Automatic Generation of World in Miniatures for Realistic Architectural Immersive Virtual Environments

Andrea Bönsch* Sebastian Freitag* Torsten W. Kuhlen*

Visual Computing Institute, RWTH Aachen University
JARA – High Performance Computing

Abstract

Orientation and wayfinding in architectural Immersive Virtual Environments (IVEs) are non-trivial, accompanying tasks which generally support the users' main task. World in Miniatures (WIMs)—essentially 3D maps containing a scene replica—are an established approach to gain survey knowledge about the virtual world, as well as information about the user's relation to it. However, for large-scale, information-rich scenes, scaling and occlusion issues result in diminishing returns. Since there typically is a lack of standardized information regarding scene decompositions, presenting the inside of self-contained scene extracts is challenging.

Therefore, we present an automatic WIM generation workflow for arbitrary, realistic in- and outdoor IVEs in order to support users with meaningfully selected and scaled extracts of the IVE as well as corresponding context information. Additionally, a 3D user interface is provided to manually manipulate the represented extract.

Index Terms: I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction Techniques;

1 Introduction

In large-scale, information-rich architectural IVEs like factory planning, urban management, or terrain simulations, it is challenging to providing supportive techniques for orientation and wayfinding. This is caused by a combination of the large distances users have to travel and the abundance of available information they have to access and remember. Additionally, occlusion and cluttering prevent users from accessing all desired information from a single position.

One technique to overcome these challenges is a WIM [3], representing a complete scene as a 3D miniature model embedded in the IVE. It provides an exocentric view from any desired viewpoint onto the scene in addition to the user's egocentric view. Thus, the WIM offers the advantages of 2D maps regarding orientation and wayfinding, yet contains more information than a 2D floor plan.

Different approaches already provide WIMs as orientation, navigation, or object manipulation aid [2, 4, 5]. However, they all have limitations: rendering the complete IVE as a replica is not sufficient. Since scale, visual clutter, and occlusions reduces the WIM's usefulness, a suitable scene extract needs to be chosen. However, these extracts are often based on fixed rectangular excerpts [5], by which important details may be excluded from the WIM, or based on manual presegmentations [2] which may be too time-consuming. Only few approaches use an automatic scene decomposition dividing the complete IVE into a set of self-contained units from which a subset is used as extract in the WIM [4] .

After selecting those extracts, their insides need to be revealed. Here, previous scene structure knowledge is used for techniques based on clipping planes or transparencies [2, 4, 5]. Furthermore, the techniques are either suitable for pure outdoor scenes [5] as they do not offer techniques to show insides, or for pure indoor scenes like multi-story buildings [2, 4] as they are solely based on room or floor decompositions. Finally, the presented scene extract contains only the user's current surrounding, hampering the established WIM's use as a global lookup.

To address this issues, we present a WIM generation workflow based on a fully automatic scene decomposition for mixed in- and outdoor scenes with interaction methods to manually adapt the presented extract.

*{boensch|freitag|kuhlen}@vr.rwth-aachen.de

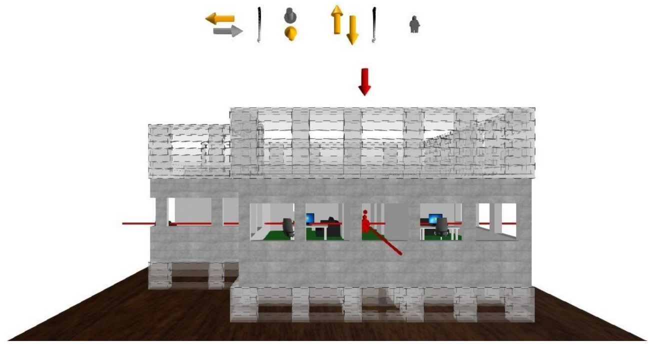


Figure 1: WIM with user avatar (red, inside building) and location hints (red lines and pointer) as well as 3D user interface (top, yellow and grey) for manipulating the presented scene extract.

2 Automatic 2-step WIM generation workflow

Our WIM shall be used as an orientation and information gathering technique. Therefore, our workflow provides a meaningful extract of the scene enriched by abstract context information and a 3D user interface (UI) allowing extract manipulations (see Figure 1). As shown in Figure 2, the IVE is first decomposed into logical units (LUs) in a preprocessing step which also generates the abstract context. In the second step, performed continuously during runtime, a meaningful set of LUs is selected as the current scene extract and context excerpts are chosen accordingly.

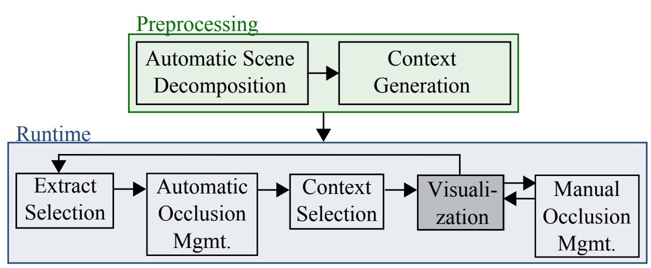


Figure 2: Overview of our automatic 2-step WIM generation workflow.

2.1 Preprocessing

The preprocessing step automatically decomposes the IVE into self-contained LUs, which—taken individually—are meaningful, e.g., complete rooms or complete floors. A set of adjacent LUs is then selected as the current scene extract for the WIM. Established approaches for identifying LUs in any 3D architectural model are based on distance fields, resulting in a cell and portal graph (CPG) representing the original scene source. A CPG can be computed in different ways, but most of them have specific limitations, e.g., the need of axis-aligned scenes or rooms without furniture as stated in [1, 4]. They are thus not suitable for a generally valid workflow. One applicable approach for indoor scenarios is the scene decomposition proposed by Andújar et al. [1], already applied successfully in [4]. As we focus on mixed in- and outdoor scenes, we extend the approach resulting in a decomposition in seven successive steps:

1. A *Free Space Map* is generated as an approximation of the floor plan and the furniture. Therefor, the IVE is divided into a regular, binary 3D grid. Each voxel indicates, whether or not it intersects with the scene geometry.
2. An unsigned *Distance Map* is generated by applying a 5×5×5 quasi-Euclidean chamfer matrix, storing the closest point distance to scene geometry per voxel. Here, the approach of [1] is extended from 2D to 3D to adapt it to outdoor scenes.
3. As a consequence, an *Optimized Neighborhood Map* has to be generated. It stores the closest distance to scene geometry regarding all six neighboring directions.
4. The *Cluster Map* is calculated in a parallel region-growing process, starting at the voxels with the maximal distances. It represents a first approximation of the LUs.
5. *Cluster Merging* is done next, based on a straightforward size heuristic in order to get rid of LUs caused by geometric noise.
6. The LUs are transformed into a more easily manageable *Polygonal Description*, since a description of the LUs based on the grid is too complex to handle.
7. A final *Polygonal Merging* reduces the amount of clusters. This leads to a scene decomposition into meaningful LUs.

In addition to a meaningful scene extract based on the LUs, the WIM needs to present enough context for users to orient themselves or to distinguish structurally similar regions. To this end, we defined an *AddOn*, showing an abstract, voxel-based space division in the form of an approximated 3D floor plan beneath and above the the scene extract, e.g., the adjacent floors in a multi-story building. For this *Context Generation*, the final set of polygonal LUs is retransformed into the voxel-based description and added to the *Free Space Map*. That way, all scene objects except for structural elements like walls, floors or ceilings, are eliminated, resulting in an approximation of the 3D floor plan. The voxels that still contain scene geometry are then rendered as semi-transparent boxes with dashed lines, emphasizing their approximative character (see Figure 1). Since too many boxes hinder a quick understanding of the content shown, the boxes are combined to larger groups beforehand.

2.2 Behavior at Runtime

At runtime, *Extract Selection* (see Figure 2) chooses a set of adjacent LUs. Per default this set represents a small extract of the user's current surrounding. Heuristics are used to extend the extract size for outdoor areas. Additionally, the extract can be manually adapted by means of a 3D UI (see Figure 1). This allows the user to manipulate the size of the context's footprint (a larger set of adjacent LUs is selected for the WIM) as well as the exact extract shown (set of LUs is not centered around users position as in the default case). Consequently, the WIM can be used as a lookup and is improved for orientation support. Before presenting the chosen extract, *Automatic Occlusion Management* is used to reveal its inside by removing ceilings and to increase its understandability by including the next available floor. Additionally, *Context Selection* determines a suitable *AddOn* for scene areas above or below the scene extract.

For the final visualization (see Figure 1), a user avatar is added. Two 3D lines, protruding from the model horizontally in four axis-aligned directions and a 3D arrow indicating the position in vertical direction, are rendered as location hints. The entire setup is centered on a table top floating in front of the user at hand height. As walls and scene objects are not excluded in the extract, a *Manual Occlusion Management* finally allows to look inside the WIM.

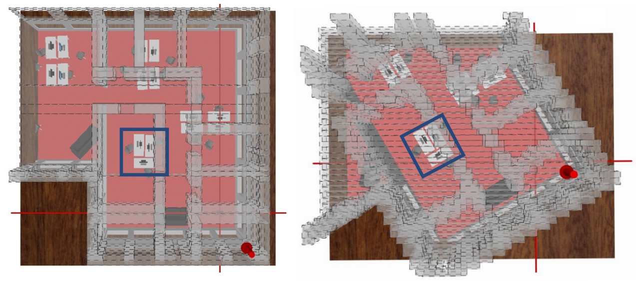
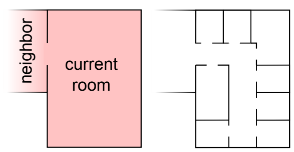


Figure 3: WIMs for identical extracts (left: axis-aligned; right: non-axis-aligned) with rectangles highlighting a set of reference objects.

3 Preliminary Results

In order to evaluate our approach, we tested it for several synthetic and real data sets. We found that, in general, the scene decomposition works well and the 3D UI supports smooth adaptions of the presented extract. However, a few restrictions were identified: (1) the LU selection and voxel-grouping for the *AddOn* are not yet optimized for non-axis-aligned scenes. Figure 3 shows the WIMs for an identical location inside a building. For clarification, the scene extract and *AddOn* floorplan are given on the left. In the axis-aligned case (Figure 3, left), the WIM shows the complete current room and a meaningful extract of the neighboring one. The room extracts presented for the non-axis-aligned case (right) are neither complete (current room) nor meaningful (neighboring room). Besides, the dashed lines of the *AddOn* impair a free view onto the inside. (2) Large, free-standing furniture is sometimes interpreted as walls, resulting in misleading *AddOn* information.



Despite these issues, our preliminary results indicate that the workflow is appropriate for the task of automatically generating meaningful WIMs for realistic architectural IVEs.

4 Conclusion

We have presented an automatic WIM generation workflow for large-scale information-rich architectural IVEs, enriched by an 3D UI to manually manipulate the scene extract presented in the WIM.

In the future, we plan to perform a quantitative user study to evaluate whether our WIM effectively supports orientation and wayfinding and whether the WIM's 3D UI is considered to be natural and easy to learn.

References

[1] C. Andújar, P. Vázquez, and M. Fairen. Way-Finder: Guided Tours Through Complex Walkthrough Models. *Comput. Graph. Forum*, 23(3):499–508, 2004.

[2] L. Chittaro, V. K. Gatla, and S. Venkataraman. The Interactive 3D BreakAway Map: A Navigation and Examination Aid for Multi-Floor 3D Worlds. In *Proc. Int. Conf. CyberWorld*, pp. 59 – 66, 2005.

[3] R. Stoakley, M. J. Conway, and R. Pausch. Virtual Reality on a WIM: Interactive Worlds In Miniature. In *CHI: Proc. SIGCHI Conf. on Human Factors in Comput. Syst.*, pp. 265–272, 1995.

[4] R. Trueba, C. Andújar, and F. Argelaguet. Multi-Scale Manipulation in Indoor Scenes with the World in Miniature Metaphor. In *Joint Virtual Reality Conf. of EGVE - ICAT - EuroVR*, pp. 93–100, 2009.

[5] C. E. Wingrave, Y. Haciahmetoglu, and D. A. Bowman. Overcoming World in Miniature Limitations by a Scaled and Scrolling WIM. In *Proc. IEEE Symp. 3D User Interfaces*, pp. 11 – 16, 2006.