

## An Embedding Method for Real Characters into the 3D Virtual Space

Ji-Seong Jeong<sup>1</sup>, Jin-A Park<sup>1</sup>, Sin Ae Kwon<sup>1</sup>, Chan Park<sup>1</sup>, Nakhoon Baek<sup>2</sup>,  
Rae-Hyun Jang<sup>3</sup> and Kwan-Hee Yoo<sup>1\*</sup>

<sup>1</sup> *Department of Digital Informatics and Convergence, Chungbuk National University,  
410 Seongbongro, Heungdukgu, Cheongju, Chungbuk, South Korea*

<sup>2</sup> *School of Computer Science and Engineering, Kyungpook National University,  
1370 Sankyeok-dong, Buk-ku, Daegu, South Korea*

<sup>3</sup> *Korea Internet Software, Cheongju, Chungbuk, South Korea*  
{farland83,\*khyoo}@chungbuk.ac.kr, oceancru@gmail.com, jrh@kis.com

### Abstract

*In this paper, we propose a method for embedding real characters in the 3D virtual space. Most systems have adopted virtual characters into the 3D virtual space to give more realistic situation, where the virtual characters have been controlled by user-steered interfaces. In order to take user immersion, we need to embed real characters into the 3D virtual space. This paper describes how to represent the real characters in 3D virtual space, and how to transform them according to the user control. The experimental results of our proposed method show that immersion and participation of learners are continuously increased.*

**Keywords:** *3D virtual space, virtual characters, real characters*

### 1. Introduction

Most e-Learning systems have been serviced based on 2D contents, including texts, 2D graphics, images and videos [1]. Various teaching and learning techniques such as cooperative learning and self-directed learning have been applied to the learning systems for enhancing the quality and effectiveness of education [1, 2]. Since such 2D contents are different from the learning and teaching space appeared in the real world, not only there might be some limitations in applications of the learning techniques, but also the educational effectiveness is hard to be achieved. Moreover, immersion, virtual experience and interaction emerging as the main issues of recent learning systems cannot be implemented in the 2D learning contents due to their constraints [3, 4]. The issues will be occurred while communicating with each other among three partners for educational services: *learners, teachers and learning systems*.

In order to solve these problems, such 2D educational systems started to migrate to 3D-based learning systems, where 3D virtual space represents the real learning world. For an example, several 3D virtual spaces modeled such as a hotel lobby, hotel rooms, coffee shops, markets, a swimming pool, an art laboratory, space travel, airport ticketing, airplane and airport customs are shown in Figure 1. The 3D virtual spaces have been used to support learners to study English effectively.

Many researchers have studied to find out technical factors, which will be operated in the 3D virtual space to provide the three issues: *immersion, interaction, and virtual experience*

---

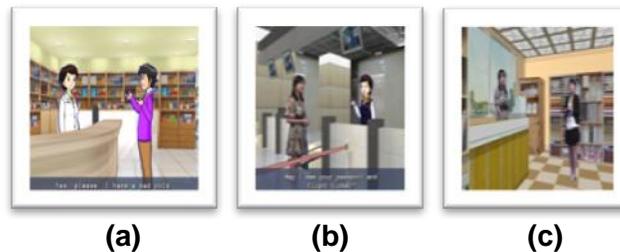
\* A corresponding author

for the educational application services [3-6]. One of them is to deal with how do learners as well as teachers participate in the 3D virtual educational space and how is their participation represented in the space.



**Figure 1. Various 3D Virtual Spaces**

As shown in Figure 2, there are three approaches for representing learners and teachers in the 3D virtual space. The first one, appeared in Figure 2(a), represents them as *avatars*, which are modeled as virtual characters so that communication with them will be appeared as behaviors of the avatars. An example for these applications is Sloodle [6], which combines the multi-user virtual environments of Second Life [7] and Moodle [8] learning management system. The second one, shown in Figure 2 (b), represents some actors by avatars and others by real-world characters. The final one is that all real-world learners and real-world teachers can be embedded into the 3D virtual space, as illustrated in Figure 2(c). To provide the educational effectiveness of learning systems to the learners, we adopt the third one, by which real-world characters in the real space can be embedded into the 3D virtual learning space. The third approach has the strong point of providing intensive immersion and interaction to learners. However, there are no results on embedding of the real-world characters into the 3D virtual space and animating them. The issues will be discussed in this paper.



**Figure 2. Representing Learners and Teachers as (a) Virtual Characters, (b) Virtual and Real Characters, and (c) Real Characters, in the 3D Virtual Space**

This paper is organized as follows: Section 2 illustrates the 3D virtual experiential system adapting real-world characters, and describes the proposed method for mapping the real

characters into the 3D virtual space and for moving them naturally in the space. Section 3 describes its implementation results, and Section 4 points out further research directions.

## 2. Embedding Real Characters into the 3D Virtual Space

### 2.1. System Configuration

In order to embed real-world characters into the 3D virtual space, first of all, images containing the real-world characters will be captured from input cameras such as Web Camera's and KINECT [9] devices. The captured images are processed with the computer vision tools such as background removal and extraction of user's skeleton depending on types of input devices. Letting *the real character space* be the space where real-world characters can move, the real character space can be constructed as a bounding box, which will be illustrated in Section 2.2. The obtained bounding box will be mapped into a box in 3D virtual space, as described in Section 2.3. With the mapping function, 3D virtual space containing real-world characters is finally rendered. The overall procedures are shown in Figure 3.

### 2.2. Capturing Data from Input Devices

To composite a 3D virtual space and real characters, the input devices to be used should be investigated. The proposed system has used the devices such as Web cameras and KINECT-like devices. In the case of Web cameras, the sequence of images will be captured and the image itself will be used as an input to remove its background. Microsoft KINECT is a movement sensing interface device for games based on the movement of users. The device consists of an RGB camera, 3D depth sensors, multi-array microphones, and a motorized tilt. The device has used IR(infrared radiation) camera and IR laser projector to take 3D depth information. Figure 4 shows a pattern image of IR laser projector output from IR camera, which shows depth information at each pixel in the captured image.

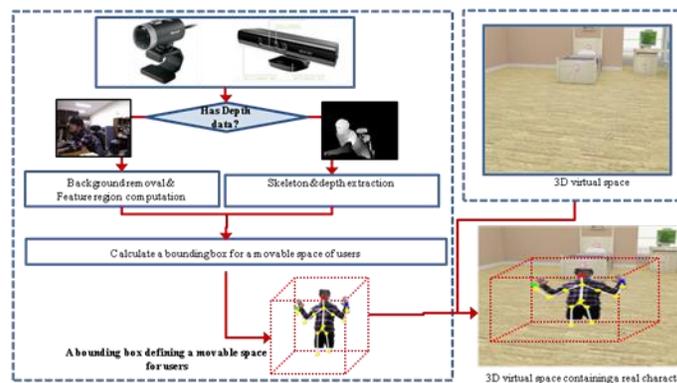
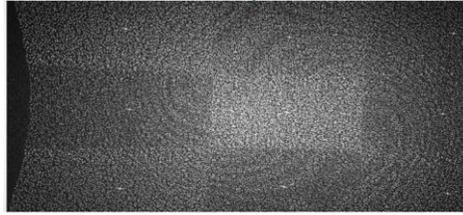
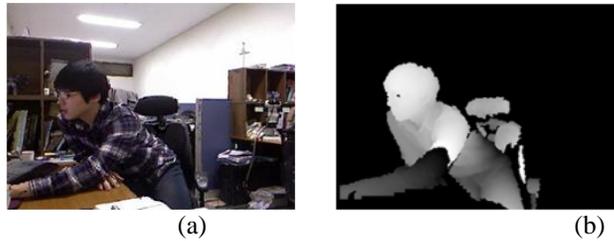


Figure 3. System Configuration for Embedding Real-world Characters into the 3D Virtual Space



**Figure 4. A Pattern Image of IR Laser Projector Output from IR Camera**

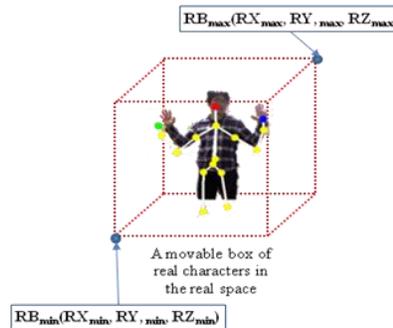
Microsoft provides KINECT SDK (software development kit) [9] for developing various KINECT-based applications. Developers can extract depth information for user's skeleton by using related functions in the SDK. The captured image and depth extraction are shown in Figure 5. In order to embed real-world characters into the 3D virtual space, first of all, the image should be preprocessed through filtering, color conversion, and background removal, which can be implemented by OpenCV [10, 11].



**Figure 5. An Image (a) Captured from the RGB Camera and a Depth Image (b) Obtained from the IR Camera**

### 2.3. Computing a Movable Bounding Box of Real Characters

The input cameras have been used to capture the images in the real 3D space. The image should be placed in 3D virtual space in order to simulate the natural movement of real-world characters in the virtual space. To do so, we take a movable box,  $MB_r$ , of real characters in the real space. Let  $(I_w, I_h)$  be the size of an image captured from a general camera or a RGB camera of Kinect.  $MRB$  is defined with two corner points  $RB_{min}(RX_{min}, RY_{min}, RZ_{min})$  and  $RB_{max}(RX_{max}, RY_{max}, RZ_{max})$  as shown in Figure 6. Clearly, since the real bounding box is made by the size of a captured image,  $RX_{min} = 0$ ,  $RY_{min} = 0$ ,  $RX_{max} = I_w$ , and  $RY_{max} = I_h$ . From now on, we will propose the method for computing two depth values  $RZ_{min}$  and  $RZ_{max}$  in the captured information from either a general camera or Kinect.



**Figure 6. System Configuration for Embedding Real Characters into the 3D Virtual Space**

A general camera is capturing an image for a scene in the 3D real space. The background removal of the image is performed through three steps: binarization of the image, noise removal of the image, and change of full transparency of the background color [10, 11]. In order to compute depth information from the captured image, the area of the region, including user's face, and one for user's hand. As shown in Figure 7, when  $FA_t$  and  $HA_t$  are the areas of smallest rectangular regions of a face and a hand of a user at a specific time  $t$ , respectively, the area  $FHA_t$  of face and hand region at  $t$  is calculated as follows:

$$FHA_t = w_f \times FA_t + w_h \times HA_t, \quad (1)$$

Where  $w_f$  and  $w_h$  are the weighting constants for face and hand, respectively. With these notations, a depth value  $RZ_t$  for the image is obtained as follows:

$$RZ_t = (FHA_t - RZ_{min}) / (RZ_{max} - RZ_{min}) \quad (2)$$

Where  $RZ_{min}$  is defined as the area of face and hand region when a user become far from the camera. The other hand,  $RZ_{max}$  is defined to be one when a user gets close enough to the camera.

Next, when a KINECT device is used, the depth information can be calculated by applying color, depth, and skeleton data as shown in Figure 8. The size of the color image is treated in the same manner as in the general input camera. Skeleton data provides 3D points containing depth data with respect to joints representing 16 articulation points of human being. The 3D points will be used to decide  $RZ_{min}$  and  $RZ_{max}$  representing a bounding box of a movable space of real characters. Similar to the previous mentioned method for a general camera,  $RZ_{min}$  is defined as the minimum among joints' depths of skeleton data when a user goes far away from the camera,  $RZ_{max}$  is defined as the maximum among joints' depths of skeleton data when a user gets close enough to the camera. The depth  $RZ_t$  at a specific time  $t$  is calculated as follows:

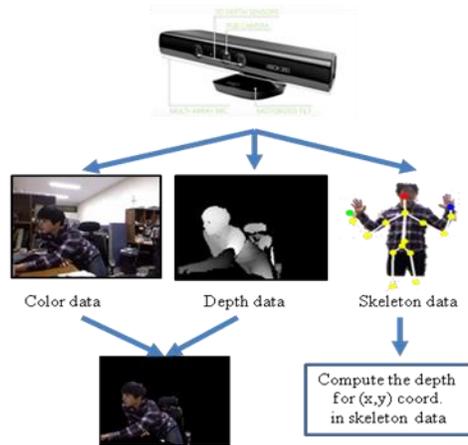
$$HLR_t = (H_t + L_t + R_t) / 3.0 \quad (3)$$

$$RZ_t = (HLR_t - RZ_{min}) / (RZ_{max} - RZ_{min}) \quad (4)$$

In Equation 3,  $HLR_t$  is the average of three depths, head one  $H_t$ , left hand one  $L_t$ , and right hand one  $R_t$  at a specific time  $t$ .



Figure 7. Computing Depth Information by using Face and Hand Regions

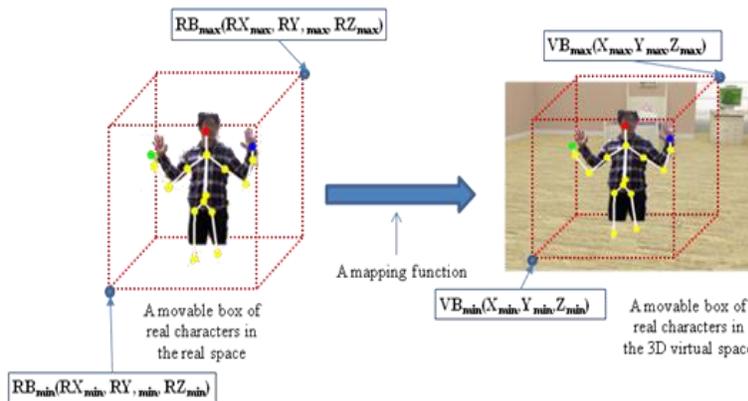


**Figure 8. Computing Depth Information by using Skeleton Data**

**2.4. Mixed Rendering of Real Character and 3D Virtual Space**

To simulate the movement of real-world characters in 3D virtual space, the bounding box *MRB* in which the movement of real characters is possible, should be placed into 3D virtual space. In this paper, we are able to give a virtual movable bounding box *MVB*, defined with two corner points  $VB_{min}(VX_{min}, VY_{min}, VZ_{min})$  and  $VB_{max}(VX_{max}, VY_{max}, VZ_{max})$  as shown in Figure 9.

Embedding real-world characters into 3D virtual space can be performed by the function for mapping from *MRB* to *MVB*. The mapping function can be easily defined so that the depth  $RZ_t$  in Equation (2) and (4) will be recalculated to embed real characters into 3D virtual space by using the mapping function. With the mapped depth value  $VZ_t$  from  $RZ_t$ , the real characters can be embedded into 3D virtual space. The embedding is executed through the texture mapping of an image containing real characters into an object with  $VD_t$  as *z*-value in 3D virtual space. The object may become either a rectangle or a human model. When the texture mapping with respect to the object is done, all pixels designated as background color of the image are assigned as full transparency for giving background removal effects. The other hand, all pixels for foreground colors are assigned to full opacity.



**Figure 9. System Configuration for Embedding Real Characters into the 3D Virtual Space**

Summarizing all the technical issues, the data structure representing real characters to be mapping into 3D virtual space can be expressed as follows:

```
RealCharacter {  
    String          Name  
    String          Description  
    Image          CaptureImage  
    BoundingBox RealSpace (RXmin, RYmin, RZmin, RXmax, RYmax, RZmax)  
    BoundingBox VirtualSpace (VXmin, VYmin, VZmin, VXmax, VYmax, VZmax)  
    Animation      NONE, LINEAR, QUAD  
    Collision      False  
    Position      Initial Position  
    Translation    Translate  
    Rotation      Rotate  
    Scaling       Scale  
    Object        MappedObject  
};
```

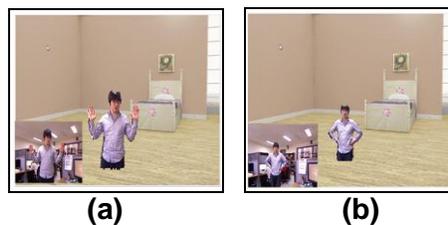
### 3. Experimental Results

Our experimental system is configured with a KINECT device and a PC with 2.66GHz Quad-core CPU and 4GB RAM. KINECT SDK, OpenCV, OpenGL, and Microsoft Visual Studio 2010 are used for development of the system. The experimental results show that the proposed method achieves 13 to 18 frames per second (fps) for embedding real characters extracted from the captured images into the 3D virtual space. Figure 10 shows the embedding result of a real character by applying the mapping function described in Section 2.4.



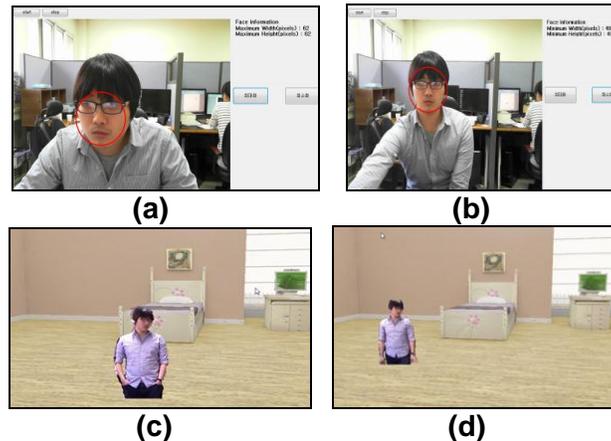
**Figure 10. The Rendering Results of a 3D Virtual Space (a) and after Embedding Real Characters (b)**

In Figure 11, the left bottom images are captured from the KINECT device. After getting depth information from the skeleton data in the image, the extracted real character is embedded in the 3D virtual space by using the mapping function described in Section 2.4. If a real character moves in the real space, his/her embedding result will be represented in the 3D virtual space while tracking the movement. Figure 11 shows the embedding results of a real character while he/she is moving backward.



**Figure 11. The Rendering Results when a Real Character is at a Specific Location (a) and after he/she Moves Back (b)**

Using KINECT, the bounding box in the real space is constructed from the depth data. On the other hand, with Web cameras only, the bounding box can be obtained from the sum of both face and hand regions as illustrated in Section 2. In Figure 12, the sizes of two face regions designated by red circles are  $67 \times 67$  pixels (Figure 12 (a)) and  $45 \times 45$  pixels (Figure 12(b)), respectively. Figure 12 (c) and Figure 12 (d) show movement results of the real character obtained by applying the mapping function after computing the real bounding box.



**Figure 12. The Movement Results Obtained by Applying the Size of the Face Region of a Real Character**

#### 4. Conclusion

Our experimental results show that the data structure described in Section 2.4 is sufficient to represent real characters obtained from both of the KINECT and Web cameras into the 3D virtual space. To perform natural animations of real characters as well as their visualization in the 3D virtual space, our data structure may show some drawbacks, since it was originally made from the 2D information not from the full 3D information. Moreover, real characters in the 3D virtual space might want to do various interactions with each other by using their gestures [12, 13]. This type of interaction is required to give immersions and will be implemented in the future.

#### Acknowledgements

This research was supported by National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (No.2012-0000479), and by the ICT standardization program of MKE(The Ministry of Knowledge Economy).

#### References

- [1] H. Srimathi and S. K. Srivatsa, "Design of Virtual Learning Environment using SCORM standards", Journal of Theoretical and Applied Information Technology, (2005-2008), pp. 542-546.
- [2] A. Soller, B. Goodman, F. Linton and R. Gaimari, "Promoting Effective Peer Interaction in an Intelligent Collaborative Learning System", In Proceedings of the 4<sup>th</sup> International Conference on Intelligent Tutoring Systems (ITS 98), San Antonio, TX, (1998), pp. 186-195.
- [3] G. Berry, "Small-wall, A Multimodal Human Computer Intelligent Interaction Test Bed with Applications", Dept. of ECE, University of Illinois at Urbana-Champaign, MS thesis, (1998).

- [4] J. S. Jeong, C. Park, J. J. Han, M. S. Im, R. H. Jang, M. Kim and K. H. Yoo, "Development of a 3D Virtual Studio System for Experiential Learning", CCIS (Communications in Computer and Information Science), vol. 150, (2011).
- [5] M. Kim, J. S. Jeong, C. Park, R. H. Jang and K. H. Yoo, "A Situated Experiential Learning System Based on a Real-Time 3D Virtual Studio", Advanced Science Letters, (in press, accepted on March 2012).
- [6] Sloodle, <http://www.sloodle.org>.
- [7] Second Life, <http://secondlife.com/>.
- [8] Moodle, <http://www.moodle.org>.
- [9] Microsoft Corp, KINECT for Windows, (2012).
- [10] G. Welch and G. Bishop, "An introduction to the Kalman filter", TR95-041, Dept. of Computer Science, Univeristy of North Carolina at Chapel Hill, (2006).
- [11] OpenCV, <http://www.opencv.org>.
- [12] J. S. Jeong, C. Park and K. H. Yoo, "Hand Gesture User Interface for Transforming Objects in 3D Virtual Space", Communication in Computer and Information Science, vol. 262, (2011), pp. 172-178.
- [13] J. O. Kim, J. S. Jeong, C. Park, N. Baek and K. H. Yoo, "A Gesture based Camera Controlling Method in the 3D Virtual Space", Advanced Science Letters, (in press, accepted on March 2012)

## Authors



**Ji-Seong Jeong**

He is a PhD candidate of department of IIE(information industrial engineering) at Chungbuk National University, Korea. He received the B.S. and M.S. in Computer Education and IIE from Chungbuk National University, Korea in 2008 and 2010, respectively. His research interests include computer graphics, e-Learning, u-Learning, and multimedia.



**Jin-A Park**

She received the B.S degree in the Computer Science from National Institute for Lifelong Education in 2005. She is currently working towards M.S degree on Information and Computer Education from Chungbuk National University, South Korea. Her main research interests include Computer Graphics, computer education.



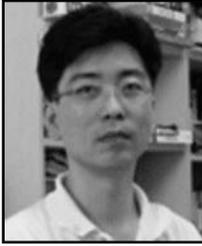
**Sin Ae Kwon**

She received the B.S degree in the Computer Science from National Institute for Lifelong Education in 2006. She is currently working towards M.S degree on Information and Computer Education from Chungbuk National University, South Korea. Her main research interests include Computer Graphics, computer education.



**Chan Park**

He is a Senior Researcher in Korea Internet Software Corporation. He received the B.S., M.S. and PhD. in Computer Education and Information Industrial Engineering from Chungbuk National University, Korea in 2003, 2007 and 2012, respectively. His research interests include computer graphics, e-Learning, u-Learning and multimedia.



**Nakhoon Baek**

He is currently an associate professor in the School of Computer Science and Engineering at Kyungpook National University(KNU), Korea. He received his BA, MS, and Ph.D degree in Computer Science from the Korea Advanced Institute of Science and Technology(KAIST) in 1990, 1992, and 1997, respectively. His research interests include real-time and mobile graphics systems.



**Rae-Hyun Jang**

He is a president in KIS (Korea Internet Software) Corporation. He received his BA in Department of Management Information System from Chungbuk National University in 1989, and his MS in Graduate of Management from Chungbuk National University in 2003. His research interests include computer graphics, e-Learning, and virtual education system.



**Kwan-Hee Yoo**

He is a professor in the Department of Computer Education and in the Department of Information and Industrial Engineering at Chungbuk National University, Korea. He received his BA in Computer Science from Chonbuk National University in 1985, and his MS and PhD in Computer Science from KAIST (Korea Advanced Institute of Science and Technology) in 1988 and 1995, respectively. His research interests include computer graphics, 3D characters animation, and dental/medical applications.