

Reflections, Shadows and Billboarding

February 16th 2009

MAE 574, Virtual Reality Applications and
Research

Instructor: Govindarajan Srimathveeravalli

The Buffers

- Rasterization readies fragments (collection of pixels based on polygon etc. info)
 - Series of tests conducted before actual pixel rendering
- Buffer :- Stores the same coherent information for all pixels from the application
 - Color
 - All drawing takes place here
 - Physically accessible from code
 - Typically contains
 - Front (left-right) and Back (left-right)
 - This arrangement needed for stereo viewing
 - Not always available/supported though
 - Depth
 - Aka Z Buffer, holds depth information for over writing of pixel information
 - Always wrt current eye position
 - Stencil
 - Provides a 'stencil' (duh) which allows drawing directly to parts of the scene without affecting the rest
 - Accumulation
 - Similar to color buffer, holds RGBA information
 - Can be used for drawing a 'composite' image

Clearing the Buffers

- `glClearColor`
 - RGBA
- `glClearIndex`
 - For Index mode coloring
- `glClearDepth`
 - Depth value
- `glClearStencil`
 - # of bytes you want cleared from the buffer, usually set to 0
- `glClearAccum`
 - RGBA
- `glClear`
 - Set the specific bit which clears the buffer to any of the above set values

Buffer Operations

- Reading/Writing
 - F.e. color buffer
 - `glReadBuffer(GL_FRONT etc.)`
 - `glDrawBuffer()`
- Masking
 - `glIndex/Color/Depth/StencilMask()`
 - Color and Depth accept a bool to enable drawing that particular component or not
 - Stencil and Index use an int to specify if a bit is to be drawn into the buffer
- Enable/Disable
 - F.e. Depth buffer for drawing translucent objects
- Example :- Drawing complex objects in background as images

Stereoscopy – aka 3D

- Generation of “apparent” 3D images
 - I.e. simulate Binocular disparity
- Multiple ways
 - Visual cues
 - Perspective, relative size, detail, occlusion and lighting/shadows
 - Stereo pairs, Holography and Autostereo

Code courtesy Paul Bourke

```
glMatrixMode(GL_PROJECTION);
glLoadIdentity();
glFrustum(left,right,bottom,top,near,far);
glMatrixMode(GL_MODELVIEW);
glDrawBuffer(GL_BACK_RIGHT); glClear(GL_COLOR_BUFFER_BIT |
    GL_DEPTH_BUFFER_BIT);
glLoadIdentity();
gluLookAt(posn);
MakeLighting();
MakeGeometry();
```

```
glMatrixMode(GL_PROJECTION);
glLoadIdentity();
glFrustum(left,right,bottom,top,near,far);
glMatrixMode(GL_MODELVIEW);
glDrawBuffer(GL_BACK_LEFT);
glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
glLoadIdentity();
gluLookAt(posn);
    MakeLighting();
MakeGeometry();
```

Analglyph Stereo

- Here R = Accumulated red (right eye), G = 0, and B = Accumulated blue (left eye)
- Use glColorMask (True, false, false, false) etc., before each pass of rendering.

Tests on Fragments

- Scissor
- Alpha
- Stencil
- Depth
- Blending
- Dithering

Scissor, Alpha and Depth Testing

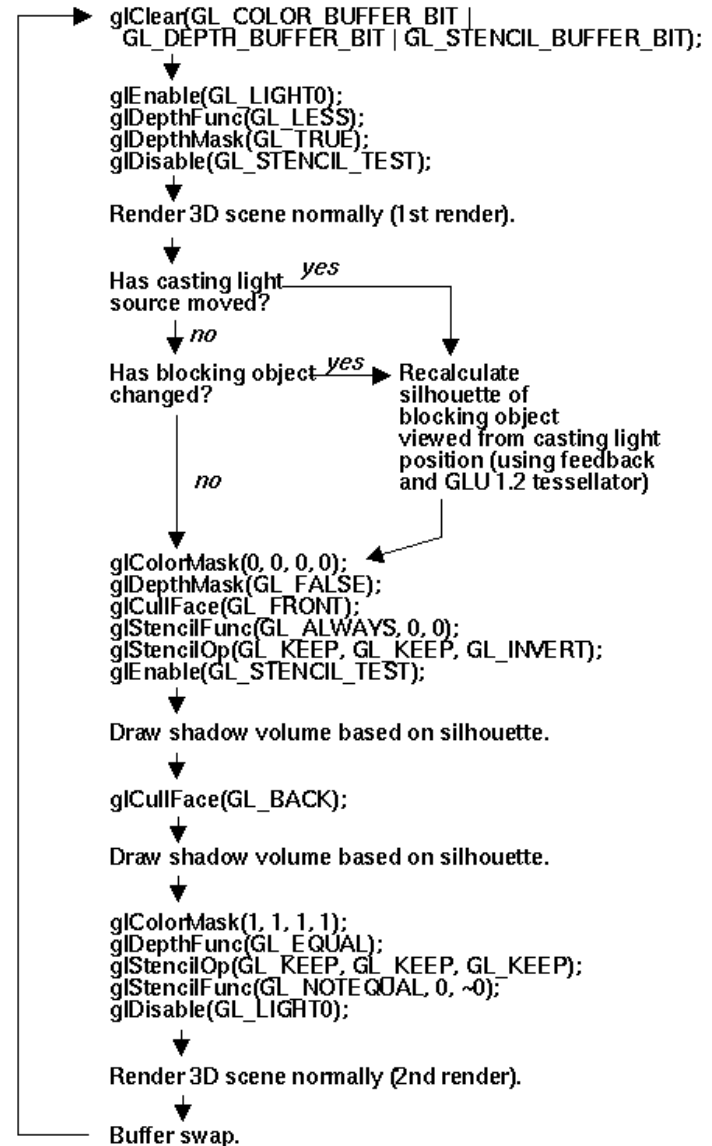
- Scissor
 - Establishes a rectangle on screen and draws only those objects that pass the scissor test.
 - `glScissor(lower left corner, size of rectangle)`
 - Activated by `enable` and `disable`
- Alpha
 - Accept or reject fragments based on their Alpha value
 - Activated by `enable` and `disable`
 - Uses `glAlphaFunc()` to determine values for testing
- Depth
 - Works similar to Alpha
 - Uses `glDepthFunc` to determine Z buffer values of pixels
 - One possible application is drawing of horizons etc.
 - Generation of your own version of fog.

The Stencil

- Similar to the scissor test
 - Lets selective drawing of/onto the scene
- Activated through enable and disable
- `glStencilFunc(func, ref, mask)`
 - Func is similar to what is used in alpha testing, blending etc.
- `glStencilOp(fail, zfail, zpass)`
 - What is to be done with fragments passes or fails the test
 - First one handles all the failed fragments, second one fragments which fail the depth test, third for those that pass the depth test.
- Can be used for
 - Masking out an irregular shaped region on screen and void all drawings to it
 - Reflections and shadows (not mathematically accurate, but something that looks nice and works)

Shadow code courtesy (morrowsoftware/APRON, opengl.org)(Mark Kilgard's sample code)

- Shadow mapping as a Z buffer problem
 - Z buffer wrt light position
 - Use that as a “depth texture” to the scene
- Shadow mapping using stencils
 - Determine location on which shadow is being cast
 - Enable stencil for that
 - Determine matrix
 - Multiply with object to be rendered and render it onto the stencil
- Other algorithms



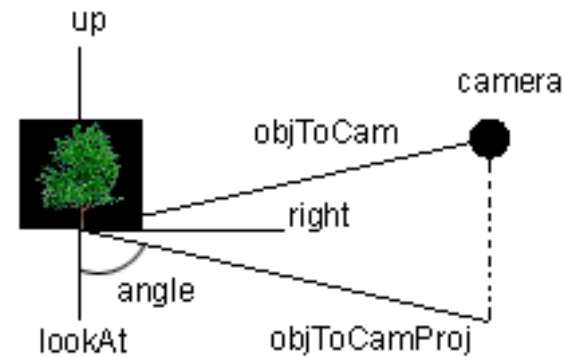
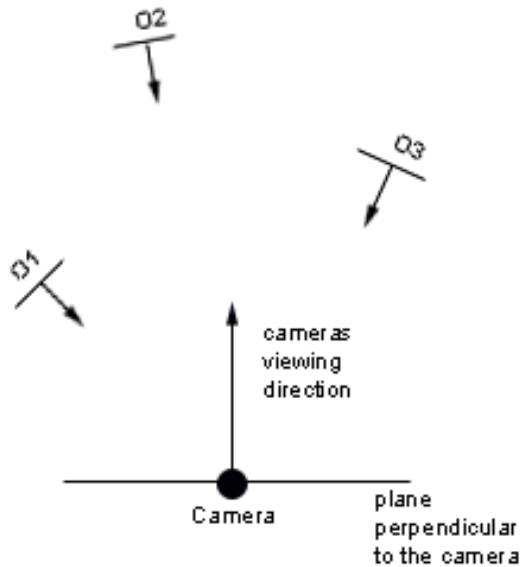
Reflections

- Similar to shadows when using Stencils
 - Chief difference is no need for calculating a “projection matrix”
 - Rendering is directly done onto the surface

Accumulation Buffer

- Series of images rendered using some other buffer viz. color buffer can be stored here to be rendered at one go
- `glAccum(open, value)`
 - Operations include `GL_ACCUM`, `GL_RETURN` etc.

Billboarding (courtesy Lighthouse 3D)

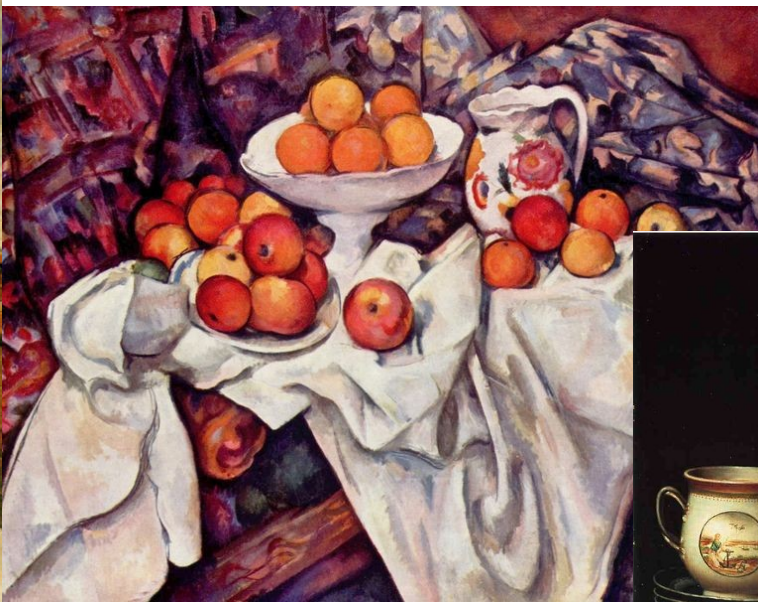


Art Imitates Life

- Large part of doing graphics requires an eye for art
 - Realism or Subjective representation
 - Colors and lights
 - Shading and brush techniques
 - Materials being painted on
 - Subjects

Still Life

- The drawing of inanimate objects
 - Vessels, food, flowers, cloth, books, furniture etc.
- Relatively easier to represent graphically
 - Lack of movement
 - No interactivity required
 - Most attention given to expressing the drawing
- Greats (~17th century onwards)
 - Van Gogh
 - Paul Cezzane
 - Raphaelle Peale
 - Renoir
 - Gaugin



Art Imitates Life

- To achieve a realistic still life rendering within VR
 - Proper object models
 - Accurate normal information
 - Material information
 - Texturing of objects
 - Opacity of materials
 - Shadows
 - Reflections
 - Brightness
 - Bump mapping for adding missing information
 - Lighting

Mini project Part 1 – Due February 27th

- Make your own still life rendering
- Try to use at least 4-5 different types of objects within your scene
- Your imagination is the only limitation
- You can always copy from the Masters and make your own rendering
- Photo – realism is not the main condition
 - Objects can appear unrealistic if there is an apparent base theme, f.e. alternate rendering like charcoal or on oil on canvas etc.