive Setup

4. City of News: an Internet City in 3-D

The language we use today to describe the Internet makes a constant reference to a place. We ask people their Internet address, we call a web page a site, and our site a home page, we meet in chat rooms, and so on. However the browsers we have currently available use the old metaphor of the hypertext and the book, with only one page visible at one time, and bookmarks to help our wayfinding. There is a mismatch, a cognitive dissonance, between the way we imagine and talk about the Net, and the means we are provided to access it.

City of News is an immersive, interactive web browser that makes use of people's strength at remembering the surrounding 3-D 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 3-D graphical world projected on a large screen. This gives the user a sense of the URLs existing in a surrounding 3-D environment and helps the user remember the previous exploration path leveraging off his/her spatial memory. The URLs are displayed so as to form an urban landscape of text and images through which the user can navigate [figures

plain 2-D hypertext. Adobe has recently launched a commercial 3-D web browser [<http://www.adobe.com/products/atmosphere>].

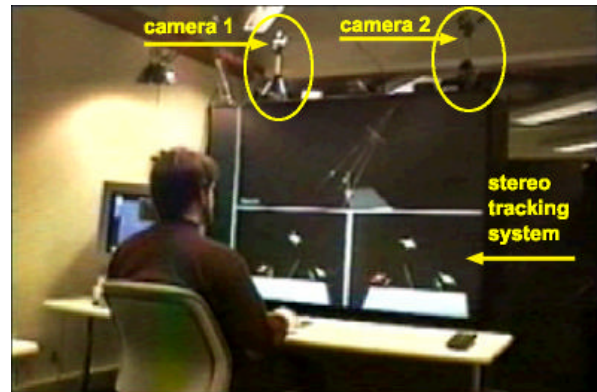


Figure 2. User at the Interactive Setup

3, 4]. The 3-D web landscape is a city. Known cities' layout, architecture, and landmarks are input to the program and are used as orientation cues and organizing geometry. This virtual internet world grows dynamically as new information is loaded, and anchors our perceptual flow of data to a cognitive map of the virtual internet city. 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 "façade".

By mapping information to familiar places, which are virtually recreated, City of News stimulates in its users association of content to geography. The 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' [23] classical memory-palace information memorization technique.

The browser currently supports standard HTML with static jpeg and gif images and MPEG movies. The program is written in C++ and SGI OpenInventor, and uses RPC to communicate with the computer vision module in a client-server architecture. Given a city support map at start, the software works in several steps:

1. For each newly loaded web page it parses the HTML file and builds an intermediate representation for the 3-D graphics render.
2. While with a separate program thread it starts loading all images associated to the target HTML file

and saves them on the local browser cache space.

3. After determining the location where to place the web page in the map it calculates the available page width and the necessary page height, based on the loaded HTML data.

4. It builds separate graphical nodes of each text and image element to render.

5. It finally assembles all the graphical information nodes into the new 3-D building, and renders it in the internet city.

To navigate this 3-D environment, users sit in front of the large screen and use hand gestures to explore or load new data. Pointing to a link will load the new URL page. The user can scroll up and down a page by pointing up and down with either arm. When a new building is raised and the corresponding content is loaded, the virtual camera will automatically move to a new position in space that constitutes an ideal viewpoint for the current page. Side-pointing gestures allow users to navigate along an information path back and forth. Both arms up drive the virtual camera 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 constrain navigation so that the user is never lost in the virtual world.

5. Real-Time 3-D Tracking

5.1. 2-D Blob Tracking

The real-time 3-D tracking is a method for estimation of 3-D geometry from blob features. The notion of “blobs” as a representation for image features has a

long history in computer vision. The term “blob” is somewhat self-explanatory (“something of vague or indefinite form”), but a useful definition from a computational point of view might be that a blob is defined by some visual property that is shared by all the pixels contained in the blob and is not shared by surrounding pixels. This property could be color, texture, brightness, motion, shading, a combination of these, or any other salient spatio-temporal property derived from the signal (the image sequence). Blobs are usually thought of as regions with dimensions that are roughly the same order-of-magnitude, in part because we have special terms for other features, e.g., “contours”, “lines”, or “points”. But these other features can also be viewed as degenerate cases of blobs, and, in fact, straight contours and points are perfectly well represented by the blob model.

Our current interest in blob models is motivated by our discovery that they can be reliably tracked even in complex, dynamic scenes, and that they can be extracted in real-time without the need for special purpose hardware. These properties are particularly important in applications that require tracking people, and we have used 2D blob tracking for real-time whole-body human interfaces [24] and real-time recognition of American Sign Language hand gestures [6]. In the setup described above, in which the upper body is used as the navigating interface to the Internet browser it is important to have a more exact knowledge of body-parts position. A monocular system would not be able to accurately recover the location pointed at by the user in the 3-D landscape. This is particularly important in our 3-D Internet browsing application for which a projection error onto the 3-D landscape can cause the user to navigate to a completely different location than what he/she intended.



Figure 3. Aerial view of City of News



Figure 4. City of News after exploration

Not having such precision available would be equivalent to having a mouse with a coarse resolution which can cause a user to click and launch undesired applications, or involuntarily click on a different link than the desired one. Using such a defective and imprecise mouse can be quite a frustrating task. A stereo tracking system, with the ability to recover the 3-D geometry of the user's input features – hands and head – is an important step towards precise and reliable man-machine interfaces to explore 3-D data.

For both 2-D and 3-D blobs, there is a useful physical interpretation of the blob parameters. The mean represents the geometric center of the blob area (2-D) or volume (3-D). The covariance, being symmetric, can be diagonalized via an eigenvalue decomposition to yield a rotation matrix and a diagonal size matrix. The diagonal size matrix represents the size of the blob along independent orthogonal object-centered axes and the rotation matrix brings this object-centered basis in alignment with the world coordinate basis. This decomposition and physical interpretation is important for estimation, because the shape is constant (or slowly varying) while the rotation is dynamic. The parameters must be separated so they can be treated differently.

We estimate useful 3-D geometry of the human body 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 goal is to recover the 3-D shape from the 2-D shapes.

5.2. Blob Finding

At start, the system looks in the image for seeds that correspond to pixels with a skin color. It visits the image along a widely spaced grid, as rastering all image pixels is unnecessary for seed-finding. A classification decision is made by log likelihood calculation in the MAP sense, as described in [24]. The skin class is defined at start by averaging several skin images taken in the same room as the setup, and by calculating the covariance matrix in YUV space. For every skin colored pixel-seed, the system applies iterative region growing centered around each seed to then determine all connected skin-colored regions in the image with the k-means algorithm. It then discards the smaller blobs until three blobs are found: left hand, right hand, and head. Simple heuristics allow the program to easily assign which blobs correspond to which body part: the head is usually on top and in the center, and at start the right hand is usually on the right-hand side of the head.

5.3. Stereo Blob Estimation

Stereo blob estimation is an ill-conditioned problem. In Figure 5, the top-left image shows a 3-D view of a simulated stereo system where each camera is represented by a virtual image plane and center of projection and there is a 3-D blob object in the scene. The projections to each camera are shown in the frames on the bottom-left. The 2-D blob moments in these images are the measurements we hope to use to estimate the parameters of the 3-D blob. However, the problem is ill-conditioned, as shown on the right. The top-right image shows the same 3-D view of another shape that has the same 2-D projections bottom-right.

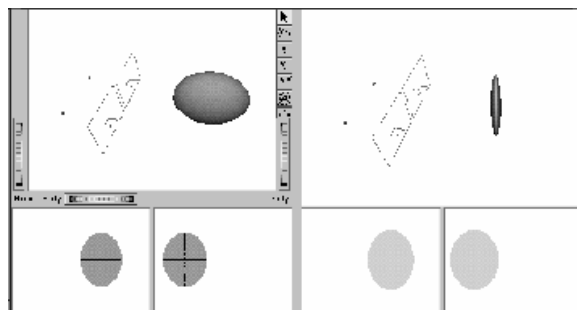


Figure 5. Stereo blob estimation is an ill-conditioned problem.

5.4. Self-calibration

When a person first enters the space, the stereo calibration is obtained recursively by incorporating the three blob correspondences (face, left hand, right hand) into the EKF/LM estimator [25]. The stereo pair shows the first image with overlaid blobs and large white boxes marking the current feature locations, and small white boxes representing the subsequent feature tracks [Figure 6]. The calibration parameters converge typically in the first 20 to 40 frames (roughly 2 to 4 seconds), if there is enough motion; longer if there is little motion. In this case, the subject waved his arms up and down to generate data and the system quickly converged to the state shown in the bottom portion of figure 7. This is a roughly overhead view showing the location of the cameras (COP and virtual image plane for each) and the 3-D trajectories of the hands and head. To evaluate the calibration quantitatively, we used the right hand as a 3-D pointer and traced the 3-D shape of known objects. We find that the error of reconstruction of a hand position is on the order of 2 to 3cm. This error is due both to estimation error and the crudeness of using the hand position to represent a point in space.

5.5. Person-tracking and shape recovery

As the camera becomes self-calibrated, the shape estimations begin to be meaningful. Here we quantitatively evaluate the steady-state shape and motion estimation by performing a physical motion after the calibration has converged in which the translation is linear and the rotation and shape are constant. There will be noise in measurement and our goal is to see how well the estimator performs over a range of object locations in the scene. Figure 7 shows a stereo pair and 3-D blob estimates for one frame of

the sequence in which the right hand is moving along the straight edge of a box with the orientation of the hand remaining (roughly) constant. Twenty frames of data were captured. Analysis on the translation was performed by fitting the 3D location estimates to a line and computing the RMS error, which was 1.5cm with a maximum error of roughly 3cm. Analysis on the rotation and shape were performed by computing the mean values and RMS errors, which were 5 degrees and 5% relative error respectively [26].

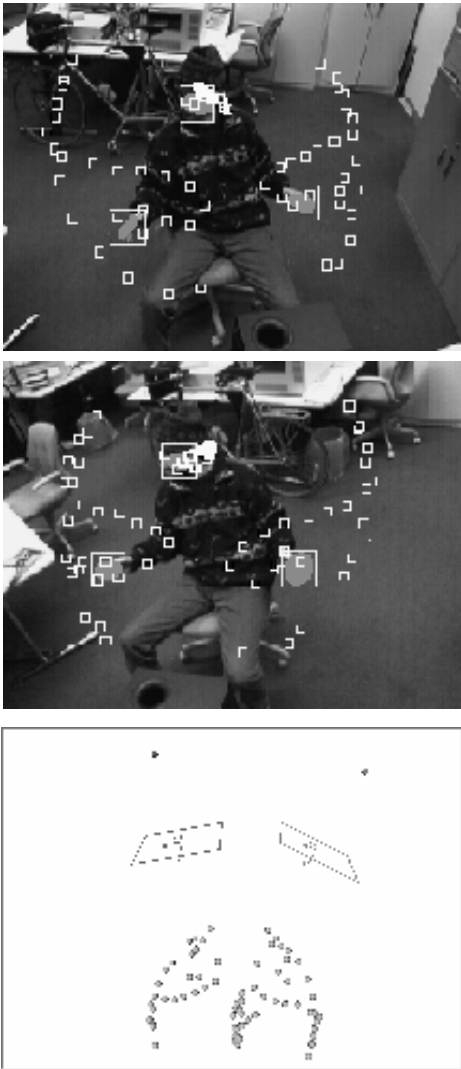


Figure 6. Calibration phase: it lasts 2-4 seconds

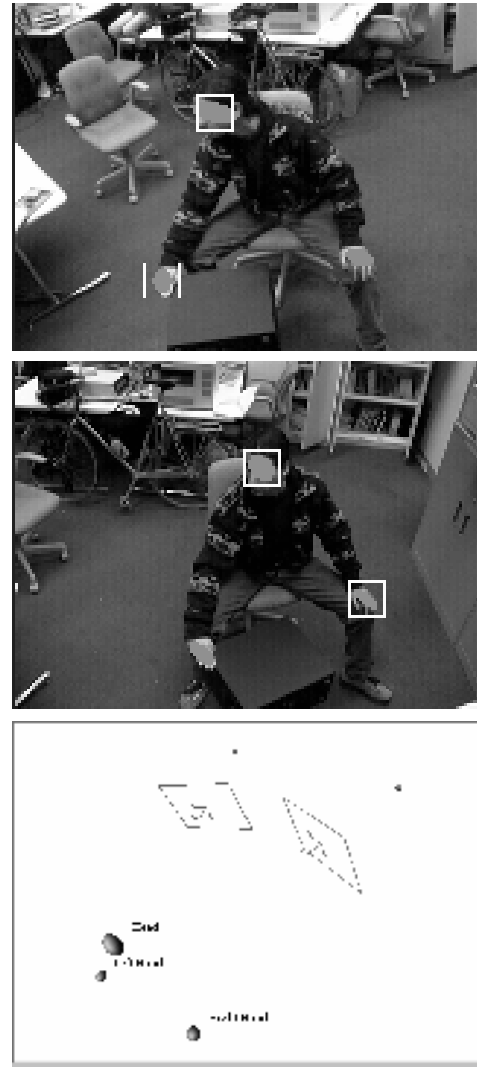


Figure 7. 3-D tracking of user's hands and head

6. Gesture Recognition

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. To navigate in the Internet 3-D city the user stands in front of the screen and uses hand gestures to navigate. Pointing to a link will load the new URL page. The user can scroll up and down a page by pointing up and down with either arm. When a new page is loaded, the virtual camera of the 3-D graphics

world will automatically move to a new position in space that constitutes an ideal viewpoint for the current page.

Recognized commands/gestures are: “follow link” → “point-at-correspondent-location-on-screen”, “go to

previous location” → “point left”, “go to next location” → “point right”, “navigate up” → “move one hand up”, “navigate down” → “move hands toward body”, “show aerial view” → “move both hands up” [figures 8,9,10,11].

Gesture recognition is accomplished by HMM modeling [27] of the navigating gestures. The feature vector includes *velocity* and position of hands and head, and blobs’ shape and orientation. We use four states HMMs with two intermediate states plus the initial and final states. Entropic’s Hidden Markov Model Toolkit [28] (HTK: <http://htk.eng.cam.ac.uk/>) is used for training. For recognition we use our real-time C++ Viterbi recognizer. All gestures start from a rest position given by the two hands on the table in front of the body.

The system runs on two SGI O2s R10,000 at 25-30 Hz. A real-time PC version using a dual processor Pentium III has already been implemented.



Figure 8. Move one hand up → navigate up

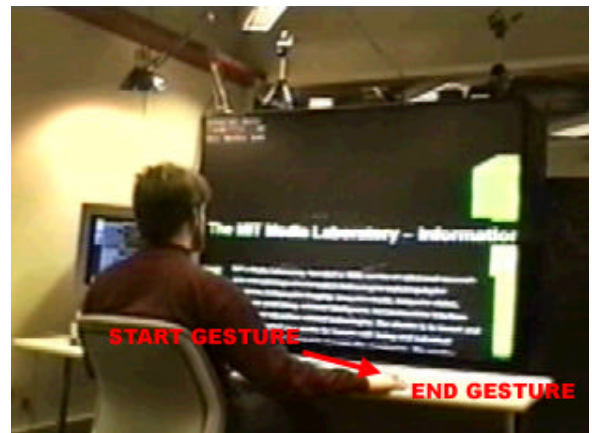


Figure 9. Point right → go to next location

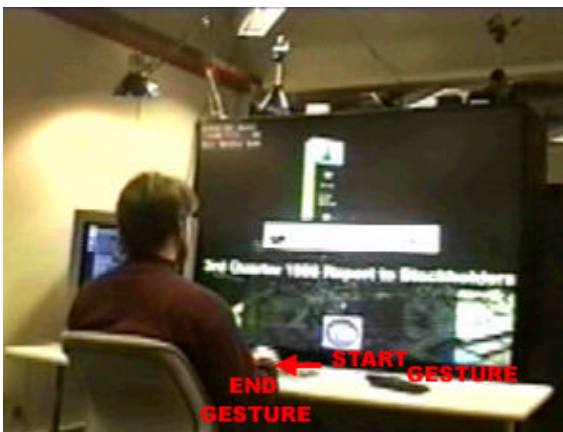


Figure 10. Move hand towards body → navigate down

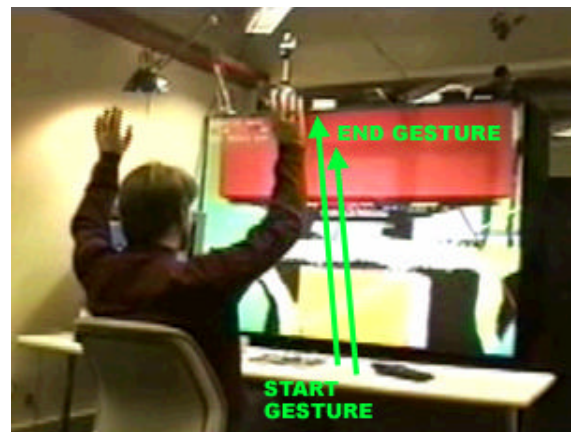


Figure 11. Move both hands above head → show aerial view

7. Discussion and Future Work

This paper describes a robust man-machine interface to navigate through a 3-D internet city. Real-time stereo hand and head tracking and 3-D position estimation ensure reliability of the interface. Our interactive space is the desktop, in which the seated user's head and hands are tracked by wide-baseline stereo pair of cameras. The virtual world is projected onto a large screen in front of the user. The user moves in the virtual 3D space with a small set of pointing gestures. HMM modeling is used to recognize the user's browsing commands. In parallel, by modeling the internet as a familiar city we build an immersive environment to better organize, visualize, and remember information. Our system establishes an important step towards natural interfaces to browse through three dimensional information landscapes, such as the proposed City of News, and 3-D computer games.

Bibliography

- [1] Pascarelli, E. and Quilter, D. *Repetitive Strain Injury: A Computer User's Guide*. John Wiley, 1994.
- [2] Aukstakalnis, S. and Blatner, D. *Silicon Mirage*. Peachpit Press, 1992.
- [3] Krueger, M. W. *Artificial Reality II*. Addison Wesley, 1990.
- [4] Pavlovic, V.I., Sharma, R., Huang T.S. Visual Interpretation of Hand Gestures for Human-Computer Interaction: A Review. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(7): 677-695, 1997.
- [5] Wu, Y. and Huang T.S. Human Hand Modeling, Analysis and Animation in the Context of Human Computer Interaction. In: *IEEE Signal Processing Magazine*, Special Issue on Immersive Interactive Technology, May 2001.
- [6] Starner, T. and Pentland, A. Visual recognition of American sign language using hidden markov models. In: International Workshop on Automatic Face and Gesture Recognition, pp. 189-194, 1995.
- [7] Brand, M., Oliver, N., Pentland, A. Coupled Hidden Markov Models for complex action recognition. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1997.
- [8] Campbell, L.W., Becker, D.A., Azarbayejani, A., Bobick A., Pentland, A. Invariant features for 3-D gesture recognition. IEEE International Conference on Automatic Face and Gesture Recognition, 1996.
- [9] Jovic N., Brumitt B., Meyers B., et al. Detection and Estimation of Pointing Gestures in Dense Disparity Maps. Fourth IEEE International Conference on Automatic Face and Gesture Recognition, 2000.
- [10] Brainard, D.H. and Freeman W.T. Bayesian Color Constancy. *Journal of the Optical Society of America*, A, 14(7), pp 1393-1411, July 1997.
- [11] Rehg, J.M. and Kanade, T. Visual tracking of high dof articulated structures: An application to human hand tracking. In: European Conference on Computer Vision, pp B:35-46, 1994.
- [12] Gavrilu, D.M. and Davis L. 3-D Model-based tracking of Humans in Action: a Multi-view Approach. Proc of IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, USA, 1996.
- [13] Pentland, A. and Horowitz, B. Recovery of nonrigid motion and structure. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13(7):730-742, July 1991.
- [14] Metaxas, D. and Terzopoulos, D. Shape and non-rigid motion estimation through physics-based synthesis. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 15:580-591, 1993.
- [15] Bichsel, M. Segmenting simply connected moving objects in a static scene. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 16(11):1138-1142, Nov 1994.
- [16] Baumberg, A. and Hogg, D. An efficient method for contour tracking using active shape models. In: Proceedings of the Workshop on Motion of Nonrigid and Articulated Objects. IEEE Computer Society, 1994.
- [17] Dodge, M. and Kitchin, R. *Atlas of Cyberspace*, Addison Wesley, 2001.
- [18] Sparacino F., Pentland A., Davenport G., Hlavac M., Obelnicki, M. City of News. Obelnicki In: Proceedings of the Ars Electronica Festival, Linz, Austria, 8-13 Sept. 1997
- [19] Sparacino F., Wren C., Pentland A., Davenport G., Hyperplex: a world of 3d interactive digital movies. In IJCAI-95 Workshop on Entertainment and AI/Alife, 1995.
- [20] Waterworth, J.A. Personal Spaces: 3D Spatial Worlds for Information Exploration, Organization, and Communication. In: Earnshaw and J. Vince (Eds.) *The Internet in 3D: Information, Images, and Interaction*. San Diego, USA, Academic Press, 1997.
- [21] Modjeska, D. and Waterworth, J. Effects of Desktop 3D World Design on User Navigation and Search Performance. In: IEEE Proceedings of Information Visualization 2000.
- [23] Yates Frances A. *The Art of Memory*. London, Routledge, 1966.
- [24] Wren, C., Azarbayejani A., Darrell, T., Pentland, A., Pfunder: Real-Time Tracking of the Human Body. *IEEE Trans. Pattern Analysis and Machine Intelligence*. 19(7): 780-785, 1997.
- [25] Azarbayejani A., Pentland, A. Real-time self-calibrating stereo person tracking using 3-D shape estimation from blob features. In: Proceedings of 13th ICPR, 1996.
- [26] Azarbayejani A., Wren, C., Pentland, A. Real-Time 3-D Tracking of the Human Body. In: Image Com, 1996.
- [27] Rabiner, L.R. and Juang, B.H. An introduction to hidden Markov Models, *IEEE ASSP Magazine*, pp 4-15, January 1986.
- [28] Young, S.J. Woodland P.C., and Byrne W.J. HTK: *Hidden Markov Model Toolkit. V1.5*. Entropic Research Laboratories Inc., 1993.