

BALANCING FREEDOM AND CONTROL FOR WALKTHROUGH IN VIRTUAL ARCHITECTURE

A Smart and Comprehensive Navigation System

WEI YAN

*Department of Architecture, Texas A&M University
wyan@archmail.tamu.edu*

Abstract. This paper presents a new navigation system that integrates a perspective view for walkthrough and an interactive map for path planning. The interactive map allows users to draw paths and improves path planning by employing either a 3D aerial view that enables users to zoom and examine or a 2D smart map that possesses knowledge about the usability of design elements in the environments. Path control is merged into users' interactive walkthrough seamlessly and intuitively to achieve a balance of freedom and control. Further more, the paths with all related information can be saved in real-time over the Internet by a user and can be loaded and replayed later by other users. Two case studies demonstrate the application of this system in Virtual Architecture.

1. Introduction

The problems of current immersive walkthrough, in terms of navigation, include the following: (1) Difficulty in path finding makes users often get lost because of the nature of complex environments and the lack of cues for path finding. (2) Excessive freedom of movement makes users difficult to have natural paths as in the real world, e.g. users often bump into walls or make unnatural sharp turns. (3) Raster-based maps used in path planning have no knowledge about the environmental elements such as buildings, lawn, fountain, etc. This causes inappropriate planned paths. For example users can draw their paths in areas that are not appropriate for walking, e.g. the roofs of buildings. (4) Designers or users of virtual spaces often have preferred paths for others to visit in order to present their design or interests, but they have no

control of where others will go, unless the paths can be stored and retrieved in real time.

Previous research work has addressed some of these issues separately, e.g. adding global maps in addition to local views (Elvins, et al, 1998; Fukatsu, et al, 1998) or adding various landmarks (Darken, et al, 1996; Vinson, 1999) to help path finding; using the "river analogy" (Galyean, 1995) or "StyleCam" (Burtnyk, et al, 2002) to guide users and enable users to deviate from the guided paths. However, each of these methods alone cannot provide a satisfactory solution, e.g. adding maps or landmarks cannot solve the natural path and the guided tour problems; the "river analogy" or "StyleCam" does not provide map knowledge thus users are still lacking context for path finding and lacking capability for path planning. Also, none of them addresses the problems of inappropriate paths and real-time path saving and retrieving.

Our solution of a new navigation system addresses the above problems comprehensively. It includes the following aspects.

1. It integrates a perspective view and an interactive map. In the perspective view, Route Knowledge (Edward and Hand, 1997) enables navigation from point to point using landmarks, and is based on an egocentric reference frame. In the map, Survey Knowledge (Edward and Hand, 1997) enables efficient planning of journeys, and is based on an exocentric reference frame.
2. Employing an aerial view as a 3D map that allows users to zoom and examine, or a 2D smart map that possesses knowledge about the usability of design elements in the environments, our system enables users to draw paths and improves path planning. The smart map can tell the users when they make mistakes thus can guarantee users to draw their paths within appropriate (e.g. walkable) areas, such as a ground, and avoid inappropriate (e.g. non-walkable) places, such as building roofs on a map of a city plaza. In addition, the system can generate natural paths with a curve-fitting algorithm. It also can indicate users' position and orientation on the map.
3. Path control is merged into users' interactive walkthrough seamlessly and intuitively. Users take a walkthrough by following the pre-defined paths and by simultaneously and interactively controlling their orientation in the perspective view. The walkthrough is partially guided by the pre-defined paths and partially controlled by users' real-time input. This integrates and balances freedom and control in walkthrough.
4. Further more, the paths with all related information, including coordinates, viewing angles, walking speeds, duration of standing, and events and people seen during the walkthrough, can be saved in real-time over the Internet by a user and can be loaded and replayed later by other users. That way, users can experience the same walkthrough

made by any other user, e.g. the space's designer who wants to present the design through her preferred paths.

2. Methodology and Implementation

We have built two prototype versions of the system: one with 3D map and the other with 2D map, demonstrating different features of the proposed solution. Their implementations are based on the common (Web-based 3D) platforms therefore the features of the two versions can be combined easily later.

2.1. NAVIGATION SYSTEM WITH 3D MAP

Version 1 (V1) of the system consists of a 3D perspective view that allows users to walk through, a 3D aerial view that enables users to draw paths and know their locations and directions, and a control panel that allows users to control their movement, e.g. moving forward or turning left. It is implemented using VRML, Java and Perl. A user can submit a VRML world URL through the Web and retrieve the user interface of the navigation system with the VRML world loaded. The diagram in Figure 1 shows the workflow of the system. The navigation interface consists of two VRML browsers and a Java applet. The viewpoint in the perspective view is controlled by the camera in the aerial view. The control panel controls both perspective view and aerial view (Figure 2). In the aerial view, users can zoom in and out and use Examiner Viewer type to see the camera's position and rotation. Users can also set the transparency of the structure to see better the camera and the paths inside buildings. Paths with all related information, including coordinates, viewing angles, and walking speeds, can be saved in real-time over the Internet by a user and can be loaded and replayed later by other users.

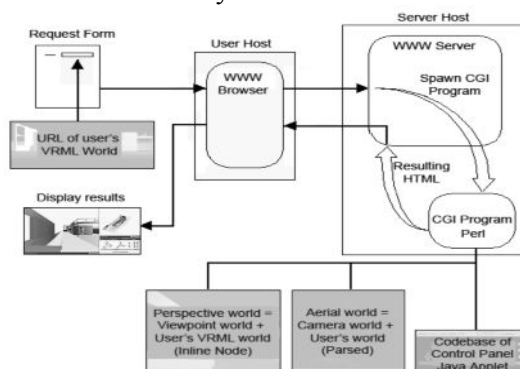


Figure 1. Workflow of the system (V1).

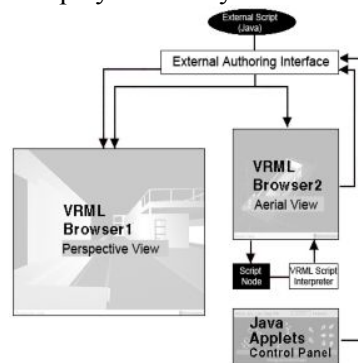


Figure 2. UI of the system (V1).

2.2. NAVIGATION SYSTEM WITH 2D SMART MAP

Version 2 (V2) of the system has focused on smart navigation. Our system can automatically convert a CAD model into a usability-based building model and generate a smart map in Scalable Vector Graphics (SVG) format, which is a subset of XML for graphical presentation (Table 1). SVG is semantically rich: the graphical information is always associated with meaningful textual information. Its element-tree can be traversed to search for a specific node, which can be checked for attributes such as usability. In our approach, as the geometries in the DXF format are converted into SVG elements, their architectural semantics are inserted into the SVG element-tree according to their layer names. Custom tags of the architectural components with their attributes, in extended architectural name-space, appear in the data structure. (A more detailed description of this model can be found in Yan and Kalay, 2005). Therefore, the smart map can possess both geometry information and usability properties of each design element, e.g. it knows a ground is walkable but a fountain is not.

Table 1. Navigation map in SVG format (partial).

```
<?xml version="1.0"?>
.....
<arch:ground id="ground1" isWalkable="yes"/>
.....
<arch:building id="building1" isWalkable="no"/>
<polygon
  points="223.5,41.8 223.5,199.8 272.5,199.8 272.5, ....."
  style="fill:rgb(128,0,0);stroke:rgb(0,0,0);stroke-width:1"
  transform="translate(-12 -21) translate(1.33333 2.66667) translate(-1.6
    0.8)"/>
.....
<arch:fountain id="fountain1" isWalkable="no"/>
<ellipse cx="368" cy="354" rx="12" ry="12"
  transform="translate(1 2) translate(-14 -22) translate(1.6 4)"
  style="fill:rgb(0,0,255);stroke:rgb(0,0,0);stroke-width:1"/>
</svg>
```

We created our system using Batik SVG package as a platform, which uses Java2D as the rendering engine. The system processes the SVG map using Java through Document Object Model (DOM). When rendering, Batik SVG skips branches of the DOM tree that use a tag it doesn't know about (e.g. arch:building, arch:ground, or arch:fountain). But when looking up architectural elements and attributes (e.g. walkability) those branches in the DOM tree will be used. The Java application will look up the DOM tree for

getting and setting architectural elements and their attributes (e.g. walkability, color, etc., see Table 2). (See Watt, A. et al. 2003 for details of Batik and DOM).

Table 2. Check for usability attribute in Java (partial).

```

public class SVGApplication extends JApplet {
    .....
    public class OnMouseUpAction implements ActionListener {
        .....
        public void handleEvent(Event evt) {
            elt = (Element) evt.getTarget();
            elt.setAttribute("style",oldAttribute);
            if (!isWalkable.equals("yes")){
                window.setTimeout(new DelayedTask(), 50);}}
        }
    public class DelayedTask implements Runnable {
        public void run() {
            window.alert(archType + " is not walkable!");
        }
    }
    .....
}

```

The user interface includes a 3D perspective view, which is a VRML browser, and a 2D map view, constructed in SVG and Java. The map enables users to click and create control points of a path. When users click within an area that is not appropriate for walking, the system will inform the users and will not allow the new path segment to be drawn.

We apply B-Spline algorithm to curve-fit the initial paths (polylines) using users input as control points. In addition to creating paths, a user can also load pre-recorded paths. Once a path has been created or loaded, the user can start the walkthrough. During walkthrough the user can click and drag in the 3D perspective view to change the viewing direction while following the path. This is done by a callback function that examines the users' mouse input in the VRML viewer through a proximity sensor. Another callback function sends the users' information in VRML viewer to the map, which then indicates the user's current location and orientation on the map.

3. Applications

Applications of this system include navigation in Virtual Reality, Virtual Architecture, etc. As case studies, we tested the two versions of the system with different architectural environments.

3.1. APPLICATION OF SYSTEM VERSION 1 TO LE CORBUSIER'S HOUSE

For Version 1, we used Le Corbusier's small house to demo how the system works. Le Corbusier designed a small house for his parents by Lake Geneva. In his book *UNE PETITE MAISON* written in 1923, he narrated his architectural design while navigating through the house. He drew paths on the floor plan and showed pictures along the paths (Figure 3).

We loaded a 3D model of this house into our system (Figure 4). First using the control panel, we can move forward or backward, and we can rotate the camera, and change viewpoints of the aerial view. We can go to any location by simply clicking on the floor plan in aerial view. We can draw paths and save them into a data file on the server and other users can load them and replay them later. We can follow Le Corbusier's path to experience the space that he tried to present in his book: entering the house from the backyard, entering the living room that has a 15 meter long window, through which we can see Lake Geneva; entering the bedroom, and then the kitchen; and finally back to the backyard through another door (Figure 5 shows two snapshots). This system improves navigation in building models with its reference to the aerial view, and it enables remote design presentation by providing guided virtual tour.

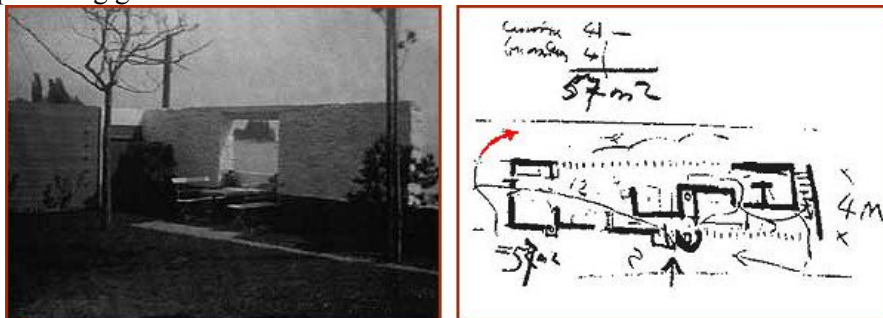


Figure 3. Photo and sketch in *UNE PETITE MAISON*, Le Corbusier, EDITION 1981, Zurich

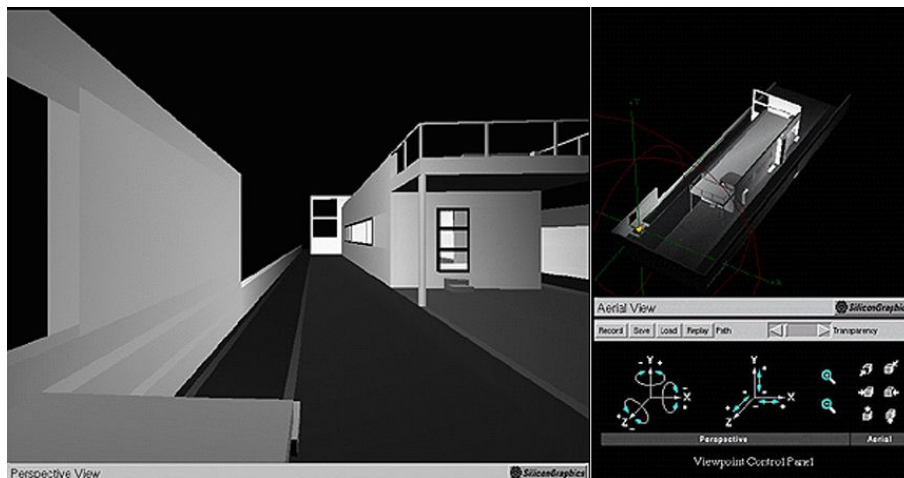


Figure 4. UI of the system version 1: perspective view on the left, aerial view on the upper right and control panel on the lower right. The camera is located at the left corner of the house in the aerial view, where users can zoom and examine. A user also can set the transparency of the house to see the camera and the paths inside the building.

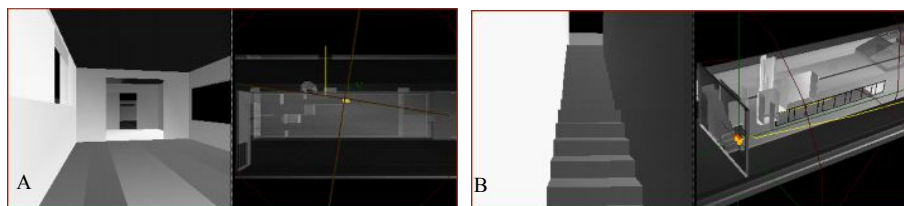


Figure 5. Two screenshots showing a camera moving the same way as in Le Corbusier's drawing and corresponding views. A: entering the house; B: on the stair. The left part of each screenshot is the perspective view and the right part is the aerial view showing the camera and the path.

3.2. APPLICATION OF SYSTEM VERSION 2 TO SPROUL PLAZA

In another case study, we applied Version 2 of the system to a campus plaza - Sproul Plaza at UC Berkeley (Figure 6 to 10). First, with a normal (non-smart) map, if a user draws a path in an area that is not appropriate for walking, e.g. the roofs of buildings, existing systems cannot tell and inappropriate paths (with segments on roofs) will be created (Figure 6).

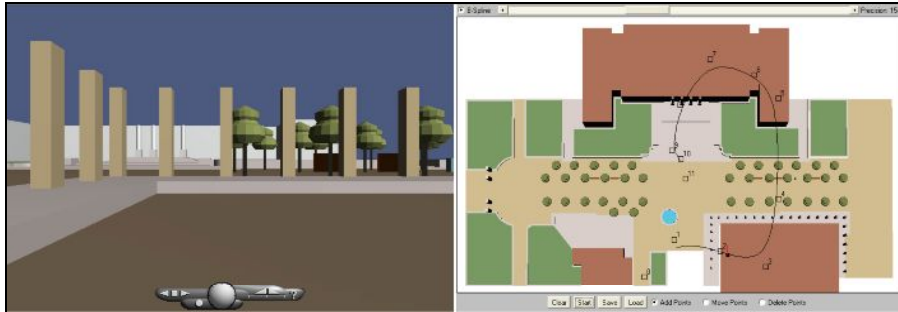


Figure 6. UI of the system version 2: a perspective view on the left and a map on the right. If users draw paths in areas that are not appropriate for walking, e.g. the roofs of buildings in Sproul Plaza, existing systems are not able to tell and inappropriate paths will be generated.

With the smart map, however, when a user clicked on a building roof (the rectangle on the lower right of the plaza), the system warned that the building roof was not walkable (Figure 7).

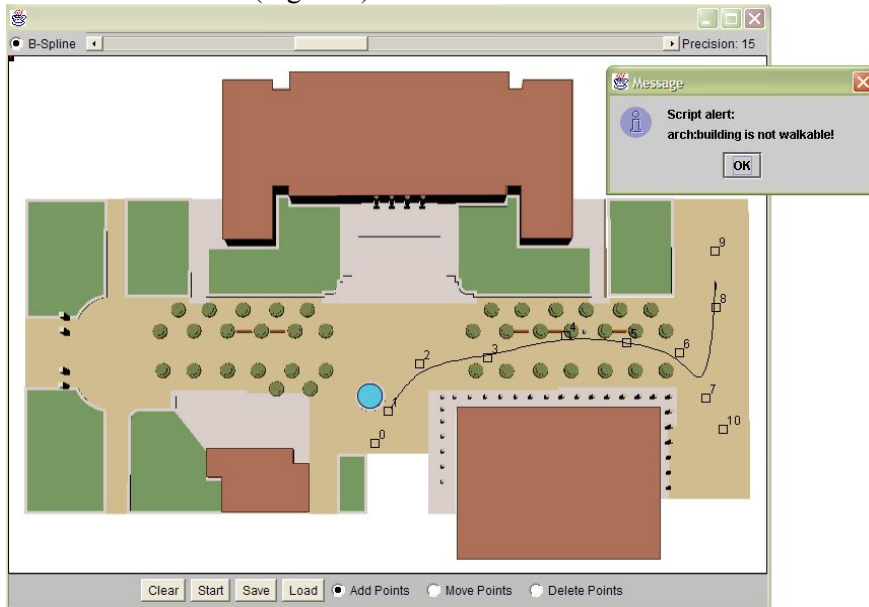


Figure 7. The smart map warned that the building roof was not walkable.

Figure 8 to 10 shows that after creating or loading paths in the map, a user started the tour following the path and at the same time controlling the viewing direction interactively in the perspective view to look around. Again, path control is merged into users' interactive walkthrough seamlessly and intuitively.

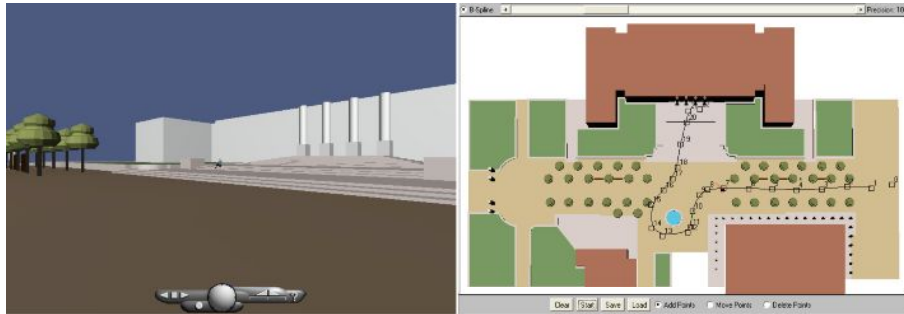


Figure 8. A user loaded a pre-saved path and started the tour in Sproul Plaza. Then the user turned and looked at Sproul Hall.

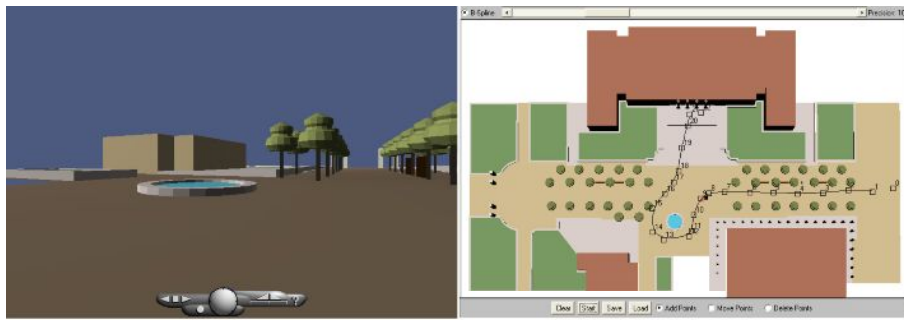


Figure 9. The user turned and looked at the fountain.

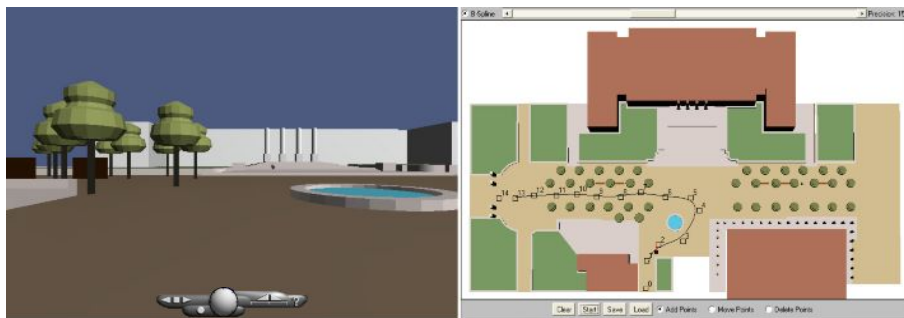


Figure 10. The user created a new path and started the tour at the Lower Sproul Plaza and looked at the direction of the fountain and Sproul Hall.

5. Conclusions and Future Work

Evaluation and review has been conducted with an architect and graduate students in a Human-Computer Interface class. Experiment with our new

system showed an improved user experience over existing VRML browsers. Our future work will include combining the two versions of the system into a coherent one and conducting more user evaluation. We expect our research on navigation will ultimately assist in offering new approaches for the future development of 3D viewers and will assist in architectural design presentation that employs Virtual Reality techniques.

Acknowledgements

Thanks to Prof. Leandro Madrazo (then at ETH Zurich) for his help with the first version of the system, Prof. Jennifer Mankoff (then at UC Berkeley) and Prof. Robert Wilensky at UC Berkeley for their comments on the second version of the system.

References

- Burtnyk, N., Khan, A., Fitzmaurice, G., Balakrishnan, R., and Kurtenbach, G.: 2002, StyleCam: Interactive Stylized 3D Navigation using Integrated Spatial & Temporal Controls. *ACM CHI Letters*, 4(2). p. 101-110. *ACM UIST 2002 Symposium on User Interface Software & Technology*.
- Darken, R., & Sibert, J.: 1996, Wayfinding strategies and behaviours in large virtual worlds. *ACM CHI'96 Conference on Human Factors in Computing Systems*. p. 142-149.
- Edwards, J. and Hand, C.: 1997, MaPS: Movement And Planning Support For Navigation In An Immersive VRML Browser, *VRML97*, Monterey CA/USA
- Elvins, T., Nadeau, D., Schul, R., & Kirsh, D.: 1998, Worldlets: 3D thumbnails for 3D browsing. *ACM CHI'98 Conf. on Human Factors in Computing Systems*. p. 163-170.
- Fukatsu, S., Kitamura, Y., Masaki, T., & Kishino, F.: 1998, Intuitive control of bird's eye overview images for navigation in an enormous virtual environment. *ACM VRST'98 Symposium on Virtual Reality Software and Technology*. p. 67-76.
- Galyean, T.A.: 1995, Guided navigation of virtual environments. *ACM I3D'95 Symposium on Interactive 3D Graphics*. p. 103-104.
- Watt, A. et al.: 2003, *SVG Unleashed*. Sams Publishing.
- Yan W. and Kalay Y.: 2005, Simulating Human Behaviour in Built Environments. *Computer-Aided Architectural Design Futures 2005*, Vienna, Austria.