

Geometry-Aware Interactive AR Authoring Using a Smartphone in a Wearable AR Environment

Jeongmin Yu, Jinwoo Jeon, Jinwoo Park, Gabyong Park,
Hyung-il Kim, and Woontack Woo^(✉)

KAIST UVR Lab., Daejeon, South Korea
{jmyu119, zkrkwlek, jinwoaa, gypark, hyungil, wwoo}@kaist.ac.kr

Abstract. This paper presents an augmented reality (AR) authoring system that enables an ordinary user to easily build an AR environment by manipulating and placing 3D virtual objects. The system tracks users' hand motions via an RGB-D camera which built-in an optical see-through (OST) head mounted display (HMD), and interactive features applied to virtual objects by the pre-defined hand gestures. In addition, the virtual objects and dynamic paths are placed by using a smartphone in a real-world space. To implement this system, three cored technologies are needed: (i) segmentation of regions of spaces and real-world objects (ii) hand tracking and gesture recognition for manipulating virtual objects, (iii) dynamic path placing of virtual objects using a smartphone wearing an OST HMD. We implement a prototype of the proposed system for testing its feasibility. To the end, we expect that our system enables simplify AR technology usage to ordinary users who are unfamiliar with professional AR authoring tools.

Keywords: AR contents authoring · Hand-augmented virtual object interaction · Wearable AR environment

1 Introduction

Augmented reality (AR) technology enables users to approach supplementary information by mixing with virtual objects in the real world in real-time with many kinds of devices such as mobile smartphones, tablet, head mounted display (HMD), and high performance PCs [1]. Using this technology, the users can be worked with digital virtual contents in various real-world spaces such as industry sites, cultural performance venues, classrooms and so forth. Owing to these attributes, AR technology is applied to various fields such as tele-educations, military, games, repair/maintenance, and gallery/exhibition [2].

For the past few decades, many researchers have been developed on AR authoring systems/tools to easily manipulate AR contents for an ordinary user. The AR authoring methods are classified two categories: AR-based libraries and GUI-based authoring tool. The popularly used AR libraries for authoring are MRT [6], DWARF [5], osgART [4], AR Toolkit [3], so forth. Among them, ARToolKit library is popularly used because it can be transferred into other computing/program languages such as FLARToolKit

(Flash ActionScript) and NyARToolKit (Java). However, these can be only accessible and practically useful for professional programmers.

GUI-based AR authoring tools give more intuitive interaction to users. Utilizing these tools, users are able to perform AR authoring process as a manner of the point-and-click. The widely used AR authoring tools are AMIRE [7], DART [8], ATOMIC Authoring Tool [10], ComposAR [9], and these are used without having to write any program code. Even though these GUI-based authoring tools are easier to use than AR libraries, users should still acquire the specialized knowledge of the tool and just work in PC environment. Meanwhile, recently intuitive AR authoring systems using smart devices or natural user interface (e.g., hand gestures) are developed which enables a user to easily build an AR world in an in-situ environment and manipulate 3D virtual content to it. [11] shows manipulation the AR contents using multi-touch interface of smart mobile device and [12] proposes an AR authoring method for unknown outdoor scene using mobile devices. Project Tango [13] is a mobile authoring system which forms a 3D map of unknown indoor scene using a depth sensor. However, these systems have a cumbersome point that a user should see the augmented spot through a narrow mobile device display.

In this paper, to overcome above mentioned shortcomings, we present a geometry-aware interactive AR authoring system which enables an ordinary user to easily build an AR world in an in-situ environment through manipulating and placing virtual objects. The proposed system tracks users' hand motions via an RGB-D camera which built-in an OST HMD, and interactive features applied to virtual objects by hand gestures. Then the user can easily add and delete dynamic paths of virtual objects to real-world environment. To develop this system, three core technologies are needed: geometry awareness by segmentation of space and object regions, manipulating virtual objects by hand tracking and gesture recognition, and placing virtual objects and dynamic paths with a smartphone wearing an OST HMD. Through a preliminary prototype system implementation, we confirm its feasibility as a future AR authoring tool. We expect that the proposed system can be applicable to many AR applications such as education/training, urban planning, games and etc.

The remainder of this paper is organized as follows. The proposed overall system is presented in Sect. 2. In Sect. 3 introduces preliminary implementation and its result. Lastly, the conclusions and future works are presented in Sect. 4.

2 Proposed Authoring System

Figure 1 shows the proposed overall system diagram of AR authoring. In this system we use a smartphone wearing OST HMD (e.g., Microsoft Hololens) for placing of AR virtual objects, and use an egocentric RGB-D camera which is built in an OST HMD and a wearable sensor (e.g., smartwatch) for accurate hand tracking and gesture recognition. Utilizing hand tracking and gesture recognition, interactive features can be applied to a virtual object (i.e., enlargement, shrinkage, rotation). Then, the authored virtual objects are placed in real-world environment using rotation and touch direction information from the sensors (e.g., IMU and touch screen) built into a smartphone.

The detail methodological description of geometry-aware interactive AR authoring is presented as follows.

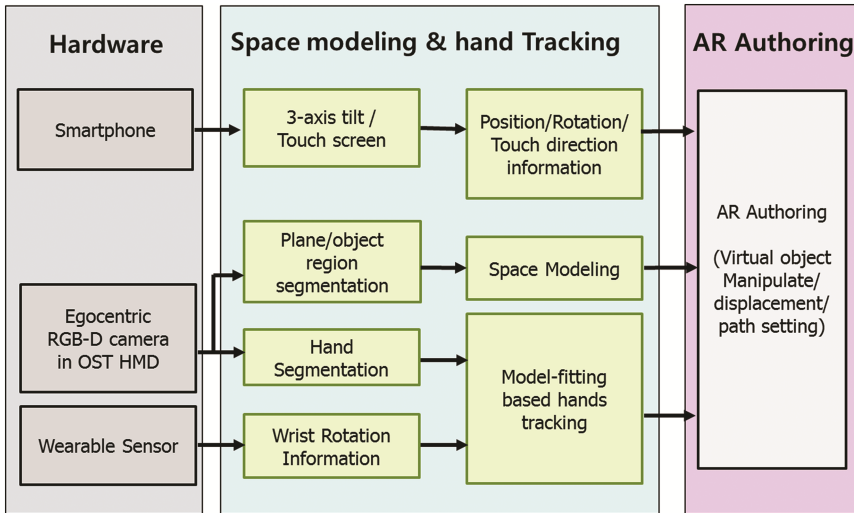


Fig. 1. The proposed AR authoring framework.

2.1 Segmentation of Space and Real-World Object Regions

The indoor space consists of objects and structures such as walls, floors, and ceilings. The object regions can be easily estimated by removing structures of planes. Figure 2 shows the procedure of the proposed irregular space/object segmentation. First, we calculate local surface normal vectors on a RGB-D image. Then, we cluster the normal vectors and calculate the plane corresponding to the structure. Finally, using the connected component labeling (CCL) algorithm, we segment the object regions of the plane regions from a RGB-D image.

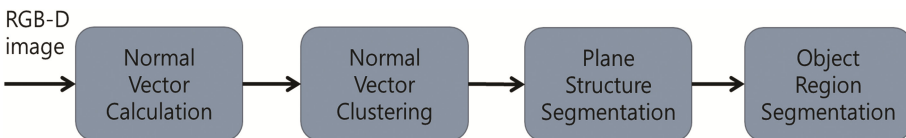


Fig. 2. Procedure of segmentation of space and object regions

2.1.1 Segmentation of Plane Regions

We first calculate the local surface normal vectors from a depth camera to estimate a planar area. After the azimuth and elevation of normal vectors are calculated, these are quantized and accumulated into a histogram. The normal vectors in the same bin are likely to exist in the same plane or parallel planes. In other words, the normal vectors

included in the local maxima and neighbors are belonging to objects that are planes of indoor structures. Otherwise, these are parallel with indoor structures. After classifying the planar regions, we segment their boundaries from the image. The boundaries are mostly generated by two cases. The first occurs at the intersection of the two planes. In this case, the normal vector of the adjacent region is different from the normal vector of the two planes. However, in most cases, when the wall and the other wall are met, they can be occluded by objects. In this case, the intersection line of two plane parameters is the boundary. The second is when the two planes are parallel. In this case, the normal vectors of the boundary have similar normal vectors to the planar regions, but the distance values are different. By using this property, it is possible to obtain the outline information of the planar regions. The planar parameters that are the furthest or lowest in the each group are used as the plane of indoor spatial structure.

2.1.2 Segmentation of Real-World Object Regions

The objects can be separated using the previously calculated plane geometry information. By calculating dot product between the 3D point and the parameter matrix main plane, the distance between the point and the plane can be calculated. If this value is less than a user-defined threshold, the point belongs to the plane. Otherwise, it belongs to the object. This process can be performed on all points to obtain the binary image of the planes and objects. Then, CCL algorithm performs to segment the objects from indoor space.

2.2 Manipulating of Virtual Object Hand Tracking/Gesture Recognition

2.2.1 Method for Hand Tracking

The proposed hand tracking algorithm employs a model-based method as Fig. 3. The method first defines a 3D geometric hand model whose joints are controlled by 26 parameters. Then, it solves an optimization problem to find the 26 parameters. To do so, we define an objective function to quantify the error between the rendered hand model and observation. The objective function E is defined as the following equation.

$$E = \sum_{i=0}^{width-1} \sum_{j=0}^{height-1} D(o(i,j), r(i,j)) \tag{1}$$

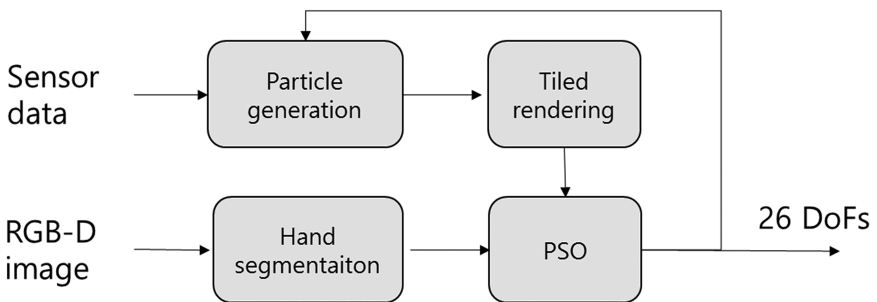


Fig. 3. Flowchart of the proposed hand tracking method.

Where $o(i, j)$ is depth value of pixel (i, j) in a depth image from a depth camera, and $r(i, j)$ is depth value of pixel (i, j) in depth image from rendered model. The function $D(\cdot)$ is Euclidean distance between two inputs. The objective function is optimized based on particle swarm optimization (PSO) algorithm [14], and the particle update rule is set as [15]. However, this method has a weakness of error accumulation after occurring tracking failure. To alleviate this, the data from wrist sensor is passed to the particle generation module. The particles are generated within the boundary decided by the data from the wrist sensor and the solution in the previous frame. This method is useful to reduce search range that the particles move.

2.2.2 3D Contents Manipulation

The optimized hand parameters are used to manipulate virtual objects as scaling, rotation, translation which shown in Fig. 4. To do this, the meshes are created by vertices transformed by the hand model parameters. The method for scaling and rotation becomes intuitive if the interaction system can detect the touch points between the mesh of the hand model and a virtual object. After detection of more than two collisions by finger models, the virtual object is translated or rotated by the wrist parameters of the hand model. With respect to the scaling, it requires a gesture to conduct it. We defines it as one click. After one collision detection of one finger, the scaling parameter of the virtual object is controlled by the area of the five fingers. To go back from the scaling mode, the user would touch the virtual object with one finger.

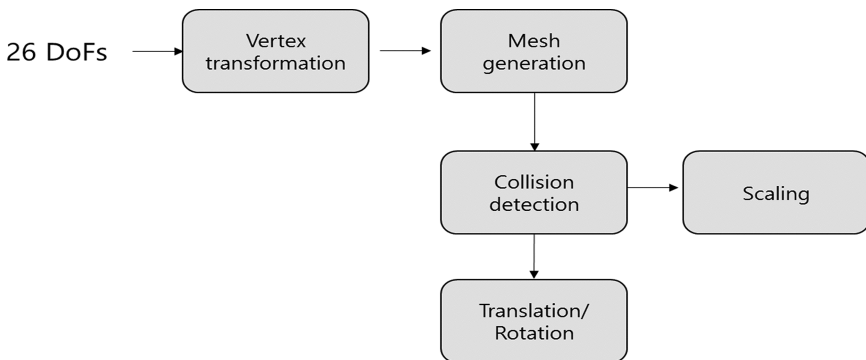


Fig. 4. Flowchart of the proposed virtual object manipulation.

2.3 Placing of Virtual Object Using a Smartphone

There are various methods using a bare hand [17, 18] and a smart device as input interface for manipulating virtual objects are developed for making an AR environment. Among them, smart devices can be used for a long time without physical fatigue, so we propose a method to manipulate virtual objects using it. Figure 5 shows the flowchart of the authoring system. The system requires two kinds of hardware such as a smartphone

and an OST HMD. Utilizing these, users can manipulate virtual objects and insert dynamic paths into real space.

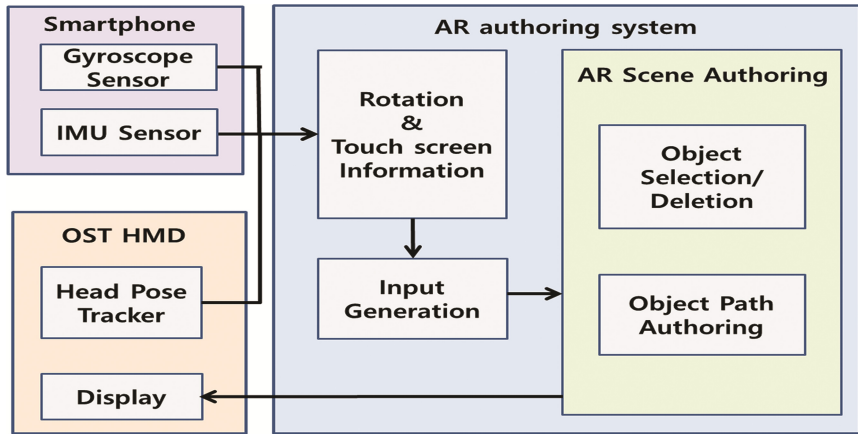


Fig. 5. Flowchart of the proposed virtual object manipulation system.

2.3.1 Rendering and Inserting of Dynamic Paths

Utilizing a casual smartphone as a tool for manipulating virtual dynamic paths and virtual objects, our system allows a user to create his/her own content in augmented reality environment. We have two advantages in terms of dealing with dynamic paths for virtual movable objects and increasing user's immersion.

First, a user can arbitrarily make and manipulate virtual paths with a simple interaction on a touch screen and a built-in IMU of a smartphone. Basically, without any special and expensive tools like an HTC VIVE controller, it is very familiar for the users to use a smartphone to interact with virtual objects. Exploiting a smart phone as a mouse, a user moves a cursor and selects a virtual object or a specific menu in a form of graphic user interface on an HMD. For example, when creating and modifying a dynamic path, it can be easily done by selecting and transferring one of key positions which compose a dynamic path as shown in Fig. 6. In this case, selection and transfer is performed by finger tap and drag and drop respectively. Because paths are managed in a list container, it is possible to insert and delete key positions stably and dynamically. After completing path manipulation, a user can assign a path to movable objects by dragging and dropping them onto the specific path's key position. Then, with a proper animation assigned by a user beforehand, the objects start to move along the path they have.

Second, in terms of rendering 3D virtual objects, we exploit static LDR image, extracted in a form of spherical environment map via a Ricoh Theta 360 digital camera, as a distant light to enhance the realism of augmented objects. Different from [16], a high dynamic range image (HDRI) is not used in our system for the simple process. However, an LDR image also gives acceptable results without extremely glossy materials. Furthermore, because users typically perform authoring in a limited indoor and

scarcely dynamic space, static distant light of an LDR image can enhance user's immersion by producing realistic visual results.

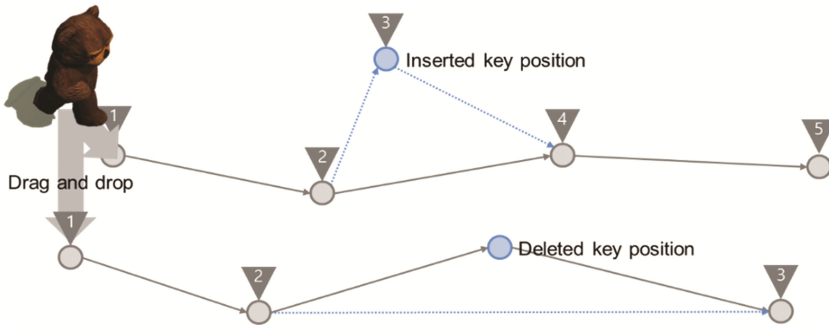


Fig. 6. Manipulation of dynamic paths and placing to movable virtual objects.

3 Implementation

3.1 Hardware and Software Configuration

We configured our prototype system using commercially available devices. Our system consists of a computing unit for computation, an OST HMD for visualization, HMD tracker for 6DOF HMD pose tracking, a near-range depth sensor for hand tracking, and a smartphone for AR authoring. Smartphone and computing unit are connected with Wi-Fi communication. We used a Microsoft Hololens, which is a computing device and an OST HMD with inside-out tracker. In addition, we tested various Android OS smartphones.

System modules are implemented and integrated with Unity Engine and Windows Universal Platform. Also, smartphone application is implemented with Android SDK. Figure 7 illustrates configuration of proposed system.

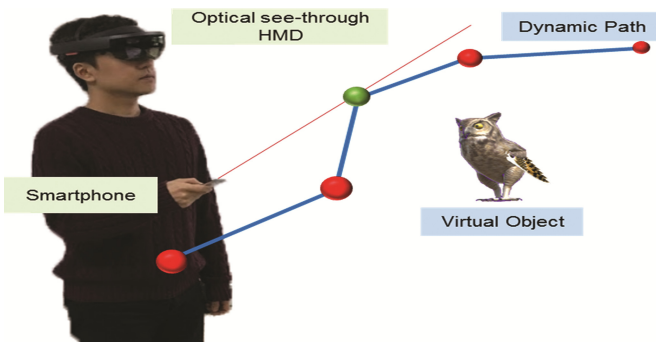


Fig. 7. Configuration of proposed system

3.2 Initial Implementation Result

Figure 8 shows initial result of our AR Authoring system prototype, which uses smartphone to author augmented space. User wearing optical see-through HMD can use one's own smartphone to select, place, and manipulate virtual objects in user's physical space. Also, user can generate dynamic path of virtual object by just manipulating key-points, and dragging virtual object into generated path. Our system enables a user to generate and organize a user-friendly augmented space without any professional programming or software skills.

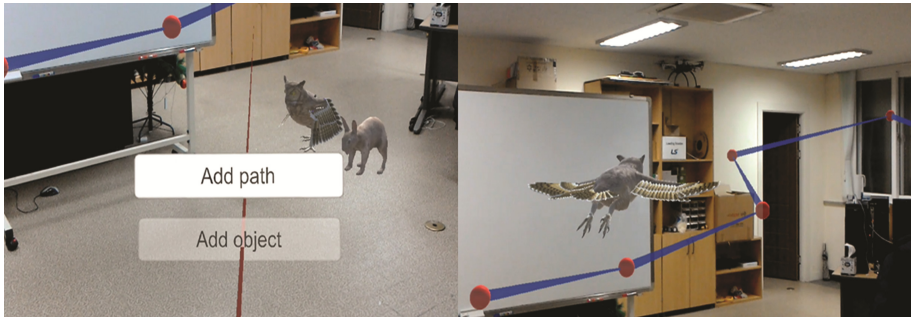


Fig. 8. User's view of our prototype system. User uses smartphone to select UI or virtual objects (left), and generate dynamic path of augmented objects (right).

4 Conclusions and Future Works

In this paper we have presented a geometry-aware interactive AR authoring system using a smartphone wearing an OST HMD, which enables an ordinary user to intuitively organize an AR space without any professional programming and tools. The proposed systems contain three core technologies: geometry awareness by segmentation of space and object regions, manipulating virtual objects by hand tracking and gesture recognition, and placing virtual objects and dynamic paths with a smartphone wearing an OST HMD. Preliminary implementation result shows its strong possibility as a future AR tool. We expect that the proposed AR system can be applicable to many AR applications such as education, training, urban planning, games, and so forth.

As the future works, we plan to develop hand tracking and recognition for manipulating virtual objects and light estimation for rendering virtual objects.

Acknowledgements. This work was supported by National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (NRF-2014R1A2A2 A01003005) and by ICT R&D program of MSIP/IITP. [R7124-16-0004, Development of Intelligent Interaction Technology Based on Context Awareness and Human Intention Understanding].

References

1. Azuma, R.: A survey of augmented reality. *Presence* **6**(4), 355–385 (1997)
2. Azuma, R., Bailloot, Y., Behringer, R., Feiner, S., Julier, S., MacIntyre, B.: Recent advances in augmented reality. *IEEE Comput. Graph. Appl.* **21**(6), 34–47 (2001)
3. Kato, H., Billinghurst, M.: Marker tracking and HMD calibration for a video-based augmented reality conferencing system. In: *Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality* (1999)
4. Looser, J., Grasset, R., Seichter, H., Billinghurst, M.: OSGART – a pragmatic approach to MR. In: *Industrial Workshop at ISMAR* (2006)
5. Bauer, M., Bruegge, B., Klinker, G., MacWilliams, A., Reicher, T., Riss, S., Sandor, C., Wagner, M.: Design of a component-based augmented reality framework. In: *ISMAR* (2001)
6. Freeman, R.: Mixed reality toolkit. MSc VIVE Final Year Project Report (2004)
7. Grimm, P., Haller, M., Paelke, V., Reinhold, S., Reimann, C., Zauner, J.: AMIRE - authoring mixed reality. In: *IEEE International Augmented Reality Toolkit Workshop* (2002)
8. MacIntyre, B., Gandy, M., Dow, S., Bolter, J.D.: DART: a toolkit for rapid design exploration of augmented reality experiences. In: *UIST* (2004)
9. ATOMIC Authoring Tool. <http://www.sologicolibre.org/projects/atomic/en/>
10. Seichter, H., Looser, J., Billinghurst, M.: ComposAR: an intuitive tool for authoring AR applications. In: *ISMAR* (2008)
11. Jung, J., Hong, J., Park, S., Yang, H.: Smartphone as an augmented reality authoring tool via multi-touch based 3D interaction method. In: *Proceedings of the 11th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry*, pp. 17–20 (2012)
12. Langlotz, T., Mooslechner, S., Zollmann, S., Degendorfer, C., Reitmayr, G., Schmalstieg, D.: Sketching up the world: in situ authoring for mobile augmented reality. *Pers. Ubiquit. Comput.* **16**(6), 623–630 (2012)
13. <https://www.google.com/atap/project-tango/>
14. Kennedy, J., Eberhart, R.C.: Particle swarm optimization. In: *International Conference on Neural Networks*, pp. 1942–1948 (1995)
15. Park, G., Argyros, A., Woo, W.: Efficient 3D hand tracking in articulation subspaces for the manipulation of virtual objects. In: *CGI* (2016)
16. Paul, D.: Rendering synthetic objects into real scenes: bridging traditional and image-based graphics with global illumination and high dynamic range photography. In: *ACM SIGGRAPH 2008 Classes* (2008)
17. Ha, T., Feiner, S., Woo, W.: WeARHand: Head-worn, RGB-D camera-based, bare-hand user interface with visually enhanced depth perception. In: *IEEE ISMAR*, pp. 219–228 (2014)
18. Jang, Y., Noh, S., Chang, H., Kim, T., Woo, W.: 3D finger CAPE: clicking action and position estimation under self-occlusions in egocentric viewpoint. *IEEE Trans. Visual Comput. Graphics* **21**(4), 501–510 (2015)