A Design Space for Linked 2D and 3D Visual Representations

Ebrar A. D. Santos* Jiayi Hong† Tobias Isenberg†
Université Paris-Saclay, CNRS, Inria, LISN, France

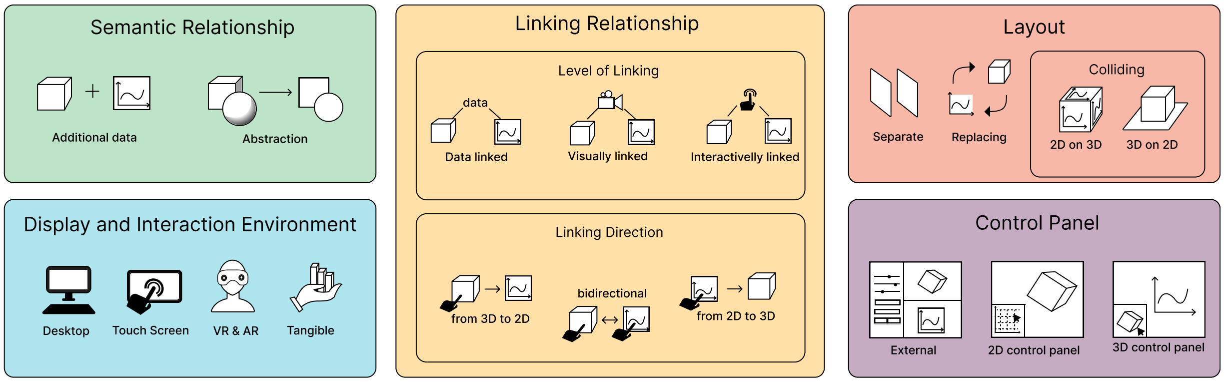

Figure 1: The design space for combining 3D representations with 2D abstract representations.

Abstract

We discuss a design space for combining, linking, and jointly presenting 3D spatial data together with related abstract data, the latter typically mapped to 2D space. Even though guidelines exist for creating linked 2D views that help designers and inspire researchers, they typically do not encompass 3D spatial data representations. We thus first reviewed the existing visualization literature and explored linking patterns in existing systems. We then extracted the dimensions of how these tools combine 2D and 3D data by their semantic relationships, display environment, placement, and the linking methods between them.

1 Introduction and Background

3D representations can depict information of real spatial phenomena such as 3D positions and shapes, while 2D representations can provide additional detailed abstract data. The latter often cannot easily be represented within the main spatial 3D representation, thus often separate representations are used. There are many system interfaces combining 3D representations with 2D abstract data and there are endless ways to layout the visuals. We thus studied the literature and summarize the systematic design choices to allow practitioners and researchers effectively represent and link both types of representations in an understandable and intuitive way.

Concepts and theories behind multiple linked views and techniques for doing so have previously been discussed [6, 9] and researchers have also investigated relations between 2D and 3D representations in VR [4]. The literature lacks, however, a general understanding of how to combine 2D and 3D representations in non-immersive environments (e. g., desktop and touchscreen interfaces). We thus specifically focus on how 3D and 2D representations can be combined. Within the timeframe of 2012–2022, we surveyed all papers from IEEE Vis, EuroVis, and TVCG and extracted 97 relevant papers. We then derived a design space for such systems based on our analysis. We organized the design considerations into four aspects: why interfaces contain both 2D and 3D representations, how they are displayed within a system, how they are linked to each other, and how users can control the system.

*e-mail:adaebrar@gmail.com
†e-mail: {jiayi.hong | tobias.isenberg}@inria.fr

2 Design Space

When surveying the literature, we excluded papers that introduce novel rendering methods of the 3D data or systems without any 2D graphs. We consider 2D abstract representations as visualizations on a planar space, we thus excluded interfaces with 3D data accompanied only by text, text fields, sliders, and buttons as well. We also excluded papers about evaluating these tools or introducing guidelines for visualizations. We thus were left with only work on interfaces with both 2D and 3D representations. Considering that many systems we reviewed include not only one 2D and one 3D window, but multiple 2D and multiple 3D windows which all linked to each other in unique ways, our foremost attempt was to capture the relationship between the 2D and 3D visualizations. When creating our design space, we started with the explicit aspects of visualizations (e. g., the devices on the visualizations are displayed and the relative layout of 2D and 3D representations). Then we went deeper into the connection methods between the 2D and 3D visualizations. We also observed the relative differences between these two parts in interaction and their inner correlation in semantics. We summarized all the considerations into the following five aspects.

2.1 Semantic Relationship

Here we mainly focus on the reasons for combining 2D and 3D representations and the added value they have for each other. For example, a second representation with a different dimension could add more information or emphasize aspects of the data.

Additional Data: One scenario where an interface combines 2D and 3D representations is that the data can be multivariate and it has additional aspects and dimensions that need to be shown as well. In this case, 2D graphs are generated as a way to show additional and inherent aspects of the 3D data, such as the temporal data behind the 3D models. 2D representations can also function as records for user history or 3D data can present additional aspects. The designers can use, e. g., stacked 3D views with temporal information [7] .

Abstraction: Its purpose is to emphasize some aspects of the 3D spatial data. Designers can use techniques such as flattening, cross-section, node-link diagrams, or other dimension reduction methods to express spatial data in the 2D space.

2.2 Display and Interaction Environment

Here we look at how a visualization is shown and the type if input.

Desktop: Traditional PC settings with keyboard/mouse input.

Touch Screen: All types of screens which capture touch input, including small tabletop screens and large wall-scale displays.

Tangible: The data is represented by physical objects, such as made from wood, plastic, or glass.

Virtual & Augmented Reality: Immersive environments that typically require specific equipment. Users usually interact with hand gestures or virtual reality controllers.

2.3 Layout

The 2D and 3D representations can be placed differently with respect to each other: separately at the same time, in the same space but at different times, or together simultaneously.

Separate: This is the most common approach to placing 2D and 3D representations. In these designs, both types of representations exist in a separate area, respectively, and do not collide with each other (juxtaposition). The window sizes can be adjusted, or new visualizations can be added to the layout via new windows.

Replacing: Unlike juxtaposition, the 2D and 3D representations can also exist in the same window but not at the same time. They replace each other and the user controls the 'switch' between them. This layout is often preferred in interfaces where 2D visualizations abstract 3D models. Transformation animations can help viewers to keep track of the corresponding data in different dimensions.

Colliding: In this layout, 2D and 3D visualizations share the same space at the same time. We consider two sub-categories within this group depending on whether the visual context is a 3D or a 2D representation. For **2D on 3D** setups the main entity is a 3D representation and 2D abstract data is represented within the 3D space, attached to the 3D data. Not only the 2D data itself but also the location of the 2D data on the 3D model conveys information. Examples of this type of layout are 2D map routes on 3D cities [8] and 2D graphs displayed within the 3D spatial data [1]. In contrast, in **3D on 2D** layouts 3D shapes and models are placed on a 2D graph. We also included cases where 3D models are formed by 2D stacked images on an image. System designers may use 2D graphs as a way to organize, classify, or show relationships between the 3D models. Examples are 3D trajectories on a 2D map [7] and small 3D multiples placed on a 2D graph [10].

2.4 Linking Relationship

The level and the direction of linking between the 2D and 3D representations is essential. First, we look at the **level of linking**:

Data Linked: In this minimum level of linking between 2D and 3D representations, both representations use the same data but users cannot manipulate the views using each other. 2D and 3D views require an external control panel for data manipulation.

Visually Linked: Both views are coordinated visually. Users can rotate or zoom in one view and the system updates or renders the other view accordingly. The system, however, does not support interactive data selection between the windows.

Interactively Linked: These systems facilitate data selection and filtering across views via user interaction, which is increasingly done in recent work. For example, one can highlight data and select objects from either 2D or 3D views [2]. Other than highlighting with hovering or pointer selection, other common data selection methods include lasso selection, brushing, and rectangle selection. In addition, system designers can create their custom tools for interaction [3]. Such diverse interaction methods heavily depend on the chosen visualization environment.

Second, in systems that allow manipulation of the views, the **direction of interaction** can be different:

From 2D to 3D: The user can change views or select data only in the 2D view. The changes made on 2D are rendered on the 3D side.

From 3D to 2D: The user can change views or select data only in the 3D view, and the 2D data is updated based on the changes.

Bidirectional: Users can change views or select data via interacting in either view. The changes are updated in both views.

2.5 Type of Control Panel

External Control Panel: In the majority of the interfaces, the data flow in the 2D and 3D visualizations is controlled by an external panel consisting of text fields, sliders, and buttons. The 2D and 3D representations do not serve as a controller.

2D control panels: Unlike the mainstream approach to control panels, system designers can also choose to design a 2D graph as the control panel. With 3D representations, the systems use 2D representations to control the rendering of the 3D data, which provides an overview to the user. The user thus manipulates the 3D model via interacting with the 2D graph.

3D control panels: In this case, the 3D model acts like a control panel for the information flow in the 2D representation. Compared to a fair amount of 2D graph control panel examples, we saw only one example: a system for visualizing pathology in blood vessels [5]. This interface uses the 3D vessel model to allow users to navigate the generated 2D pictures. The 3D model thus provides an overview and works as a map, whereas the generated 2D images provide detail.

3 Conclusion

We found that many systems that deal with both 3D spatial and abstract data link respective representations in various and innovative ways. With such linked views they help users with extracting information, summarizing, getting an overview, and being aware of the data as a whole. The interactive linking between both representations largely enhances the information delivery and expands hidden data. We thus hypothesize that it increases people's general engagement in a system. We are currently working on creating a web-based overview of our survey, including a detailed code book to summarize the different design choices.

References

[1] A. Bock, A. Pembroke, M. L. Mays, L. Rastaetter, T. Ropinski, and A. Ynnerman. Visual verification of space weather ensemble simulations. In *Proc. SciVis*, pp. 17–24. IEEE, Los Alamitos, 2015. doi: 10.1109/SciVis.2015.7429487

[2] J. Hong, A. Trubuil, and T. Isenberg. LineageD: An interactive visual system for plant cell lineage assignments based on correctable machine learning. *Computer Graphics Forum*, 41(3):195–207, June 2022. doi: 10.1111/cgf.14533

[3] T. Klein, F. Guéniat, L. Pastur, F. Vernier, and T. Isenberg. A design study of direct-touch interaction for exploratory 3D scientific visualization. *Comput Graph Forum*, 31(3):1225–1234, 2012. doi: 10.1111/j.1467-8659.2012.03115.x

[4] B. Lee, M. Cordeil, A. Prouzeau, B. Jenny, and T. Dwyer. A design space for data visualisation transformations between 2D and 3D in mixed-reality environments. In *Proc. CHI*, pp. 25:1–25:14. ACM, New York, 2022. doi: 10.1145/3491102.3501859

[5] G. Mistelbauer, A. Morar, A. Varchola, R. Schernthaner, I. Baclija, A. Köchl, A. Kanitsar, S. Bruckner, and E. Gröller. Vessel visualization using curvicircular feature aggregation. *Comput Graph Forum*, 32(3):231–240, 2013. doi: 10.1111/cgf.12110

[6] J. C. Roberts. Exploratory visualization with multiple linked views. In *Exploring Geovisualization*, pp. 159–180. Elsevier, Amsterdam, 2005. doi: 10.1016/B978-008044531-1/50426-7

[7] C. Tominski, H. Schumann, G. Andrienko, and N. Andrienko. Stacking-based visualization of trajectory attribute data. *IEEE Trans Vis Comput Graph*, 18(12):2565–2574, 2012. doi: 10.1109/TVCG.2012.265

[8] J. Waser, A. Konev, B. Sadransky, Z. Horváth, H. Ribičić, R. Carnecky, P. Kluding, and B. Schindler. Many plans: Multidimensional ensembles for visual decision support in flood management. *Comput Graph Forum*, 33(3):281–290, 2014. doi: 10.1111/cgf.12384

[9] G. Wills. Linked data views. In *Handbook of Data Visualization*, pp. 217–241. Springer, Berlin, 2008. doi: 10.1007/978-3-540-33037-0_10

[10] C. Zhang, M. W. Caan, T. Höllt, E. Eisemann, and A. Vilanova. Overview + detail visualization for ensembles of diffusion tensors. *Comput Graph Forum*, 36(3):121–132, June 2017. doi: 10.1111/cgf.13173