MODELING FROM REALITY

edited by
Katsushi Ikeuchi
Yoichi Sato

KLUWER ACADEMIC PUBLISHERS

MODELING FROM REALITY

THE KLUWER INTERNATIONAL SERIES IN ENGINEERING AND COMPUTER SCIENCE

ROBOTICS: VISION, MANIPULATION AND SENSORS

Consulting Editor

Takeo Kanade

Other books in the series:

PERCEPTUAL METRICS FOR IMAGE DATABASE NAVIGATION
Y. Rubner, C. Tomasi
ISBN: 0-7923-7219-0
DARWIN2K: An Evolutionary Approach to Automated Design for Robotics
C. Leger
ISBN: 0-7923-7979-2
ENGINEERING APPROACHES TO MECHANICAL AND ROBOTIC DESIGN FOR MINIMALLY INVASIVE SURGERIES
A. Faraz, S. Payandeh
ISBN: 0-7923-7792-3
ROBOT FORCE CONTROL
B. Siciliano, L. Villani
ISBN: 0-7923-7733-8
DESIGN BY COMPOSITION FOR RAPID PROTOTYPING
M. Binnard
ISBN: 0-7923-8657-4
TETROBOT: A Modular Approach to Reconfigurable Parallel Robotics
G.J. Hamlin, A.C. Sanderson
ISBN: 0-7923-8025-8
INTELLIGENT UNMANNED GROUND VEHICLES: Autonomous Navigation Research at Carnegie Mellon
M. Hebert, C. Thorpe, A. Stentz
ISBN: 0-7923-9833-5
INTERLEAVING PLANNING AND EXECUTION FOR AUTONOMOUS ROBOTS
Illah Reza Nourbakhsh
ISBN: 0-7923-9828-9
GENETIC LEARNING FOR ADAPTIVE IMAGE SEGMENTATION
Bir Bhanu, Sungkee Lee
ISBN: 0-7923-9491-7
SPACE-SCALE THEORY IN EARLY VISION
Tony Lindeberg
ISBN 0-7923-9418
NEURAL NETWORK PERCEPTION FOR MOBILE ROBOT GUIDANCE
Dean A. Pomerleau
ISBN: 0-7923-9373-2
DIRECTED SONAR SENSING FOR MOBILE ROBOT NAVIGATION
John J. Leonard, Hugh F. Durrant-Whyte
ISBN: 0-7923-9242-6

Preface

This book summarizes the results of our modeling-from-reality (MFR) project which took place over the last decade or so. The goal of this project is to develop techniques for modeling real objects and/or environments into geometric and photometric models through computer vision techniques. By developing such techniques, time consuming modeling process, currently undertaken by human programmers, can be (semi-)automatically performed, and, as a result, we can drastically shorten the developing time of such virtual reality systems, reduce their developing cost, and widen their application areas.

Originally, we began to develop geometric modeling techniques that acquire shape information of objects/environments for object recognition. Soon, this effort evolved into an independent modeling project, virtual-reality modeling, with the inclusion of photometric modeling aspects that acquire appearance information, such as color, texture, and smoothness. Over the course of this development, it became apparent that environmental modeling techniques were necessary when applying our techniques to mixed realities that seamlessly combine generated virtual models with other real/virtual images. The material in his book covers these aspects of development.

The project has been conducted while the authors were/are at the Computer Science Department of Carnegie Mellon University (CMU) and the Institute of Industrial Science at the University of Tokyo. Many fellow researchers contributed various aspects of the projects. Raj Reddy, Takeo Kanade, and Masao Sakauchi guided us in conducting this project in the first, middle, and last phases of this project, respectively. Steve Shafer and Shree Nayar were our leaders in photometric modeling. Hideyuki Tamura introduced us to the necessity of environmental modeling.

Several funding agencies supported this project. At CMU, the ARPA Image Understanding program was the main sponsor of this project. A similar role was played at the University of Tokyo by the Shin program, Ministry of Education. Now this project has grown into an independent, JST Ikeuchi CREST program,

with the goal of developing techniques for modeling Japanese cultural heritage objects (as was introduced in the Epilogue of this book).

Publication of this book would not be realized without the editorial help of Daisuke Miyazaki, Toru Takahashi, Yuko Saiki, Jennifer Evans, and Marie Elm. Many thanks go to them.

KATSUSHI IKEUCHI

YOICHI SATO

Introduction

Katsushi Ikeuchi

Virtual reality systems have wide application areas, including 3D catalogues for e-commerce, virtual museum virtual museums, and movie making. The systems are also one of the most important interfaces between human operators and computers in interactive games, flight simulators, and tele-operations.

One of the most important issues in virtual reality research is how to create models for virtual reality systems. Currently, human programmers create those models manually, a tedious and time-consuming job. The model creation period is long, and its developing costs are very high.

Many, if not all, application areas of virtual reality systems have real objects and/or environments to be modeled. For example, a virtual museum often has real objects to be displayed in the museum. 3D catalogues for e-commerce have real merchandise to be modeled and sold through internets. A flight simulator has a real environment in which a virtual plane flies for simulation purposes.

The goal of the modeling-from-reality (MFR) project is to develop techniques for the automatic creation of virtual reality models through observation of these real objects and environments. Recently, the computer vision field has developed techniques for determining shapes of objects and measuring reflectance parameters by observing real objects and environments. The MFR project aims to apply these newly developed methods to VR model creations and to achieve automatic model creations through these techniques. As for the benefits to be gained from this work, MFR will allow us to drastically reduce both programming efforts and developing costs; in turn, the cost reduction will enable us to widen possible application areas.

The MFR spans three aspects as shown in Figure I.1. First, the shape and size of objects should be correctly represented. We will refer to this acquisition of shape and size information from real objects/environments as *geometric modeling*. Geometric modeling, however, provides only partial information for virtual reality models. For final virtual reality models, photometric models, such as color and smoothness, are also necessary. *Photometric modeling* deals

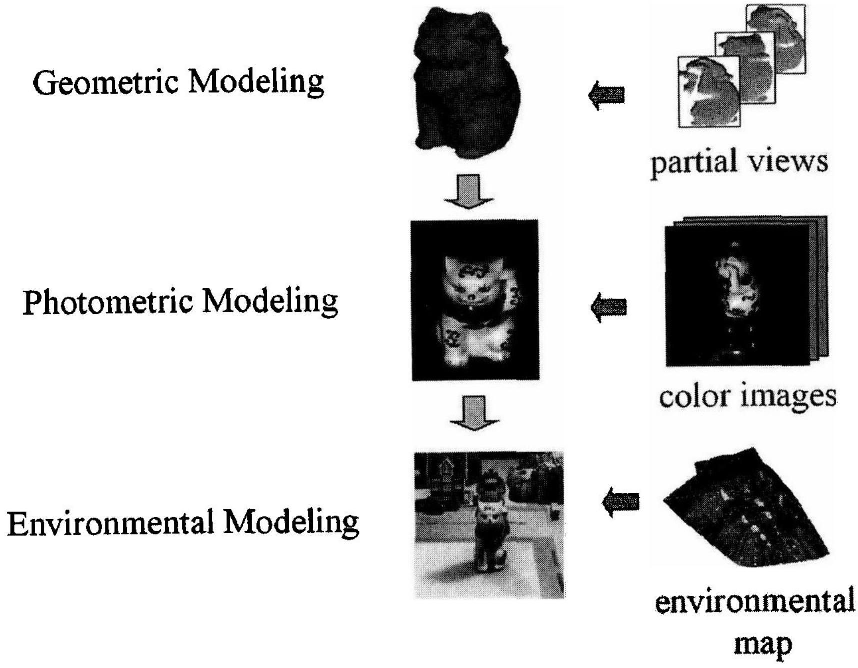

Figure I.1 Three aspects of MFR

with how to create such photometric/appearance models of virtual objects through observation. Further, for seamless integration of virtual objects with real/virtual environments, it is necessary to establish geometric and photometric consistency, including lighting conditions and viewing directions, between them. *Environmental modeling* deals with acquiring such an environmental model of the real background for seamless integration of a virtual object with its background.

Geometric Modeling

Geometric modeling acquires the shape and size of objects through observation. In one sense, the vision community has a long history of geometric modeling research. For example, shape-from-shading [1] and binocular stereo [2, 3] both aim to obtain such shape information from images. Recently, various types of range sensors have also become widely available. These computer vision techniques and range sensors provide a cloud of points that poses their own three dimensional coordinates, or so-called 2-1/2D representations [4, 5].

This cloud of points, however, provides only partial information. A cloud of points representation consists of a set of unstructured points, observed from one single viewing direction. In a cloud of point representation, adjacent points are not always connected to each other. For a complete geometric model, it is necessary to establish connection information among points, i.e. which points are connected to which points, through triangular meshes. Also, one cloud of points is obtained from a single observation, corresponding to only a part of an object. It is necessary to combine those partial data into one single representation corresponding to the whole surface of an object.

Complete geometric modeling requires a three step operation as shown in Figure I.2. The first step is to generate a mesh representation, for each view, from a cloud of points. The second step, the alignment step, is to determine the relative configuration between two meshes from two different viewing directions. Although we can use various means, such as GPS or a rotary table, to determine the sensing configuration, geometric modeling needs far better accuracy in this alignment step; thus, it is necessary to determine the configuration by using image data. Then, using the configuration obtained from this alignment step, we can set all partial mesh representations in one coordinate system. The third and final step is the merging step, to combine these aligned mesh representations into a single consistent representation corresponding to the whole surface of the object. This process is accomplished with the consideration of data accuracy and reliability.

Shum, Ikeuchi, and Reddy, in Chapter 1, propose a method to simultaneously conduct both the second and the third steps, the alignment and merging, by assuming that the object to be modeled consists only of planar faces. First, they segment input range images into planar faces; then, they extract face equations, and establish correspondences among planar faces from different viewing directions. Using these correspondences, they set up an observation matrix. Here the components of the matrix are the equation parameters of the faces. The rows correspond to viewing directions, and the columns correspond to face numbers. By using the weighted least square minimization, Shum et al. decompose this matrix as a product of an interframe-transformation matrix and a face-equation matrix, of which equations are represented with respect to a one world coordinate system.

Higuchi, Hebert, and Ikeuchi, in Chapter 2, describe how they developed an alignment algorithm for free-formed objects. One of the difficulties encountered when handling free-formed objects is that there are no clear entities for establishing correspondences. For example, in the previous chapter, Shum et. al. employ planar faces for correspondence entities for matching. Free-formed objects do not have such convenient units. Higuchi et al. divided a free-formed surface into uniformly distributed triangular patches by using the technique originally developed by Hebert, Delingette, and Ikeuchi [6]. Each

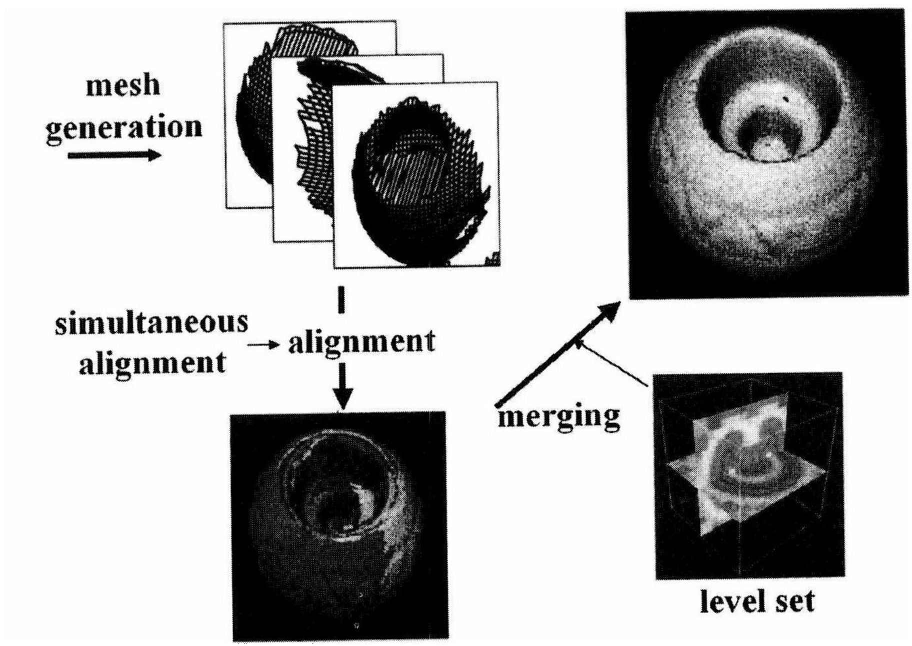

Figure I.2 Three steps for geometric modeling: mesh generation, alignment, and merging triangular patch obtained by this method has roughly the same area and the same topological structure. They use these triangular patches as matching entities. At each triangular patch, they measure color and curvature. For the sake of convenience, they map those curvature and color distributions over the unit sphere, and compare two spherical representations, given from two viewing directions, for establishing correspondences and alignment of views.

Wheeler, Sato, and Ikeuchi, in Chapter 3, continue the discussion of free-formed objects. They propose a robust merging method for creating a triangulated surface mesh from multiple partial meshes. Based on the alignment algorithm discussed in the previous chapter, Wheeler et al. first align all partial meshes in one coordinate system, and convert them into a volumetric implicit-surface representation. From this implicit-surface representation, they obtain a consensus surface mesh using a variant of the marching-cubes algorithm [7]. Unlike previous techniques based on implicit-surface representation [8], their method estimates the signed distance to the object surface by first finding a consensus of local coherent observations of the surface. Due to this consensus operation, the method is very robust against noise existing in the range data.

Photometric Modeling

Photometric modeling aims to acquire the appearance of the object [9, 10]. One of the common methods for representing appearances is the texture mapping that pastes one single texture color at each mesh, usually given from a frontal direction of the mesh. This method is a simple and handy way to acquire the textural appearance of an object. However, because each mesh possesses only one single color value, the method provides neither subtle color differences nor the shift of specular points caused by the movement of the viewer.

In order to generate such appearance differences, the MFR project developed two methods, *model-based* and *eigen-texture rendering*. The model-based rendering method analyzes the surface of an object and extracts reflectance parameters under the assumption of an underlying surface reflectance model. This method is compact and efficient for appearance generation, provided that the surface follows a certain type of reflectance model. For exceptional surfaces that do not follow such typical reflectance models, we have also developed the eigen-texture rendering method. This is an extension of texture mapping. A usual texture mapping pastes only one single texture at each point, while this method pastes all possible textures at each point. Since pasting all the possible textures requires a huge amount of data, we have developed an efficient compression method, which we refer to as eigen-texture rendering.

Both the model-based and eigen-texture rendering methods employ a sequence of color images of an object generated by either the movement of the light source or the object, or both. For an image sequence given by a moving light source, image correspondence is relatively easy, because the relative relation between the viewer and the object never changes during the imaging process, and the same pixel in the image sequence corresponds to the same physical point. For a sequence given by the movement of an object, we first calibrate the movement of the object and the color TV camera, and from this relation we can track image pixels corresponding to the same physical point over the image sequence.

Model-based rendering, described in Chapter 5, estimates surface reflectance parameters at each point of an object. Color variance at each physical point, caused by the different illumination geometry, enables us to separate the surface reflection component from the body reflection component. Here the basic assumption is the Shafer's dichromatic reflection model, which assumes that the reflected light consists of surface and body reflection components [11]. As the result of this separation operation, sequences of body and surface reflection are obtained at each position of an object. The Torrance-Sparrow reflection model is independently applied to both sequences and reflectance parameters are estimated [12, 13]. This method is much more robust than the previous method that directly fit the Torrance-Sparrow models to the data through the non-linear minimization [14].

Figure I.3(b) shows synthesized images with the reflectance parameters obtained by the model-based rendering, while, for comparison, Figure I.3(a) shows original input images. This demonstrates the effectiveness of model-based rendering. The necessary information to be stored is the reflectance parameters at each point on the object surface. The method achieves a quite compact representation of an object.

Model-based rendering can be applied to a class of objects. Model-based rendering employs the dichromatic reflectance model as the underlying assumption; the method cannot be used for those objects that do not follow the dichromatic model. Those excepted objects, which account for 30-40 % of our daily life objects, include clothes and fur. For such classes of objects, Nishino, Sato, and Ikeuchi developed the eigen-texture rendering method.

As does the model-based rendering, the eigen-texture rendering method, described in Chapter 5, also employs a 3D geometric model. Figure I.4 shows an overview of eigen-texture rendering. The method pastes all possible textures, given under either the movement of a light source or the object, or both, onto the 3D surface of the object model. Unlike standard texture mapping, which pastes only a single texture at each point onto the 3D surface, the eigen-texture method pastes all the possible textures at each point. Obviously, this is a large amount of data; but the method compresses those textures, through the eigenspace method. The compression is achieved along the object coordinate system defined on the surface of the 3D geometric model; all the textures are compared and compressed at the same physical position; there is high

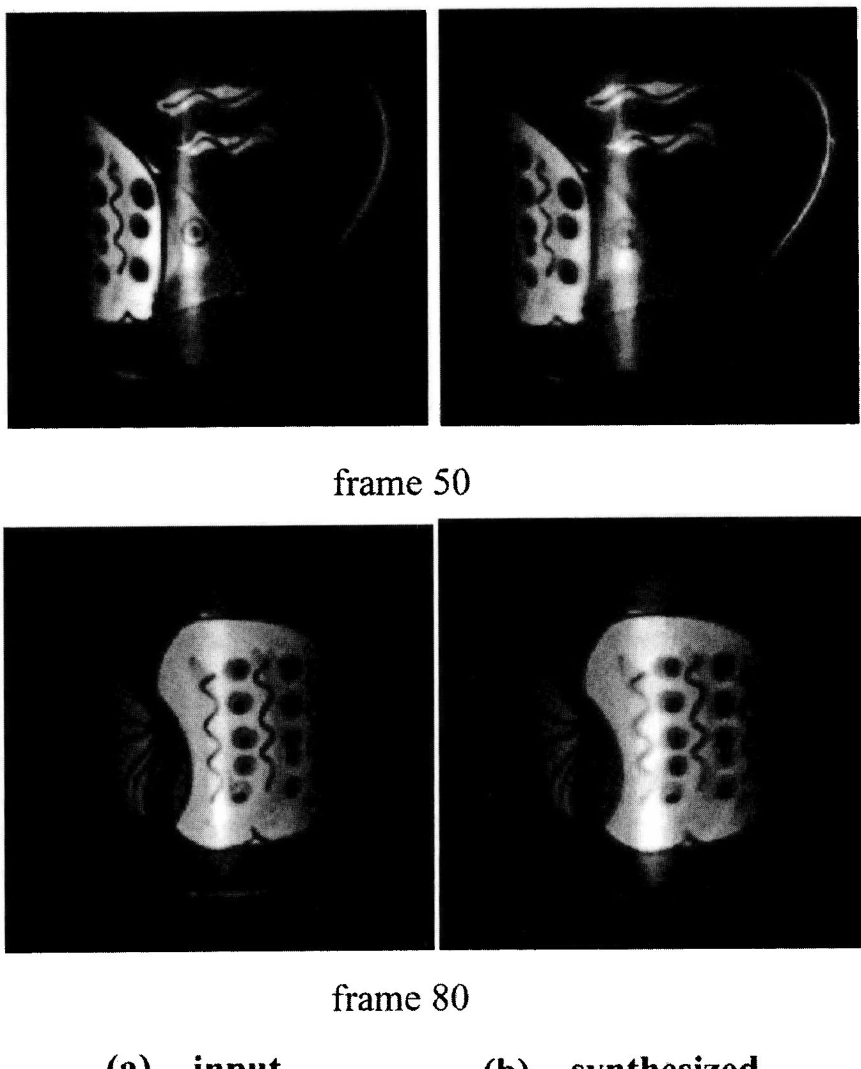

Figure I.3 Real and synthesized images correlation between textures among images - texture difference is due only to the difference of lighting geometry, but the underlying body color is the same. Thus, we can achieve high compression ratio. For example, an image sequence consisting of 360 images can be satisfactorily synthesized using only eight images. Moreover, it is known that, if the surface is Lambertian, only 3 eigen-images are required in order to recover 360 original images.

Environmental Modeling

For most virtual reality systems, it is quite rare for a single virtual object to be displayed alone; rather, a virtual image is often superimposed onto a real or virtual image [15, 16]. For example, in a virtual 3D catalogue, it is preferable to display virtual merchandise on the shelf of a shop in a virtual mall rather than showing it simply floating in air. And it is far better to display virtual pieces of fine art in a virtual museum environment.

Such superimposition requires that consistency between the virtual object and its environment be established in several aspects. One of these aspects is geometric consistency. Both virtual objects and background images are displayed in the same scale, and their coordinate systems are aligned so that the virtual object is displayed in the right position. However, geometric consistency is not enough. In Figure I.5, the two dodecahedra are displayed in the same position. Namely, both images are equivalent in terms of geometric consistency. In the left image, the dodecahedron appears to be floating, while in the right image, it seems to be sitting on the table. The left one does not have shadows, while the right one does; this difference is due to photometric inconsistency. For the human perceptual system, such photometric consistency plays an important role.

For establishing photometric consistency, we have developed two methods: direct and indirect. Sato, Sato, and Ikeuchi describe the direct method in Chapter 6. The direct method measures the illumination distribution of the background environment. A pair of TV cameras fitted with fish-eye lenses acquire images at two different locations as shown in Figure I.6. By using this pair of images, the three dimensional structure of the surrounding environment is constructed using triangularization. Once a rough 3D geometric model of the environment is constructed, the method pastes illumination brightness over the 3D geometric structure to complete the radiance map of the environment. Note that, for soft shadows, not only direct light sources such as incandescent bulbs or florescent lights, but also indirect sources such as walls or ceilings are included in the radiance map. By using the completed radiance map, Sato et. al. established the method for calculating brightness of virtual objects and projected soft shadows from the virtual object to the real background.

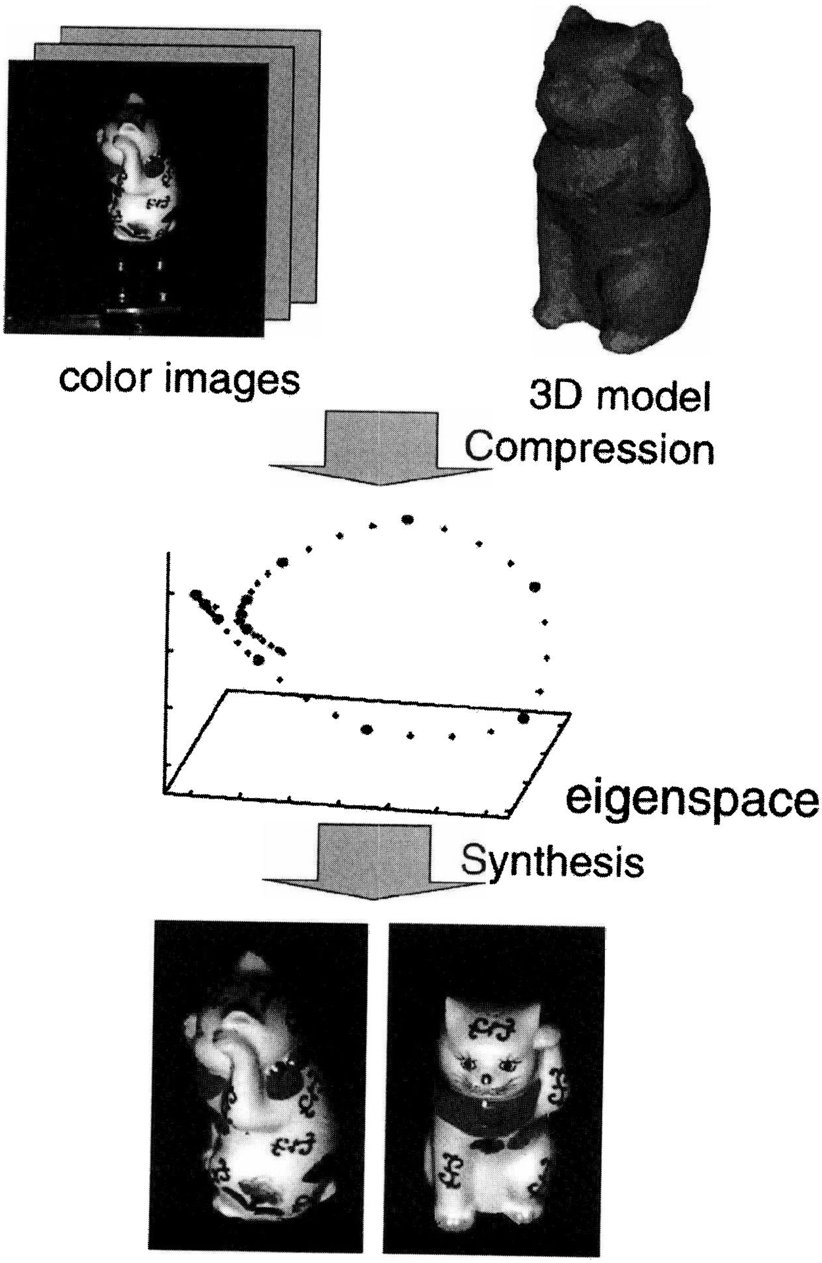

Figure I.4 Eigen-texture rendering

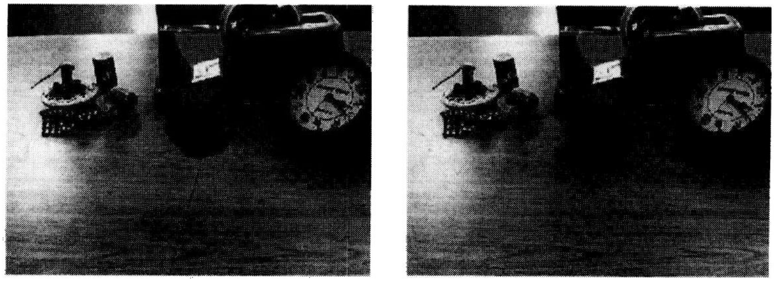

Figure I.5 Two dodecahedra without and with shadows

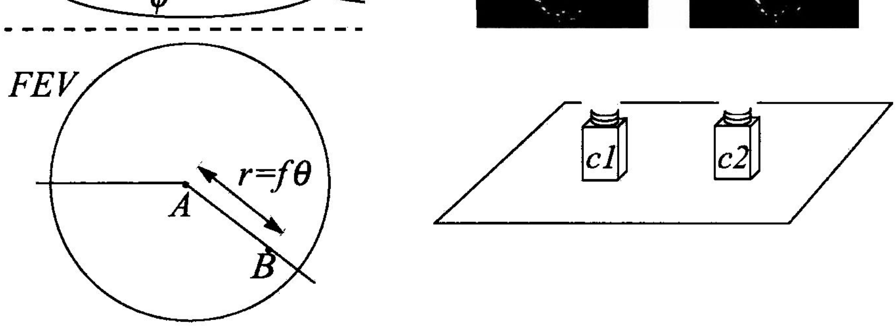

Figure I.6 The direct method for environmental modeling

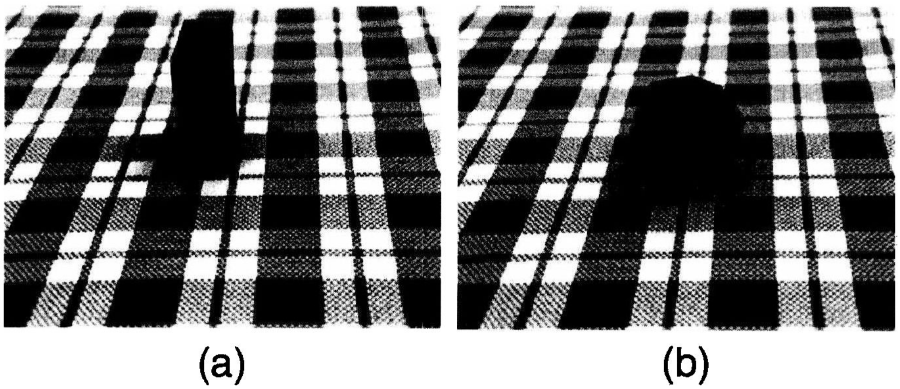

Figure I.7 The result of the indirect method

One of the difficulties of the direct method is that we have to bring such equipment to the real environment. Some modeling tasks require estimating the illumination environment from a given single image to create a seamless image integrated with a virtual object and a real background image. In Chapter 7, Sato, Sato, and Ikeuchi describe the indirect method, which estimates an illumination environment from a given single image. They employ the linearity of the image brightness such that the image brightness of one point is represented as a linear combination of image brightness given from all possible light sources. From this image linearity and the assumption that one object shape in the image is known, Sato et. al. set up a system of linear equations, whose coefficients are known from the shape of the objects, whose independent variables are unknown light source brightness, and whose dependent variables are the observed image brightness at each pixel around the object. By solving the set of linear equations, they estimate the illumination environment of the input image and generate soft shadows around a virtual object superimposed on the image. Figure I.7(a) is the input image. The method estimates the illumination environment from the image brightness around the central object and generates a soft shadow around the virtual object as shown in Figure I.7(b).

In the Epilogue, we present a future direction of the MFR: modeling all Japanese cultural heritage objects through the use of these MFR techniques. As a kick-off project for our efforts, Ikeuchi et al. digitized the great Buddha of Kamakura. The digitization consists of three aspects: how to create geometric models of the great Buddha; how to create photometric models of the great Buddha; and how to integrate such a generated digital Buddha with a virtual main hall of the Buddha, whose real counterpart was destroyed in the 12 century.

Through this project, we have demonstrated effectiveness of these techniques as well as the importance of this line of research.

References

[1] B. K. P. Horm, and M. J. Brooks, *Shape-from-Shading.* MIT Press, Cambridge, MA, 1989.

[2] W. E. L. Grimson, *From Image to Surfaces: a Computational Study of the Human Early Visual System,* MIT Press, Cambridge, MA, 1981.

[3] O. Faugeras, *Three-Dimensional Computer Vision: a Geometric Viewpoint,* MIT Press, Cambridge, MA, 1993.

[4] D. Marr, *Vision,* Freeman, San Francisco, CA, 1982.

[5] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle, "Surface reconstruction from unorganized points," *Proc. SIGGRAPH '92*, pp. 71-78, 1992.

[6] M. Hebert, K. Ikeuchi, and H. Delingette, "A Spherical Representation for Recognition of Free-Form Surfaces," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 17, No. 7, pp.681-690, 1995.

[7] W. Lorensen, and H. E. Cline, "Marching cubes: a high resolution 3D surface construction algorithm," *Proc. SIGGRAPH '87*, pp. 163-169, 1987.

[8] B. Curless, and M. Levoy, "A volumetric method for building complex models from range images," *Proc. SIGGRAPH '96*, pp.303-312, 1996.

[9] S. A. Nayar, K. Ikeuchi, and T. Kanade, "Extracting Shape and Reflectance of Hybrid Surfaces by Photometric Sampling," *IEEE Trans. Robotics and Automation*, Vol. 6, No. 4, pp.418-431, 1990.

[10] K. D. Gremban , and K. Ikeuchi, "Appearance-Based Vision and the Automatic Generation of Object Recognition Programs," *3D Object Recognition Systems*, pp.229-258, A. Jain and P. Flynn (eds.), Elsevier, 1993.

[11] S. A. Shafer, "Using color to separate reflection components," *Color research and Application*, Vol 10, No. 4, pp.210-218, 1985.

[12] K. E. Torrance, and E. M. Sparrow, "Theory for off-specular reflection from roughened surfaces," *Journal of the Optical Society of America*, Vol. 57, pp.1105-1114, 1967.

[13] S. K. Nayar, K. Ikeuchi, and T. Kanade, "Surface reflection: physical and geometrical perspectives," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 13, No. 7, pp.661-634, July, 1991.

[14] K. Ikeuchi, and K. Sato, "Determining reflectance properties of an object using range and brightness images," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 13, No. 11, pp. 1139-1153, 1991.