3D Contents Browser Using a Cluttering-Avoided Visualization Technique

Reiko Miyazaki*
Ochanomizu Univ.

Takayuki Itoh†
Ochanomizu Univ.

ABSTRACT

We propose a 3D visualization technique which realizes browsing of hierarchical or metadata-based link structures intelligibly while avoiding 3D cluttering. The poster describes the algorithm which relocates the metaphor of contents mapped onto a 3D space so that cluttering is avoided on a 2D projection space. We apply the visualization technique "HeiankyoView" which represents hierarchical structure as nested rectangular regions, and enables to directly access to the contents which place at the lowest level of hierarchy. Furthermore, when a user chooses arbitrary content, the technique efficiently represents the classification result based on another metadata simultaneously.

1 INTRODUCTION

Many of existing contents browsers dynamically display hierarchy of contents according to multiple metadata. For example, typical music browsers display the contents while they form multiple levels of categories, such as genre, artists, and albums. Such interface often bothers users, because it may require many steps of selections. Also, it often occurs that users feel difficult to return to the previous display, since they often browse the contents while randomly selecting the metadata.

To solve the problem, we propose a new contents browser, which hierarchically place the groups of contents by applying HeiankyoView [1], and three-dimensionally pile the groups of contents. The technique realizes all-in-one display of large number of contents, and reduces the operations for users since it allows directly accessing each of the contents.

Generally, three-dimensional visualization system has a capability of display of large-scale information in one display space. However, it has a drawback of cluttering problem; it often causes occlusions among data elements, and therefore it often requires complicated operations to access the occluded contents. Cluttering is a general problem in 3D applications, and several studies have addressed the problem. One of the work applied a dynamics model to avoid the cluttering [2]. The poster also focuses on the technique which avoids cluttering among the three-dimensionally piled contents, so that it can make users easily browse the hierarchy of contents dynamically categorized according to user-specified metadata.

2 PROPOSED TECHNIQUE

The proposed browser consists of the following three features: 1) avoidance of cluttering, 2) switch of metadata for dynamic construction of hierarchy, and 3) management of history of display.

2.1 Avoidance of cluttering

Figure 1 show an example of the contents display by the proposed technique, which realizes the cluttering-avoided layout of contents, by the following three steps.

*e-mail:reiko@itolab.is.ocha.ac.jp
†e-mail:itot@is.ocha.ac.jp

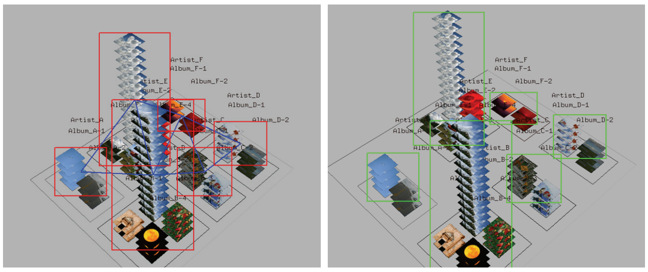


Figure 1: Avoidance of cluttering. (Left) Before avoidance of cluttering. (Right) After avoidance of cluttering.

2.1.1 Definition of bounding rectangle

HeiankyoView represents hierarchy of the data as nested rectangular frames. This representation style is suitable for users to recognize the hierarchical structure. However, cluttering due to the occlusions among the three-dimensionally piled contents may prevent the recognition of the hierarchical structure.

Here, the proposed technique defines "bounding rectangle" which is the rectangle enclosing the piled contents on the display, shown as red rectangles in Figure 1(Left), or green rectangles in Figure 1(Right). We treat the cluttering problem as an overlap problem of boundary rectangles on a two-dimensional display space.

2.1.2 Positioning of overlap-avoided boundary rectangles

The technique calculates positions of boundary rectangles defined in Section 2.1.1, while it reduces the overlap among the boundary rectangles by the following procedure on the two-dimensional display space. The procedure is similar to a technique [3] which reduces the overlap of icons.

1. The technique first constructs a Delaunay triangular mesh which connects centers of boundary rectangles.

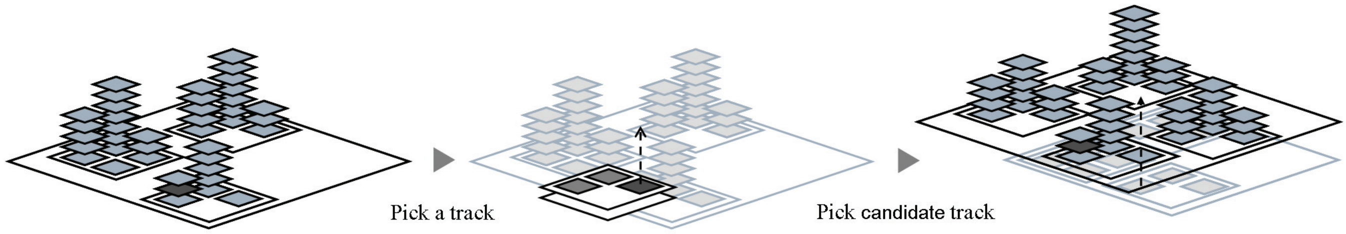
2. The technique then selects an edge of the triangular mesh, and calculates the ideal positions of two boundary rectangles connected by the edge. It defines the positions as is, if the two boundary rectangles do not overlap each other. Otherwise, it translates them along the edge at minimum, so that they do not overlap each other. The technique applies the above procedure to all possible edges of the triangular mesh.

3. The technique finally minimizes the sum of distances between current and calculated positions of all boundary rectangles, by applying least square method. If the procedure is applied while the viewing operations, it also refers the calculated positions in the previous moment.

The technique repeats the above procedure until the overlaps among the boundary rectangles are enough reduced.

2.1.3 Feedback onto the three-dimensional contents layout

The technique calculates the positions of contents in the 3D space by the following procedure, from the ideal positions of boundary rectangles calculated by the procedure described in Section 2.1.2.



	Attribute	Focused track
Frame	Artist	A
Pile	Album	A-1
(Invisible)	Genre	Jazz
(Invisible)	Mood	Healing

(a) Default pile structure

	Attribute	Focused track
Frame	Artist	A
Pile	Album	A-1
Metadata pile	Genre	Jazz
Metadata pile	Mood	Healing

(b) Show Metadata pile

	Attribute	Focused track
Pile	Artist	B
(Invisible)	Album	B-1
Frame	Genre	Jazz
(Invisible)	Mood	Fiery

(c) Show new pile structure and History pile

Figure 2: Switch of metadata for dynamic construction of hierarchy.

1. Let the ideal position of a boundary rectangle in the 2D display space as (x', y').

2. Transparently display a place defined as y =(average of heights of the highest and lowest piles), and then obtain the value of z' at (x', y') from a depth-buffer.

3. Calculate the position (x, y, z) in the 3D space by inverse projection transformation, from (x', y', z') in the display space.

2.2 Switch of metadata for dynamic construction of hierarchy

Generally, music contents files can contain various metadata, including genre, mood, artist name, album name, and composer name. It is possible that users expand their association of interested music based on metadata: for example, users may be interested in other music composed by the same people, or other music which are categorized as the same mood but the different genre.

The proposed technique supports the effective contents browsing, which allows users dynamically switching metadata for hierarchy reconstruction, and therefore assists them to expand their association. Figure 2 denotes an example of switch of metadata for hierarchy reconstruction. As shown in Figure 2(a), the first level of the hierarchy (represented as rectangular frames) is initially based on artist names, the second level of the hierarchy (represented as piles) is initially based on album names, and nodes in this example denote tracks. When a user picks a track, the browser displays "metadata pile" representing the set of metadata of the node except the metadata used for hierarchy construction. In particular case of Figure 2(b), two small planes (drawn by inky black) represents genre and mood; they do not represent other metadata used for hierarchy construction, artist and album in this case. Each of the planes in the metadata pile also displays tracks which have same metadata (same genre and mood respectively), one for each of first level groups (one for each artist in this case), including the picked track.

When a user picks a track in a metadata pile, the browser constructs a new hierarchy, where frames are based on the picked metadata (genre or mood in this case), and piles are based on the metadata originally referred for frame construction (artist in this case). As shown in Figure 2(c), the browser displays all tracks based on the new hierarchy. Also, users can return to the previous display as shown in Figure 2(a), by picking the originally selected track.

2.3 Management of history of display

When the browser displays the new hierarchy as shown in Figure 2(c), it keeps displaying the previous hierarchy. The browser flattens the pile of the previous hierarchy, and displays below the new hierarchy. It places the picked track at the same position, and marks the picked track on both the hierarchy so that users can visually recognize the linkage between the two hierarchies, shown as a dotted arrow in Figure 2(c). This style makes users easier to visually recognize how they jumped to the new hierarchy.

Also, the browser allows users returning to the previous hierarchy by one operation. When the user picks the previous hierarchy, the browser restores the pile of the previous hierarchy and displays above the new hierarchy; at that same time the browser flattens the new hierarchy and displays below the previous hierarchy.

3 Future Works

We would like to extend the proposed browser as a music recommendation browser. It is desirable to use feature values calculated by signal processing techniques as additional metadata, since they are effective for the sort of the piled contents by calculating the score of recommendation from the metadata.

Also, we would like to extend the contents layout technique as follows. We would like to bring in importance-based weighted calculation of overlap, so that interested groups can be well-clutter-avoided. Then we would like to add an interactive mechanism for dynamic definition of bounding rectangles of interested area, so that users can interactively input their interests.

References

[1] T. Itoh, H. Takakura, A. Sawada, and K. Koyamada. Hierarchical visualization of network intrusion detection data in the ip address space. In *IEEE Computer Graphics and Applications*, volume 26, pages 40–47, 2006.

[2] S. Takahashi, K. Yoshida, K. Shimada, and T. Nishita. Occlusion–free animation of driving routes for car navigation systems. In *IEEE Transactions on Visualization and Computer Graphics (Proceedings of IEEE Visualization / Information Visualization 2006)*, volume 12, pages 1141–1148, 2006.

[3] N. Watanabe, M. Washida, and T. Igarashi. Bubble clusters: An interface for manipulating spatial aggregation of graphical objects. In *ACM User Interface Software and Technology*, pages 173–182, 2007.