Virtual Photography – A Framework for Teaching Image Synthesis

Joe Geigel and Nan C. Schaller
Computer Science Department
Rochester Institute of Technology
jmg@cs.rit.edu and ncs@cs.rit.edu

Abstract

The camera has long served as a metaphor for teaching three-dimensional graphics in introductory computer graphics courses. We extend this metaphor to include the complete photographic pipeline as a framework for teaching image synthesis in a second graphics course. We present the correspondence between photographic processes and the areas of study in image synthesis, and discuss the success of using this framework in an image synthesis course at our university for the past two years.

1 Introduction

The Computer Science Department at Rochester Institute of Technology has offered two computer graphics courses since the early 1980s. These courses are co-listed for undergraduates and graduates. Computer Graphics I introduces two-dimensional and three-dimensional graphics including theory, algorithms and the use of an API, currently OpenGL. Since its inception, Computer Graphics II has been a projects course, using a collaborative learning approach. Students would choose a term project, often working on a team. Each student would research a topic pertinent to their project, write a paper about their findings and present them to the class. While students were happy with this approach, they felt that they needed another, more focused, theoretical course between Computer Graphics I and Computer Graphics II. Two years ago, Computer Graphics II was modified to specifically focus on image synthesis and rendering.

The camera has long served as a metaphor for teaching three-dimensional graphics in introductory courses. In the modification of our second course, we extend this metaphor by using the complete photographic pipeline as a framework for teaching image synthesis. In the following sections, we lay out the correspondence between photographic processes and the areas of study in image synthesis, and discuss the success of using this framework in this course at our university for the past two years.

2 Computer Graphics as Virtual Photography

Images are a two-dimensional projection of a three-dimensional world. This is true for computer graphics as well as for photography. As the goals of image synthesis are similar to those of photography, computer graphics has borrowed extensively from photography in developing its rendering pipeline.

Photography begins with a physical scene. A camera is used to capture and focus light from the scene and project it onto film. The film is then processed, producing a print. The structure of the pipeline is essentially the same in computer graphics. However, in computer graphics the scene is virtual, and both the lighting and the processing are simulated. The correspondence of the processes is shown in Figure 1.

Many of the fundamental algorithms in image synthesis involve the simulation of the transport of light within a virtual scene. Even early graphics techniques, i.e., those developed before global illumination, rely on a photographic metaphor. Students are exposed to the graphics rendering pipeline and the camera metaphor in Computer Graphics I. Being that most students are somewhat familiar with rudimentary photography, we feel that building on this knowledge by teaching Computer Graphics II using a photographic framework is a natural way to organize and convey the additional complexities of complete image synthesis.

3 Course Content

An extensive web search has led us to conclude that, although there is an overlap in the topics we cover and those in image synthesis courses taught at other universities, our organization and presentation with an analogy to the complete photographic pipeline is unique. This is true as well with respect to our emphasis on tone reproduction.

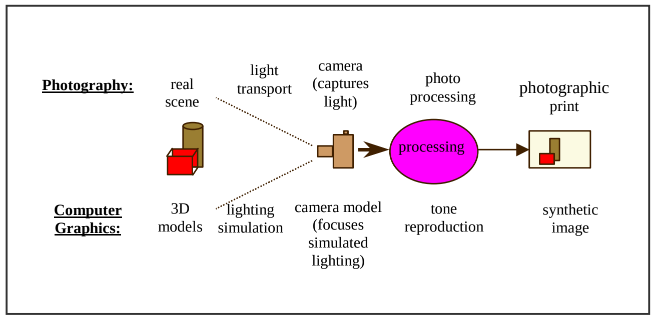

Figure 1. Photographic and image synthesis pipelines.

The course content is divided into four units, each corresponding to an area of the pipeline. These units, along with the topics covered in them, are summarized below.

UNIT 1: Setting the scene. This unit focuses on object modeling and transformations. Some of the topics in this section are review from Computer Graphics I. They not only serve as a refresher for those who have already taken the first course, but also as a means to bring those who have entered the course through a non-traditional path up to speed. Topics in this unit include object modeling (geometric primitives; polygonal models; surfaces, including NURBS; hierarchical modeling; constructive solid geometry; and procedural models, including fractals) and object transformations.

UNIT 2: The camera. Most students will have been exposed to the pinhole camera model as well as perspective and orthographic projections. This unit introduces more advanced camera models, e.g., [Kolb et al. 1995], developed from photographic science. Visual effects, such as focus, depth of field, and motion blur, made possible by using these extended models are discussed. This unit also emphasizes the point sampling nature of the image synthesis process and the problems inherent in this method, namely aliasing.

UNIT 3: Illumination. Photographs are the result of focusing light on film. In computer graphics, photorealistic images are produced by simulating light transport within a virtual scene. In this unit, we explore methods by which this simulation is performed. Topics include the physics of light, color science basics, shading (BRDFs, traditional shading models, e.g., Phong and Cook-Torrance, texture mapping, and procedural shading), global illumination (the rendering equation, ray tracing, radiosity, distributed ray tracing, and photon mapping), and advanced methods such as the two pass global illumination method and REYES.

UNIT 4: Tone Reproduction. The dynamic range of a scene can far exceed the dynamic range of the device that will display its image; thus, it is essential to address the issue of tone reproduction. Indeed, tone reproduction is fundamental to photography, being a driving force in its development. In computer graphics, it has been given increasing attention over the past decade [Devlin et al. 2002]. In keeping with the photographic analogy, tone reproduction must be treated as an integral part of the image synthesis pipeline, rather than just as another advanced topic in computer graphics. Topics in this unit include: introduction to tone reproduction, perceptual models, photographic tone reproduction, and color management.

These units are presented first by focusing on general theory and then by progressing to specific topics and algorithms. For example, in presenting shading and material properties, the theory and definition of bi-directional reflection distribution functions (BRDFs) is covered first. Then, traditional shading models are presented as examples of general BRDFs. When discussing global illumination, the rendering equation is discussed first, then ray tracing and radiosity are presented as example solutions to this equation. The rationale behind this approach is to separate the conceptual components of the framework from any particular implementation. One of our goals is that students be able to use the conceptual framework provided to identify where to "plug in" the algorithms and techniques that are discussed. It should be noted that units may not necessarily be covered in a linear fashion, but rather are visited as is required in preparation for programming assignments.

4. Student Learning

Students are expected to master the fundamental rendering framework and illustrate acquired depth in an advanced technique of their choice. There are four components employed in the course to achieve this goal: lectures, assignments, readings and projects. Each is briefly described below:

4.1. Lecture

The lecture component conveys basic knowledge about image synthesis. Lectures are used to present the material with three different focuses: conceptual, technical, and advanced topics. The conceptually focused lecture provides an overview, explaining the purpose of an area of the pipeline or of a class of algorithms. The lectures with a technical focus present the fundamental techniques or algorithms of an area, including the details required to complete programming assignments. The advanced topic lectures provide a survey of more sophisticated techniques, intended to provide breadth, as well as motivation for student projects.

4.2. Programming Assignments

As it is our firm belief that the best way to fully understand a technique in computer graphics is to program it, this is the area of our course that has evolved the most and, is indeed, still evolving.

The first offering of the revised course maintained the single term project approach, using a final examination to assess learning from lecture topics. However, since lecture topics were not necessarily related to their projects, students found the final examination jarring. Choosing examination questions was difficult as well: Should they be easy enough to answer without reinforcing learning activities, or difficult enough to require extensive and intensive study? As neither seemed reasonable, programming assignments replaced the examination in the offering that followed as a mechanism to reinforce the concepts discussed during lecture.

Initially, the elements of the (virtual) photography pipeline were discussed in order. This naturally placed most of the programming assignment topics in the last half of the term. Having all programming assignments due then caused a severely unbalanced workload. Topic order was then adjusted to allow meaningful programming assignments earlier in the course. The next pass added some intermediate checkpoints for the main programming assignment. The current offering is fine-tuning the topic order and adding even more checkpoints to facilitate the alignment of lecture topics and assignments.

The main programming assignment is designed to step students through the image synthesis pipeline with the goal that they will understand it well enough to code it. Like the course presented in [Schweitzer 1990], we choose ray tracing to motivate this process. This assignment is divided into a number of checkpoints; each checkpoint builds on the previous and focuses on a particular topic. Collectively, the goal of this assignment is to reproduce the classic image first seen in the seminal ray tracing paper [Whitted 1980] and shown in Figure 2, and to experiment with tone reproduction operators.

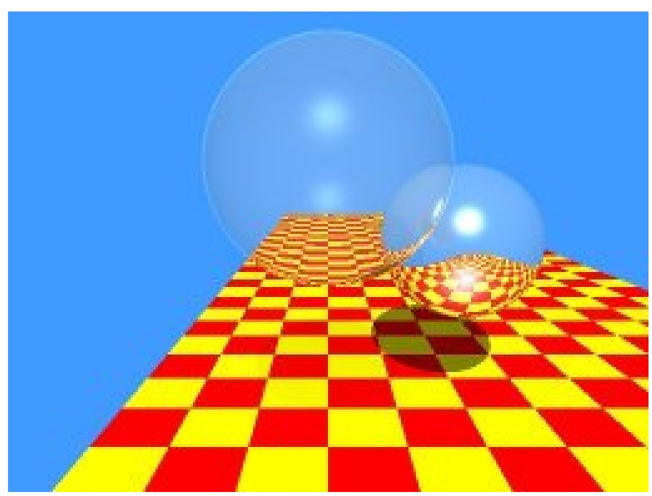

Figure 2. A classic raytraced image.

The checkpoints for this programming assignment follow:

1) SETTING THE SCENE – Construct the scene using a three-dimensional API. The goal is to determine the correct locations and transformations of the objects, as

well as the camera parameters necessary to produce the model for this image.

2) CAMERA – Using the mathematics of ray geometry, cast rays using a simple perspective pinhole camera model for visible surface determination.
3) SHADING – Add basic Phong illumination.
4) PROCEDURAL SHADING – Implement the checkerboard floor as a procedural texture.
5) REFLECTION – Recursively trace the reflection rays.
6) TRANSMISSION – Recursively trace the transmission rays.
7) TONE REPRODUCTION – Scale the pixel values of the ray-traced image by three different maximum scene luminances, and apply two different tone reproduction operators [Ward 1994, Reinhard et al. 2002], to the result. Explain and compare the effects of the operators.

With the recent advances in and focus on procedural shading, students are provided the opportunity to further experiment with procedural shading in the second assignment. Noting the successes presented in [Owen 1992], we chose RenderMan™ as the means for exploring this topic. In this assignment, students are asked to create three simple shaders using the RenderMan™ shader language [Upstill 1990], either from scratch or by modifying existing shaders, and to apply them to a simple scene. They are encouraged to go beyond this with bonus points being awarded to that student who produces the "best image", selected by a vote of the class. Figures 3 and 4 show two recent winning images.

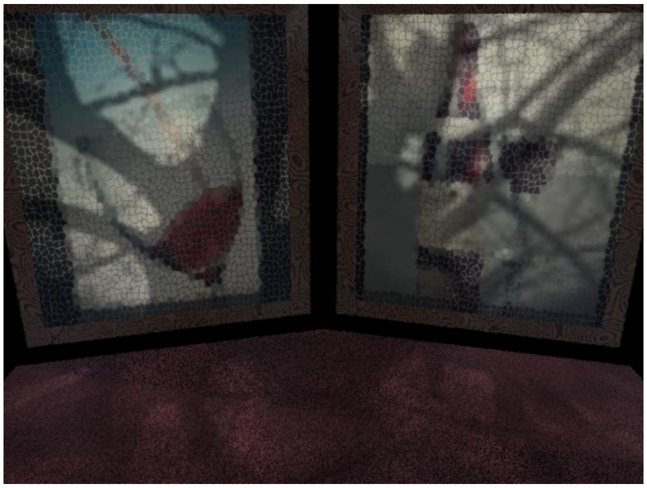

Figure 3. Shader assignment winner.

4.3 Projects

Whereas the programming assignments are designed to give the students a firm foundation in the basic rendering pipeline, the project component allows students to explore an advanced topic in depth. Projects are proposed at the beginning of the term and worked on throughout. The final product includes a report and a presentation. These projects can be done in groups. To help those students who do not have an idea for their project, suggestions are solicited from faculty from other departments. In the recent past, faculty members in the School of Design, Imaging Science, and Chemistry, among others, have mentored such interdisciplinary projects.

4.4 Readings

We do not require a textbook. Instead, we provide copies of notes and an extensive required reading list consisting of seminal graphics papers. Supplemental textbooks are available on reserve at the campus library. Details for the lectures come from these required readings as well as from the supplemental texts.

Graduate students are required to summarize a paper of their choice from the reading list for each class. Undergraduates are invited to do the same for extra credit.

The required readings serve three purposes. First, they reinforce the material presented in lecture. Secondly, they provide motivation and ideas for student projects. Finally, they introduce students to the graphics literature and help them learn to comprehend technical papers from scholarly journals.

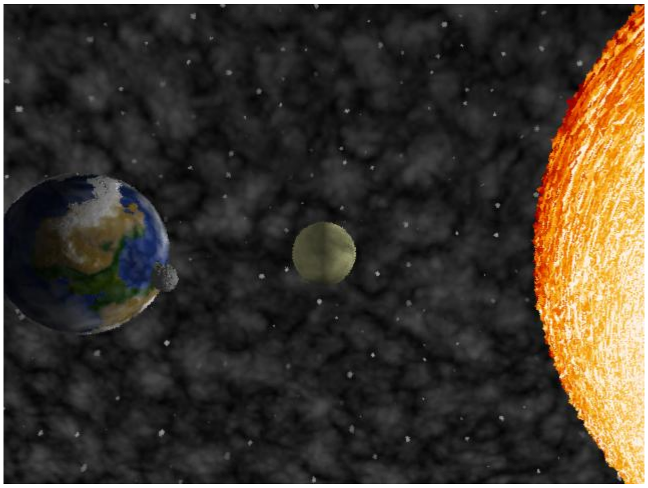

Figure 4. Shader assignment winner.

5 Results

Student interest in the course has increased with the change in format. Formerly the course was offered twice a year (winter and summer), but due to popular demand, it is now offered each quarter. While not all of the lecture material may be pertinent to each student project, it serves to provide a complete overview of image synthesis; something that was lacking in the projects only course.

Both the ray tracing and RenderMan™ assignments generate a lot of enthusiasm. The students appreciate seeing the ray traced image come together. Figures 3 and 4 indicate their response to the RenderMan™ challenge!

The required reading assignments brought some interesting comments. One graduate student said: "The paper summaries are the best part of the course." An undergraduate, who did them for extra credit, stated: "I like that I can now take a research paper and understand it, and if I want to implement the authors' ideas into a project I am doing, I can!"

The favorite component of the course, however, is the project. Many students indicate that the best thing about the course is the "freedom to focus on what we are personally interested in." The project presentation days are the most anticipated classes as the projects are generally quite interesting and well done. Student enthusiasm for their work is contagious!

Framing the project in the context of the complete photographic pipeline has clearly affected the definition and direction of the projects. Some students chose to extend programming assignments, such as adding caustics to the ray tracer as shown in Figure 5. Others are defined outside of the scope of the assignments yet clearly fall somewhere within the rendering framework. For example, one student's urban landscape builder, *City Builder*, shown in Figure 6, explored the modeling area of pipeline.

Work on projects often continues after the course is completed. For example, *City Builder* grew into an open source project (*http://citybuilder.sourceforge.net/*), freely available on the web. A three-person team collaborated with eleven others to produce *MegaMonkey* (*http://www.rit.edu/~jpw9607/m3/)*, a game based on their project, that was entered in the annual IGDC student gaming competition. For others, projects form the basis for further exploration through independent study or through interdisciplinary collaboration with other RIT departments

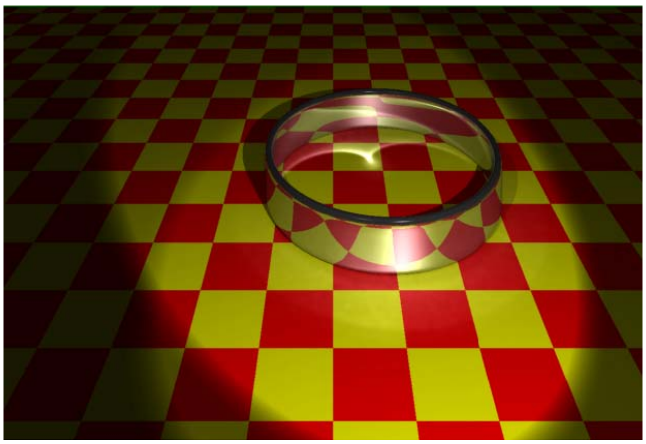

Figure 5. Adding caustics to the ray tracing assignment.

6 Conclusions

We have found that virtual photography provides an effective framework for teaching image synthesis. The mix of learning components, presented in the context of the basic and familiar photographic framework has provided a rich, intuitive, and exciting learning experience for students.

Course materials for the most recent offering of the course are available at ***http://www.cs.rit.edu/~graphics/cgII***. We invite others to try it and provide us with feedback. Student excitement and involvement in the course has encouraged us to further extend our graphics course offerings to include a course focusing on the algorithms and techniques used in computer animation and another focusing on theatrical applications of virtual reality.

Acknowledgements

We want to express our appreciation to Pixar for donating their suite of RenderMan™ software, and to our students, Matthew Bellinger and Patrick McCormick (Figure 3), Kevin Pazirandeh (Figure 5), Kristina Rozanski (Figure 4), and Andrew Sampson (Figure 6) for allowing us to share their images.

Figure 6. Screenshot from *City Builder.*

References

DEVLIN, K., CHALMERS, A., WILKIE, A., PURGATHOFER, W., 2002, "Tone Reproduction and Physically Based Spectral Rendering", In *State of the Art Reports, Eurographics 2002*, Dieter Fellner and Roberto Scopignio, editors, pp. 101-123.

KOLB, C., MITCHELL, D., HANRAHAN, P., 1995, "A realistic camera model for computer graphics", *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pp. 317-324.

OWEN, G., 1992, "Teaching computer graphics using RenderMan", *SIGCSE Bulletin*, Volume 24, Issue 1, Proceedings of the twenty-third SIGCSE technical symposium on Computer science education, pp. 304– 08, Kansas City, Missouri, United States.

REINHARD, E., STARK, M., SHIRLEY, P., FERWERDA, J., 2002, "Photographic tone reproduction for digital images", *ACM Transactions on Graphics (TOG), Proceedings of the 29th annual conference on Computer graphics and interactive techniques,* Volume 21, Issue 3, pp. 267–276, July 2002.

SCHWEITZER, D., 1990, "Ray tracing: a means to motivate students in an introductory graphics course", *SIGCSE Bulletin*, Proceedings of the twenty-first SIGCSE technical symposium on Computer science education, Volume 22, Issue 1, pp. 157 – 161, February 1990.

UPSTILL, S., 1990, The RenderMan™ Companion, Addison-Wesley.

WARD, G., 1994, "A Contrast-based Scalefactor for Luminance Display", *Graphics Gems IV*, P.S. Heckbert (Ed), Boston: Academic Press Professional, pp. 415-421.

WHITTED, T., 1980, "An Improved Illumination Model for Shaded Display", *Communications of the ACM*, pp. 343-349, June 1980.