ICAT 2003
December 3-5, Tokyo, JAPAN

SpaceSensor: Real-time Gesture Tracking for I-NEXT*

Dongpyo Hong and Woontack Woo

KJIST U-VR Lab.
Gwangju 500-712, S.Korea
{dhong, wwoo}@kjist.ac.k

Abstract

In this paper, we propose a real-time gesture tracking technique for the personalized user interface exploiting non-contact 3D vision technique. We first separate a user of interest as precisely as possible from the acquired natural background scenes and then estimate the disparity of the segmented user with a multi-view camera. Given the depth information of the user, we construct *SpaceSensor*, 3D box-based space sensor, around the user. Using the proposed *SpaceSensor*, the user can express his/her intension with natural gestures. The proposed technique overcomes the constraints of gesture tracking based on 2D vision techniques. In addition, the computation complexity is reduced when it compare to those of traditional real-time gesture tracking techniques. According to the experimental results, the proposed gesture tracking technique can be applied to various applications requiring real-time interaction, without loss of generality.

Key words: Image Segmentation, 3D Vision-based Gesture Tracking, Interactive Virtual Environment

1. Introduction

With the rapid progress of technologies in the areas of computers and communications, the future computing environments will support "seamless support from ubiquitous computers and pervasive networking", that is, users could do just-in-time access to any (invisible) computers anytime and anywhere [1][2]. Consequently, this environment will require users to interact with computers through more natural and comfortable interfaces. In addition, it will be an important issue to deliver the user's intensions or emotions over the network.

In the spite of these requirements, the previous user interfaces have not yet overcome the restrictions of 2D, such as keyboard, mouse, etc. The 2D user interfaces make it hard for users to interact with 3D virtual objects or virtual environment. In consequence, they provide users with uncomfortable and unnatural interactions. Thus, in the last few decades, there have been many studies which substitute the conventional user interfaces with new types of interfaces like vision, gesture, voice, and other sensors. Especially, the advantages of a vision-based interface over other sensors are calibrated easily, interact with the systems naturally and remove the cumbersome devices from users.

In general, the vision-based user interfaces are categorized as two types [3]. One is the contact vision-based interface which generally uses markers worn by the user. The other is the non-contact vision-based interface which generally uses the background subtraction techniques. The contact vision-based interface is able to extract information of interest by simply tracking markers. However, it has a few drawbacks. For example, when markers are occluded or when multiple markers are used, it is hard and error prone to track them [4]. Furthermore, the contact interface requires the user to wear markers. To the contrary, the non-contact vision-based interfaces overcome the limitations of the contact vision-based interfaces. Although these advantages over contact vision-based interfaces, they still suffer from the interference of lighting sources. To resolve these restrictions, there have been many research activities [5][6][7][8]. And also traditional vision-based gesture tracking techniques use complex computational algorithms. Due to complexity of tracking algorithm, it is hard to apply into real-time applications [9][10][11][12].

In this paper, we propose a real-time gesture tracking technique for the personalized user interface based on the 3D vision technique. For the real-time user's gestures tracking, we exploit the user segmentation method which extracts the user information from natural background scene without special facilities or devices, such as blue screen or chroma-keying. In addition to the user segmentation, we construct invisible 3D box-based sensor, *SpaceSensor*, around the user dynamically, in which we utilize both disparity map acquiring from multiple view camera and the extracted user information. The proposed technique overcomes the constraints of gesture tracking based on 2D vision techniques as well as the computation complexity of traditional real-time gesture tracking.

In order to show the effectiveness of the proposed

* *This work is sponsored by ICU Digital Media Lab.*

technique, we applied the proposed technique to the interface of an interactive media system, what is called, "I-NEXT: An Interactive Networked Expression eXperience Testbed." I-NEXT allows users to express their intentions or emotions interactively in the networked VE through a 3D vision-based user interface [13]. Users are able to experience the interactive expressions and share the experience over the network in real- time through I-NEXT.

This paper is organized as follows. In Section 2, we introduce the key components of the proposed technique, i.e., user segmentation algorithm, design of *SpaceSensor* and gesture tracking. Experimental results and some applications are shown in Section 3. The discussion and future works are mentioned in Section 4.

2. SpaceSensor: 3D Vision-based User Interface

2.1 User Segmentation

The proposed user segmentation method, like general background subtraction techniques, has two stages. One is training background and the other is subtracting from the trained background. In the first stage, as shown in Figure 1, we train background images and make the reference image in RGB and normalized RGB color space, respectively. Then in the second stage, we do subtract the current image from the reference image in each color space.

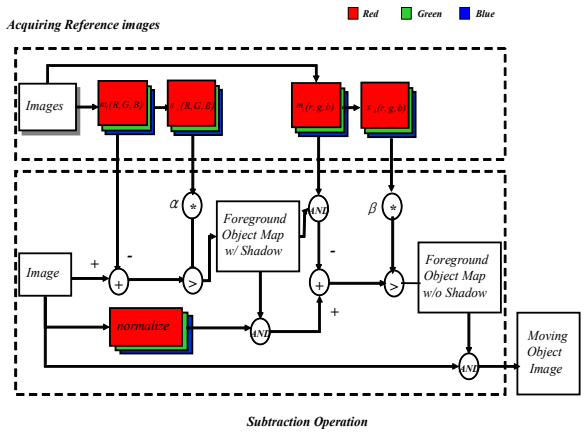


Fig 1. The proposed user segmentation algorithm

In the training background stage, we model the background and determine the pixel-wise threshold. After background modeling in each color space, i.e. RGB color space and normalized RGB color space, we separate the user with cast shadows from the background scene in RGB color space. As shown in Figure 1, the generated binary map is used as a mask image in normalized RGB color space. When we apply the mask image into the reference image and current image in normalized RGB color space at the same time, we simply discard cast shadows from the user because shadows have only effects on luminance [14].

2.2 SpaceSensor Design

It is natural for users to use their dynamic gestures to interact with virtual objects and environment instead of 2D interfaces. However, it is time consuming process to track dynamic gestures in 3D space. Therefore, we adopt a 3D box-based sensor, which improves tracking accuracy while maintaining its simplicity.

For the design of *SpaceSensor*, we need to acquire the segmented information as well as the depth information of the user. The design of the proposed *SpaceSensor* focused on tracking of natural movements of the user by making it dynamically augmented around the user instead of fixing the *SpaceSensor* at a certain location [3]. Furthermore, it is augmented into a reachable space from the movements of the user in order to track the gestures of the user correctly. To implement the proposed *SpaceSensor*, we first calculate the center position of the user ($H_c = \{H_x, H_y, H_z\}$) as follows.

$$H_c = \frac{1}{N_j}\sum_{i=1}^{N_j} SUD_{j,i} \qquad , j = \{x, y, z\} \tag{1}$$

where SUD_j (Segmented User Depth information) represents the 3D points in each coordinate within the segmented user disparity image. N_j is the number of 3D points in each coordinate within the segmented user disparity image. From Equation (1), we can compute requisite parameters for the implementation of *SpaceSensor* as the following equation.

$$SS_{width} = SS_{height} = SS_{depth} = SUD_{\max\{y_i\}} - SUD_{\min\{y_i\}} \tag{2}$$

where SS_{width} , SS_{height} and SS_{depth} represent width, height, and depth of *SpaceSensor* which is based on the user's center point, respectively. $SUD_{\max\{y_i\}}$ and $SUD_{\min\{y_i\}}$ represent the top and bottom points of regions being occupied by the user, respectively. The Equation (2) is based on "*Leonardo da Vinci: The Vitruvian man*" [15]. From Equation (1) and (2), the proposed *SpaceSensor* is augmented around the user as shown in Figure 2.

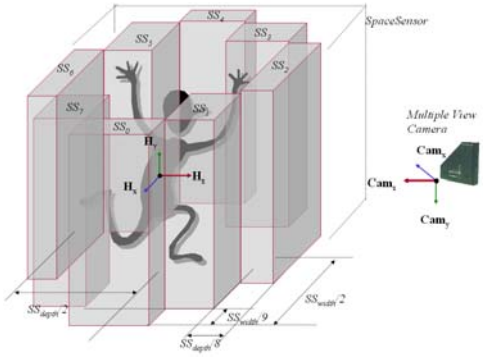


Fig 2. Allocation of SpaceSensor.

In our design of *SpaceSensor* , we allocate invisible eight 3D box-based sensors into the personal space in order. The more 3D box-based sensors, the more accurate the tracking of movement will be. However, there is a tradeoff between accuracy and computation complexity because the increased latency or time-delay caused by an increased number of boxes will distract the user. This allocation of *SpaceSensor* is based on the height of segmented user.

2.3 Dynamic Gesture Tracking

As explained previous subsections, the proposed tracking technique is simple but accurate. To maintain tracking technique as simple as possible, we adopt *SpaceSensor* which has 8 different regions and states. Let the state of *SpaceSensor* be $SS = \{ss_0, ss_1, \ldots, ss_N\}$, where N denotes the number of 3D box-based sensors. The status of i-th box, ss_i, is denoted by 1 or 0, where 1 indicates that the user is touching the box when it is observed. Unlike other vision-based approaches, we keep track of the state of *SpaceSensor*, *SS*, to track dynamic gestures, instead of tracking the movement of user's limbs such as arms, legs, feet and torso.

As shown in Figure 2, $SS_{\{0,\ldots,7\}}$ represents the regions of *SpaceSensor* which covers user's personal space. Through the sequence of states of $SS_{\{0,\ldots,7\}}$, a user is able to manipulate virtual objects directly for explicit interactions as well as make gestures for implicit interactions. When a user touches one of *SpaceSensor*, its state changes 1 and the touched position is calculated as follows.

$$P(x,y,z) = \left(\frac{1}{N_x}\sum_{i=1}^{N_x} x_i, \frac{1}{N_y}\sum_{i=1}^{N_y} y_i, \frac{1}{N_z}\sum_{i=1}^{N_z} z_i\right) \quad (3)$$

where *P(x,y,z)* represented the touched position in *SpaceSensor* . N_x, N_y and N_z is the number of touched points in *SpaceSensor* .

Figure 3 shows the measurement of touched position in *SpaceSensor*. Given the segmented user with the depth information and *SpaceSensor*, the movement of gestures can be tracked by observing how the touched position moves through *SpaceSensor*.

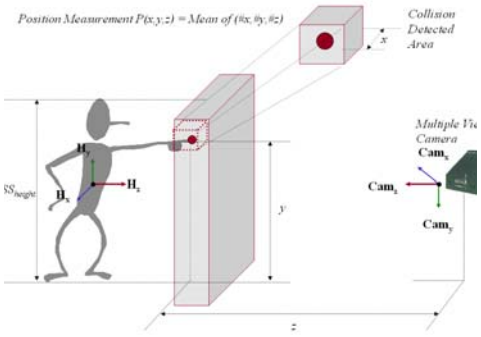
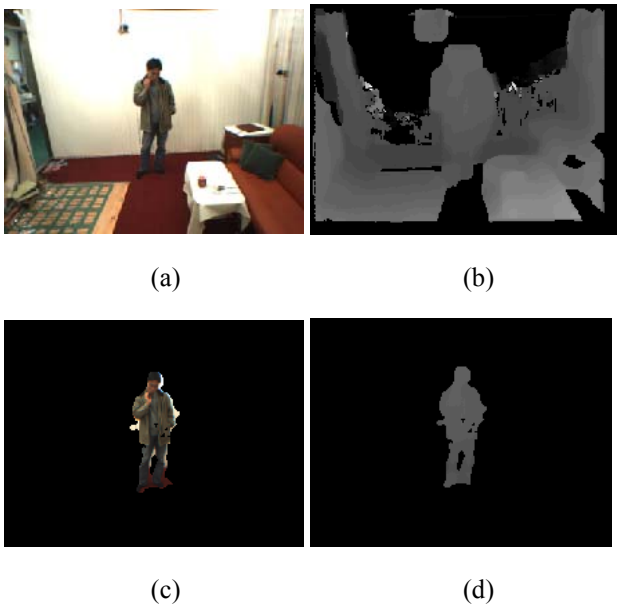


Fig 3. Position measurement of SpaceSensor.

The proposed *SpaceSensor* is able to track user's gestures as well as extract additional information using combination of states of *SpaceSensor* on the fly. For example, the movements of the user (Speed), the usage of the personalized space (Large or small), the weight of movements (Acceleration[1]) and so on. Therefore, it is applicable to the new type of a user interface in virtual environment.

3. Experimental Results

Figure 4 shows the extracting procedure of 3D information of the segmented user with the depth information.



(a) (b)

(c) (d)

Fig 4. Extraction 3D information of the user. (a) Captured live image (b) Disparity image of the live image (c) The segmented user (d) The disparity image of the segmented user

As shown in Figure 4(b), it is hard to extract the 3D information of the user directly from the disparity image acquired by a multiple view camera. It includes not only the user but also background or other objects information. As shown in Figure 4(d), however, we can extract more accurate 3D information of the user of interest if we exploit both the segmented user image and the disparity image. It shows that the accuracy of extracting 3D information of a user is proportional to the user segmentation algorithm[2].

The following Figure 5 shows the augmentation of *SpaceSensor* around a user.

[1] *We applied the Newton's law, F = ma, so the power is proportional to the acceleration.*

[2] *This is, of course, associated with the accuracy of disparity map. In this experiment, however, we exploited the supplied disparity image and libraries by Digiclops and Triclops.*

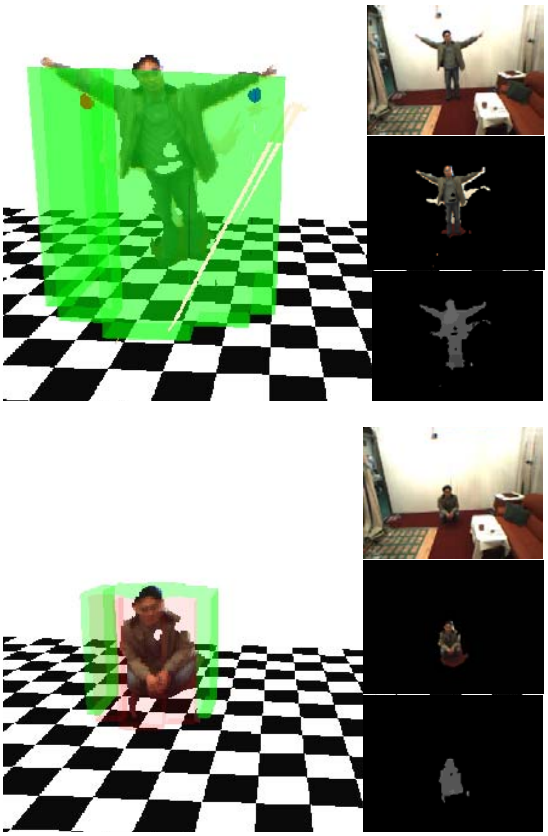


Fig 5. Augmentation of SpaceSensor around a user.

As shown in Figure 5, we back-projected the 3D information of the user in the virtual environment in order to show that the proposed *SpaceSensor* is exactly allocated around the user [16]. The result indicates that *SpaceSensor* is able to track gestures any direction around the user. In addition, it can construct *SpaceSensor* dynamically using the requisite parameters, for example, shrinking and growing *SpaceSensor* itself. These parameters are automatically calculated on the fly.

Figure 6 shows the experimental results of how to detect the touched position in *SpaceSensor* and to track gestures.

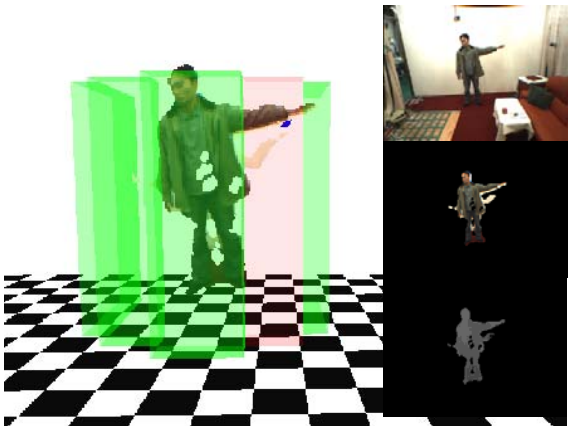


Fig 6. Tracking and Detection of SpaceSensor

As shown in Figure 6, it calculates the 3D information of touched position using Equation (3) when the user touches one of areas of *SpaceSensor*. As explained in previous section, we trace the trajectory of the touched positions in order to track user's gestures. However, in our current scheme, a user has to touch certain region of *SpaceSensor* to track gestures continuously. Currently we let the hand position be (0,0,0) when there is no collision with *SpaceSensor* .

Figure 7 depicts that users interact with either virtual objects or each other in I-NEXT using the proposed personalized emotional user interface, *SpaceSensor*.

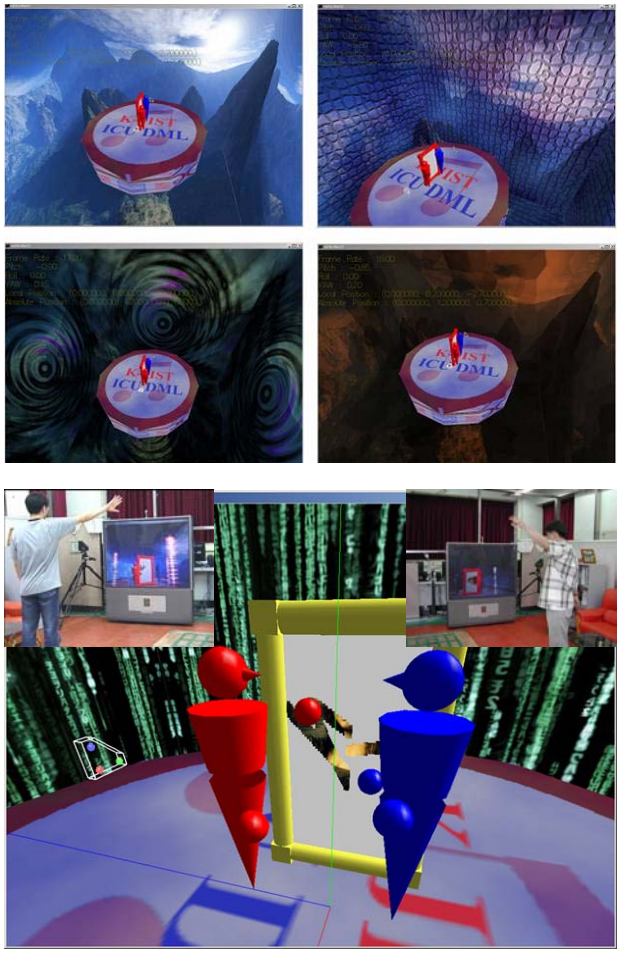


Fig 7. Virtual environments of I-NEXT and User Interactions through the network

As shown in Figure 7, there are several background scenes, which are changeable by users' interactions, in I-NEXT. Through the experiments, the participants were able to express their intentions interactively in real-time over the network. In the explicit interactions, however, they had a difficulty scratching the hidden layer on the virtual object due to the restriction in the accuracy of *SpaceSensor*. When users are in 3m~5m from the 3D camera, the minimum and maximum errors are about from 0.04m to 0.07m. On the contrary, in the implicit interactions, the participants are enough to express their intentions through *SpaceSensor*. Therefore, we need to

improve the collision detection algorithm and accuracy of *SpaceSensor* to provide interactive expressions.

4. Discussion and Future works

In this paper, we proposed a real-time gesture tracking technique for the personalized emotional user interface based on 3D vision technique. The proposed *SpaceSensor* overcomes the restrictions of 2D vision-based user interface as well as resolves the complexity of real-time gesture tracking. However, the proposed *SpaceSensor* is yet dependent on both user segmentation and disparity estimation method. Therefore, we need to investigate more deeply both on user segmentation and disparity estimation method.

References

1. M. Weiser, "The Computer for the 21st Century," Scientific American, pp. 94-104, Sep. 1991
2. S. Jang, S. Lee and W. Woo, "Research activities on Smart Environment," Magazine of the KITE, vol. 28, pp.1359-1371, Dec. 2001
3. W. Woo, N. Kim, K. Wong and M. Tadenuma, “Sketch on Dynamic Gesture Tracking and Analysis Exploiting Vision-based 3D Interface,” in Proc. SPIE PW-EI-VCIP'01, vol. 4310, pp. 656-666, Jan. 2001
4. Kida, K., Ihara, M., Shiwa, S., Ishibashi, S., “Motion tracking method for the CAVETM system”, Signal Processing Proceedings, 2000 WCCC-ICSP 2000. 5th International Conference on , 859 -862 vol.2, 2000
5. T. Horprasert, D. Harwood, and L.S. Davis, “A Statistical Approach for Real-time Robust Background Subtraction and Shadow Detection,”Proc. IEEE ICCV'99 FRAME-RATE Workshop, Kerkyra, Greece, September 1999
6. Ahmed Elgammal, David Harwood, and Larry Davis, “Non-parametric Model for Background Subtraction,” 6th European Conference on Computer Vision, Dublin, Ireland, June/July 2000.
7. A. Elgammal, R. Duraiswami, D. Harwood and L. S. Davis “Background and Foreground Modeling using Non-parametric Kernel Density Estimation for Visual Surveillance”, Proceedings of the IEEE, July 2002.
8. C. Kim, W. Woo, and H. Jeong, “Determination of Optical Flow by Stochastic Model,” Journal of the Korea Information Science Society (KISS), vol.19, no.6, pp.581-594, Nov., 1992.
9. Soren Lenman, Lars Bretzner, Bjorn Thuresson, “Computer Vision Based Hand Gesture Interfaces for Human-Computer Interaction,” Technical report TRITANA-D0209, CID-report, June 2002
10. Atsushi Nishikawa, Akio Ohnishi, Fumio Miyazaki, “Description and Recognition of Human Gestures Based on the Transition of Curvature from Motion Images,” In Face and Gesture Recognition, pages 552--557, 1998
11. William T. Freeman, and Craig D. Weissman, “Television control by hand gestures”, IEEE Intl. Wrkshp. on Automatic Face and Gesture Recognition, Zurich, June, 1995
12. Markus Kohler. “System Architecture and Techniques for Gesture Recognition in Unconstraint Environments”, In Nadia Magnenat Thalmann, editor, International Conference on Virtual Systems and Multimedia VSMM'97, pages 137-- 146, University of Geneva, Switzerland, September 10--12th 1997
13. D. Hong, W. Lee, J. Jeong, J. Kim, W. Woo, “I-NEXT: An Interactive Networked Expression eXperience Testbed”, Ninth International Conference on Virtual Systems and MutilMedia(VSMM03), pp. 455-462, 2003
14. D. Hong, W. Woo, “A Background Subtraction for a Vision based User Interface”, Pacific-Rim Conference on Multimedia (PCM2003), accepted, 2003
15. http://www.aiwaz.net/Leonardo/
16. Y. Suh, D. Hong, W. Woo, “"2.5D Video Avatar for Networked VRPhoto System," HCII03, pp. 533-537, 2003