Interactive Annotation on Mobile Phones for Real and Virtual Space Registration

Hyejin Kim[*]
GIST U-VR Lab

Gerhard Reitmayr[†]
Graz University of Technology

Woontack Woo[‡]
GIST U-VR Lab

ABSTRACT

Registration of real space and virtual information is a fundamental requirement for any augmented reality system. This paper presents an interactive method to quickly create a 3D room model and annotate locations within the room to provide registration anchors for virtual information. The method operates on a mobile phone and uses a visual rotation tracker to obtain orientation tracking for in-situ applications. The simple interaction allows non-expert users to create models of their environment and thus contribute marked-up representations to an online AR platform.

Keywords: Annotation, mobile AR, interactive modeling, visual rotation tracking

Index Terms: H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems—Artificial, augmented, and virtual realities; H.5.2 [Information Interfaces and Presentation]: User Interfaces—Interaction styles

1 INTRODUCTION

Current work for creating annotations in mobile AR applications focuses on outdoor environments based on the ubiquitous tracking solution which combines GPS and orientation sensors. AR Browsers such as Junaio[1] allow users to place 3D objects or information tags at GPS locations and move them on a virtual ground plane relative to the user's current location. Langlotz et al. [3] demonstrate annotation authoring using panoramic images of the user's location, providing more accurate placement of the annotations. However, no system is currently designed for indoor environments.

In this paper, we focus explicitly on annotations for indoor environments and individual rooms. Our system allows to quickly capture the dimensions of a room, approximated as a box, and to annotate and mark-up locations and items in the room. It operates at interactive frame-rates on a mobile device and provides simple touch-interaction to specify room dimensions, location and extends of rectangular areas on the room's surface. These areas serve as anchors for linking virtual information to the real space represented by the room.

2 INTERACTION AND ANNOTATIONS

The interactive annotation method comprises of four interaction modes, one for each of the tasks: room modeling, marking up locations, annotating, and viewing in AR and VR. In this section, we assume that we have an estimate of the camera orientation R relative to an unknown reference rotation, but with the direction of gravity aligned to the -z axis. The details of this orientation tracking are described in section 3.

[*]e-mail: hjinkim@gist.ac.kr

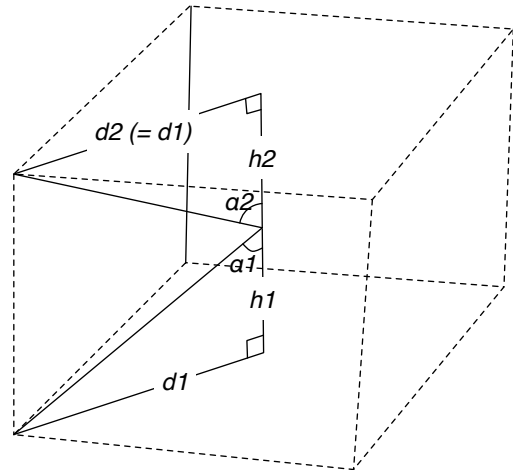[†]e-mail:reitmayr@icg.tugraz.at

[‡]e-mail:wwoo@gist.ac.kr

Figure 1: Picking algorithm setup with person height.

2.1 Room modeling

In room modeling mode, user measures the dimension of a room, approximated as a box. Standing in a fixed position, and looking at a corner or edge of a wall with the floor computes the distance of that point from the user to the intersection with the floor using the human height and the current camera rotation R. We assume that the user's operating height h_1 from ground to mobile device is calibrated and known to the system (see Figure 1). Then the modeling system computes an angle α_1 from

$$\cos \alpha_1 = \frac{\vec{f}_w \cdot \vec{g}}{|\vec{f}_w||\vec{g}|}. \qquad (1)$$

Here, $\vec{f}_w$ and $\vec{g}$ denote forward vector with respect to the reference coordinate system and the gravity vector, $(0,0,-1)$. The forward vector can be obtained from $\vec{f}_w = R^{-1}(0,0,-1)^T$, where, R^{-1} is inverse of current camera rotation. Then, by using the height h_1 and the angle α_1, a distance d_1 is computed as

$$d_1 = h_1 \tan \alpha_1. \qquad (2)$$

Distances on the floor like d_1 can be used to define the width and depth of the box from the user's position. However, to obtain a 3D model, we need the height as well. Thus, the system requires one more interaction to measure the height h_2 by pointing at the ceiling edge of the first wall. d_1 and d_2 are the same because ceiling and floor are parallel planes. Then, using equation (1) the angle $\alpha_2 = \pi - \alpha_1$. Finally, h_2 is computed using d_2 and α_2,

$$h_2 = \frac{d_2}{\tan \alpha_2}. \qquad (3)$$

After picking a floor and ceiling line in the first wall, the user rotates and looks at each of other three walls clockwise while picking only one floor or ceiling line.

Figure 2: An interactive modeling application for mobile phones captures the shape of a room and allows for annotation of locations. (Left) Marked-up location for the window. (Middle) Virtual objects added to the AR view. (Right) A VR exocentric viewpoint of the scene.
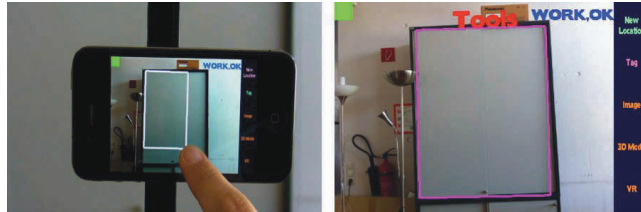

Figure 3: Screenshot of (Left) selecting a rectangle and (Right) final rectangle after mark-up.

2.2 Mark-up locations

After room modeling, the user is able to mark-up locations with a rectangle that represents links to virtual content (see Figure 3). A marked-up location has four kinds of attribute: global and local coordinates to present locations, transformation from local to global coordinate system for annotating content, and a texture for the resulting 3D model and VR view.

2.3 Annotating locations

Three types of virtual content are supported: text, images and 3D models can be added as annotations to marked-up locations. To annotate marked-up locations, some attributes about the virtual content and the link are needed as described in [1]. In case of text, virtual content has only one attribute, the text field, while an image requires texture and geometry to be textured. For 3D model, more attributes are needed such as vertex buffer object handles, 3D model, textures for each material in the model. Finally, link attributes are defined as a pair of id and transformation to link marked-up location and virtual content.

3 TRACKING

All interaction in our annotation method is operating relative to the current room. Therefore, the system needs to continuously estimate the pose of the mobile device with respect to that room. We restrict the pose estimation to a pure rotation with a fixed position of the camera to obtain a reliable tracking solution. Thus, our tracking solution only estimates orientation of the device.

3.1 Visual Rotation Tracker

The visual rotation tracker estimates the device's camera orientation with respect to an arbitrary starting orientation. As the camera pans around, the tracker builds a map of the environment consisting of keyframes that captured at various orientations, very much in the spirit of systems like PTAM [2]. The restriction to a pure rotation allows us to only estimate orientation between keyframes based on point correspondences without the need to estimate a 3D map for these points. The design is similar to recent work by Lovegrove et

al.[4] with the difference that we only use sparse 2D-2D correspondences, instead of full template matching between keyframes.

At any point in time, the track system maintains a set of keyframes C_i that consist of: an orientation R_i mapping from the first keyframe C_0 to the keyframe C_i, an image pyramid of images created by repeatedly half-sampling the keyframe image, and a set of interest point locations $\{p_{i,j}\}$. Furthermore, the tracker maintains an estimate of the rotation R_t at the current timestamp t. At the initial keyframe C_0 the device's rotation R_g with respect to gravity is also measured from the device sensors. The full rotation R that is passed to interaction system consists then of the combined rotation $R = R_t \cdot R_g$ relative to gravity.

4 CONCLUSION AND FUTURE WORK

In this paper, we presented our method for modeling several rooms and annotating locations with virtual content. As shown in Figure 2, we made a 3D room model and marked-up locations by adding keywords of the object, replacing images for planar objects and added virtual plants. Thanks to rotation tracker and 3D room model, we can navigate the scene in VR view.

Future steps will include the integration with an online system to store and retrieve models in a larger environment. This would make the models persistent and allows creating larger annotated areas through the collaboration of users. Another aspect is to retain the map information from the visual tracker as well. This information should be stored with the 3D models to enable further tracking for a user revisiting the area. Finally, through the annotations of planar features on the wall, explicit planar tracking targets can be created as well as to enable independent and full 6DoF tracking for such locations.

REFERENCES

[1] H. Kim, H. Yoon, A. Choi, W. Baek, I. Lee, D. Kim, and W. Woo. Data markup representation for mixed reality contents. In *The International AR Standard Meeting 2011*, 2011.

[2] G. Klein and D. Murray. Parallel tracking and mapping for small ar workspaces. In *Proc. ISMAR 2007*, Nara, Japan, Nov. 13–16 2007.

[3] T. Langlotz, S. Mooslechner, S. Zollmann, C. Degendorfer, G. Reitmayr, and D. Schmalstieg. Sketching up the world: in situ authoring for mobile augmented reality. *Personal and Ubiquitous Computing*, page Online First, 2011.

[4] S. Lovegrove and A. J. Davison. Real-time spherical mosaicing using whole image alignment. In *Proc. ECCV 2010*, volume 6313, pages 73–86, Heraklion, Crete, Greece, Sept 5–11 2010.