Semantic-driven Visualization Techniques for Interactive Exploration of 3D Indoor Models

Alessandro Florio
University of Trento,
now at
TOL GmbH,
Bolzano, Italy
alessandro.florio@tol.info

Matthias Trapp
Hasso Plattner Institute,
Faculty of Digital Engineering,
University of Potsdam
Germany
matthias.trapp@hpi.de

Jürgen Döllner
Hasso Plattner Institute,
Faculaty of Digital Engineering,
University of Potsdam
Germany
juergen.doellner@hpi.de

***Abstract*—The availability of detailed virtual 3D building models including representations of indoor elements, allows for a wide number of applications requiring effective exploration and navigation functionality. Depending on the application context, users should be enabled to focus on specific Objects-of-Interests (OOIs) or important building elements. This requires approaches to filtering building parts as well as techniques to visualize important building objects and their relations. For it, this paper explores the application and combination of interactive rendering techniques as well as their semantically-driven configuration in the context of 3D indoor models.**

***Keywords*-Building Information Models, BIM, Industry Foundation Classes, IFC, Interactive Visualization, Real-time Rendering**

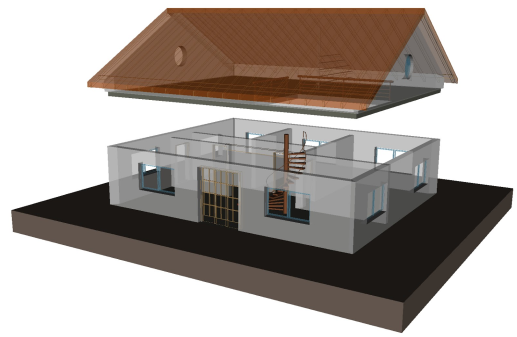


Figure 1: Combination of ghosted and exploded view visualization techniques: all wall surfaces are rendered semi-transparent, while the furniture is rendered opaque; the roof elements are displaced vertically.

I. Introduction

A. Motivation

Building Information Models (BIMs) are intended to include and provide information for all the stake holders that are involved during a buildings life, e.g., planning, construction, maintenance. Thus, BIM models are often detailed, geometrical complex, include a vast amount of different information, and form a rich data source for 3D indoor models as well as systems and applications based on these.

However, 3D indoor models are used mainly in applications based on virtual 3D city models, and thus currently are considered with a low Level-of-Detail (LOD), making only use of their external shape and their position in space leaving most of the information unexploited. Despite the increasing quality and availability of BIMs, most of current interactive viewers are limited with respect to their visualization capabilities. In particular this concerns the exploration BIMs by users that want to focus their attention only on specific parts of an indoor model, but that at the same time want to be able to maintain a spatial reference of these elements.

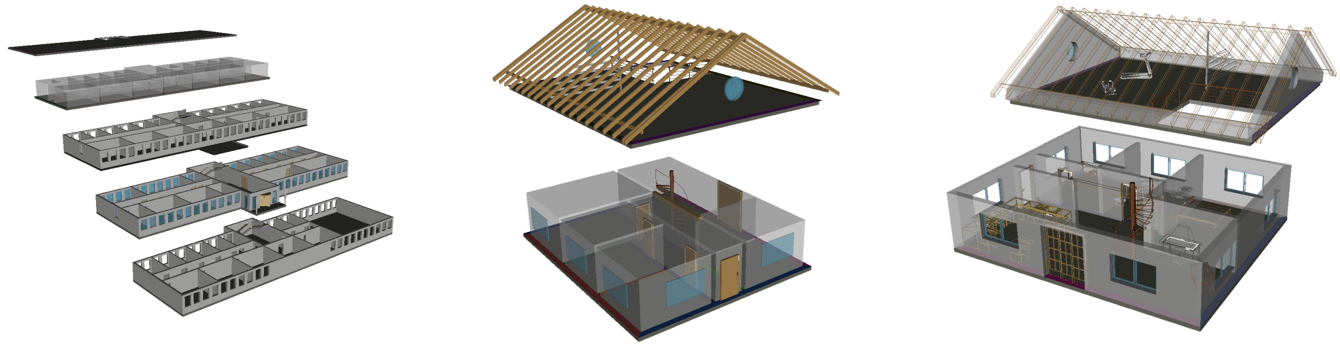
B. Challenges for Visualization of 3D Indoor Models

Effective indoor visualization is a crucial component building-related information systems, applications, and processes. For it, full interactivity is required to provide the full capabilities of information rich indoor models to the users hands. More specifically, special techniques are required to allow users to gather exactly the information that they need in their actual situation or task. This yield the questions of how to enable effective visualization for highly-detailed 3D indoor models, based, e.g., on BIM models.

Exploiting a 3D indoor building model in this terms means handling many elements that are visualized together, when in most applications only few of them, or the relations between them, are relevant for the use-case. Therefore, visualization techniques are required to filter single unimportant elements or an entire set of elements based on their semantics, as well as to set the focus to the desired ones, rendering the remaining parts just as context in order to keep the spatial information to relate all the data. Moreover, focus should be given interactively depending on what the user is most interested in, providing a flexible approach that suits various possible requirements.

C. Applications of 3D Indoor Visualization

Besides map providers that start to including indoor models into 2D maps, BIM are increasingly used by modern Facility Management (FM), which is an interdisciplinary field deal-



(a) (b) (c)

Figure 2: Exemplary results achieved by semantic-driven combination of exploded, cut-away, and ghosting views of automatically generated indoor LOD to reveal inner building structures.

ing with the coordination of space, infrastructure, people, and organizations. In particular, it manages safety, security, maintenance, cleaning, efficiency, and space management of a facility. With respect to this, visualization of virtual 3D indoor models can support these task, e.g., by removing uninteresting elements using cut-aways, by using exploded views to relate floor plans and obtaining a complete overview of all the Region-of-Interest (ROI) simultaneously, featuring ghosted views to retain the context of the 3D scene.

For instance, if a building has to be refurnished, an interior designer could load a model of the building, cut-away the existing furnishment, explode floors, and by considering the placement of windows, doors, beams, heating and eventually electrical as well as water system, he/she could plan to arrange new pieces of furniture or how to relocate the already existing furnishing. Further, disaster management concerns the effort of communities or businesses to plan for and coordinate personnel and materials required to either mitigate the effects of, or recover from, natural or man-made disaster. With respect to this use case, it is important to obtain a 3D representation of the respective building(s) in order to support rescue teams in understand the facility structure, e.g., to highlight possible entrances or exits.

D. Approach & Contributions

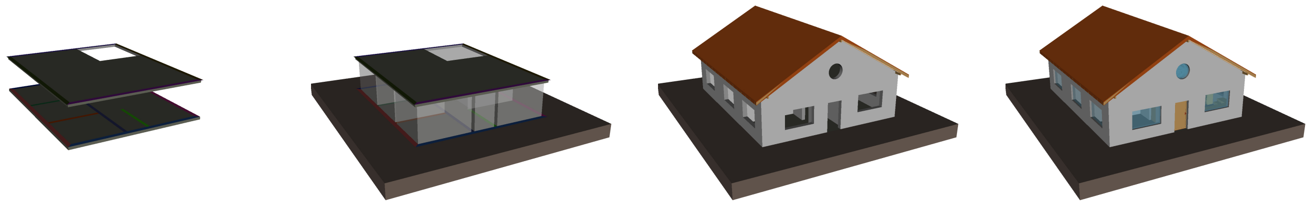
Concerning the challenges and possible applications of 3D indoor visualization, this paper presents a concept and prototypical implementation for combining state-of-the-art interactive visualization techniques (e.g., cut-away- ghosted and explosion views) to highlight use-specific, important features of 3D indoor models using their represented object semantics (Figures 1 and 2). Therefore, a configurable mapping between suitable visualization and rendering techniques allows for rapid building exploration by enabling interactively dissemination of available building model information, not only their external shape and appearance. In addition thereto, it automatically computes LOD representations in a preprocessing step and integrate these into the visualization process. The remainder of this paper is structured as follows. Section II reviews related work with respect to indoor model representations and interactive visualization techniques suitable for 3D indoor models. Section III describes how LOD variants can combined with different interactive techniques for 3D indoor model visualization. Section IV discusses the runtime performance of a prototypical implementation of the proposed concept. Finally, Section V concludes this paper.

II. Related Work

A. Indoor Model Representations & Indoor Navigation

Most of the recent works on 3D indoor models do not use the semantics and detailed information for the inspection of a construction, but focus on the applications to indoor navigation systems. For instance, Li & He show how to derive from an building indoor model the blueprints and, once rooms, doors, and stairs are identified by means of the semantics, a graph derivation phase occurs, in which the 3D indoor network is produced [1]. When routing information is computed, navigation techniques are applied within the original virtual 3D model.

The main field of application in the existing work is emergency situation, where a navigation system could be exploited. Even if focusing more on the management of rescue teams, Rüppel *et al*. [2] base their study on BIMs and on their semantic information, highlighting the importance of having an accurate model. The need of detailed 3D model in these emergency cases is stressed by [3] who state that 2D blueprints are not sufficient to find accurate paths in situations where no mistake is allowed. Different types of data models are instead evaluated by [4] against requirements specified for the implementation of a sufficient indoor navigation system, with the result that City Geography Markup Language (GML) (CityGML) and in particular IFC are versatile data models that aim at spatio-semantic coherent models but also allow the representation of 3D models at various degrees of geometric and semantic complexity. To conclude this section, [5] propose a hybrid spatial



(a) LOD-1 (b) LOD-2 (c) LOD-3 (d) LOD-4

Figure 3: Examples of different level-of-detail representations that are automatically derived based on the semantic information encoded within an Industry Foundation Class (IFC) data set.

model for indoor environments that hierarchically structures spaces in a building to have an easier understanding of the distribution of elements in space and to better support wayfinding algorithms.

B. Visualization Techniques for 3D Indoor Models

There are three major categories of visualization techniques that are applicable for 3D indoor models: *cut-away*, *ghosted*, and *exploded* views; which are briefly described as follows.

1) Cut-away Views: A detailed work on cut-away views is presented in [6]. It shows how removing outside-parts of a 3D scene, or use transparent rendering to reveal occluded parts, which in several cases is more important. This approach finds an application for example in anatomy: to analyze the interior of the human body many layers need to be removed in order to inspect a 3D model. Another approach to cut-away views is the one presented by Lidal *et al.* : the 3D scene is cut by means of clipping planes [7] or simple clipping solids [8]. It it applied to geological illustrations for underground area examination. View-dependent cut-away views are presented by Burns *et al.* , which are based on clipping surfaces that cut-off the occluding elements according to the shape of the OOIs to retain the maximum amount of context possible [9].

2) Ghosted Views: Occluded OOIs can be made visible, as [10] stated, by rendering the occluding surfaces semi-transparent. The portion of occluding elements that has to be rendered in such a way is computed according to the position of the objects that should be visible, so that only the necessary parts are rendered with ghosted views.

3) Exploded Views: Li *et al.* shown, how exploded views are crucial for explaining the internal structure of complicated objects [11] . They provide a framework for creating and viewing interactive exploded view diagrams using static images of arbitrary objects as input. Following to that, Li *et al.* present a system for creating and viewing interactive exploded views of complex 3D models [12]. Such a model is automatically structured into an explosion graph that encodes how parts are displaced with respect to each other. On the other hand, Bruckner *et al.* use exploded views in anatomy to separate the various concentric layers of the human body to have all the details visible simultaneously but without losing much spatial information by allowing to relate one layer to the neighboring ones [13]. Finally, Tatzgern *et al.* show a system that automatically displaces only a subset of the parts of the 3D model to reduce the complexity in explosion diagrams, which otherwise may suffer from clutter caused by the excess of displaced parts [14] .

There are few works which combine advanced visualization techniques for 3D indoor models. Niederauer *et al.* apply exploded views to 3D models such as multi-floor buildings to enable viewers simultaneously depict internal and external structures [15]. However, in the analyzed cases information about floors is not provided, but has to be computed for every model. In fact, in a later work, Houston *et al.* highlight the need of higher-level semantic knowledge of architectural environments that facilitates the development of exploded-view visualization techniques [16]. Another visualization technique uses LODs to suit the different requirements that can be encountered in different application scenarios of 3D building models [17]. Four LODs are defined that represent a building model at different abstraction levels. In this work, LODs are computed based on the semantic information of the 3D indoor model.

III. Interactive Visualization of Indoor Models

Building parts and structures are information rich (in terms of multiple attributes) and can be considered as instance as multi-dimensional data. This section describes how to automatically derived LOD representations for IFC indoor models (Section III-A) and their semantics can used be to parametrize and control interactive visualization techniques (Section III-B), which can be *combined* to facilitate communication of spatial relations.

A. Automatic Level-of-Detail Generation of Indoor Models

A building can be visualized using LODs according to how much detail the user is interested in w.r.t. a certain use-case. Our approach uses four LODs, each one defined as a set of visible IFC elements. Advancing a lower to a higher LOD generally implies the addition of new or detailing of existing elements to the 3D scene. However, the only exceptions are *room volumes* and *wall footprints*, which are removed from LOD-2 to LOD-3 because of substitution by

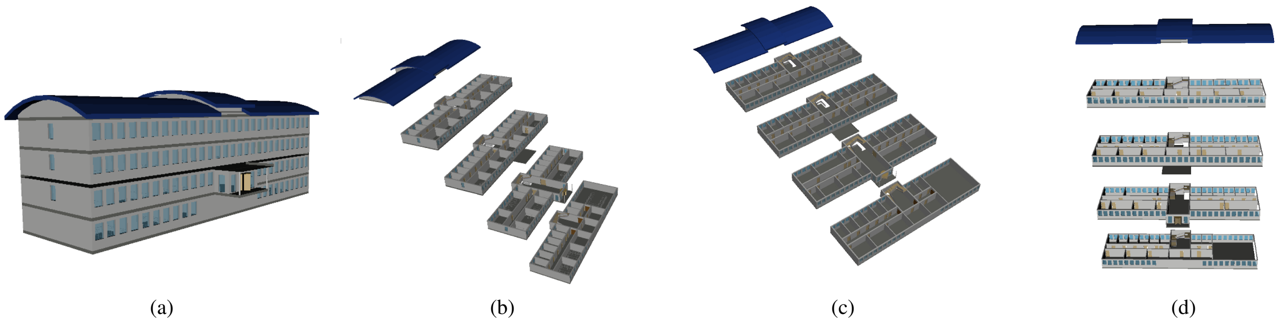


Figure 4: Exploded-view visualizations of a multi-floor building (a) using different LODs and viewing angles (b)-(d).

more accurate elements, e.g., the walls, introduced in LOD-3. The four different LODs supported can be distinguished as follows (cf. Figure 4):

LOD-1: provides basic information about the structure of the building, i.e., only the floors and wall footprints;

LOD-2: adds the site where the building is built and the room volumes to LOD-1.

LOD-3: removes rooms volumes and wall footprints, and adds wall, structural, and roof geometry.

LOD-4: adds remaining elements (doors, windows, furnishment), but excludes the ones removed previously.

LODs can be a feasible alternative to using multiple cut-away views or they can be used as a shortcut to reach a configuration close to one of the levels. For example, instead of focusing on particular objects in a ROI, a user could be interested in an entire building floor while retaining the others as context. It should then be possible to select on which floor to focus, and render it using LOD-4 settings. Subsequently, the upper and lower floors can be rendered with LOD-3, their respective upper and lower floors with LOD-2 and the remaining ones with LOD-1; this way, a gradual filtering can be achieved (Figure 2a).

B. Interactive Visualization Techniques for Indoor Models

1) Cut-away Views: Cut-away views allow for removal of scene elements to simplify depiction and to hide unnecessary objects that might distract the attention from OOI. They are usually implemented at element type level, i.e., the user can hide a complete set of elements according to their semantics specified in the IFC model. For example, if a user is interested in building is furnishment, all occluding walls or roofs can be hidden by the visualization.

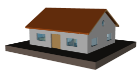
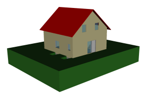
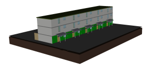
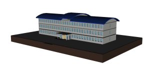
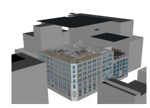
2) Ghosted Views: Removing building parts using cut-away views may result into losing information about the context in which the remaining elements are placed. Simplifing an indoor model depiction but keeping the context simultaneously requires a compromise: *ghosted views*. Instead of removing elements, they are presented as so-called ghosts, i.e., they are rendered using transparency of wire-framing. The usage of transparency retaines structural aspects, but allows for partial depiction of obstructing elements, while wireframe rendering show only the polygon contours of these elements. This basically preserves structural aspects, but allowing the possibly obstructed elements to be almost completely visible.

3) Exploded Views: In our prototypical implementation, every semantic element can be assigned to one of the two visualization techniques described above. In general, semi-transparency is applied to elements that have large planar surfaces such as walls, roofs, and floors, since wireframe rendering often cannot preserve their structural aspects having insufficient vertex density.

However, wireframing is suitable to elements representing small objects of more complex shape. The individual techniques described previously may not be sufficient to visualized all OOIs. In fact, if elements on every floor of the building are required to be visualized, numerous wall and floor elements may be affected and contextual information can be lost easily. To counterbalance this, exploded views can be used to set an individual offset vector to every floor (also defined in the IFC semantics as a set of elements) relative to the previous one. Thus, these floors are displaced in space, similar to an explosion at the center of the building in the direction specified by that vector. Choosing a suitable offset vector, floors do not overlap each other, and thus it is possible to reveal containment relationships (Figure 4).

4) Combination of Viewing Techniques: The individual techniques described above can be combined to provide a custom visualizations and with various of possibilities for configurations, depending on the use-case. For example, Figure 1 shows a 3D building model rendered at LOD-4, from which only furniture has been cut away. Walls and the roof are depicted semi-transparent: in particular the beams and the railing of the upper floor can perceived easily; windows and doors are rendered using wireframing, giving a hint of their presence while not occluding other parts of the building. Floors are depicted using exploded views to expose the spiral staircase, which would be occluded otherwise.

Table I: Performance parameters of different BIM datasets.

IFC Model	House	Jasmin Sun	Smiley	Office	210 King
Preview					
#Lines	82 237	98 515	62 678	49 675	1 845 439
IFC Elements	117	92	483	782	14 882
Drawables	17 201	24 148	47 134	15 932	1 504 707
Vertices	77 523	96 261	220 153	74 367	7 629 873
Memory	110 MB	145 MB	64 MB	55 MB	2 GB
Parsing	12 s	13 s	9 s	8 s	26 min
Optimization	7 s	7 s	12 s	5 s	57 min
Total Loading	19 s	20 s	21 s	13 s	83 min
FPS (LOD4)	8	9	3	3	0.12
FPS (LOD3)	17	24	6	10	0.47

C. Preprocessing of Index Data Structures

Prior to rendering, the IFC model is preprocessed and converted into data structures (geometry and appearance) supported by the rendering system. Our implementation uses OpenSceneGraph (OSG). As described previously, IFC elements are organized in hierarchically structure that supports different LOD. The most important is the *object level* that defines common objects such as doors, walls, and furniture. IFC models are parsed on a line-by-line basis: each line corresponds to an object at a certain LOD or to a relationship between objects. During parsing, every element is stored in main Random Access Memory (RAM) to be available if an element higher-level is defined by it, in order to link them, to create a hierarchy, or to construct a new object given its basic properties.

Since the described visualization techniques are applied at a per-object level, their references are stored using two additional data structures (Figure 5) in order to gain a fast access for configuration changes at run-time. For it, in addition to inserting an IFC element into the scene graph, it is added into separate indexing table. Also floors, if present, are saved separately to facilitate the configuration of explosion-views. The resulting scene graph is handled by the viewer for real-time rendering, while the index data structures are used to provide efficient access for the respective visualization techniques.

IV. Performance Evaluation

This section describes the results of a performance evaluation for a prototypical implementation of the concepts described previously using different test data sets. Table I (Line 2 to 6) shows the complexity of five different IFC test datasets used for the performance evaluation. The following system was used for performance evaluation: 3.1 GHz Intel Core i5-2400 processor with 8 GB RAM, running Windows 7 Enterprise 64 bit and mounting an NVIDIA GeForce GT 630 graphic card with 2 GB of Video RAM (VRAM). We

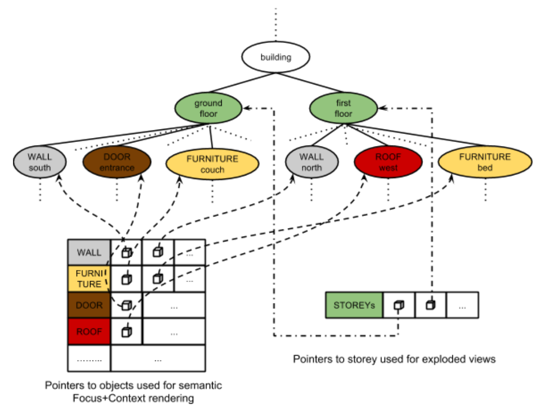


Figure 5: Index data structures for storing associations between indoor building elements instances used for run-time configuration of visualization techniques.

evaluated the loading and processing performance (Line 7 to 9) as well as the run-time performance for rendering (Line 10 to 11).

A first parameter that has to be considered when evaluating usability is the loading time, in particular the two phases: (1) *parsing* the IFC model (2) *optimizing* the scene graph once it has been created from the model (performed by OSG automatically). Parsing timings depends primarily on the complexity of the 3D indoor model. For the first four datasets of low to medium complexity, the average parsing speed is about 7000 lines-per-second. However, IFC models can contain a number of forward references, i.e., elements depending on another that has not been defined already during file parsing. On these occurrences, the respective line is queued with all their ancestors result to be incomplete. These will then be re-parsed when the complete document has been read. Thus, the procedure may be repeated several times, because in the worst case one depth level at at a time can be updated. In fact, the last dataset has a significant lower parsing speed due to high amount of forward references.

The optimization time depends on how the scene is structured in the model representation: many similar elements share the same geometries and result in fewer nodes in the scene graph. Thus, the optimization operation performs faster. However, if geometries are copied instead of being referenced, then the scene graph can become more complex, e.g., in the case of the Smiley West dataset the optimization phase takes longer than the parsing one.

With respect to rendering speed, the measurements show two datasets rendered at approx. 3 Frames-per-Second (FPS) in LOD-4, while the 210 King model generates a new frame only every 8 to 9 seconds. The rendering performance basically depends on the number of individual drawables represented in the scene graph, which in turn depends on how geometries are organized: datasets that share nodes at the highest level possible yield only small numbers of drawables. However, switching to LOD-3 in the worst case doubles the frame rate, mostly because LOD-3 does not include highly detailed furniture that usually has complex geometry.

V. Conclusions

This paper presents an approach to semantic-driven visualization of 3D indoor models. It discusses how combinations of interactive visualization techniques such as cut-away views, ghosted views, and exploded views can be used to support exploration of potentials complex 3D buildings by exploiting the object or element semantics, e.g., provided by IFC model representations. Application examples demonstrate how a user's focus-of-attention can be guided to instances of specific element classes by counterbalancing potential occlusions.

Acknowledgments

We would like to thank the anonymous reviewers for their valuable feedback and the University of Trento (Italy). This work was funded by the Federal Ministry of Education and Research (BMBF), Germany, within the InnoProfile Transfer research group "4DnDVis"

References

[1] Y. Li and Z. He, "3D Indoor Navigation: a Framework of Combining BIM with 3D GIS," in *44th ISOCARP congress*, 2008, pp. 1–10.

[2] U. Rüppel and K. M. Stuebbe, "BIM-based Indoor-emergency-navigation-system for Complex Buildings," *Tsinghua Science & Technology*, vol. 13, pp. 362–367, 2008.

[3] U. Isikdag, S. Zlatanova, and J. Underwood, "A BIM-Oriented Model for Supporting Indoor Navigation Requirements," *Computers, Environment and Urban Systems*, vol. 41, pp. 112–123, 2013.

[4] G. Brown, C. Nagel, S. Zlatanova, and T. H. Kolbe, "Modelling 3D Topographic Space against Indoor Navigation Requirements," in *Progress and New Trends in 3D Geoinformation Sciences*. Springer, 2013, pp. 1–22.

[5] B. Lorenz, H. J. Ohlbach, and E.-P. Stoffel, "A Hybrid Spatial Model for Representing Indoor Environments," in *Proceedings of the 6th International Conference on Web and Wireless Geographical Information Systems*, ser. W2GIS'06. Berlin, Heidelberg: Springer-Verlag, 2006, pp. 102–112.

[6] W. Li, L. Ritter, M. Agrawala, B. Curless, and D. Salesin, "Interactive Cutaway Illustrations of Complex 3D Models," *ACM Trans. Graph.*, vol. 26, no. 3, Jul. 2007. [Online]. Available: http://doi.acm.org/10.1145/1276377.1276416

[7] M. Trapp and J. Döllner, "2.5D Clip-Surfaces for Technical Visualization," *Journal of WSCG*, vol. 21, pp. 89–96, 2013.

[8] E. M. Lidal, H. Hauser, and I. Viola, "Design Principles for Cutaway Visualization of Geological Models," in *Proceedings of the 28th Spring Conference on Computer Graphics*. ACM, 2012, pp. 47–54.

[9] M. Burns and A. Finkelstein, "Adaptive Cutaways for Comprehensible Rendering of Polygonal Scenes," *ACM Transactions on Graphics (Proc. SIGGRAPH ASIA)*, vol. 27, no. 5, pp. 124:1–124:9, Dec. 2008.

[10] N. Elmqvist, U. Assarsson, and P. Tsigas, "Employing Dynamic Transparency for 3D Occlusion Management: Design Issues and Evaluation," in *Human-Computer Interaction–INTERACT 2007*. Springer, 2007, pp. 532–545.

[11] W. Li, M. Agrawala, and D. Salesin, "Interactive Image-based Exploded View Diagrams," in *Proceedings of the 2004 Graphics Interface Conference*, ser. GI '04. School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada: Canadian Human-Computer Communications Society, 2004, pp. 203–212.

[12] W. Li, M. Agrawala, B. Curless, and D. Salesin, "Automated Generation of Interactive 3D Exploded View Diagrams," in *ACM SIGGRAPH 2008 Papers*, ser. SIGGRAPH '08. New York, NY, USA: ACM, 2008, pp. 101:1–101:7.

[13] S. Bruckner and M. E. Gröller, "Exploded Views for Volume Data," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 5, pp. 1077–1084, 9 2006.

[14] M. Tatzgern, D. Kalkofen, and D. Schmalstieg, "Compact Explosion Diagrams," in *Proceedings of the 8th International Symposium on Non-Photorealistic Animation and Rendering*. ACM, 2010, pp. 17–26.

[15] C. Niederauer, M. Houston, M. Agrawala, and G. Humphreys, "Non-invasive Interactive Visualization of Dynamic Architectural Environments," in *Proceedings of the 2003 Symposium on Interactive 3D Graphics*, ser. I3D '03. New York, NY, USA: ACM, 2003, pp. 55–58.

[16] M. Houston, C. Niederauer, M. Agrawala, and G. Humphreys, "Visualizing Dynamic Architectural Environments," *Commun. ACM*, vol. 47, no. 8, pp. 54–59, Aug. 2004.

[17] B. Hagedorn, M. Trapp, T. Glander, and J. Döllner, "Towards an Indoor Level-of-Detail Model for Route Visualization," in *Proceedings of the 2009 Tenth International Conference on Mobile Data Management: Systems, Services and Middleware*, ser. MDM '09. Washington, DC, USA: IEEE Computer Society, May 2009, pp. 692–697.