MVA'94 IAPR Workshop on Machine Vision Applications Dec. 13-15, 1994, Kawasaki

Virtual Gun
A Vision Based Human Computer Interface Using the Human Hand

James J. Kuch* and Thomas S. Huang
Beckman Institute, University of Illinois

405 N. Mathews Urbana, IL 61801 USA
*now with TouchVision Systems, Inc. Chicago IL, 60631

ABSTRACT

Herein, we present an unique vision based computer interface entitled *Virtual Gun*. In *Virtual Gun*, a person sits at a computer and points his index finger toward the screen with his thumb pointing up (similar to using one's hand as a gun). Movement of the hand and index finger moves the cursor on the screen. In order to *click* down on the cursor, one gestures as if he were shooting a gun (e.g., bringing his thumb down to the palm). Releasing the mouse button is equivalent to bringing the thumb back up and away from the palm.

The uniqueness in the presented tracking system is in the use of the entire 3-D hand model. This method is in contrast to tracking methods which use only a set of features of the model such as finger edges and tips or other methods which use an internal representation of the hand as is done in neural networks. In addition, there is no need for the user to wear a special glove or other physical items, which allows complete freedom to the user.

INTRODUCTION

If one could track the human hand, it could replace the computer mouse. The hand can provide a larger number of degrees of freedom (DOF) in which a person could interact with a computer instead of the two DOF provided by a mouse or trackball. Tracking the human hand in three dimensions (3-D) will thus allow for a more intuitive interaction of 3-D computer scenes.

Currently, the only commercial way to convey full hand movement to a computer is by the use of external devices such as VPL's DataGlove [1]. These devices must be physically attached to the person wanting to communicate to the computer. Such an interface is not only a hindrance to the user, but it also takes away from the natural free-flowing feel of communicating with the hands. Even though some HCI [2, 3] has been accomplished with such physical devices, their cost has prohibited widespread use.

Recently, much research has been performed on using computer vision to track human hands in an effort to provide a more natural setting for human communication with machines. Most of the research has been for HCI and has entailed either gesture recognition or using the hand as a 3-D mouse. Kjeldsen [4] gives a thorough review of many of these methods as they pertain to gesture recognition. Of the methods reported by Kjeldsen, a few are briefly reviewed here.

Wirtz and Maggioni [5] developed a real-time hand tracker that is able to track hand position and orientation but no finger movement. The user is required to wear a special glove that aids in the tracking.

Davis and Shah [6] also have a real-time recognition system that requires the user to wear a specially marked glove. The hand starts from a known position, and the fingers are then tracked to their final position. Recognition is based on the motion vectors of the fingers that are matched against a library of motion vectors organized into gestures.

Segen [7] is able to discern ten distinct poses of a hand using edge based techniques. The system operates in real-time and uses only a 2-D silhouette.

Kjeldsen [4] obtained marginal results in recognizing gestures for HCI. His approach used a neural network to recognize specific gestures that would control a computer's window manager. A grammar was also developed which detailed specific actions on how to interact with the window manager. The grammar is designed to aid in recognition, since it restricts the hand's movements.

Other recent vision based systems not reviewed by Kjeldsen, yet of considerable interest, are briefly reviewed in the following paragraphs.

Rehg and Kanade [8] demonstrate full hand tracking in real-time using special hardware and a model based approach. Image features, which included finger and palm edges along with finger tips, are detected using edge based operators. Tracking is performed by searching the image for image features with the aid of a motion estimate and the internal model of the hand. One tracking scenario presented uses the palm's movement in a plane along with the rotation of the index finger to provide three degrees of freedom. The hand is then used to maneuver an object in a 3-D graphical scene.

Darrel and Pentland [9] implemented a system that recognizes hand gestures. A view based approach is used to match space-time patterns of a human hand against precomputed gesture patterns. The matching process, a form of correlation, is performed using dynamic time warping that adjusts for the length of the space-time pattern. Real-time recognition is achieved by the use of special hardware. Only two examples of

successful tracking are presented, a hand waving "hello" and "good-bye.

Lee and Kunii [10] present a semiautomatic hand recognition system in which the user wears a specially marked glove. A stick figure of a hand is automatically articulated to a real hand's pose, following an initialization stage that places the stick figure at the proper orientation with respect to the real hand. The stick figure is adjusted based on natural hand movement constraints. This technique eliminates unrealistic hand movements and decreases the recognition time. The system uses only one real image yet takes over 30 minutes for convergence. They are able to reasonably fit their model to a real hand for 16 letters of the manual alphabet of the American Sign Language.

A NEW APPROACH

In this paper, the authors attempt to eschew the short comings of the previous methods and present a unique method for using the hand as a computer interface. In this approach, vision was chosen for the tracking for a number of reasons. First, it provides the most natural, noninvasive form of tracking. Second, computers are now being sold with cameras mounted on top of the screen,[1] therefore making it a very cost-effective tracking method. Last, it corresponds to the method in which the hand model was calibrated (see [11]), which enables both the calibration and tracking procedures to share the same hardware, thereby eliminating the need for additional hardware for either the calibration or tracking systems. This is the topic of the next section.

THE HAND MODEL

In [11], a new method for building lifelike hand models which articulate in a realistic manner is presented. Calibration of the model is based on anatomical studies of the human hand and on the specific method of tracking to be employed in the HCI scenario. The calibration method is done visually and requires only three views of the real hand to be modeled. The calibration system is designed to be accurate, to be easy to use and to allow for a short calibration time. These characteristics are all desirable when one is working in the realm of human computer interfacing. For a detailed description of the model see [11].

The fully calibrated hand model for the first author is shown in wireframe from two views in Figure 1.

TRACKING

The current method of tracking relies entirely on the 3-D structure of the calibrated hand model and incorporates no motion prediction, gray level gradient descent or similar, sophisticated tracking methods. Thus, the emphasis here is on demonstrating the power and robustness of an accurate 3-D hand model in tracking. Application of more advanced tracking methods can then only improve the speed and accuracy of the current method.

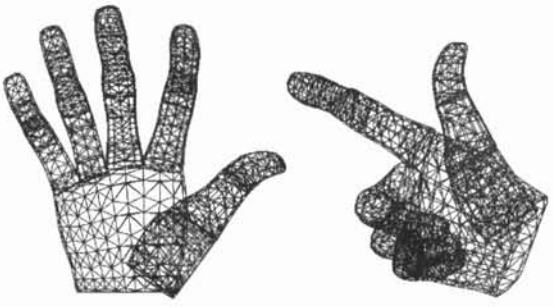


Fig. 1. Two views of the fully calibrated hand model.

The tracking scenario consists of having a person holding his hand in a predefined orientation within the field of view of a camera looking at a uniform background. The hand model is then interactively orientated within the computer in order to replicate the real world scene. Next, the sequence of images to be tracked are thresholded to segment the hand from the background, and then they are displayed. The hand model is then rendered flat shaded in black (binary). The resultant image from the model rendering is then exclusive-ORed (XOR) with the real binary image to yield a residual error image. The number of high pixels in the residual error image is then summed to give a single scalar error value, E. It is this scalar error that is then minimized to orient the hand model to the same configuration as the hand in the real image.

To arrive at the minimum value of E, the DOF of the hand model are independently and locally perturbed. After each perturbation, the preceding procedure for acquiring the value of E is repeated until E is minimized. The algorithm to perturb the individual fingers, thumb and palm for minimization of E is as follows:

(1) Get E (error value),
(2) Rotate joint by +1 degree and render,
(3) Get new_E (new error value),
(4) If new_E > E then goto 5 else goto 9,
(5) Rotate joint by -2 degrees and render,
(6) Get new_E,
(7) If new_E > E then goto 8 else goto 9,
(8) Flex joint by +1 degree and render,
(9) Minimize other DOF at same joint,
(10) Get new_E,
(11) If E > new_E then goto 1 else stop.

Minimization of any additional DOF (line 9) is achieved in exactly the same manner as described for the first DOF.

Figure 2 shows the experimental test bed for *Virtual Gun,* showing the camera orientation as well as the hand's orientation. From the first frame, the hand model is interactively adjusted into the same configuration as the real hand. The viewing point of the hand model

[1] Silicon Graphics' low-end machine, Indy, now comes standard with a digital color camera.

Fig. 2. *Virtual Gun* test bed.

within the camera is also adjusted at this time to correspond to the real world. Tracking is now ready to begin.

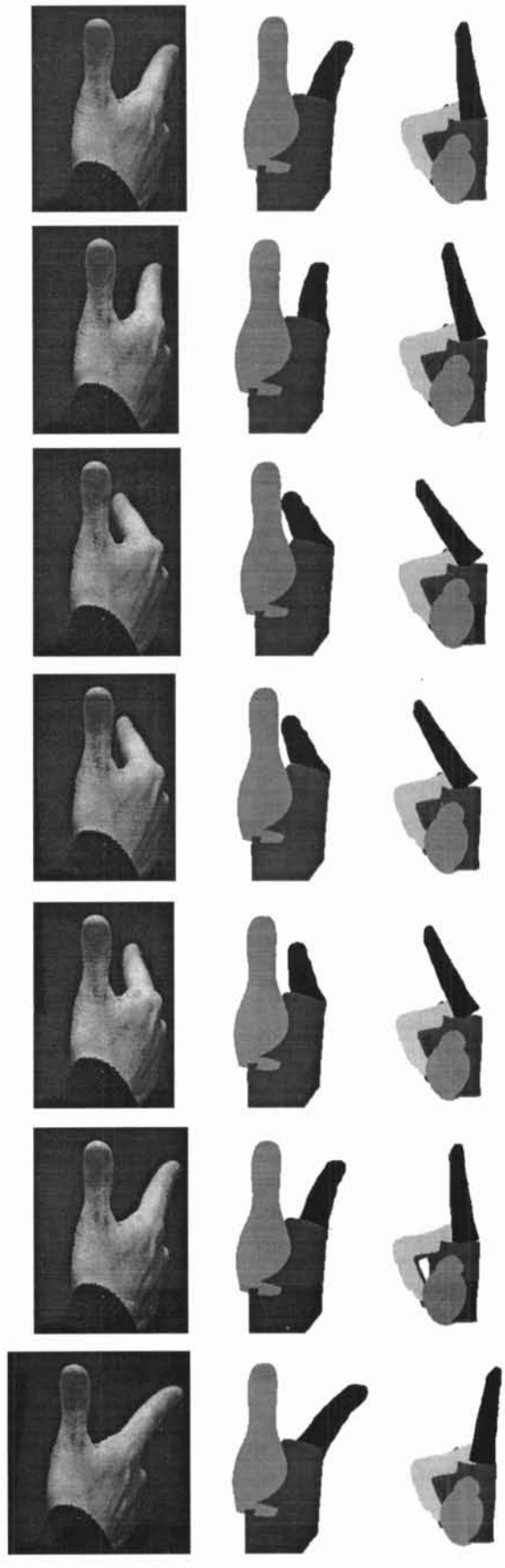
In the *Virtual Gun* scenario, the only parts of the hand allowed to move are the palm, the index finger and the thumb. The other fingers are curled inward until they reach the palm, just as one would make a gun with their hand. In addition, the index finger is assumed to always point directly outward; thus, the distal joints do not flex. Tracking then starts with the palm.

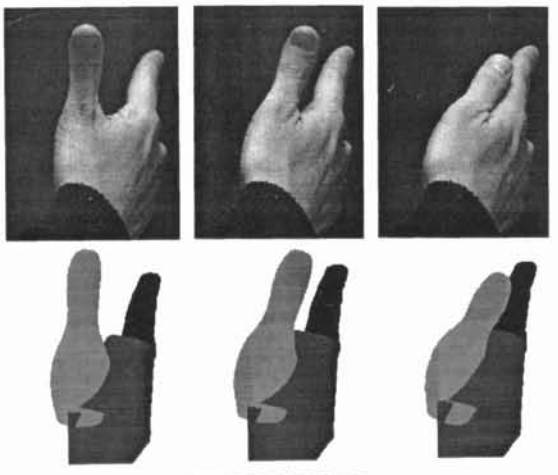
The palm's orientation is first obtained by rendering only the palm and perturbing the palm's DOF until E is minimized. The finger's orientation is then obtained in the same manner as the palm's, with the exception that both the palm and finger are rendered during the perturbation. Finally, the thumb's orientation is obtained by perturbing the thumb, while rendering the thumb, the palm and the finger.

RESULTS

During each tracking sequence, frames were recorded at 60 fields per second or 30 frames per second interlaced. Each frame was digitized at a resolution of 720 pixels horizontally by 486 pixels vertically (NTSC standard). Odd fields were dropped, and each field was then aspect corrected for computer displaying purposes for a final size of 640 pixels by 480 pixels Since the hand typically occupied only half of the image in all cases, all the images shown here were cropped and then reduced to optimize paper space. The tracking system used every third field for a temporal resolution of ten frames per second.

A 13.3 second sequence of a hand performing several pointing actions followed by two clicks at a computer screen was recorded and tracked. Figure 3 shows the results for the first second of tracking (some frames are not shown due to space). Beside the real image is the corresponding hand model as seen during the tracking procedure. To the right of that, is the same hand model except viewed from above the thumb (palm motion is not shown in the last view). Figure 4 shows a few frames corresponding to the thumb being pressed downward, which denotes a click of the *Virtual Gun*.





TIME ISSUES

The tracking time for each frame was approximately four seconds. When large motions were present, the times were slightly higher due to the number of renderings that were required for convergence. Likewise, when the motion was relatively small, convergence was obtained in two to three seconds.

Tracking was implemented on a Silicon Graphics Crimson, with a R4000 processor running at 100 MHz. Although the tracking was not real-time, the *Virtual Gun* could be implemented in real time on a multiprocessor Silicon Graphics Onyx with R4400 processors running at 250MHz.

An eight processor Onyx is able to increase rendering by a factor of ten. This increase, along with motion estimation and other more sophisticated tracking methods, will be able to achieve a speed increase of 40 over the current equipment and tracking method. Thus, the current four second analysis time will be reduced to 0.1 seconds, or real time.

In addition, with computers increasing in performance and simultaneously decreasing in cost, the *Virtual Gun* system will be practical in the near future for both its performance as well as its cost.

CONCLUSIONS

The preceeding results clearly demonstrate the power of the presented hand model and tracker. The tracking has shown to be accurate and robust.

Verification of the quantitative results of an experiment of this type is often impossible to evaluate, since the real values of the joint angles of the real hand are unknown and are practically impossible to obtain. The results obtained for several experiments showed strong correspondence between the motion of the real hand and the motion of the model with no visible error.

The tracked finger movement can now be used to drive the cursor on a computer screen as described in the *Virtual Gun* interface. This task is simple, since camera position and hand position are known. The only missing parameter is the distance between the hand and the monitor. This distance can either be predetermined or entered in by the user. Basic trigonometry will then give the location on the computer monitor in which the user is pointing, and thus the position of the cursor.

With texture, gray scale image processing methods could be incorporated into the tracking algorithm. Methods such as gradient descent can provide quicker convergence rates and therefore faster tracking.

Other improvements in tracking consist primarily of motion estimation of the individual fingers. Finger movements consist primarily of a repeated series of an acceleration followed by a deceleration. This movement is typical for pointing, where the hand moves from one position on the screen to another. Such movement can give rise to a simple prediction method to determine where a particular finger will be in the next frame.

ACKNOWLEDGMENTS

This work was supported in part by National Science Foundation Grant IRI-89-08255 and in part by a Grant from Sumitomo Electric, Inc.

REFERENCES

[1] J. Foley, A. van Dam, S. Feiner and J. Hughes, *Computer Graphics: Principles and Practice*. Reading, MA: Addison-Wesley Publishing Company, 1990.

[2] T. Baudel and M. Beaudouin-Lafan, "Charade, remote control of objects with freehand gestures," *Communications of the ACM*, vol. 36, no. 7, pp. 28-35, 1993.

[3] D. Sturman, D. Zeltzer and S. Pieper, "Hands-on interaction with virtual environments," in *UIST: Proceedings of the ACM SIGGRAPH Symposium on User Interfaces*, Williamsburg, VA, pp. 19-24, November, 1989.

[4] R. Kjeldsen, "Visual hand gesture interpretation," *IEEE Computer Society Workshop on Non-Rigid and Articulate Motion*, Austin, TX, November 1994.

[5] B. Wirtz and C. Maggioni, "ImageGlove: a novel way to control virtual environments," in *Proceedings of Virtual Reality Systems*, New York, April 1993.

[6] J. Davis and M. Shah, "Gesture recognition," University of Central Florida, Department of Computer Science, Tech Rep CS-TR-93-11, 1993.

[7] J. Segen, "Controlling computers with gloveless gestures," in *Proceedings of Virtual Reality Systems*, April 1993.

[8] J. Rehg and T. Kanade, "DigitEyes: vision-based human hand tracking," CMU-CS-93-220, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, December 1993.

[9] T. Darrell and A. Pentland, "Space-time gestures," *Computer Vision and Pattern Recognition*, pp. 335-340, 1993.

[10] J. Lee and T. Kunii, "Constraint-based hand animation," in *Models and Techniques in Computer Animation*, Tokyo: Springer-Verlag, pp. 110-127, 1993.

[11] J. Kuch, T. Huang, "Human Computer Interaction via the Human Hand: A Hand Model," in *Proceedings of the 28th Asilomar Conference*, Pacific Grove, CA, November 1994.