

Spatial Parsing within the Topos 3D Environment

Michael Bang Nielsen & Peter Ørbæk

Dept. of Comp. Sci., University of Aarhus, Aabogade 34, 8200 Aarhus N, Denmark

{bang, poe}@daimi.au.dk

Abstract

Spatial parsers augment spatial hypermedia systems by letting the computer perceive the informal – but visually apparent – groupings formed by humans working with a spatial hypermedia tool. A number of research systems implementing 2D spatial parsing have been described in recent years. This position paper describes an implementation of a tailorable 3D spatial parser for the Topos system: a 3D information organization tool for use on desktops, interactive whiteboards and tables. A number of issues pertaining to 3D parsing as opposed to 2D parsing are discussed.

1 Introduction

Traditional hypertext considers text and multimedia linked by explicit links. Spatial hypertext considers the process of creating such links by mere placement of documents and materials close to each other on a canvas. Spatial hypertext thereby supports informal, ad hoc, organization of electronic materials. Within the last decade the virtues of spatial hypertext for informal information organization have been explored in the Aquanet [8, 9] and VIKI [11] systems, among others.

Spatial parsing augments spatial hypertext by letting the computer automatically recognize hierarchical spatial groupings on the canvas using the placement (created by users) of work materials for group detection. With help from the parser, such groups can be manipulated as a whole and the visual structure may subsequently be made permanent if the user so desires.

This position paper introduces the notion of spatial parsing in 3D and describes the implementation of a 3D spatial parser for the Topos¹ [2, 3, 12] system. Topos is an industrial prototype of a 3D information management system that supports informal groupings of related documents and 3D models. It has been designed for use on desktops as well as for interactive whiteboards and tables. Topos has been designed from the outset to alleviate the burdens of finite 2D screen space via the use of 3D. The 3D interface of Topos provides a very natural and physical interface metaphor [1] by exploiting users' intuition about 3D space.

The paper discusses how to implement and use a spatial parser for 3D spaces. Several questions arise when considering spatial parsing in 3D: Do we parse the space in true 3D or just the 2D projection of the space? Does the parse depend on the user's current viewpoint? Also a number of more technical questions arise, such as how to implement the parser efficiently in the face of true three dimensional objects oriented at arbitrary angles as opposed to the 2D parsers that dealt with groups of axis-aligned rectangles only.

This workshop position paper is a small edited excerpt of a full paper by the authors, submitted for publication elsewhere.

2 Topos and Parsing in 3D

Our 3D parser is implemented as a component inside the Topos system [2, 3, 12]. Topos allows for manipulation and maintenance of relationships among materials in a 3D environment. It integrates with existing applications, supports collaboration among co-workers across the Internet and runs on Windows 2000, SGI IRIX and Linux. Figure 1 shows Topos in use at interactive whiteboards. Figure 2 shows two early screen

¹Topos is a complete re-write of the Manufaktur prototype described in earlier papers.



Figure 1: The Topos tool in use at interactive whiteboards.

shots of a Topos client. It depicts an open workspace (with some of the edges created by the parser highlighted) containing a set of documents and a sub-workspace with more documents. Double clicking any of the documents will launch the document in its appropriate application, and changes to the document will be reflected within Topos in near-real time. The semi-transparent box is a sub-workspace. The pointy cylinder to the left is a “proxy” referring to the sub-workspace. The 3D objects can be moved, sized, rotated etc; light effects may be applied; documents may be made semi-transparent; manipulated as groups and so forth.

Grouping materials in 3D space is familiar to us all from our daily lives: we build heaps of related documents on our desks, we place related materials together on shelves, etc. A main point of the Topos prototype is to mimic and support this grouping of materials in digital form. Topos also supports grouping and organization of materials by means that are not possible in the physical world: whole groups can be moved and placed as a whole, sub-groups can be closed to unclutter the space etc.

Topos supports informal grouping of related documents and models within 3D workspaces. In fact, this is the basic mechanism for information management within Topos. Being three dimensional, the workspaces allow for much more freedom in placement of work materials and allows the users to oversee much larger collections of materials at a time than would be possible in two dimensions. The 3D workspaces also allow users to push groups of materials into the background to be peripherally aware of changes to them while working on something else.

The way Topos is often used, both by its developers and users (architects and landscape architects), is to create loose groups of work materials by placing materials together, and when significant groups emerge, the groups are formalized by creating sub-workspaces for them. The informal and imprecise, “direct-manipulation” way of interacting with the materials within Topos does not lead users to create nicely aligned and finely structured groups, as done in VIKI and other 2D environments. Rather, materials are typically heaped together in rough clusters for later fine-grained organization. One of the uses for Topos and the spatial parser by landscape architects is to sort and organize a set of pictures taken at an initial site visit. Within Topos a large number of pictures can be overseen at once, and the pictures can be grouped easily via the powerful direct manipulation interface. Using the parser whole groups of pictures can be manipulated at once instead of individually.

A spatial parser attempts to recognize structures and groupings from users’ placements of documents, notes, images or models close to each other in space. A good parser should find the same groupings that the user thought of when creating or inspecting the groups. However, different people find different groupings

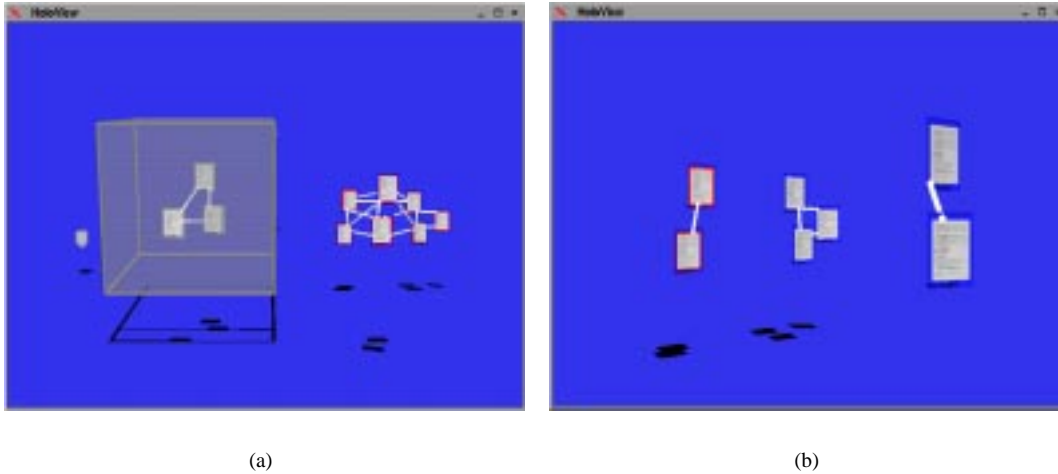


Figure 2: One Topos workspace seen from two different viewpoints with some of the edges created by the parser shown.

in the same geometric constellations of objects. This lead us to consider parsers customized for a particular user, quite in line with [5] that provided user-customization by letting users correct the parser when it found the “wrong” structure.

Groupings in 3D space may look very different from different angles, and different in different contexts: while looked on from afar, a set of objects may seem to belong to a homogeneous group, but when looked on from a closer vantage point, and in isolation, the group may appear divided into smaller sub-groups. Figure 2.a shows an overview of a workspace. From this viewpoint the objects in the right part of the view seem to constitute one homogeneous cluster which the parser recognizes. Figure 2.b on the other hand shows the same objects from another viewpoint closer to the objects, and from this viewpoint the objects seem divided into three disjoint groups (also recognized by the parser).

The above observations lead us to consider parsers that would use the user’s current viewing frustum while parsing, in effect parsing the scene from the user’s viewpoint. One implication of this decision is that a complete parse cannot be re-used as the user moves his viewpoint around in the virtual world. We can, however, re-use some data structures.

Our first, crude, interface to the parser lets the user shift-click on an object which will engage the parser and multi-select the group of objects found around the clicked object. Multiple clicks invokes hierarchical click selection. To cater for the cases where the parser does not automatically select all of the desired group, the user can select additional objects (perhaps using the parser) and add them to the current (multi-)selection.

3 Related Work

Spatial parsing was first introduced by Marshall and Shipman in [10], where an implementation of a spatial parser in Aquanet was described. A more mature spatial parser inside the VIKI system was described in [11]. Other systems have explored different aspects of spatial parsers: CAOS [14] implemented an incremental spatial parser for a generic hypermedia middle-ware layer [13] intended for distributed collaborative work.

Igarashi, Matsuoka and Masui [5] implemented a standalone, adaptive, spatial parser using genetic algorithms to tune the parser to the user’s expectations. Much of the work in this paper is inspired by their approach and extend it to 3D. The primary differences between their parser and ours are that they work with axis-aligned rectangles in 2D whereas we work with objects oriented arbitrarily in 3D; they use a local edge selection strategy whereas ours is global; and they utilize genetic algorithms to tune parameters

whereas we currently rely on user configuration.

The above systems all focus on spatial parsing in a two dimensional canvas. However, it has been repeatedly noted that for large document collections, the limitations of 2D screen space is problematic when organizing large amounts of information [7]. Shipman, Marshall and LeMere [17] added multiple fish-eye views to VIKI to try to overcome the lack of 2D screen space, and also support hierarchical spaces that can be individually closed to unclutter the screen.

Several 3D environments for document and task management have been developed in recent years. The WebBook and the Web Forager [4] used a book metaphor in a 3D environment to help organize and provide access to web pages efficiently and intuitively. Data Mountain [16] is another 3D environment for organizing web pages. Data Mountain explored how to exploit human spatial memory (ie. the ability to remember where you put something) in organizing a large number of web pages and verified that users were able to use this ability in a 3D virtual environment. The Task Gallery [15] is a 3D window manager for organizing tasks (a task being a collection of documents and applications) that exploits human spatial memory to keep track of a large number of tasks at once.

In [6] Light et al. presented a 3D environment called Miramar for organizing documents, relationships and tasks. They focused on the navigational system of Miramar and stated the importance of being able to navigate the 3D space intuitively. The user experiments conducted emphasized both the success of 3D graphics as a means of obtaining intuitive graphical scaling and the ease of use of a 3D environment when provided with an intuitive navigational system.

The above mentioned 3D environments all take advantage of the intuitive graphical scaling and depth cues that the use of 3D graphics provides, and they verify that by using 3D environments the amount of information presented on the screen can be increased without increasing the cognitive load of the users.

4 Discussion

In pure 2D, there is no real concept of a view point. However, a 2D interface with scrollbars, where the user only sees part of the canvas at a time, might benefit from a spatial parser taking the visible area into account. In a zoom-able 2D interface or a 2D interface with multiple fish-eye views a spatial parser might also benefit from taking the current viewpoint, as well as the placements of the fish-eye views into account. However, changing the 2D view point does not affect the apparent clustering of objects as much as changing the view point in a true 3D world.

The decision whether the spatial parser should parse the apparent – subjective – view or some objective reality of a complete workspace depends on the intended application of the parser results. In 2D the difference between objective and subjective placements of objects is fairly small, but in 3D the difference can be large, as some of the figures have shown. We think that parsing our 3D workspaces from an objective global view point would much too often confuse users as structures which seem apparent on the screen do not necessarily exist from an objective global view point, and would result in them struggling to place materials precisely to fit the parser instead of getting on with their inevitably subjective work.

In the shared 2D parser of CAOS [14] the point was to make the parser incremental and share the parse result among many cooperating users. We do not think that such sharing would be beneficial for our 3D parser as cooperating users (fully supported by Topos) do not typically share the same view point, and would therefore require different parses. Also, users cooperating with other Topos users may only be sharing parts of the scene and as a whole the scene seen by one user may be very different from the scene seen by another.

No formal user tests have yet been carried out, but the users of the Topos system that have tried to use the parser have given very positive feedback. The parser allows treating informal groups of objects as a whole without making their relationship formal and more permanent by explicitly grouping the objects into a sub-workspace.

Acknowledgments. This work has been funded by the Danish Research Council's Centre for Multimedia (project no. 9600869), the EU Esprit LTR project DESARTE and the EU IST project WorkSPACE.

References

- [1] B.-B. Bederson and J. D. Hollan. Pad++: A zooming graphical interface for exploring alternate interface physics. In *Proc. ACM UIST'94*, pages 17–26. ACM, ACM Press, 1994.
- [2] M. Büscher, M. Christensen, K. Grønbæk, P. Krogh, P. Mogensen, D. Shapiro, and P. Ørbæk. Collaborative augmented reality environments: Integrating VR, working materials and distributed work spaces. In *Proc. Collaborative Virtual Environments 2000*, pages 47–56, San Francisco, Sept. 2000. ACM, ACM Press.
- [3] M. Büscher, M. Christensen, P. Mogensen, D. Shapiro, and P. Ørbæk. Creativity, complexity and precision: Information visualization for landscape architecture. In *Proc. ACM Intl. Symp. on Information Visualization 2000*, pages 167–171, Salt Lake City, Oct. 2000. ACM, ACM Press.
- [4] S. K. Card, G. G. Robertson, and W. York. The webbook and the web forager: An information workspace for the world-wide web. In *Proc. ACM SIGCHI'96*, pages 111–117. ACM, Apr. 1996.
- [5] T. Igarashi, S. Matsuoka, and T. Masui. Adaptive recognition of implicit structures in human-organized layouts. In *Proc. 11'th Intl. Symp. on Visual Languages (VL'95)*. ACM, ACM Press, 1995.
- [6] J. Light, J. D. Miller, D. Sorensen, and D. Harris. Cinematic dynamics in the information savanna. In *Submitted for publication*, 2001.
- [7] J. D. Mackinlay, G. G. Robertson, and S. K. Card. The perspective wall: Detail and context smoothly integrated. In *Proc. CHI'91*, pages 173–179, New Orleans, 1991. ACM.
- [8] C. C. Marshall, F. G. Halasz, R. A. Rogers, and W. C. H. Jr. Aquanet: A hypertext tool to hold your knowledge in place. In *Proc. ACM Hypertext'91*, pages 261–275. ACM, ACM Press, Dec. 1991.
- [9] C. C. Marshall and R. A. Rogers. Two years before the mist: Experiences with aquanet. In *Proc. ACM ECHT'92*, pages 53–62, Milano, Nov. 1992. ACM, ACM Press.
- [10] C. C. Marshall and F. M. Shipman. Searching for the missing link: Discovering implicit structure in spatial hypertext. In *Proc. ACM Hypertext'93*. ACM, ACM Press, Nov. 1993.
- [11] C. C. Marshall, F. M. Shipman, and J. H. Coombs. VIKI: Spatial hypertext supporting emergent structure. In *Proc. ACM ECHT'94*, pages 13–23. ACM Press, Sept. 1994.
- [12] P. Mogensen and K. Grønbæk. Hypermedia in the virtual project room – toward open 3D spatial hypermedia. In *Proc. 11th Conf. on Hypertext and Hypermedia*, pages 113–122, San Antonio, TX, 2000. ACM.
- [13] P. J. Nürnberg, J. J. Leggett, and E. R. Schneider. As we should have thought. In *Proc. 8th. ACM Conf. on Hypertext*, pages 96–101, Southampton, UK, Apr. 1997. ACM, ACM Press.
- [14] O. Reinert, D. Bucka-Lassen, C. A. Pedersen, and P. J. Nürnberg. CAOS: A collaborative and open spatial structure service component with incremental spatial parsing. In *Proc. ACM Hypertext'99*, pages 49–50, Darmstadt, 1999. ACM, ACM Press.
- [15] G. Robertson, M. V. Dantzich, D. Robbins, M. Czerwinski, K. Hinckley, K. Ridsen, D. Thiel, and V. Gorokhovskiy. The task gallery: A 3d window manager. In *Proc. ACM SIGCHI'00*, pages 494–501. ACM, 2000.
- [16] G. G. Robertson, M. Czerwinski, K. Larson, D. C. Robbins, D. Thiel, and M. van Dantzich. Data mountain: Using spatial memory for document management. In *ACM Symposium on User Interface Software and Technology*, pages 153–162, 1998.
- [17] F. M. Shipman, C. C. Marshall, and M. LeMere. Beyond location: Hypertext workspaces and non-linear views. In *Proc. ACM Hypertext'99*, pages 121–130, Darmstadt, Feb. 1999. ACM, ACM Press.