Real-Time Input of 3D Pose and Gestures of a User's Hand and Its Applications for HCI

Yoichi Sato
Institute of Industrial Science
The University of Tokyo
7-22-1 Roppongi, Minato-ku
Tokyo 106-8558, JAPAN
Tel: +81-3-3402-6231 ext. 2365
ysato@iis.u-tokyo.ac.jp

Makiko Saito Hideki Koike
Graduate School of Information Systems
University of Electro-Communications
1-5-1 Chofugaoka, Chofu
Tokyo 182-8585, JAPAN
Tel: +81-424-43-5651
{makiko, koike}@vogue.is.uec.ac.jp

Abstract

In this paper, we introduce a method for tracking a user's hand in 3D and recognizing the hand's gesture in real-time without the use of any invasive devices attached to the hand. Our method uses multiple cameras for determining the position and orientation of a user's hand moving freely in a 3D space. In addition, the method identifies predetermined gestures in a fast and robust manner by using a neural network which has been properly trained beforehand. This paper also describes results of user study of our proposed method and its application for several types of applications, including 3D object handling for a desktop system and 3D walk-through for a large immersive display system.

1. Introduction

With the rapid increase of computer usage in every aspect of our daily lives, the role of human computer interaction, or HCI, is becoming more important. Thus, a tremendous amount of effort has been made to provide more natural and intuitive means for interacting with computers. In particular, an approach that involves adapting the way humans communicate with each other for HCI is considered to be the most promising; this type of approach has resulted in development of important technologies such as speech recognition.

On the other hand, graphical user interface, or GUI, has been widely accepted as a standard interface framework.In recent decades, GUI has been predominantly used for commercially available systems; it provides an efficient interface for employing various kinds of applications on a computer, e.g., word processing or web browsing. One of the main reasons for the wide acceptance of GUI is that it can provide direct manipulation of objects on a computer monitor by means of input devices such as a mouse [16]. This property provides a user a clear model of what commands and actions are possible and what their affects will be.

Unfortunately, however, GUI is not a suitable option for some types of applications which inherently require controls with a high degree of freedom. For instance, manipulation of a 3D object by using a mouse is not an easy task because its motion is limited to 2D. In this case, direct manipulation with a user's hand, rather than with a mouse, can offer an ideal alternative for such applications. In this way, users can control the position and orientation of a 3D object directly by simply moving their hands.

This observation motivated us to develop a new method for tracking a user's hand in 3D and recognizing hand gestures in real-time without using any invasive devices attached to the hand. In this work, we propose a new technique for estimating the 3D pose of a user's hand by using multiple cameras in real-time. In addition, the proposed technique is able to recognize predetermined hand gestures in a fast and robust manner by using a neural network which has been properly trained beforehand.

This paper is organized as follows. In Section 2, we describe the previously proposed methodqs for estimating the 3D position and orientation of a user's hand, and discuss the limitations of these methods. In Section 3, we explain our proposed method for estimating the 3D pose of a hand and recognizing hand shape patterns. In Section 4, we report experimental results for evaluating the performance of the proposed method. In Section 5, we describe preliminary user study for using the proposed method for several kinds of applications including 3D object handling for a desktop system and 3D walk-through for a large immersive display system. Finally, in Section 6, we present our conclusions.

Proceedings of the Virtual Reality 2001 Conference (VR'01)
0-7695-0948-7/01 $17.00 © 2001 IEEE

2. Related Works

In this section, we give a brief overview of the previously proposed methods for tracking a user's hand in 3D, and examine the limitations of these methods.

The use of glove-based devices for measuring the location and shape of a user's hand has been widely studied in the past, especially in the field of virtual reality. Angles of finger joints are measured by some sort of sensor, typically mechanical or optical. An additional sensor determines the position of a hand. One of the most widely known examples of such devices is DataGlove by VPL Research [21] which uses optical fiber technology for flexion detection and a magnetic sensor for hand position tracking. A good survey of glove-based devices can be found in [17].

In general, glove-based devices can measure hand postures and locations with high accuracy and high speed. However, the use of glove-based devices is not suitable for some types of applications such as human computer interfaces because those devices may limit a user's motion due to the physical connection to their controllers.

For this reason, other researchers have studied a number of methods based on computer vision techniques in the past. One approach is to use some kind of markers attached to a user's hand or fingertips, so that those points can be easily found. For instance, color markers attached to the fingertips are used in the method reported in [2] to identify locations of fingertips in input images. Maggioni [9] presented the use of a specially marked glove for hand tracking. The glove has two slightly off-centered, differently colored circular regions. By identifying those two circles with a single camera, the system can estimate hand position and orientation.

Unfortunately, this approach with markers is not preferred as a method for human-computer interaction for the same reason as for the approach with glove-based devices. Although use of markers is less restrictive than glove-based devices which are physically connected to their controllers, it can be prohibitively cumbersome for users to attach markers onto their hands every time they use a computer.

For this reason, other researchers have investigated techniques for determining the 3D pose of a hand and gestures without any markers. In these techniques, image regions corresponding to human skin are extracted typically either by color segmentation or by background image subtraction. After the images regions are identified in input images, the regions are analyzed to estimate the location and orientation of a hand, or to estimate locations of fingertips. For instance, in the method by Maggioni et al., [10], the shape of the contour of an extracted hand region is used for determining locations of fingertips. Segen and Kumar [15] introduced a method which fits a line segment to a hand region contour to locate the side of an extended finger. Interestingly, their method can estimate a 3D position of a user's fingertip reliably by making use of the shadow cast by the finger on a table. However, a table needs to have a constant color so that a shadow can be detected.

Most of the previously proposed computer vision techniques for tracking a user's hand use a single camera. Therefore, the problem of 3D hand pose estimation becomes an ill-posed problem since 3D information of a hand is lost in a 2D image and cannot be recovered unless some sort of assumptions are made. To overcome this difficulty, some researchers have studied techniques for 3D hand pose estimation by using multiple cameras. For instance, Fukumoto et al. introduced a method called Finger-pointer, which can estimate the position of a user's pointing finger in 3D space as well as the number of extended fingers by using two cameras [6]. However, hand gestures treated with their method were rather simple, and the main focus was placed on integration of voice commands and hand gestures. Utsumi and Ohya recently developed a method which can estimate 3D poses of two hands and recognize several hand shape patterns in real-time by using five cameras [20]. To our knowledge, their method has the best performance so far in the sense that two hands can be tracked simultaneously and their hand shape patterns are recognized at the speed of approximately 10 frames per second.

Another approach used in hand gesture analysis is to use a three-dimensional model of a human hand. In this approach, in order to determine the posture of the hand model, the model is matched to a user's hand images which have been obtained by using one or more cameras. The method proposed by Rehg and Kanade [14] is one example based on this approach. Unlike other methods which do not use a three dimensional hand model, the method proposed by Rehg and Kanade can estimate three-dimensional posture of a user's hand.

However, this approach faces several difficulties such as self-occlusion of a hand or high computational cost for estimation of hand posture. Due to the high degrees of freedom of the hand model, it is very difficult to estimate the hand configuration from a two-dimensional image even if images are obtained from multiple viewpoints.

In addition to the methods mentioned in this section, a large number of methods were proposed in the past. A good survey of hand tracking methods as well as algorithms for hand gesture analysis can be found in [8] and [13].

3. Proposed Method

The goal of our study is to develop a new method which can estimate the pose of a user's hand in 3D and recognize hand shape patterns in real-time. In particular, our method is intended to include all of the following features ideal for applications in human-computer interaction.

- Robust and accurate estimation of the 3D position and orientation of a user's hand by using multiple cameras
- Real-time processing at or close to video-frame rate

- Robust recognition of hand gestures which can be easily adapted to different users
- Algorithm design conceptually straightforward and easy to implement

The proposed method consists of three major parts: extraction of an image region corresponding to a user's hand; estimation of the 3D position and orientation of the hand; and recognition of hand gestures. Figure 1 shows the overview of the proposed method.

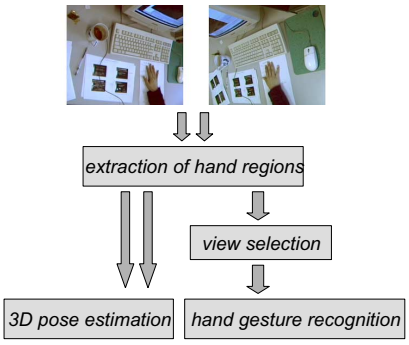

Figure 1. Overview of the proposed method

In the proposed method, multiple input images taken from different locations are used for observing a user's hand. Examples of those input images are shown in Figure 1. As described in the previous section, it is generally difficult to estimate the 3D pose, i.e., 3D position and orientation, of a hand from a single input image. To avoid this difficulty, our method utilizes multiple input images taken from different locations as previously proposed by several other researchers. In our current implementation of the proposed method, two cameras are used.

First, an image region corresponding to a hand is extracted in each of the input images. For this purpose, image processing techniques such as background subtraction or region extraction-based color are commonly used. Since we consider use of our method for a desktop interface system in a cluttered office environment, backgrounds of input images might be fairly complex and even dynamically changing. On the other hand, illumination is expected to be reasonably stable.

For this reason, image regions corresponding to a hand are extracted based color in our method. Input images from two color cameras are captured as a YUV color image with 256×220 pixels, and then converted to a HSV (hue, saturation, value) color image. After a median filter is applied for suppressing effects of image noise, image regions where saturation values are sufficiently high and hue values are close to that of human skin are identified as candidate regions. Then, the largest connected region is selected as a hand region based on the assumption that other image regions whose colors happen to be similar to skin color are generally much smaller than a user's hand. In this way, image regions for a user's hand can be obtained even in a fairly cluttered background.

3.1. Estimation of 3D Position and Orientation of a Hand

After hand regions are found in each input image taken from multiple cameras, a 3D position and orientation of a user's hand are determined based on triangulation [1]. Here each camera is calibrated beforehand by using the Tsai camera calibration method [18] to determine its camera parameters such as a 3D location of its projection center and the orientation of the camera in the world coordinate system defined with the real 3D space.

Once the camera calibration has been done, we can determine a 3D line that extends from a camera projection center through the center of gravity for the extracted hand region in the input image taken by the camera. Figure 2 illustrates such 3D lines as $L1$ and $L2$. Then the 3D position of the hand can be obtained as an intersection of these two lines. However, due to various kinds of errors such as error in hand region extraction, these two lines may not intersect. Therefore, the hand position is estimated as a point where the distance between the two lines becomes minimal.

After the 3D position of the hand is obtained, we need to determine the 3D orientation, i.e., roll, pitch, and yaw angles, of the hand as illustrated in Figure 3.

To estimate these three angles, three more feature points as well as the center of gravity of a hand region are used. Those points are the tip of a hand region, and both right and left end points of the region as illustrated in Figure 2. By using the center of gravity and the tip point, roll and pitch are determined as the direction of the hand in 3D space. Similarly, yaw is determined by using the right and left end points.

3.2. Hand Gesture Recognition with Neural Network

In addition to the 3D pose of a user's hand, hand gestures are recognized in our method. Here we are concerned with recognition of static hand gesture for each input image frame. In particular, we would like to identify which of the pre-defined hand shapes, e.g., closed, open, or pointing, is observed, so that these hand shapes can be used as commands for controlling various types of computer applications. Recognition of dynamic gestures with temporal analysis of motion of a user's hand is not considered in this work.

When we consider the use of hand gesture recognition for human computer interaction, it is important that a recognition method can be easily adapted for different users performing various tasks.

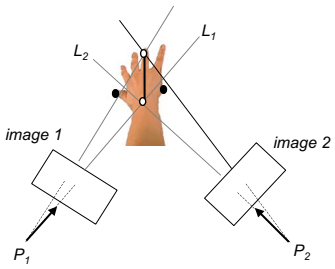

Figure 2. Estimation of 3D position and orientation of a hand

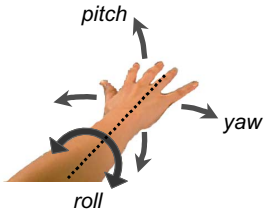

Figure 3. Roll, pitch, yaw orientation of a hand

For this reason, we decided to use a neural network [5] for hand gesture recognition. Neural networks are known to be effective for distinguishing different signal patterns, and have been widely used for various tasks of pattern recognition with both spatial and temporal signals, e.g., images and sounds. Another advantage of using a neural network is its low computational cost. Training a neural network using a large set of training data requires a significant amount of time. However, the training is required only once as an off-line process, and once a network is properly trained, on-line recognition for new input data can be performed quite efficiently.

Since we consider the case where a user's hand is moving freely in 3D space, a large portion of the hand might be self-occluded if it is observed from a single camera position. Therefore, it is necessary to select an appropriate view for hand pattern classification from multiple input images. In such an appropriate image, a hand should not be observed too far from or too close to a camera. In addition, a hand should not be observed from its side; rather, it should be seen from its front side, i.e., facing straight toward a palm, to avoid self-occlusion.

In our method, the best view is selected based on the area and aspect ratio of an extracted hand region such as the one shown in Figure 4 (a). In principle, an image which contains a hand region whose area is sufficiently large, and whose aspect ratio is close to 1 is selected as the best view. Area and aspect ratio of an extracted hand region as well as the center-of-gravity and the orientation of its principal axis can be computed from image moments of the region as described in [4].

After a best view is selected, a hand region in the view needs to be normalized before it is further provided to a neural network for hand shape classification. Otherwise, slight misalignment of input data may affect recognition results significantly.

This normalization is done by using the center-of-gravity, orientation, and aspect ratio of a hand region. First, the hand region is translated so that the center of gravity of the region is placed at the center of the image. Then, the region is rotated based on the orientation of the principal axis of the region so that the axis is aligned to one of the image axis, e.g., the row direction in our case. Finally, the region is scaled in two orthogonal image directions to make the aspect ratio close to 1. For example, the hand region in Figure 4 (a) is normalized as shown in Figure 4 (c) after translation, rotation, and scaling as described here.

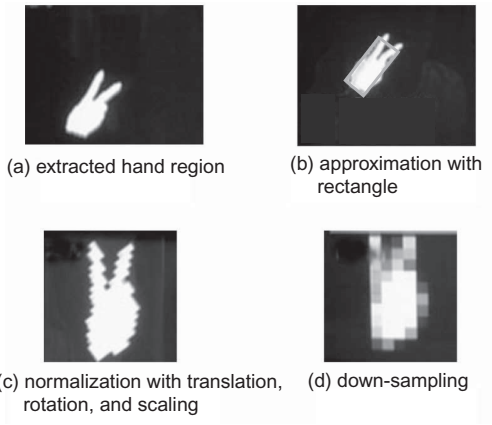

Figure 4. Normalization of input images

Then, the normalized image is down-sampled with averaging to a smaller gray-level image to produce input data for a neural network. This is necessary for reduction of the amount of data provided to a neural network to avoid unnecessarily high computational cost. We also found that the down-sampling with averaging is helpful for reducing effects of small illumination variation. In our current implementation, the size of input data is 12×12 pixels. Figure 4 (d) shows an example of down-sampled input data.

Finally, a gray level value of each pixel of the down-sampled input data is given to a corresponding input node of the neural-network. Our method uses a three-layer neural network. The neural network is trained beforehand with a set of training data by using the back-propagation algorithm [5] so that an output node with the highest score represents a recognized hand shape pattern. Figure 5 illustrates the three-layer neural network used in our method.

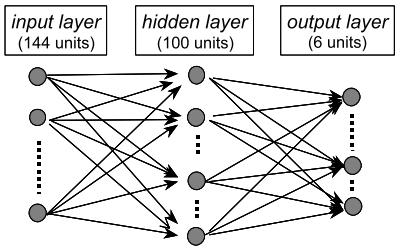

Figure 5. Neural network model

4. Experimental Result

We have tested our proposed method by using a desktop-type experimental setup shown in Figure 6. In this experimental setup, two 3CCD color video cameras (JVC KY-57B) are placed facing toward the center of the workspace where a user's hand is moved. The distance from the center to those two cameras is approximately 1.5 meters. The angle subtended by those two cameras with respect to the workspace center is about 90 degrees as shown in Figure 6. This setting of the two cameras was empirically determined to avoid self-occlusion of a hand in input images. Here we mainly considered situations where users keep their hands pointing toward a computer monitor and rotate hands around the pointing direction. The question of how to place input cameras for other types of applications will require further investigation depending on the nature of target applications.

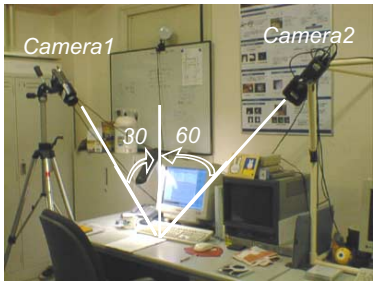

Figure 6. Experimental setup

Input images from the two cameras are digitized and processed with a personal computer (PentiumII 450MHz, 384MB memory, WindowsNT4.0 OS) with two sets of a general-purpose image processing board (Hitachi IP5005). With this hardware configuration, our current system can process more than 20 input image frames/sec.

4.1. Experimental Results for 3D Pose Estimation

To evaluate the performance of the proposed method for tracking a user's hand in 3D, we have conducted a user study with 10 individuals. In this study, users were asked to move their hands in front of a computer monitor which displayed a CG model of a hand at the estimated 3D pose of their hands in real-time. Then, users were asked to evaluate how intuitively they could move the CG hand model with their own hands. The result of this study is shown in Figure 7.

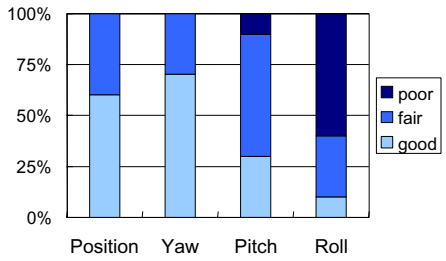

Figure 7. Performance evaluation of estimation 3D position and orientation

From these result, we see that our proposed method has a satisfactory performance for estimating the 3D position of a hand. Most users reported that they were able to control the 3D position of a CG hand model accurately. Similarly, users were able to control the yaw and pitch angles of a CG hand model with their hands very intuitively. On the other hand, the result show that estimation of roll angles is worse than estimation of other parameters. The commonly observed problem was that the direction of roll orientation could not be determined correctly even though a value of the roll angle was correctly estimated. One way to avoid this problem is to use a more sophisticated algorithm for estimation of roll angles. For instance, Utsumi et al. used a probabilistic approach [20] so that roll angles could be estimated reliably even in the presence of image noise.

The user study presented here is still preliminary, and we have not yet done any quantitative evaluation of 3D pose estimation. More detailed study with quantitative evaluation using ground truth provided with additional sensors such as a Polhemus magnetic 3D position sensor is left for further study.

4.2. Experimental Results for Hand Shape Classification

We have evaluated the accuracy of hand shape pattern classification by our proposed method with a neural network. In this study, 10 individual users were asked to make 6 different hand shapes which are shown in the bottom of Figure 8. Each test took 1 minute and therefore, approximately 1200 frames were used for evaluation. The neural network used in this experiment was trained with a set of training data created by observing one particular user selected from the 10 individual users.

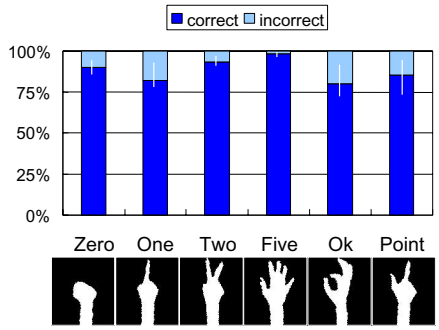

Figure 8. Experimental results for hand shape pattern classification

Figure 8 shows recognition accuracy for each hand shape pattern with a standard deviation indicated by a white bar. As we can see from this diagram, correct hand shape patterns were recognized with more than 85% accuracy overall in this experiment. The typical mode of failure was misclassification happening during transition from one hand shape to another. We consider that this result is quite reasonable since the neural network was not trained to recognize those intermediate hand shapes.

In this experiment, several hand shape patterns such as *Zero*, *Two*, and *Five*, were recognized very accurately. On the other hand, recognition results were worse for the rest of hand shape patterns. Several possible reasons can be given for explaining for this performance. For instance, we found that hand shapes were fairly different from person to person for *Ok* and *Point*. Thus those hand shape patterns were classified to a wrong shape pattern in some cases, and that resulted in lower recognition accuracy and higher deviation. Recognition accuracy for *One* is also slightly lower than the others. The main reason was the shape of *One* was close to that of *Two*, and therefore *One* was sometimes confused with *Two*. However, the opposite case was not seen in our experiment.

Nevertheless, we consider that the result of this experiment is quite encouraging because the neural network was trained using training data created for a particular person. To improve the hand shape classification performance further, we need to train the neural network used in our method with a larger set of training data so that slight variation of hand shape patterns among different users are taken into account.

5. Example Applications

We have examined the use of our proposed method for several kinds of applications including a desktop type interface system and an interface system for a virtual reality environment with a large immersive display. In particular, we aimed to examine how intuitively users could perform their tasks using our method.

5.1. 3D Object Handling for Desktop System

To evaluate the effectiveness of the proposed method for a desktop type interface system, we considered applications for which 3D hand pose estimation and hand gesture recognition were expected to be particularly useful. In our experiment, a task of handling a CG object in a virtual reality environment was examined as an example of such applications. For this test, the same computer and camera configuration were used as the one shown in Figure 6.

In this test, users could execute commands by changing their hand shape patterns, and they could manipulate a CG object by simply moving their hands. Both the target CG object and a CG model of a user's hand were displayed on a monitor. Each of six hand shape patterns shown in Figure 8 was assigned to a different control mode in this experiment. The mapping between hand shape patterns and control modes is shown in Table 1. With the mode *NoMode*, a CG object is not attached to a user's hand, e.g., being released.

Table 1. Control modes used for 3D object handling

hand gesture	control mode
zero	NoMode
one	Yaw
two	Pitch
five	Move
point	Size
ok	Hold

To avoid unwanted change of control modes when hand shape patterns are not classified reliably, e.g., during change from one shape to another, a filtering process was incorporated. A new hand shape was considered to be identified only if the shape is found in more than 2 out of 5 last video frames. We found that this filtering process improved stability of our method significantly.

Ten individuals were asked to manipulate a target CG object by moving their hands for a fixed duration of time. Then they were asked to evaluate with three scores, i.e., good, fair, and bad, how intuitively they could manipulate the CG object using our proposed method. The result of this user study is shown in Figure 9.

Most users reported favorable evaluation for translation of their hands and for the grabbing operation corresponding to three control modes *NoMode*, *Move*, and *Hold*. Similarly, pitch rotation by the control mode *Pitch* received high scores.

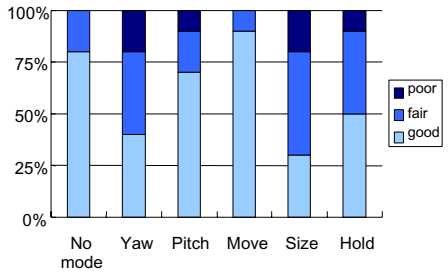

Figure 9. Performance evaluation of 3D object handling

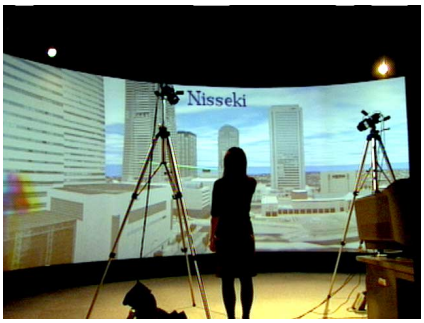

Figure 10. Prototype system for 3D navigation in a virtual reality environment

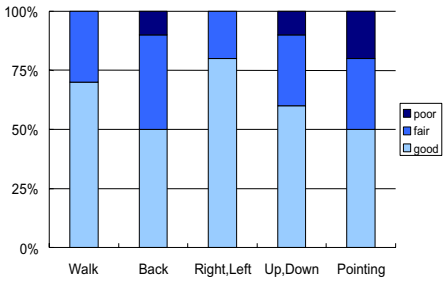

Figure 11. Performance evaluation of 3D navigation in a virtual reality environment

On the other hand, the rest of the control modes, i.e., *Yaw* and *Size*, received a worse evaluation. The hand shape pattern assigned to the control mode *Yaw* was *One* in Figure 8. Thus the shape was often incorrectly classified to *Two* as seen in the previous experiment for hand shape classification. That explains lower scores for yaw rotation operation. For zooming operation with the control mode *Size*, several users reported that they felt somewhat awkward since they were using only one hand for stretching a CG object. As a result, the evaluation for the operation was lower than the others.

Even though several users felt direct manipulation of a CG object with our proposed method was not so intuitive, overall impression about the use of our method for 3D object handling was quite favorable. For further justification of effectiveness of our method for 3D object handling, we need to conduct more careful user study including comparison of our proposed method with other means, e.g., a conventional mouse and keyboard.

5.2. 3D Navigation for a Large Immersive Display

We applied our method for 3D navigation in a virtual reality environment generated with a large immersive display. The same computer and camera configuration were used as in the other experiments except that the distance from the center of workspace to two cameras was longer. In addition, a graphics workstation (SGI Onyx2 Infinite Reality) was used for generating images of a virtual reality environment displayed on a large cylindrical screen (radius: 4 meters, height: 2.7 meters, horizontal field of view: 150 degrees). Figure 10 shows the large immersive display and the two input cameras used in this experiment.

In this example application, three different hand shape patterns are used to switch between two different control modes. When a hand is fully opened (*Five* in Figure 8), navigation mode is activated so that users can move on a ground plane and change its view direction in a virtual reality environment simply by changing the position and orientation of their hands. The other control mode is pointing mode, which is activated by extending a pointing finger (*Point* in Figure 8). With the pointing mode, users can point to various buildings in a virtual reality environment to obtain the buildings' names. To make transition between two modes reliably, the third hand shape pattern *OK* was used to switch from one control mode to the other.

Similar to the user study described in the previous section, 10 individual users were asked to rate how intuitively they could control this application. As seen in the result shown in Figure 11, most users reported favorable scores for both control modes: 3D navigation and pointing in a virtual reality environment. The main reason is that the orientation of a user's hand was not used in this application; only the 3D position of the hand was used. As seen in the previous experiments, estimation of the 3D position of a user's hand is much more reliable than that of the 3D orientation. It is also interesting that several users pointed out that they felt direct control with their hands was more intuitive in this 3D navigation application than in the previous 3D object handling because of immersion experience with a large display. This suggests that direct manipulation with a user's hand is more suitable for applications with a large immersive display than desktop type

applications.

6. Conclusions

In this work, we have proposed an efficient method for estimating 3D hand pose and recognizing hand shape patterns in real-time. In particular, the use of a neural network for hand gesture classification gives our proposed method a distinct advantage over other related techniques in that our method is computationally less expensive and it can be easily adapted to work with different users by simply adding appropriate training data. The current system of the proposed method can process images up to 20 to 25 frames per second.

In addition, we have conducted several preliminary user studies to evaluate how well a user can use the proposed method for different types of applications including 3D object handling with a desk-top system and 3D navigation in a virtual reality environment with a large immersive display. We are encouraged with the initial results of those user studies. While users reported that the stability of hand orientation estimation needed further improvement, most users found direct manipulation with their hands with our method easy to use for the both applications tested in our experiments.

References

[1] Ballard, D. H. and Brown, C. M. *Computer Vision*, Prentice-Hall, 1982.

[2] Cipolla, R., Okamoto, Y. and Kuno, Y. Robust structure from motion using motion parallax, In *Proc. 1999 IEEE International Conference on Computer Vision*, 1993, pp. 374-382.

[3] Cipolla, R. and Pentland A. (ed.) *Computer Vision for Human-Machine Interaction*, Cambridge University Press, 1998.

[4] Freeman, W. T. and Anderson, D. B. Computer Vision for Interactive Computer Graphics, *IEEE Computer Graphics and Applications*, May/June 1998, pp.42-53.

[5] Fu, L. *Neural Networks in Computer Intelligence*, McGraw-Hill, 1994.

[6] Fukumoto, M., Suenaga, Y., and Mase, K. Finger-pointer: pointing interface by image processing, *Computers and Graphics*, Vol. 18, No. 5, 1994, pp. 633-642.

[7] Glassner, A. S. (ed.) *Graphics Gems*, AP Professional, Cambridge, MA, 1990.

[8] Huang, T. S. and Pavlovic, V. I. Hand gesture modeling, analysis, and synthesis, In *Proc. 1995 IEEE International Workshop on Automatic Face and Gesture Recognition*, September 1995, pp. 73-79.

[9] Maggioni, C. A novel gestural input device for virtual reality, In *Proc. 1993 IEEE Annual Virtual Reality International Symposium*, 1993, pp. 118-124.

[10] Maggioni, C. and Kammerer, B. GestureComputer - history, design and applications, In *Computer Vision for Human-Machine Interaction* (R. Cipolla and A. Pentland, eds.), Cambridge University Press, 1998, pp. 23-51.

[11] Moghaddam, B. and Pentland, A. Maximum Likelihood Detection of Face and Hands, In *Proc. 1995 IEEE International Workshop on Automatic Face-and Gesture-Recognition*, 1995, pp.122-128.

[12] Numazaki, S., Morishita, A., Umeki, N., Ishikawa, M., and Doi, M., A kinetic and 3D image input device, *Proc. ACM SIGCHI'98*, August 1998, pp. 237-238.

[13] Pavlovic, V. I., Sharma, R., and Huang, T. S. Visual interpretation of hand gestures for human-computer interaction: a review, *IEEE Transactions on Pattern Aanalysis and Machine Intelligence*, Vol. 19, No. 7, July 1997, pp. 677-695.

[14] Rehg, J. M. and Kanade, T. Visual tracking of high DOF articulated structures: an application to human hand tracking, In *Proc. European Conferende on Computer Vision '94*, 1994, pp. 35-46.

[15] Segen, J. and Kumar, S. Shadow gestures: 3D hand pose estimation using a single camera, In *Proc. 1999 IEEE Conference on Computer Vision and Pattern Recognition*, June 1999, pp. 479-485.

[16] Shneiderman, B. Direct manipulation: a step beyond programming language, *IEEE Computer*, Vol. 16, No. 8, 1983, pp. 57-69.

[17] Sturman, D. J. and Zeltzer, D. A survey of glove-based input, *IEEE Computer Graphics and Applications*, Vol. 14, January 1994, pp. 30-39.

[18] Tsai, R. Y. A versatile camera calibration technique for high accuracy machine vision metrology using off-the-shelf TV cameras and lenses, *IEEE Journal of Robotics and Automation*, Vol. 3, No. 4, August 1987, pp.323-344.

[19] Turk, M. Perceptual user interfaces, *Communications of the ACM*, Vol. 43, No. 3, March 2000, pp.33-34.

[20] Utsumi, A. and Ohya, J. Multiple-hand-gesture tracking using multiple cameras, In *Proc. IEEE Conference on Computer Vision and Pattern Recognition '99*, 1999, pp.473-478.

[21] Zimmermann, T. G., Lanier, J., Blanchard, C., Bryson, S., and Harvill, Y. A hand gesture interface device, In *Proc. ACM Conf. Human Factors in Computing Systems and Graphics Interface*, 1987, pp. 189-192.