1

Introduction

One of the primary goals in computer graphics is photorealistic rendering. Much progress has been made over the years in graphics in a bid to attain this goal, with significant advancements in 3D representations and model acquisition, measurement and modeling of object surface properties such as the bidirectional reflectance distribution function (BRDF) and surface subscattering, illumination modeling, natural objects such as plants, and natural phenomena such as water, fog, smoke, snow, and fire. More sophisticated graphics hardware that permit very fast rendering, programmable vertex and pixel shading, larger caches and memory footprints, and floating-point pixel formats also help in the cause. In other words, a variety of well-established approaches and systems are available for rendering models. See the surveys on physically-based rendering [232], global illumination methods [69], and photon mapping (an extension of ray tracing) [130].

Despite all the advancements in the more classical areas of computer graphics, it is still hard to compete with images of real scenes. The rendering quality of environments in animated movies such as *Shrek 2* and even games such as *Ghost Recon* for Xbox 360™ is excellent, but there are hints that these environments are synthetic. Websites such as http://www.ignorancia.org/ showcase highly photorealistic images that were generated through raytracing, which is computationally expensive. The special effects in high-budget movies blend seamlessly in real environments, but they typically involved many man-hours to create and refine. The observation that full photorealism is really hard to achieve with conventional 3D and model-based graphics has led researchers to take a "short-cut" by working directly with real images. This approach is called *image-based modeling and rendering*. Some of the special effects used in the movie industry were created using image-based rendering techniques described in this book.

Image-based modeling and rendering techniques have received a lot of attention as a powerful alternative to traditional geometry-based techniques for image synthesis. These techniques use images rather than geometry as the main primitives for rendering novel views. Previous surveys related to image-based rendering (IBR) have suggested characterizing a technique based on how image-centric or geometry-

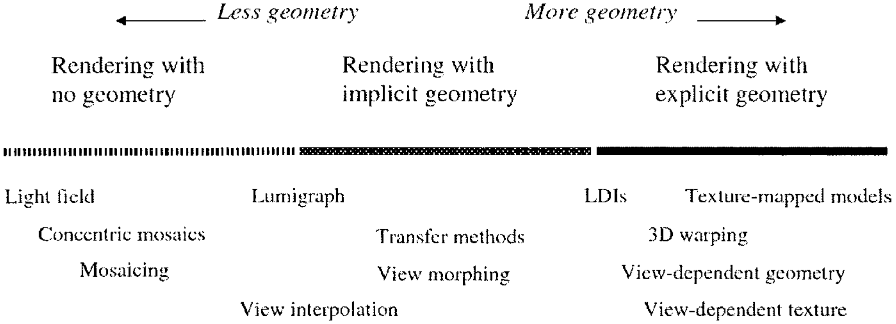

Fig. 1.1. IBR continuum. It shows the main categories used in this book, with representative members shown. Note that the Lumigraph [91] is a bit of an anomaly in this continuum, since it uses explicit geometry and a relatively dense set of source images.

centric it is. This has resulted in the image-geometry continuum (or *IBR continuum*) of image-based representations [155, 134].

1.1 Representations and Rendering

For didactic purposes, we classify the various rendering techniques (and their associated representations) into three categories, namely rendering with no geometry, rendering with implicit geometry, and rendering with explicit geometry. These categories, depicted in Figure 1.1, should actually be viewed as a continuum rather than absolute discrete ones, since there are techniques that defy strict categorization.

At one end of the IBR continuum, traditional texture mapping relies on very accurate geometric models but only a few images. In an image-based rendering system with depth maps (such as 3D warping [189], and layered-depth images (LDI) [264], and LDI tree [39]), the model consists of a set of images of a scene and their associated depth maps. The surface light field [323] is another geometry-based IBR representation which uses images and Cyberware scanned range data. When depth is available for every point in an image, the image can be rendered from any nearby point of view by projecting the pixels of the image to their proper 3D locations and re-projecting them onto a new picture. For many synthetic environments or objects, depth is available. However, obtaining depth information from real images is hard even with state-of-art vision algorithms.

Some image-based rendering systems do not require explicit geometric models. Rather, they require feature correspondence between images. For example, view interpolation techniques [40] generate novel views by interpolating optical flow between corresponding points. On the other hand, view morphing [260] results in-between camera matrices along the line of two original camera centers, based on point correspondences. Computer vision techniques are usually used to generate such correspondences.

At the other extreme, light field rendering uses many images but does not require any geometric information or correspondence. Light field rendering [160] produces a new image of a scene by appropriately filtering and interpolating a pre-acquired set of samples. The Lumigraph [91] is similar to light field rendering but it uses approximate geometry to compensate for non-uniform sampling in order to improve rendering performance. Unlike the light field and Lumigraph where cameras are placed on a two-dimensional grid, the Concentric Mosaics representation [267] reduces the amount of data by capturing a sequence of images along a circle path. In addition, it uses a very primitive form of a geometric impostor, whose radial distance is a function of the panning angle. (A geometric impostor is basically a 3D shape used in IBR techniques to improve appearance prediction by depth correction. It is also known as geometric proxy.)

Because light field rendering does not rely on any geometric impostors, it has a tendency to rely on oversampling to counter undesirable aliasing effects in output display. Oversampling means more intensive data acquisition, more storage, and higher redundancy.

1.2 Sampling

What is the minimum number of images necessary to enable anti-aliased rendering? This fundamental issue needs to be addressed so as to avoid undersampling or unnecessary sampling. Sampling analysis in image-based rendering, however, is a difficult problem because it involves unraveling the relationship among three elements: the depth and texture information of the scene, the number of sample images, and the rendering resolution. Chai *et al.* showed in their plenoptic sampling analysis [33] that the minimum sampling rate is determined by the depth variation of the scene. In addition, they showed that there is a trade-off between the number of sample images and the amount of geometry (in the form of per-pixel depth) for anti-aliased rendering.

1.3 Compression

Because image-based representations are typically image-intensive, compression becomes an important practical issue. Compression work has been traditionally carried out in the image and video communities, and many algorithms have been proposed to achieve high compression ratios. Image-based representations for static scenes tend to have more local coherence than regular video. The issues associated with dynamic scenes are similar for regular video, except that there is now the additional dimensions associated with the camera viewpoint. As a result, image-based representations have a significantly more complicated structure than regular video because the neighborhood of image samples is not just along a single time axis as for regular video. For example, the Lumigraph is 4D, and it uses a geometric impostor. Image-based representations also have special requirements of random access and selective decoding for fast rendering. As subsequent chapters will reveal, geometry has been used as a means for encoding coherency and compressing image-based representations.

1.4 Organization of book

This book is divided into four parts: representations and rendering techniques, sampling, compression, and systems and applications. Each part is relatively self-contained, but the reader is encouraged to read the Part I first to get an overall picture of IBR. In a little more detail:

Part I: Representations and Rendering Techniques

The chapters in this part survey the different representations and rendering mechanisms used in IBR. It starts with a survey of representations of static scenes. In this survey, important concepts such as the plenoptic function, classes of representations, and view-dependency are described. Systems for rendering dynamic scenes are subsequently surveyed. From this survey, it is evident that the design decisions on representation and camera layout are critical. A separate chapter is also devoted to rendering; it describes how rendering depends on the representation and what the common rendering mechanisms are.

Part II: Sampling

This part addresses the sampling issue, namely, the minimum sampling density required for anti-aliased rendering. The analysis of plenoptic sampling is described to show the connection between the depth variation of the scene and sampling density. Three different interpretations are given: using sampling theorem, geometric analysis, and optical analysis. A representation that capitalizes on the sampling analysis to optimize rendering performance (called layered Lumigraph) is also described in this part.

Part III: Compression

To make any IBR representation practical, it must be easy to generate, data-efficient, and fast to render. This part focuses on the sole issue of compression. IBR compression is different from conventional image and video compression because the non-trivial requirements of random access and selective decoding. Techniques for compressing static IBR representations such as light fields and Concentric Mosaics are described, as are those for dynamic IBR representations such as panoramic videos and dynamic light fields.

Part IV: Systems and Applications

The final part of the book showcases four different IBR systems. One system demonstrates how Concentric Mosaics can be made more compact using the simple observation about perception of continuous motion. Another system allows customized layout of representations to large scene visualization so as to minimize image capture. The layout trades off the number of images with the viewing degrees of freedom.

Segmentation and depth recovery are difficult processes—the third system was designed with this in mind, and allows the user to help correct for areas that look perceptually incorrect. This system automatically propagates changes to the user inputs to "pop-up" layers for rendering. Finally, the fourth system allows a light field to be morphed to another through user-assisted feature associations. It preserves the capability of light fields to render complicated scenes during the morphing process.