

Spatial-Temporal Data Management for a 3D Graphical Information System

Minoru Maruyama¹, Teruhiko Teraoka¹, Takashi Tamada¹, and Shogo Nishida²

¹ Central Research Laboratory, Mitsubishi Electric Corporation
Amagasaki, Hyogo 661, Japan

² Faculty of Engineering Science, Osaka University
Toyonaka, Osaka 560, Japan

Abstract. In this paper, we propose a spatio-temporal data management method for a 3D graphical facility management system. To provide highly interactive environment, we propose a 3D spatial data management method based on multiple 2D-trees. With the method, interactive operations such as smooth walkthrough and interactive information retrieval in 3D space are made possible. Then, we augment the data structure to deal with the temporal information as well as the spatial information. MP-tree is proposed for spatio-temporal data management. With the use of the data structure, efficient temporal search is made possible keeping the performance of spatial search. We also present a 3D electric facility management system using our data management method.

1 Introduction

In most of the geographical information systems (GIS), 3D objects are represented by 2D shapes. However, due to the growth of the computational power of workstations, the system that can directly handle 3D shapes themselves is coming to within the scope. The 3D graphical information system that enables users to interactively explore the virtual 3D environment is very useful for many tasks such as management, analysis and planning etc. In 3D systems, as conventional 2D systems, various interactive operations including interactive information retrieval, smooth walkthrough in 3D space are required.

In this paper, we describe the data management method for a 3D graphical facility management system. To realize interactive operations, rapid spatial search based on the hierarchical data structure is very helpful. Based on the hierarchical data structure, we propose a data management method for spatial information of city area to realize various kinds of interactive operations. Then, we augment the data structure to deal with the temporal information (e.g. the installation/update history of facilities) as well as the spatial information. MP-tree (Multidimensional Persistent tree) is proposed for spatio-temporal data management. With the use of the data structure, efficient temporal search is made possible keeping the performance of spatial search. We also present a 3D electric facility management system using our data management method.

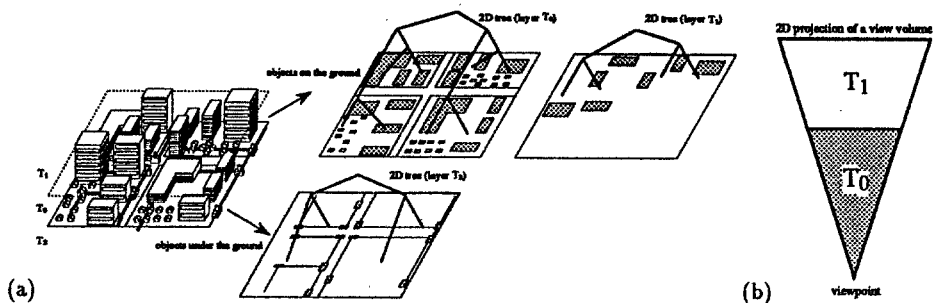


Fig. 1. (a)Management of 3D objects by multiple 2D trees (b)Division of the view volume and the corresponding layers

2 Spatial data management for interactive operations

We propose a spatial data management method for interactive operations in 3D space such as smooth walkthrough. To realize smooth walkthrough, the frame update rate around 10 frames/sec is required. During the walkthrough, usually the number of visible objects is very small compared to the total number of objects. Therefore, if the potentially visible objects are found very rapidly, it prunes unnecessary objects to render and can accelerate the rendering process. To find the potentially visible objects rapidly, efficient range search based on the hierarchical data structure is quite helpful[4]. In our method, each 3D object is managed by using its 2D projection onto the ground. The projected 2D shape is managed by the 2D kd-tree[1]. Potentially visible objects whose 2D projections intersect with the 2D projection of the view volume can be retrieved rapidly by the range search.

For the efficient display control, the objects on the ground and the objects under the ground are managed separately by distinct 2D trees (layers) as shown in Fig.1. Moreover, the space on the ground is further sliced into multiple layers depending on the height. As Fig.1 shows, all the objects on the ground are managed by the tree (layer) T_0 . If the object is higher than the given threshold, it is also managed by another tree T_1 . In the city area, if an object is far from the viewpoint, it is likely that the object is occluded by the other objects which are closer to the viewer. This implies that an object far from the viewer is visible only if it is tall enough. Based on the heuristics, the search area is divided into two regions (Fig.1(b)). During the walkthrough operation on the ground, in the region which is close to the viewpoint, the potentially visible objects are searched using the layer T_0 . In the other region, which is far from the viewpoint, visible objects are searched using the *higher* layer T_1 . With this technique, we can expect to reduce the number of objects to render without degrading the image quality.

3 Augmenting the data structure for spatio-temporal data management

3.1 Possible methods to manage spatio-temporal information

Time is an important aspect of real-world phenomena. In this section, the multidimensional data structure is augmented to handle the geometric objects with temporal information. In this paper, we consider spatio-temporal data which consists of location and its insertion/deletion version. It is required to design a data structure that manages temporal information efficiently keeping the performance of the spatial search. We consider several possible methods and discuss about their advantages and disadvantages. The methods considered are :

– **Single tree structure (ST)**

Geometric objects are managed by a single tree. Version (temporal) information is dealt with as one of the attributes of each object.

– **Multiple trees (MT)**

For each version, a distinct tree is constructed. Thus, to handle all the data, multiple trees are employed.

– **Multidimensional tree (3DT)**

N-dimensional data with version information can be regarded as (N+1)-dimensional data. Multidimensional data structures (e.g. kd-tree) can be directly applied to the (N+1)-dimensional data.

Each of the above three methods has its own advantages and disadvantages with respect to the storage cost and the time (search) cost. For example, if a version number is specified, the search performance is very efficient with MT. However, it is not so efficient for spatio-temporal range search, where multiple versions are involved. Moreover, since long-lived data gives rise to many copies, MT requires huge storage cost. On the other hand, ST does not require any *additional* storage cost. However, since the structure of ST does not reflect any time information, ST can be inefficient for time involving search. A 2D point with its insertion/deletion version information can be regarded as a line segment in 3D space. If the data set is static, where time domain is bounded, then 3DT may

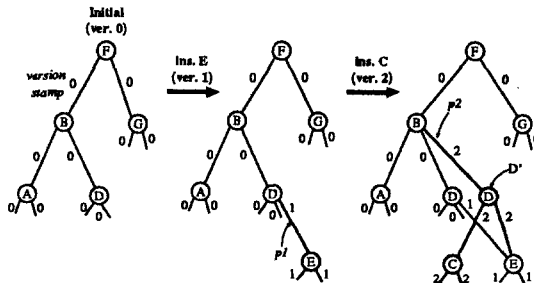


Fig. 2. Persistent binary search tree (node copying method)

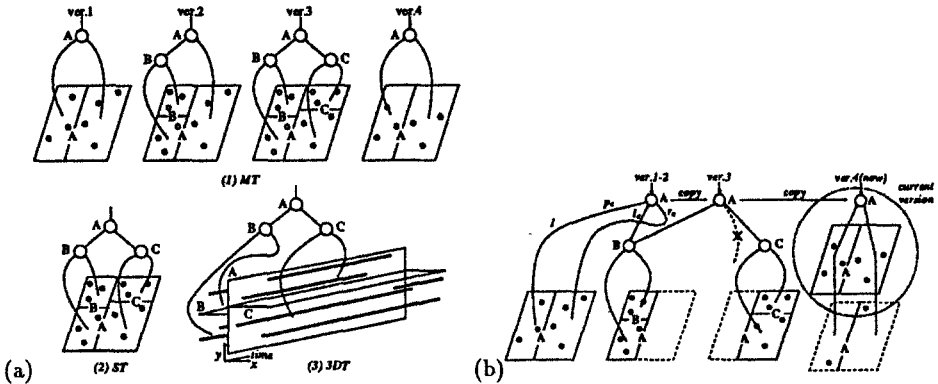


Fig. 3. (a)Data management by ST, MT and 3DT (b)Data management by MP-tree (In the figures, A,B,C show the discriminator values, that specify the partitions)

be a very good data management method. However, if the data set is dynamic, where time domain is unbounded, then 3DT cannot be directly applied.

Based on the above analysis, we try to design a new data structure that satisfies the following requirements: 1. Storage cost is not so expensive as MT, 2. Rapid spatio-temporal search can be performed with the data structure, 3. A dynamic data set can be handled without difficulty, 4. Spatial search for the current version can be performed as fast as in the case of MT.

3.2 The multidimensional persistent tree

To satisfy the above requirements, we use MP-tree (Multi-dimensional persistent tree)[5]. MP-tree is the multidimensional extension of the persistent search tree [3]. MP-tree is constructed by extending the conventional spatial data structures (e.g. kd-tree) using the node copying method[3] (see Fig.2) to save the changes. In Fig. 3 we show data management methods by ST, MT, 3DT and the MP-tree.

The MP-tree is a multi-way tree that consists of two types of nodes : internal nodes and leaf nodes. In MP-tree, spatial data partitioning procedure is same as the conventional spatial data structure (in our case kd-tree). Thus, each internal node has a spatial key. The configuration of the tree reflects not only the spatial information but also the temporal information (i.e. changes of data). Temporal information is managed by the pointers of nodes storing *the time stamp*. In the MP-tree, each historical state is stored in a subtree of the MP-tree. In the spatio-temporal search, the spatial key of each node is examined to get the spatial information and the time stamp of each pointer is examined to get the temporal information while traversing the tree. The features of MP-tree include : 1. A bucket method [2] is applicable, 2. At each version update, the number of

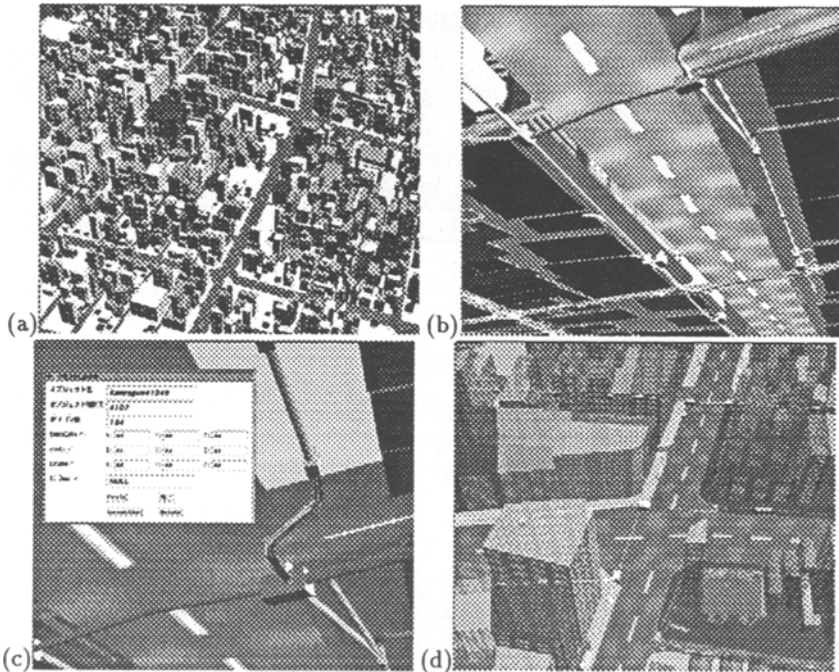


Fig. 4. (a)Entire region to be managed (b) A scene from the viewpoint under the ground (c)Interactive information retrieval by picking the 3D object (d)A scene using transparency control

data to be inserted or deleted is not limited to one, 3. A spatial search for the current version can be performed as fast as in the case of MT, 4. A multiversion range search for the short time interval can be performed faster than ST, MT, 5. The storage cost is much less expensive than MT.

4 A 3D facility management system

We have implemented our spatio-temporal data management method to a 3D facility management system. The system manages 3D objects on the ground such as buildings, houses, roads, and the electric facilities installed under the ground such as pipes, cables, man-holes. In Fig4 (a),(b), we show a scene of the buildings of the area, and a scene of the electric facilities under the ground. By our data management method, users can explore the 3D space freely and interactively retrieve the information of facilities (see Fig.4(c)). Due to the use of 3D models users can intuitively understand the spatial relations, such as proximity. In addition, exploiting some visualization techniques, such as transparency control, understanding of spatial relations between the ground facilities and the

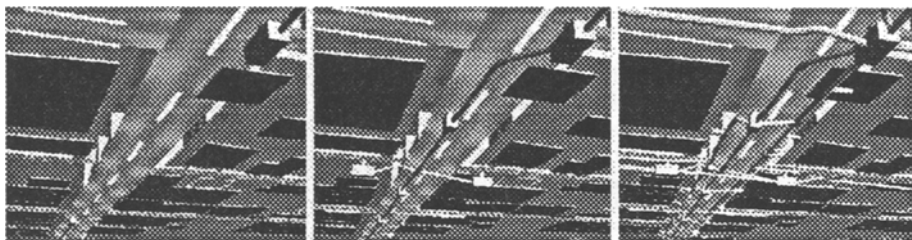


Fig. 5. An example of temporal search. Visualization of the facility installation history

underground facilities gets easier (see Fig.4(d)). Moreover, with the use of MP-tree, our system provides efficient spatio-temporal search. In Fig. 5, we show an example of visualization of facility installation history.

5 Conclusion

We have presented an efficient data management method for a 3D graphical facility management system. To provide highly interactive environment, we propose a 3D data management method based on multiple 2D-trees. With the method, interactive operations such as smooth walkthrough and interactive information retrieval are made possible. For facility management, not only the spatial information but also the temporal information such as facility installation history is very important. We have proposed the MP-tree for efficient spatio-temporal data management. Using the MP-tree, efficient temporal search is carried out keeping the performance of spatial search. We have implemented the data management method for electric facility management system. The 3D graphical facility management system that enables users to explore the virtual 3D environment will become a strong tool for not only data management but analysis and planning.

References

1. J.L.Bentley, "Multidimensional binary search trees used for associative searching" , Comm. ACM, Vol.18, pp. 509 - 517 (1975).
2. H.Samet, "The design and analysis of spatial data structures" , Addison-Wesley (1989).
3. N.Sarnak and R.E.Tarjan, "Planar point location using persistent search trees" , Comm. ACM, Vol.29, pp. 669 - 679 (1986).
4. T.Tamada, Y.Nakamura and S.Takeda, "An efficient 3D object management and interactive walkthrough for the 3D facility management system", Proc. IECON '94, pp. 1937 - 1941 (1994).
5. T.Teraoka, M.Maruyama, and S.Nishida, "The MP-tree : A data structure for spatio-temporal data", Proc. International Phoenix Conference on Computers and Communications '95, pp. 326 - 333 (1995).