An Interactive Introduction to OpenGL Programming

Dave Shreiner, Course Organizer
shreiner@siggraph.org

November 28, 2000

Course Level

Beginner

Proposed Length

We propose the course for a full-day. Ideally, we would like to be able to utilize the classroom section of the Creative Applications Laboratory (CAL) for our presentation to make it interactive for the students. Alternatively, we are prepared to present in the more traditional non-interactive classroom, while still making our interactive exercises available in the CAL.

In order to properly provide due attention to all of the relevant topics, the course truly needs to be presented as a full-day course.

Summary Statement

OpenGL is the leading computer graphics programming library. This course provide an introduction to almost all of OpenGL features including drawing images and 3D objects, simulating lighting, and texture mapping objects.

By the end of the day, students will be able to author simple interactive animated OpenGL programs.

Expanded Statement

OpenGL is the mostly widely supported programming library in the computer graphics industry. It is used in almost every area of computer graphics: academic research, scientific visualization, interactive entertainment, computer-aided design and engineering, gaming, and many more.

This course provides a through introduction to programming with the OpenGL API, including advanced topics like simulating light effects and texture mapping, by utilizing interactive tutorial programs and techniques, in addition to reviewing source code and running demo programs. By the end of the session, participants will be able to create interactive OpenGL programs that render animated, shaded objects in real-time.

Prerequisites

Our only prerequisite is that students should be able to read simple computer programs written in the "C" computer language.

Topics List

Our course will present a wide variety of concepts from basic computer graphics theory presented in the context of OpenGL. The topics we intend to cover include:

- Geometric rendering primitives, and how the can be assembled into 3D objects.
- Matrix operations for virtual camera manipulation (viewing and projection transformations) and modeling transformations.
- Animation of objects and interactive program manipulation.
- Depth (z) buffering.
- Simulating lighting effects for geometric objects.
- Texture mapping of geometric objects, and utilizing texture mapping for simple image manipulation.
- Anti-aliasing.
- Using the accumulation and stencil buffers for advanced rendering techniques.
- Image blending and simple image processing techniques.

We also intend to utilize the OpenGL Utility Toolkit (`GLUT`) for the creation of our demonstration programs and tutorials to remove any specific machine or window system dependencies.

Speakers

Name	Title	E-mail
Vicki Shreiner	Technical Education Specialist	`vshreiner@sgi.com`
Ed Angel	Professor of Computer Science	`angel@cs.unm.edu`
Dave Shreiner	OpenGL Development Engineer	`shreiner@siggraph.org`

Course Syllabus

Time	Speaker	Topic
8:30 AM	Shreiner, D.	Welcome and Introduction
8:35 AM	Shreiner, V.	OpenGL Overview & Rendering Basics • Overview of the OpenGL System • A minimal OpenGL program utilizing `GLUT` • OpenGL's rendering primitives • *Interactive exercise: Shapes* • Shading models
9:30 AM	Angel, E.	Projection, Viewing, and Modeling Transformations • Virtual camera analogy • Projection matrix transformations • *Interactive exercise: Projection*
10:00 AM		*Break*
10:15 AM	Angel, E.	Projection, Viewing, and Modeling Transformations (cont'd) • Viewport, aspect ratios, and window resizes • Viewing and modeling matrix transformations • *Interactive exercise: Viewing*
10:45 AM	Shreiner, V.	Depth Buffering • Requesting a depth buffer • Enabling the depth test • Depth buffer and projection transformations
11:00 AM	Shreiner, D.	Lighting • Lighting principles • Material properties • *Interactive exercise: Material Properties* • Lighting characteristics • *Interactive exercise: Lighting Properties* • Attenuation and spotlights
12:00 PM		*Lunch*
1:30 PM	Shreiner, V.	Animation & User Interaction • Animation variables • Double buffering • User input – mouse, keyboard
2:00 PM	Angel, E.	Texture Mapping • Texture mapping principles • Specifying texture maps for OpenGL • Texture coordinates • *Interactive exercise: Texturing* • Texel filtering modes
3:00 PM		*Break*
3:15 PM	Shreiner, D.	Rendering Images and Image Processing • Dealing with image file formats • Positioning images • Image manipulation • Image processing using texture mapping

Course Syllabus (cont'd)

Time	Speaker	Topic
3:45 PM	Shreiner, V.	Pixel Blending • Blending principles • Blending Modes • *Interactive exercise: Blending*
4:10 PM	Angel, E.	Using the Accumulation Buffer • Requesting an accumulation buffer • Accumulating images • Viewing the result • *Interactive exercise: Accumulation*
4:45 PM	Shreiner, D.	Using the Stencil Buffer • Requesting a stencil buffer • Creating arbitrary shaped pixel masks
5:20 PM	All	Summary and Questions & Answers
5:30 PM		*Close*

Course History

An Interactive Introduction to OpenGL Programming has been presented in various forms at SIGGRAPH for the last five years. Originally, a half-day version of the course was presented by Kathleen Danielson and Tom McReynolds in 1996 in a lecture only format. In 1997, the course was presented again as a half-day with Ed Angel replacing Tom McReynolds.

In 1998, Mason Woo (organizer) and Dave Shreiner presented the first interactive version of the course utilizing the first set of tutorials (originally created by Nate Robins). This presentation was also lecture only, and conducted in a hurried half-day.

In 1999, Ed Angel, Dave Shreiner (organizer), and Mason Woo, made the first presentation of the interactive version of the course in the CAL. In order to allow the course attendees to have some time to personally interact with the tutorials, the course was expanded to a full day.

At SIGGRAPH 2000, Ed Angel, Vicki Shreiner, and Dave Shreiner (organizer) presented the course in the CAL. The course was very well received, even though some attendees where unable to utilize the tutorials until late afternoon due to networking problems in the CAL.

As OpenGL has become the *lingua franca* for creating applications in almost every field that utilizes interactive computer graphics, demand for the course has been consistently high year to year. At SIGGRAPH 2000, hopeful students were turned away due to limited seating in the CAL, and almost every seat was filled at the end of the day. Additionally, here are some comments from students who attended last year's presentation:

> "Good overview of OpenGL basics. Cool snippets and interactive tutorials, especially helpful-made the course much better."
>
> "Demo programs were very useful! This is one of the few courses where I felt the course notes were worth the money."
>
> "Great course! Good jump start to openGL. Will be able to utilize knowledge immediately."

For this year, we're hoping to be able to utilize new tutorials demonstrating additional useful features of OpenGL. As

always, the source code and executables for these tutorials, as well as other demonstration programs and source code, will be made available on the Courses CD-ROM.

Continuity Statement

Consistent presentation is essential for the students to be able to reap the most benefit from the course. Each of our speakers is a professional educator with experience and training in presentation skills and course design. Additionally, this team has worked together multiple times in the past, presenting the course not only to SIGGRAPH audiences, but also at Game Technology 2001 and other venues.

As important as the interactive presentation is, we strive to produce a set of course notes that are easy to follow during the class, and useful as a reference long after SIGGRAPH. Our course notes are reviewed, reorganized, and refined from year to year to include new material, smooth over trouble topics, and keep up-to-date with OpenGL changes. Additionally, we utilize the Microsoft Powerpoint templates provided by SIGGRAPH for the formatting of our notes to provide a consistent presentation from chapter to chapter. Finally, the organizer does the final integration and editing of the other presenters materials to guarantee consistency in form and flow of the notes.

A sample page from our notes in included in the **Special Notes** section.

Course Presenter Biographies

Dave Shreiner

Member of the Technical Staff
SGI

E–mail	shreiner@siggraph.org	c/o SGI
Work Phone	(650) 933-4899	1600 Amphitheatre Pkwy.
Home Phone	(650) 969-7750	M/S 590
Fax Number	(650) 932-0681	Mountain View, CA 94043

Biography

Dave Shreiner is a Member of the Technical Staff at SGI, in the OpenGL development group, and an adjunct professor at the Santa Clara University in the computer engineering department. He's also the co-author of the *OpenGL Programming Guide* (Third Edition, Addison Wesley, 1999), and editor of the *OpenGL Programming Manual* (Third Edition, Addison Wesley, 2000). In addition to presenting courses about OpenGL at SIGGRAPH and other conferences, Dave has 13 years of experience developing interactive graphics applications, including 10 years at SGI. Dave has a bachelors degree in Mathematics from the University of Delaware.

Ed Angel

Professor of Computer Science and Computer and Electrical Engineering
University of New Mexico

E–mail	angel@cs.unm.edu	Computer Science Department
Work Phone	(505) 277-6560	University of New Mexico
Home Phone	(505)266-6036	Farris Engineering Center, Rm 157
Fax Number	(505) 277-6927	Albuquerque, NM 87131

Biography
Ed Angel is a Professor in the departments of Computer Science, and Computer and Electrical Engineering at the University of New Mexico. He has over 25 years of experience in research and teaching in computer graphics and image processing. He is the author of the recent textbook: *Interactive Computer Graphics: A Top-Down Approach using OpenGL* (Second Edition, Addison Wesley, 2000). He has taught over 100 professional short courses worldwide, including at SIGGRAPH.

Vicki Shreiner
Technical Education Specialist
SGI

E–mail	vshreiner@sgi.com	c/o SGI
Work Phone	(650) 933-6706	1600 Amphitheatre Pkwy.
Home Phone	(650) 969-7750	M/S 138
Fax Number	(650) 932-0681	Mountain View, CA 94043

Biography
Vicki Shreiner is an Technical Education Specialist at SGI in the Technical Education Department. She has over 16 years of experience in teaching computer-related technical topics at both SGI and Hewlett Packard. Vicki has presented numerous courses on OpenGL at SIGGRAPH and other forums. Vicki has a bachelors degree in Mathematics and Computer Science from Lebanon Valley College.

Special Notes

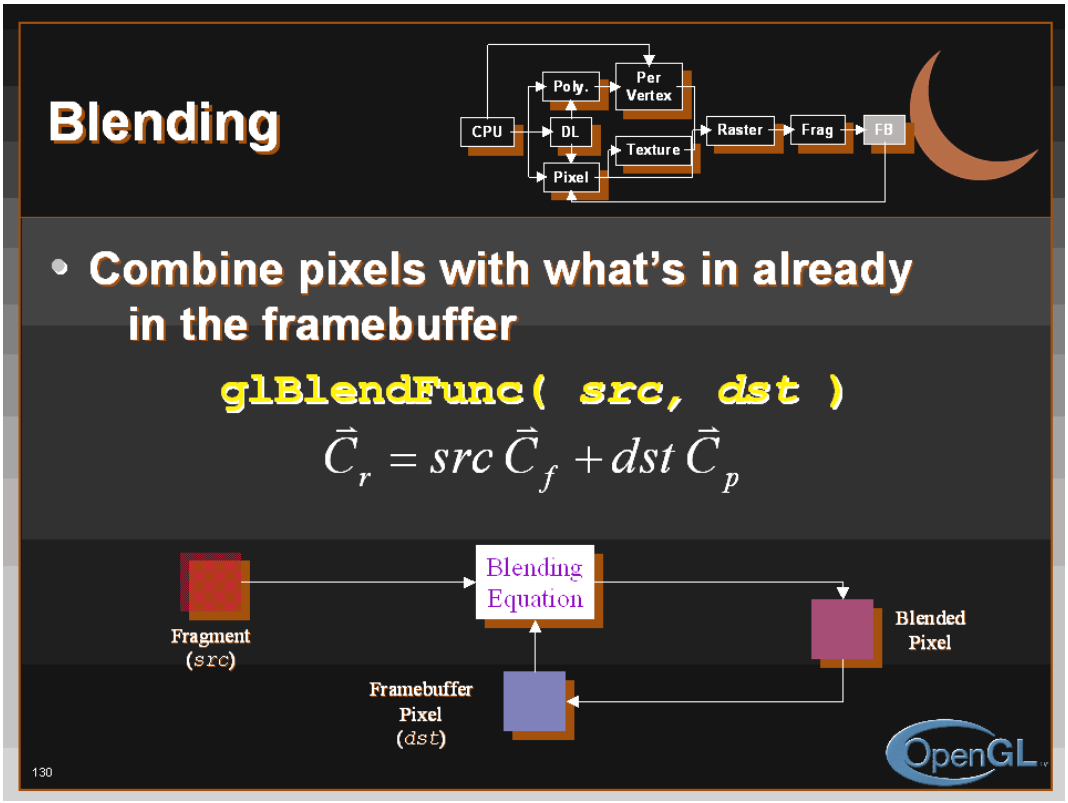
Speaker Commitments

Dave Shreiner has also submitted a proposal for a short tutorial (course_0028 - *Performance OpenGL: platform-independent techniques*) and is included as a speaker for a half-day course (authonly_0123 - *Scalable Interactive Graphics Software for Clustered Architectures*).

Course Note Format

As mentioned previously, we strive to make our course notes easy to follow along with in class, as well as a reference after SIGGRAPH. We use the Microsoft Powerpoint template provided by SIGGRAPH for our presentation. For each content page, we include notes to help clarify the bullet points on the slide.

For example, here's an example slide from our SIGGRAPH 2000 course notes:



Additionally, here is the text that was included as the supplemental note for the slide:

Blending combines fragments with pixels to produce a new pixel color. If a fragment makes it to the blending stage, the pixel is read from the framebuffer's position, combined with the fragment's color and then written back to the position.

The fragment and pixel each have a factor which controls their contribution to the final pixel color. These blending factors are set using `glBlendFunc()`, which sets the source factor, which is used to scale the incoming fragment color, and the destination blending factor, which scales the pixel read from the framebuffer. Common OpenGL blending factors are:

```
GL_ONE                GL_ZERO
GL_SRC_ALPHA          GL_ONE_MINUS_SRC_ALPHA
```

They are then combined using the *blending equation*, which is addition by default.

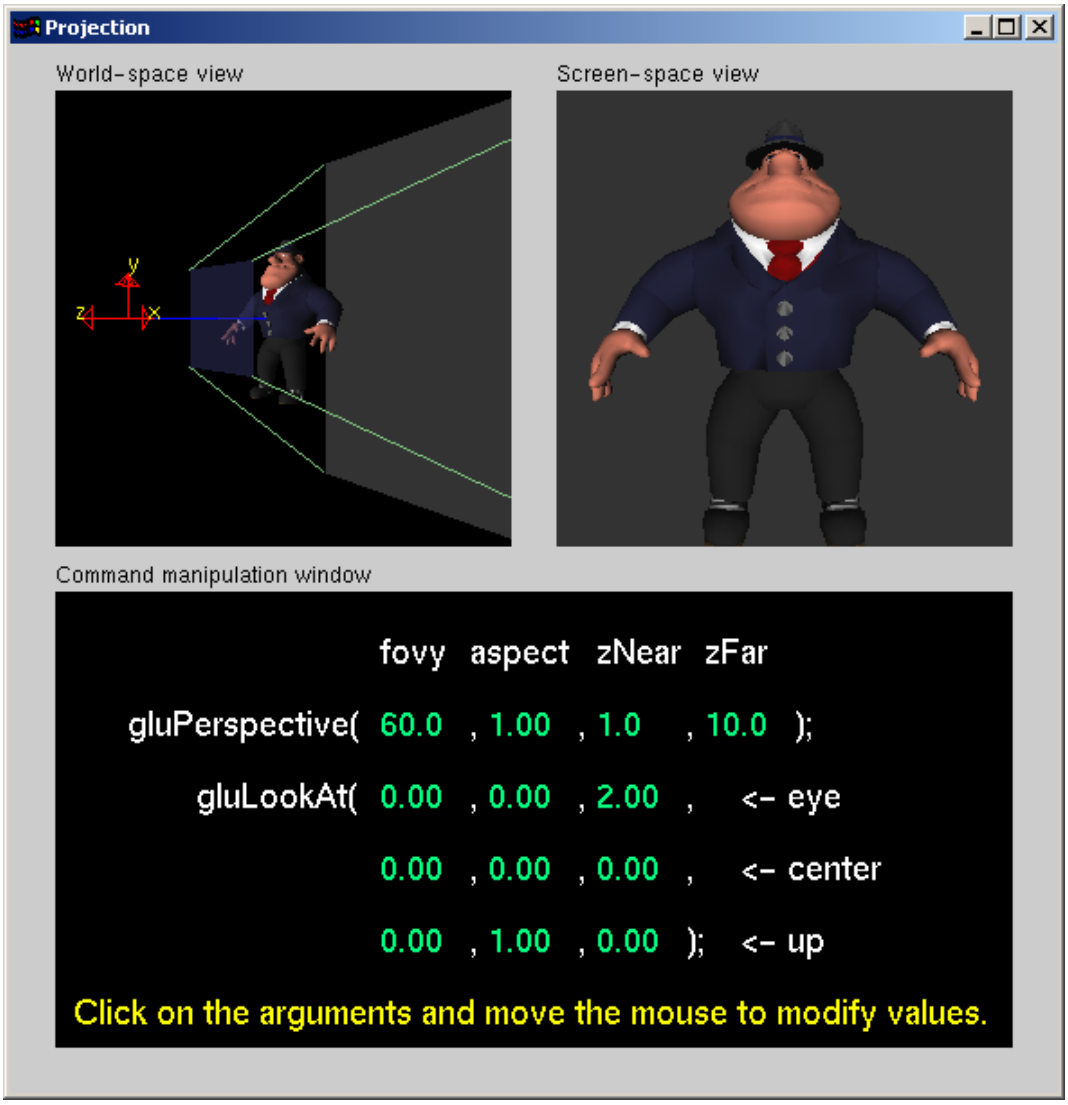
Blending is enabled using `glEnable(GL_BLEND)`.

Note: If your OpenGL implementation supports the `GL_ARB_imaging` extension, you can modify the blending equation as well.

Presentation Techniques and Equipment Requirements

As compared to most courses at SIGGRAPH, where the attendee basically attends a lecture for some number of hours, we hope to make the process of being introduced to OpenGL more interactive and intuitive. To that end, we are hoping to be able to utilize the CAL for our presentation. We have set aside over an hour for the attendee to use our tutorials during the course. As mentioned, shortening the course to a half-day format would remove the possibility for interactivity, as well as forcing the removal of some topic areas.

Below is a screen shot of one of the tutorial programs that we use in our course. This particular tutorial demonstrates the OpenGL 3D projection routines, and the effects of their parameters.



Fortunately, due to the operating and windowing system independence OpenGL, we do not have any specific equipment requirements that need to be met.