Proceedings of the Seventh International
Conference on Human-Computer Interaction,
(HCI International'97), Vol.2, pp869-872, 1997

Presentation and Interaction of Virtual 3D Objects without Geometric Model

Shinji Uchiyama, Hiroyuki Yamamoto, Akihiro Katayama, and Hideyuki Tamura

Mixed Reality Systems Laboratory Inc.,
6-145 Hanasaki-cho, Nishi-ku, Yokohama 220, Japan
{uchiyama, ymmt, katayama, tamura}@mr-system.co.jp

We discuss how to locate and manipulate the multi-viewpoint data set (MVDS) in virtual environment. Since MVDS has no geometric information in it, additional information, which links a virtual object represented by MVDS and geometry-based virtual space, should be attached to it. Without having that kind of information, no manipulation in virtual space is possible. Four types of link information are studied from the point of interactivity. In addition, experimental examples confirm the usefulness of MVDS with link information.

1. Introduction

Most of recent virtual reality (VR) systems utilize geometric models to represent virtual spaces and objects. To make virtual space more realistic, people seek ways to utilize captured images of real scenes. Recently, several methods have been proposed (from [1] to [3]) in which multiple images at given viewpoints are used to reconstruct views at various viewpoints without any explicit geometric data. Authors call this image data set as MVDS (Multi-Viewpoint Data Set) and the method to render images by this approach is widely called “image-based rendering.”

We have already proposed a method to construct a virtual space with traditional geometry-based space and objects represented by ray-space data [4][5]. Note that the ray-space data is one type of representation of MVDS. With this method, MVDS data and geometry data can be seamlessly rendered into an image, though manipulation of MVDS objects is limited.

In this paper, we systematically study and extend the method in order to treat MVDSs and the geometric models in the same representation level for virtual environment. In this process, knowledge types, which links MVDS objects and geometry-based virtual space, and interaction class are categorized. Based on the categories, the relations between interaction and link information are clarified. As well as the methodical study, experimental examples confirm the usefulness of MVDS with link information in virtual reality system.

2. MVDS to represent objects

Geometric models represent objects by their shapes. On the other hand, MVDSs describe objects by recording data to reconstruct views at various viewpoints. Images observed at all the possible viewpoints is an ideal case of MVDS, though it is not practical because of its huge data. Thus methods to reconstruct any views from a set of raw images at given viewpoints are proposed. In this paper, the term “MVDS” is used to represent a data set which is converted from the set of raw images and is suitable for view reconstruction. Although methods to recover an object shape from MVDS can be used to reconstruct views, we consider reconstruction methods without recovering explicit geometry.

Let's assume getting an MVDS from images at N viewpoints. In this case, N images record rays reflected or emitted from an object surface through N image planes. Therefore, the target is explicitly or implicitly assumed to exist in the region restricted by the N image planes.

If the target is a solid object, the region becomes a closed space (region R) surrounded by the N image planes as shown in figure 1. Throughout the discussion, raw images for MVDSs are assumed to be acquired with this approach.

In this approach, there can be two types of viewpoint arrangement, two dimensional and three dimensional, as shown in figure 2. The difference of arrangement affects the spatial resolution and freedom of view reconstruction.

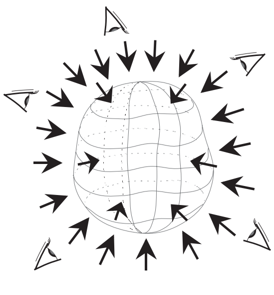
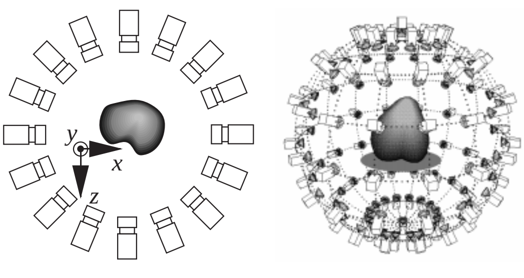


Figure 1. Observing a solid object



(a) Two dimensional (b) Three dimensional
Figure 2. Capturing views of an object

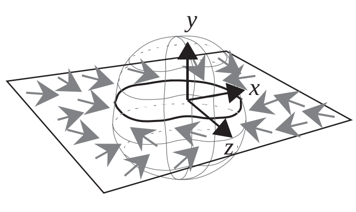


Figure 3. Limitation on viewpoint placements

(a) Two dimensional observation

In this method, viewpoints are allocated on a circle with its center placed within a small region in the object as illustrated in figure 2 (a). In addition, all the views are directed to the center. This is realized with a moving camera along a circle with an object at its center, or an object on a turning table with a camera viewing toward. MVDSs created from a set of images at these viewpoints are much smaller than those obtained by the three dimensional arrangement. But it is impossible to reconstruct views at points not being on the circle plane (*x*-*z* plane) since this type of MVDS has no vertical (*y* direction) disparity information. Thus the freedom on reconstruction is limited to 3 (*x*, *z*, and yaw) as shown in figure 3.

(b) Three dimensional observation

On the other hand, viewpoints can be allocated on the sphere with its center placed within a small region in the object in this arrangement as illustrated in figure 2(b). MVDSs obtained with this viewpoint arrangement provide perfect views with 6 degree of viewpoint freedom at the sacrifice of the amount of data.

In either case, relative positions between the viewpoints and the object must be calibrated before taking the images. The calibration can be achieved by computer vision techniques tracking some feature points in the scene. It is also possible to obtain the positions directly with a motion controllable imaging device.

3. MVDSs in virtual space

3.1 Interaction with virtual environment

Interaction with the virtual space is indispensable to make people feel its presence. There are three levels of the interaction as shown below, and if higher level of interaction is realized, we can increase the sence of presence.

- Interaction level 1: (1.a) Realtime rendering (1.b) Free placement of viewpoint
- Interaction level 2: (2.a) Translation and rotation of objects (2.b) Interference and collision detection
- Interaction level 3: (3.a) Deformation of objects (3.b) Introduction of gravitational field, calculation of elasticity, etc.

Interaction of level 1 is basic and indispensable for realizing the functionality to walk-through the virtual space. Interaction of level 2 is required to manipulate objects in the space. This interaction includes basic manipulation of objects (2.a) and collision detection (2.b) to provide more natural virtual experience.

Interaction of level 3 is more complicated and includes deformation of objects (3.a) and kinetic simulation (3.b) such as gravitational field or elastic deformation.

The purpose of introducing MVDS into virtual space is to realize photo-reality in the space. As described in Section 2, we have limited the introduction of MVDSs only to solid objects (items) in the space. Therefore, our system uses geometric models for the environmental entity such as walls and shelves. This method has already been proposed in [4]. figure 6 is an example of rendering of virtual space using this method. As shown in the figure, the space has higher photo-reality than a space build only from geometric models. Since it has already been stated in [4] that this method satisfies interaction of level 1, we will discuss only about interaction levels of 2 and 3 in this paper.

3.2 Geometric information attached to MVDSs

Since MVDSs have no geometric information about the object, it is not possible to place them into the space nor to manipulate them in the space. With the spatial information about region R, however, it becomes possible to place and manipulate them.

What kind of spatial information should be attached to MVDSs? First of all, the relation between a coordinate system

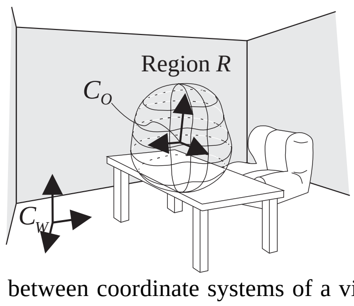
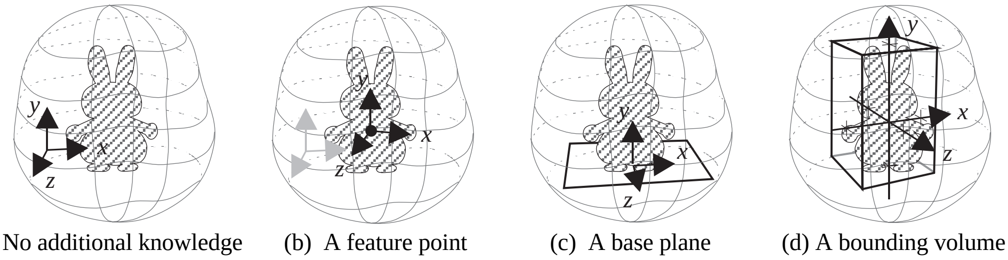


Figure 4. Interrelationship between coordinate systems of a virtual space and an MVDS



(a) No additional knowledge (b) A feature point (c) A base plane (d) A bounding volume

Figure 5. Attaching geometric data to an MVDS

(C_o) set in the region R and that (C_w) of the virtual space is required in order to place and manipulate MVDS objects in virtual space while getting consistent views (see figure 4). This is because the relations between MVDS objects as well as virtual space and camera coordinate system at the viewpoint are determined by the information above.

In addition, it is possible to extend placement and manipulation with the additional knowledge about coordinate system C_o. This knowledge includes the origin, direction, and range of C_o which in principle are not available in MVDSs. Thus the following four levels are categorized as the information attached to MVDSs. These four levels are illustrated in figure 5.

(L1) No additional knowledge

This is the case that any knowledge is not attached to MVDS other than the region R. Since the C_o can be set arbitrarily in the region R, C_o is not related to the object described by the MVDS. Thus the place and range of the item in the region R are not determined.

(L2) A feature point

If a certain point of the object is attached to the MVDS as the origin of coordinate C_o, the position of the item's feature point can be specified but the attitude can not.

(L3) A base plane

A base plane attached to MVDS as the *x-z* plane of C_o can set the attitude of the item but cannot define its range.

(L4) A bounding box

A bounding volume, such as a bounding box to show the spatial area occupied by the item, can define the placement and range of the item, though they are not strictly specified.

4. Placement and manipulation of MVDSs

In this section, we clarify what kind of interaction is possible for each level of attached knowledge described above.

4.1 Locating MVDSs

(L1) The item represented by MVDS can be roughly placed in a virtual space with only the information about the region R. The object oriented location, however, such as placing an object on another object cannot be performed since the placement and range of the item are not determined. Although the locating operations are limited, MVDS object of this kind is still useful. Consider the case when a flower vase is placed on a table. In such a case, it is sufficient that the flower vase is seen as if it is placed on the table. Thus, adjusting the transformation from C_o to C_w manually by observing the rendered images, placement can be achieved without visual inconsistency.

(L2) The same kind of locating can be achieved as the L1 above.

(L3) The MVDS object can be placed on top of another object since interference of the MVDS object against the upper plane of the other, represented by MVDS or geometric model, can be computed. MVDS data of this kind is specially useful in typical virtual reality experience where placing objects on top of planes according to the gravitation is a major operation.

(L4) Not only base plane interference but also interference of an MVDS object against other objects can be computed, thus the MVDS item can be placed as it does not overlap other objects.

4.2 Translation and rotation of MVDSs

(L1) Though MVDS objects can be moved and rotated by changing the translation and rotation matrix between the coordinates C_o and C_w, it is impossible to control the movement and rotation considering the existence of other objects. This is because the placement and range of the items are not described. Object oriented rotation is also impossible since the position is undetermined.

(L2) In addition to the possible operation described in L1, an MVDS object can be rotated as the item since the attached feature point indicates the center of rotation.

(L3) Movement and rotation of MVDS objects can be constrained on the plane where they are placed.

(L4) As well as the properties described in L3, the movement and rotation are performed so as not to overlap others.

4.3 Mutual occlusion

(L1) It cannot be judged whether an MVDS item or its part is occluded by another object or not since the placement and range of the item are undetermined.

(L2) Limited mutual occlusion can be achieved considering the distance between the viewpoint and the feature point of the MVDS as the existent range of the item. It is impossible, however, to represent mutual occlusion in such a case where the item is placed so that it passes through the opening of a coffee cup handle. That is, the mutual occlusion is calculated when the MVDS item is nearly convex and it does not conceal another object occluding the item itself.

(L3) The same kind of mutual occlusion can be achieved as the L2 above.

(L4) Assuming the bounding volume of the item, more realistic mutual occlusion is performed compared to the case L2.

4.4 Picking an MVDS

There are two kinds of methods available to pick an MVDS item in a virtual space: one is a two dimensional method such as clicking an item rendered on a screen and the other is to grasp an item virtually with some VR 3D device such as a glove. In either case, some geometrical information which shows the spatial range of the item is necessary. For this reason, the MVDS item must have a bounding volume (L4) or at least a rectangular plate showing the two dimensional range of the item (L3).

Note that it is usually used a simplified shape to simulate the picking of a geometric model, since the original complex shape is computationally expensive to calculate the interference. Thus above condition for picking MVDSs is not a drawback in the virtual object interaction.

4.5 Interference

(L1) It cannot be determined whether two items interfere each other.

(L2) Overlap judgment is not possible as L1.

(L3) Interference of the MVDS object against the upper plane of the other, represented by MVDS or geometric model, can be computed.

(L4) Though interference cannot be strictly judged, rough interference computation can be achieved by considering the bounding volume of the MVDS item as the shape of the item.

4.6 Deformation of MVDS objects

In all cases, an item cannot be partially deformed since the exact surface of the item is not determined. However, its whole size can be increased or decreased by changing the scaling factor of the coordinate system C_O.

4.7 Application example of MVDSs in virtual reality

Figure 6 is a design example to show how the six functionalities studied above are implemented. This figure is a rendering result of a virtual space in which geometric models are mixed with MVDSs using the method described in [4].

Three items, one toy rabbit and two lady's suits, are represented by MVDSs and placed inside a virtual mall represented using geometric models.

Figure 7 shows an example of realtime manipulation of MVDSs. Each MVDS has its own bounding box defined. Three kinds of manipulators are superimposed as shown in figure 7 when you pick up this box. These are a ring shaped belt for rotation, four plates placed outside of the belt to control horizontal translation and a tiny cube to control vertical translation. You have to switch to the scaling mode when you want to change the scale of an item. If you pick an item in the scaling mode, other manipulators (large wire frame of a cube consisting of eight tiny cubes and connecting lines) as shown in figure 8 are displayed. Figure 8 shows an enlarged toy rabbit.

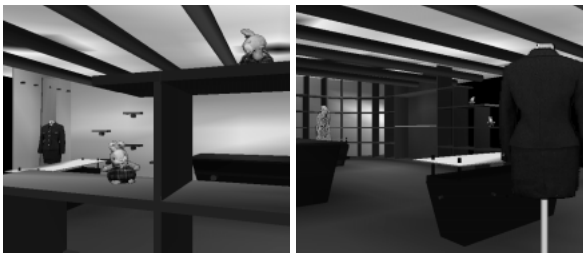


Figure 6. Virtual mall

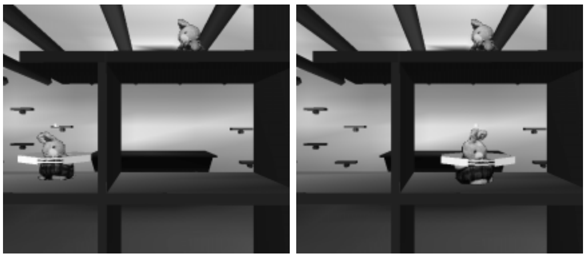


Figure 7. Manipulating an MVDS object

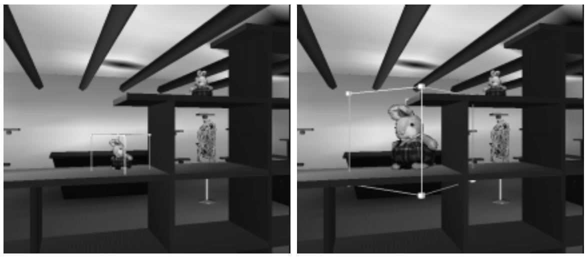


Figure 8. Scaling an MVDS object

5. Concluding remarks

This paper describes the characteristics the Multi-Viewpoint Data Set (MVDS) for image-based rendering has at the interaction process in a virtual environment.

We often think it is impossible to manipulate our MVDSs since they do not have explicit geometric models. However, breaking the interaction activities adopted by conventional virtual reality systems into the three levels explained in this paper, it is revealed that the level 1 and 2 can also be applied to our MVDSs.

The essential purpose of the MVDSs is to realize high quality photo-realism using images captured from real world. Although it is impossible to realize the third level of interaction with the MVDSs, there are a lot of application fields that can be benefited from their photo-reality.

References

[1] A. Katayama, K. Tanaka, T. Oshino, and H. Tamura: “Viewpoint-dependent stereoscopic display using interpolation of multi-viewpoint images,” Proc. SPIE, Vol.2409, pp.11-20, 1995.

[2] M. Levoy, and P. Hanrahan: “Light field rendering,” Proc. SIGGRAPH, pp.31-42, 1996.

[3] S. J. Gortler, R.Grzeszczuk, R. Szeliski, and M. F. Cohen: “The lumigraph,” ibid., pp.43-54, 1996.

[4] S. Uchiyama, A. Katayama, H. Tamura, T. Naemura, M. Kaneko, H. Harashima: “Building a cyber-space by mixing real data based on ray-space theory with computer graphics models,” Proc. 3D Image Conference’96, 1-3, pp.13-18, 1996 (in Japanese).

[5] http://www.x-zone.canon.co.jp/CyberMirage/index-e.html