

Image-Based Virtual World Generation

Michitaka Hirose
University of Tokyo

The methodologies introduced here generate photographically realistic 3D worlds from 2D photographic images instead of from 3D geometrical models. If we call conventional methods “algorithm intensive,” these methods are “data intensive.” Several prototype systems, including the Virtual Dome and Camera with GPS, serve as examples. Of course, both types have their advantages, so trade-offs and combinations of the two are also discussed briefly.

When the term “virtual reality” debuted at the end of the 1980s, the technology gained popularity because of its strange and interesting interface devices. However, with the technology now being considered more seriously, quality has become an important issue.

At the very beginning, the virtual world displayed in primitive head-mounted displays seemed a “toy-like world” consisting of simple polygons. This naive computer graphics technology was based on 3D geometrical models. However, if we want to implement more complex worlds, this straightforward methodology has a serious limitation: Defining a 3D model from huge numbers of polygons is exhaustive, time-consuming work, and real-time drawing of the 3D model requires an expensive graphics workstation with a powerful geometry engine. When my laboratory implemented a world consisting of several blocks of downtown Tokyo, we defined 100,000 polygons—but the quality of the virtual landscape generated was far from satisfactory. To develop serious applications, we must greatly improve the quality of such worlds to move beyond toy-like.

Of course, we currently have many sophisticated 3D graphics tools such as 3D modeling systems, 3D scanning systems, and 3D “clip art.” These tools combined with hours of labor let us generate sophisticated CG images as seen in movies, but we can’t use these 3D worlds to generate “interactive” worlds. Image quality comes at the cost of great development effort and slow rendering speed.

Various research efforts have attempted to improve the quality of “interactive” virtual

worlds. This article introduces several attempts conducted in my laboratory.

Algorithm intensive to data intensive

One promising strategy for world generation employs image-based rendering technology. This methodology uses 2D photographic images instead of 3D geometrical models. Worlds generated using this method have an advantage compared to those created using the polygon-based method—generating the same image quality is much easier. In addition, world generation is relatively easy following preparation of the 2D images. For example, the Virtual Dome system discussed later in this article can generate a photo-realistic world by rotating a camera once. The quality of the image remains basically independent of the rendering speed because everything is a simple bitmap. You can consider various sources for images, even existing 2D sources such as movies, videos, and prerendered CG images.

This kind of technology has become quite popular recently, with increasingly active research in various institutes.¹⁻³ Most of you probably know about Apple Computer’s QuickTime VR, used on the World Wide Web to display 3D objects from various viewpoints.⁴ Several other companies reportedly are also ready to distribute similar tools.

What differentiates the conventional polygon-based technology and the 2D image-based technology? Each has advantages and disadvantages. For example, the former is very generic, capable of generating any worlds and objects by using a geometrical model from the beginning. Users encounter no limitation in interacting with the world. We can call this an “algorithm-intensive methodology.” In contrast, the geometrical model is implicit in the latter. It produces quality as good as conventional 2D media, but interaction with the world is limited. A good example is a table look-up. We can call this a “data-intensive methodology.”

The former strongly depends on CPU capability, and the latter depends on memory capacity. In other words, the latter requires a huge amount of data space because it has to handle redundant data. It also has disadvantages in networking, requiring very high bandwidth to share an image-based virtual world. For this reason, most current image-based systems in my laboratory are designed for just one user. However, given communication lines with broader bandwidth (say ATM networks), the systems can be extended to multiple users. (Several tens of megabits per second per person will suffice.)

Figure 1. Spectrum of world generation methods.

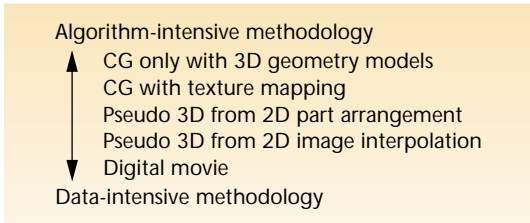


Figure 2. Conceptual image of the Virtual Dome.

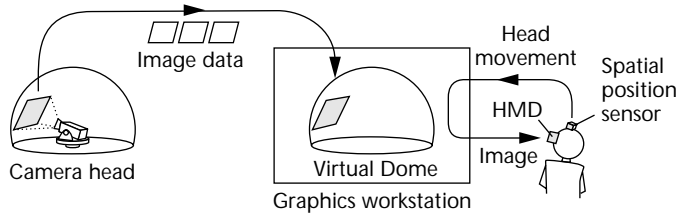
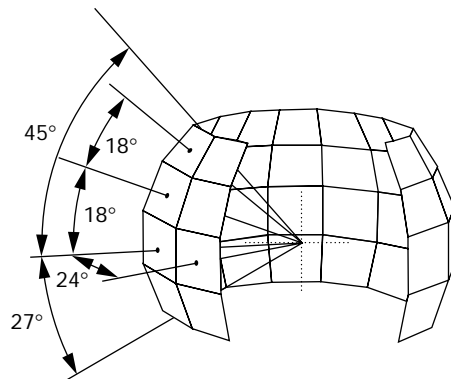


Figure 3. Basic principle behind the Virtual Dome.

One reason I am still interested in this image-based technology is because I believe improving memory capacity and communication channel capacity (which are quantitative problems) is much easier than improving CPU performance (which is a qualitative problem). In fact, current PCs already have enough memory and capability for transmitting or handling bitmap image data—although they need further development.

Consider speech synthesis. The algorithm-

Figure 4. Shape of the virtual spherical screen.



intensive methodology corresponds to methods that synthesize every phoneme by modeling vocal chord motion. On the other hand, the data-intensive methodology corresponds to the use of edited and spliced recordings of human voices. Although in principle the former can generate unlimited voices, the quality of the synthesized voice is not as good as with the latter. In contrast, if we can obtain a huge memory device at a reasonable cost, no technical difficulty will hinder our obtaining a realistic voice. As a result, most real application systems for voice synthesis seem to employ the prerecorded method.

The analogy to virtual-world generation is clear. The algorithm-intensive method corresponds to polygon-based computer graphics. As an example of a data-intensive method, we have digital image editing. Figure 1 shows the spectrum of world generation stretching between algorithm-intensive and data-intensive synthesis. By combining these methods, we should be able to generate more complex scenes of better quality.

Original Virtual Dome

One of the simplest uses of the data-intensive method is just to arrange photographs, as with the Virtual Dome developed in my laboratory in the late 1980s.⁵ We originally developed this system to provide a wide field of view to a remote user (see Figure 2).

The main concept behind the Virtual Dome is to disconnect the movement of the HMD and the camera head, as shown in Figure 3. The camera head continuously scans the surrounding space in order to capture a complete image of the area. The captured images are transmitted to the graphics workstation via a communication line. In the graphics workstation, a spherical shell is prepared as a virtual Omnimax screen onto which the transmitted images are texture mapped.

With this configuration, a user wearing the HMD should be able to look around the rotating camera's world, even with a very large time delay caused by the distance. If an HMD and a camera head are directly connected via a communication line with a large time delay, the camera does not respond to the user's head movement immediately, and the user does not get the sensation of presence in the remote location. The permissible delay between HMD and camera movement is around several hundred milliseconds. This critical value is very constraining because a 1-second (1,000 milliseconds) delay can easily occur when we use a satellite communication link.

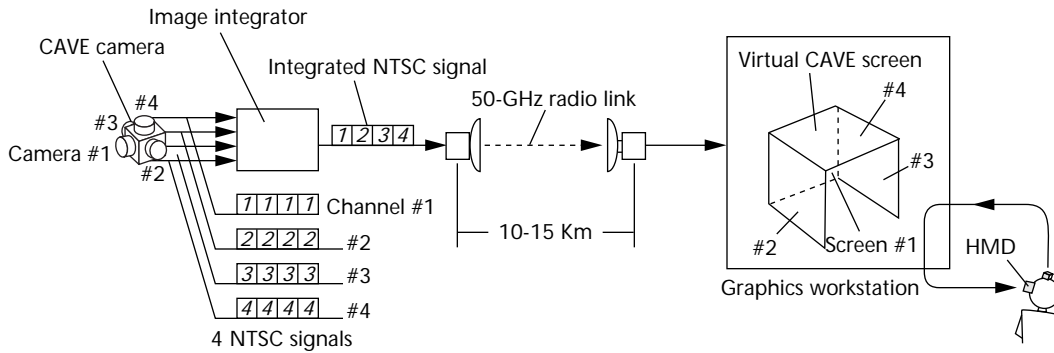


Figure 5. Virtual Dome for live-image video.

As shown in Figure 4, the screen is composed of 90 polygons total. This virtual screen can support a view field of 45 degrees above and 27 degrees below the horizon. At the very beginning of development, handling live images in the Virtual Dome environment was extremely difficult, so the first implementation was limited to still images. However, thanks to this shortcoming of the prototype implementation, it became clear that the Virtual Dome could be used as a tool for virtual-world generation.

The Virtual Dome can generate virtual worlds easily because just rotating the camera head obtains a photorealistic virtual world. As seen in the previous figures, the image displayed in the Virtual Dome system differs from a “toy” world. Of course, today in 1997, implementation for live images is also available even without special expensive hardware such as Synthevision.⁶ Figure 5 shows the actual implementation for live video images.

Extended Virtual Dome

The original Virtual Dome environment did not support a changing image caused by translational movement of the user’s head. In other words, the user would encounter limitations in interacting with the 3D world. One idea to allow translational head movement involves making the surface of the spherical screen uneven,⁷ similar to a relief map (see Figure 6). This extended Virtual Dome system requires the partial use of 3D geometrical models again. A similar idea (combining a digital image and a 3D model) occurs in augmented reality such as Immersive Video² and the 3D Virtualized Studio.⁸

As Figure 7 shows, motion parallax easily produces the sensation of 3D. The left image is an original view. If the user slightly changes the viewpoint to the right, the right image appears. Interestingly enough, even if the screen were

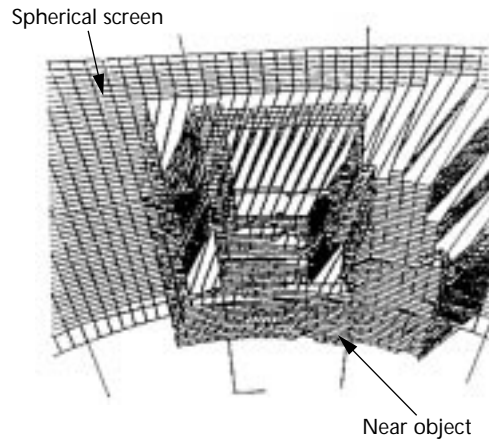


Figure 6. Uneven screen of the extended Virtual Dome. Part of the near object is sticking out from the spherical screen.

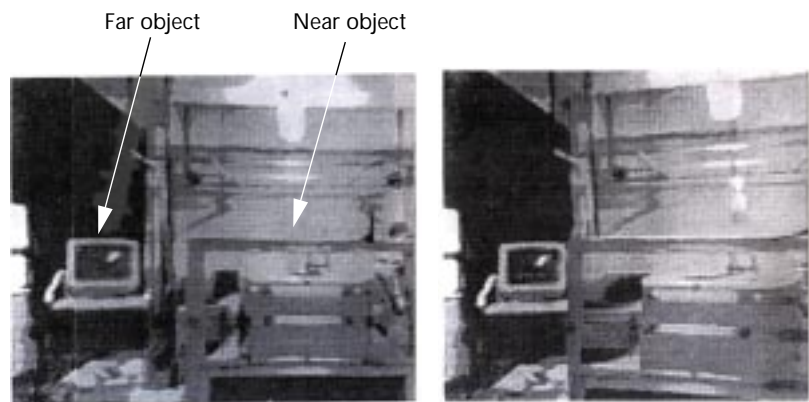


Figure 7. Motion parallax synthesized by an uneven screen. The original image is on the left. If you move the viewpoint to the right, the near object image (projected as sticking out) moves more than the far object image projected on the spherical screen, as shown on the right.

made partially uneven, users could get very good 3D sensations because of the illusions. Using the 3D geometrical model, it should be simple to obtain real-time interaction; the exact 3D geometry, as measured by a sensor such as a laser range finder, is not needed.

We can imagine a simplified method. Just arranging 2D texture

Figure 8. A 3D world generated by arranging 2D textures.



parts in 3D space as shown in Figure 8 can generate a pseudo-3D world. This method resembles a *set*—scenery used in theatrical performances.

As shown in Figure 9, even using such a very simple 3D model produces excellent results. In generating this kind of panel-like world, interactive image handling tools that support object extraction (cutting objects out of the background), removal of perspective effects, or touching up of occluded areas would be very useful.

However, this kind of 3D extension of the Virtual Dome works only for small changes in the user's viewpoint. If we want to walk around wider areas, we will need other ideas for extension.

Figure 9. Motion parallax synthesized by 2D textures, moving from left to right. The first image shows the view from the left. The last image shows the view from the right.

Camera with GPS

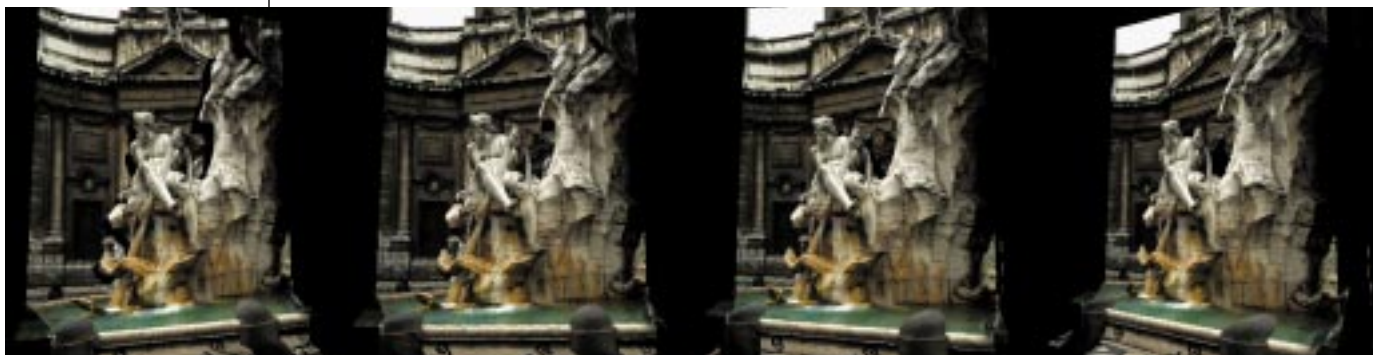
MIT's Aspen Movie Map was a pioneering system that let users browse a large 3D image world. This system used images prerecorded on a laser disc. One problem was the fairly large effort of

preparing images; the path for image capture had to be carefully planned and many images prepared to ensure a smooth change in the displayed images.

Recently, my lab developed a new camera system with a position sensor.⁹ Although a conventional video camera only produces a noninteractive sequence of images, this sequence can produce interactive images following the principle shown in Figure 10. A camera equipped with a position sensor can produce a huge sequence of image data with viewpoint information having six degrees of freedom (three for position and three for direction). For this wide-range position data acquisition, you should use a positioning sensor with a wider measurement range such as GPS (Global Positioning System) instead of a conventional magnetic position sensor such as the Polhemus sensor, which has a measurement range of several cubic meters. Although the GPS measurement error is around several tens of meters, when integrated into a car navigation system, its error can be reduced to several meters.

The image data are transmitted to the graphics workstation and arranged as an image database, which is a simple data array just for image storage, made from scratch in my laboratory. A major difference between this system and the Aspen Movie Map is the use of image interpolation. Using image interpolation greatly reduces the required number of prerecorded images because we should be able to estimate images from arbitrary viewpoints. In other words, we can synthesize infinitely many images from the finite data set.

Suppose we have two photographs taken from slightly different viewpoints. Based on these images, we should be able to estimate the image from a third viewpoint. In the field of computer graphics, many interpolation technologies have been developed and tested.¹⁰ However, we cannot use overly sophisticated algorithms because the



interpolation should take place in real time.

Morphing is one promising technique. As shown in Figure 11, it involves selecting several major points as reference points. Comparing the positions of reference points in two images lets you interpolate the other points. Using texture mapping, you can then shrink and stretch areas surrounded by the reference points to synthesize the intervening image from two neighboring images.

Figure 12 (next page) shows the browsing of a virtual world generated using this technology. It takes only a few hours to prepare an image database augmented with reference points.

Let's estimate the errors caused by simple interpolation such as zooming. Suppose the distance from the viewpoint to the standard object is D . When the viewpoint moves toward the object by as much as dx , the image of the object moves from s to $s + ds$ on the screen (s is measured from the screen center). When D is sufficiently bigger than dx , ds can be estimated as follows:

$$ds \sim s (dx/D)$$

This transformation is valid only for objects at distance D and not for other objects. The difference of ds will cause distortion of the screen image. If the object lies at distance D^* , the error of ds will be

$$ds - ds^* \sim s dx (1/D - 1/D^*) \\ \sim s dx (D^* - D)/DD^*$$

If we think of $(ds - ds^*)$ as the magnitude of distortion e and $(D^* - D)$ as the variance of D dD , e can be

$$e \sim s (dx/D)(dD/D)$$

This equation tells us that the error will be small when D is large enough. And if the images are sampled densely, distortion will be small because the dx become small. However, dx cannot be too

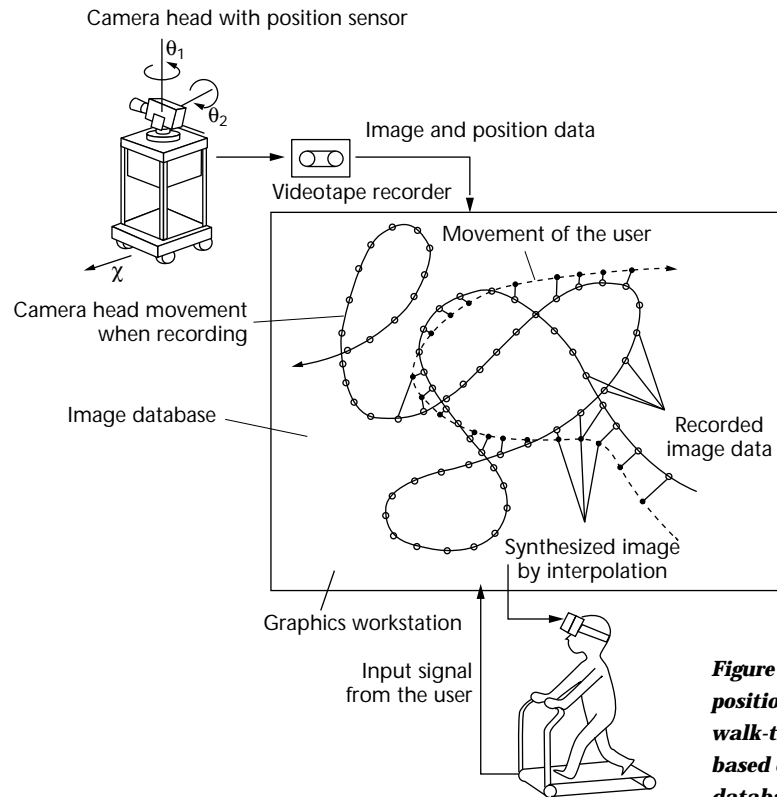


Figure 10. Camera with position sensors. This walk-through system is based on an image database.

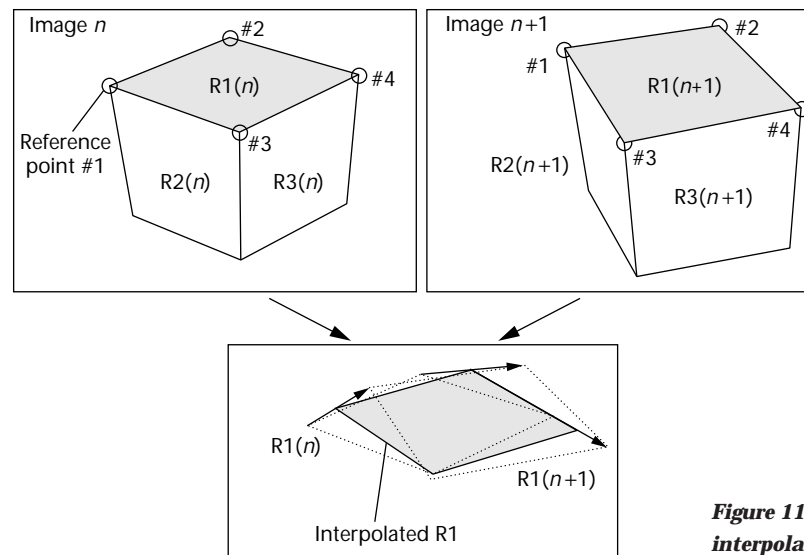


Figure 11. Image interpolation by morphing.

small because the current resolution of GPS is several meters. Some kind of registration technique such as using fiducial marks will help improve the resolution. (Much "registration" research can be found in the field of augmented reality.¹¹)

Trade-offs occur between image distortion and sampling. Improving the interpolation, such as

Figure 12. Motion parallax synthesized by image interpolation. The upper left and lower right images are the originals. The scenes in between are the morphed images.



dividing a scene into several domains having different D (as in Figure 11), proves essential in reducing distortion.

According to our experimental data, dx/D should be less than 0.1 or so. This means we need several gigabytes of memory storage to generate

200-meter by 200-meter street blocks (at 400 by 400 pixels resolution, without image compression). So, this methodology should be closely linked to image-compression technology. In fact, image-based world generation, which includes structured video technology, is one of the most important topics of MPEG-4 discussion.¹²

Geometry and image-handling engines

In this article I have introduced several methodologies for generating realistic scenes of virtual 3D space. Many applications will require photorealistic virtual worlds. For example, exhibits in virtual museums must demonstrate fidelity to real exhibits. Design tools for architects or urban designers should generate realistic landscapes to create the sensation of actually being on the street represented.

Trade-offs also occur between quality and interactivity. By combining digital video technology and computer graphics technology, we should be able to achieve both high quality and high interactivity. Currently most people, including both users and system builders, focus on high-performance machines as measured in terms of polygons drawn per second. Recently my laboratory also began looking at performance in terms of pixels per second. I believe the combination of 3D graphics and digital image technology will give the best results.

Several years ago the data-intensive methodology mentioned in this article was impossible because memory devices and high speed data links cost so much. Then amazing advances in semiconductor technologies, including reductions in memory device cost, made it possible. File servers with several 10-Gbyte hard disks are already available off the shelf. Of course, we still have to use expensive high-end graphics workstations for some of the prototype systems introduced here. But clearly we can look forward to using low-cost personal computers to handle these tasks within a few years. MM

References

1. L. McMillan and G. Bishop, "Plenoptic Modeling: An Image-Based Rendering System," *Siggraph 95 Conf. Proc.*, ACM Press, New York, 1995, pp. 39-46.
2. S. Moezzi et al., "Immersive Video," *Proc. IEEE VRAIS 96*, IEEE Computer Society Press, Los Alamitos, Calif., 1996, pp. 17-24.
3. S. Becker and V.M. Bove, "Semiautomatic 3D Model Extraction from Uncalibrated 2D Camera Views," *SPIE Symp. on Electronic Imaging: Science Technology*, SPIE, Bellingham, Wash., Feb. 1995.

4. S.E. Chen, "QuickTime VR—An Image-Based Approach to Virtual Environment Navigation," *Siggraph 95 Conf. Proc.*, ACM Press, New York, 1995, pp. 29-38.
5. M. Hirose et al., "A Study on Synthetic Visual Sensation through Artificial Reality," *Proc. 7th Symp. on Human Interfaces*, Society of Instrument and Control Engineers (SICE), Tokyo, 1991, pp. 675-682.
6. S. Shimoda, M. Hayashi, and Y. Kanatsugu, "New Chromakey Imaging Technique with Hi-Vision Background," *IEEE Trans. on BT*, Vol. 35, No. 4, 1989, pp. 357-361.
7. M. Hirose, K. Yokoyama, and S. Sato, "Transmission of Realistic Sensation: Development of Virtual Dome," *Proc. IEEE VRAIS 93*, IEEE Neural Networks Council, Piscataway, N.J., 1993, pp. 125-131.
8. T. Kanade, P.J. Narayanan, and P. Rander, "Virtualized (Not Virtual) Reality," *Proc. 15th Int'l Display Research Conf.*, Institute of Television Engineers (ITE) of Japan, Tokyo, Oct. 1995, pp. 799-802.
9. M. Hirose et al., "An Alternate Way to Generate Virtual Worlds: A Study of Image Processing Technology for Synthetic Sensations," *Presence*, Vol. 5, No. 1, 1996, pp. 61-71.
10. S. Ullman and R. Basri, "Recognition by Linear Combinations of Models," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 13, No. 10, 1991, pp. 992-1006.
11. K.N. Kutulakos and J. Vallino, "Affine Object Representations for Calibration-Free Augmented Reality," *Proc. IEEE VRAIS 96*, IEEE Computer Society Press, Los Alamitos, Calif., 1996, pp. 25-36.
12. C. Reader, "MPEG-4 at Present," *Proc. MPEG-4 and Virtual Reality Object Coding Symp.*, Information Processing Society of Japan (IPSI), Tokyo, 1995, pp. 31-35.



Michitaka Hirose is an associate professor of systems engineering in the Department of Mechano-Informatics at the University of Tokyo. His research interests include human interfaces, inter-

active computer graphics, and virtual reality.

Hirose received BE, ME, and PhD degrees from the University of Tokyo in 1977, 1979, and 1982, respectively. He is a member of the ACM, IEEE, SICE, and JSME.

Readers may contact Hirose at the Dept. of Mechano-Informatics, University of Tokyo, 7-3-1, Hongo, Bunkyo-ku, Tokyo, 113, Japan, e-mail hirose@ihl.t.u-tokyo.ac.jp or visit his Web site at <http://www.ihl.t.u-tokyo.ac.jp/index-j.html>.