

Rotbav: A Toolkit for Constructing Mixed Reality Apps with Real-Time Roaming in Large Indoor Physical Spaces

Huan Xing¹ Xiyu Bao¹ Fan Zhang¹ Wei Gai¹ Meng Qi² Juan Liu¹ Yuliang Shi^{1,4}
Gerard de Melo³ Chenglei Yang¹ Xiangxu Meng¹

¹School of Software, Shandong University, Jinan, Shandong, China

²Shandong Normal University, Jinan, Shandong, China

³Rutgers University, Piscataway, New Jersey, USA

⁴Dareway Software Co., Ltd., Jinan, Shandong, China

ABSTRACT

This paper presents a toolkit called Rotbav for easily constructing mixed reality (MR) apps that can be experienced in real time in large indoor physical space via HoloLens. It resolves the problem that existing MR devices, e.g. HoloLens, are unable to scan and model an entire large scene with several rooms at once. We introduce a custom data structure called VorPa, based on the Voronoi diagram, to implement path editing, accelerated rendering and location effectively. Our experiments and applications show that the toolkit is convenient and easy to use for constructing MR apps targeting large indoor physical spaces, in which users can roam in real time.

Keywords: Mixed reality, Toolkit, HoloLens, Real-Time Roaming, Voronoi

Index Terms: Human-centered computing—Human computer interaction (HCI)—Interactive systems and tools—User interface toolkits; Human-centered computing—Human computer interaction (HCI)—Interaction paradigms—Mixed / augmented reality

1 INTRODUCTION

Microsoft HoloLens, a popular simple mixed reality (MR) heads-up display, can enable users to interact with 3D holograms blended with the real world. For many MR Apps, how to construct virtual scenes for large physical spaces is of principal concern. At present, the virtual scenario models in most mixed reality applications are designed by professionals through specialized tools such as Google SketchUp or Autodesk's 3D Studio Max [3], which is difficult for general users. To create a convincing MR experience, users need to perceive the real environment spatially. While HoloLens provides a spatial mapping module, it suffers from several limitations: Users are required to complete a scan of all parts of the environment [3], which is time-consuming and laborious for a large-scale interior space with complex room structure. Additionally, in large interior spaces, as the design content is enriched, the system performance declines when users roam around because of the limited computing resources provided by HoloLens [2].

This paper proposes a toolkit called Rotbav that aids in constructing mixed reality applications in a convenient and effortless manner. The toolkit can establish and maintain accurate mapping relationships between the virtual and physical spaces, and set appropriate transparency values according to the occlusion degree of the walls to provide a realistic mixed reality effect. It resolves the problem that existing MR devices cannot scan and model an entire large scene with several rooms at once and also provides a custom data structure called VorPa, based on the Voronoi diagram, to implement path editing, optimize the layout of virtual objects and accelerate rendering effectively.



Figure 1: Overview of our system. (a) The toolkit running on a Microsoft Surface; (b) Experiencing a scenario via a HoloLens.

2 SYSTEM DESIGN AND IMPLEMENTATION

The toolkit is composed of two parts (see Figure1): one is a 2D interface supporting users in designing virtual scenes using multi-touch technology, so that the 2D map of the space can be drawn and the virtual objects can easily be manipulated; the other is the experience device, which is a HoloLens.

2.1 Constructing MR Environments

In the following, we introduce how our toolkit enables constructing MR environments for interior spaces, especially for large-scale spaces with several rooms, such as museums and art galleries. Our method consists of the following steps.

Step 1: Constructing the initial virtual scenes. In this stage, only walls are considered. Firstly, we obtain the digital 2D map of the real space from three kinds of data: structure parameters of the real space, architectural drawings and existing 3D models. After obtaining the 2D map, we can construct the 3D scene via a sweeping operation, and then arrange the virtual objects in it.

Step 2: Setting mapping relationships between the virtual and physical space. We calculate transformation relationship of coordinates between the virtual space (points from the toolkit) and the real space (points from the HoloLens coordinate system).

Step 3: Setting transparency values of walls. Specifically, if a real wall is opaque, we set the transparency value as 1. For a transparent or semi-transparent wall such as a glass wall, we set the transparency value between 0 and 1. We will introduce how to use the transparency value to accelerate rendering of realistic MR scenes in Section 2.2.

Step 4: Setting spatial anchors to correct deviations. There may be deviation errors between the virtual model and the real space resulting from inaccurate tracking. We use spatial anchors provided by HoloLens to resolve this problem. Some spatial anchors are arranged at certain intervals in the MR environment. When a user walks into the scope of one spatial anchor, the system will obtain the updated coordinates of the spatial anchor.

2.2 Data Structure VorPa

We propose a custom data structure called VorPa to effectively implement path editing, optimize the layout of virtual objects and

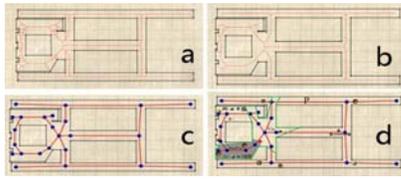


Figure 2: VorPa. (a) Voronoi path V in P ; (b) Initial roaming path R ; (c) Path editing; (d) The visible region of the current view point.

accelerate rendering. We define the structure of VorPa based on Voronoi diagrams as follows:

```

Struct VorPa {
  List <Edge> P; // 2D design wall list
  List <Path> V; // Voronoi path in polygon
  List <Path> R; // Roaming path in polygon
}
Struct Path {
  Edge edge; // Current Voronoi edge
  List<Model> modellist; // Models bound to path
  List<Edge> Vis; // Weak visibility region of path
  Bool direction; // Direction of edge
}

```

Path Editing. In VorPa, Voronoi path V is the Voronoi diagram of wall polygon P (see Figure 2a). We obtain initial roaming path R by deleting all edges that contain vertices of the polygon P (see Figure 2b). Designers can modify the nodes in the path to optimize the roaming path R , as shown in Figure 2c, e.g. adding, moving or merging vertices. In addition, designers can also set a recommended tour path for the users on the experience side by adding a direction.

Optimizing the Layout of Virtual Objects. By calculating the weak visibility region Vis_i of R_i in polygon P [1], VorPa provides recommendations for the model layout. Arranging models in Vis_i can avoid unreasonable layouts, such as dead corners or large variations in density in different fields of view. VorPa also pre-processes model data during the layout phase. When the designer places a model, the model is bound to all sub-paths which contain this model in the weak visibility region. This pre-processing operation serves to accelerate rendering.

Accelerating Rendering. VorPa can locate a user with regard to the appropriate sub-path R_i and obtain the model list M_i bound to that sub-path. The visible regions of the field of view $RealVis_i$ are calculated in real time [4], as shown in Figure 2d. First, the HoloLens stops rendering all models that do not belong to M_i . Then, for models in M_i , HoloLens also stops rendering the models in Vis_i but not in the $RealVis_i$. Finally, we calculate the model complexity F that is in M_i and $RealVis_i$. According to the distance between the model and the user's location, the accuracy of the model is tuned from far to near so as to guarantee $F < threshold$ and that the average frame number is at least 15. For transparent or semi-transparent walls, some special handling is introduced. As mentioned before, we set a transparency value between 0 and 1 as a parameter to select different levels of detail for the models. When users see through transparent or semi-transparent walls, we reduce the complexity of the models behind the walls according to the distance between models and users' locations as well as the transparency value of the wall.

3 EXPERIMENTS

To evaluate our toolkit and method, we designed a MR museum to conduct a series of experiments. In our experiments, we used the total number of triangles in a scene to quantify its capacity and frame rates to evaluate the user experience in terms of smoothness. In the design stage, we obtained the model layout and route planning according to the model layout strategy provided by VorPa. In the

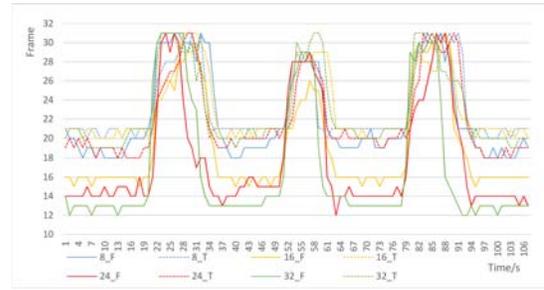


Figure 3: We present all results in a single chart. In the diagram, solid lines depict results without our algorithm; while dotted lines represent results using our algorithm.

experience stage, we first tested the threshold in advance and obtained the result that when the total number of triangles is no more than 10^5 , the frame rate is sufficient for a smooth user experience. Subsequently, we compared the performance of VorPa on four different capacities of scenes, namely 3.2×10^6 , 6.4×10^6 , 9.6×10^6 , and 12.8×10^6 . In each case, we experimented under two conditions, with and without our VorPa algorithm.

We plotted a line chart to compare the different experimental results in Figure 3. By considering the participants routes, the peaks and troughs in every line chart can be analyzed. When the participant walks past areas that feature numerous models, the frame rate declines and reaches a low point (trough). In circumstances with fewer models, the frame rate is higher. Overall, we observe that with an increase in capacity, the difference between the troughs is larger, which shows that our algorithm succeeds in enhancing the rendering effect and keeping it real-time. The effect is more pronounced as the capacity of the scene increases.

4 CONCLUSION

This paper presents a toolkit called Rotbav to aid general users in designing MR applications rapidly and conveniently, especially in large indoor space. Above all, the MR applications can provide a smooth and immersive roaming experience in real time. To satisfy this requirement, we introduce VorPa, a data structure based on Voronoi diagrams, which is beneficial to the user experience with regard to the strategies of path editing and accelerated rendering. This toolkit is demonstrated in an application and evaluated in a set of experiments. Our evaluation shows that it improves the real-time rendering efficiency of HoloLens effectively.

ACKNOWLEDGMENTS

This work was supported by the National Key Research and Development Program of China (2018YFC0831003, 2016YFB1001403), the National Natural Science Foundation of China (61802232), and the Natural Science Foundation of Shandong Province of China for Major Basic Research Projects (ZR2017ZB0419), the TaiShan Industrial Experts Program of Shandong Province of China (tscy20150305).

REFERENCES

- [1] D. Z. Chen and H. Wang. Weak visibility queries of line segments in simple polygons. *International Symposium on Algorithms - Computation*, 48(6):443–452, 2015.
- [2] A. Shearer, L. Guo, J. Liu, M. Satkowski, and R. LiKamWa. Characterizing bottlenecks towards a hybrid integration of holographic, mobile, and screen-based data visualization.
- [3] L. Yang, N. R. B. Stiles, and M. Meister. Augmented reality powers a cognitive prosthesis for the blind. 2018.
- [4] A. Zarei and M. Ghodsi. Query point visibility computation in polygons with holes. *Computational Geometry*, 39(2):78–90, 2008.