

CS 4204 Computer Graphics

OpenGL Practice

Yong Cao
Virginia Tech

Demo

- *Load *.bmp texture files*
- *Display bitmap font*
- *Play simple *.wav sound*
- *Fog in OpenGL*

Load BMP textures

- ***Step 1: Load bitmap file using any image library.***

- Example: GLU library (include “glu.h”)

```
AUX_RGBImageRec* auxDIBImageLoad (char *Filename);
```

```
typedef struct _AUX_RGBImageRec {  
    GLint sizeX, sizeY;  
    unsigned char *data;  
} AUX_RGBImageRec;
```

Load BMP textures (2)

•*Step 2: Create texture using the image data*

```
Imagedata = auxDIBImageLoad(Filename);
```

```
glGenTextures(1, textureid);
```

```
glBindTexture(GL_TEXTURE_2D, textureid);
```

```
glTexImage2D(GL_TEXTURE_2D, 0, 3,  
             Imagedata->sizeX, Imagedata->sizeY, 0, GL_RGB,  
             GL_UNSIGNED_BYTE, Imagedata->data);
```

```
glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MIN_FILTER, GL_LINEAR);
```

```
glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MAG_FILTER, GL_LINEAR);
```

Display bitmap font

- *You need a bitmap texture for font.*
- *For the string you want to display, pick the character one by one from the bitmap.*



Display bitmap font (2)

➤ *Build the display lists for all the characters.*

```
base=glGenLists(256); // Creating 256 Display Lists
glBindTexture(GL_TEXTURE_2D, texture[0]); // Select Our Font Texture
for (loop1=0; loop1<256; loop1++) // Loop Through All 256 Lists
{
    float cx=float(loop1%16)/16.0f; // X Position Of Current Character
    float cy=float(loop1/16)/16.0f; // Y Position Of Current Character

    glNewList(base+loop1, GL_COMPILE); // Start Building A List

    glBegin(GL_QUADS);
        // Use A Quad For Each Character
        glTexCoord2f(cx, 1.0f-cy-0.0625f); // Texture Coord (Bottom Left)
        glVertex2d(0, 16); // Vertex Coord (Bottom Left)
        glTexCoord2f(cx+0.0625f, 1.0f-cy-0.0625f); // Texture Coord (Bottom Right)
        glVertex2i(16, 16); // Vertex Coord (Bottom Right)
        glTexCoord2f(cx+0.0625f, 1.0f-cy); // Texture Coord (Top Right)
        glVertex2i(16, 0); // Vertex Coord (Top Right)
        glTexCoord2f(cx, 1.0f-cy); // Texture Coord (Top Left)
        glVertex2i(0, 0); // Vertex Coord (Top Left)
    glEnd(); // Done Building Our Quad (Character)
    glTranslated(15, 0, 0); // Move To The Right Of The Character
    glEndList(); // Done Building The Display List
} // Loop Until All 256 Are Built
```

Display bitmap font (3)

➤ *Call display lists*

```
// Choose The Font Set (0 or 1)
```

```
glListBase(base-32+(128*set));
```

```
//Execute a list of display lists
```

```
glCallLists(strlen(text),GL_UNSIGNED_BYTE, text);
```

Play *.wav sound file

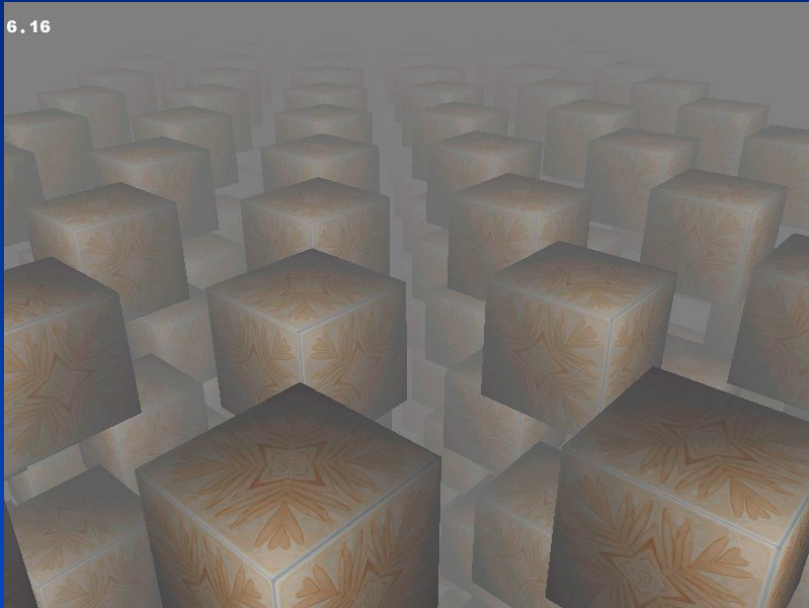
Simple:

```
PlaySound ("filename.wav", NULL, SND_SYNC);
```

- `SND_SYNC` `PlaySound` returns after the sound event completes.
- `SND_ASYNC` `PlaySound` returns immediately after beginning the sound.

Fog in OpenGL

6.16



Fog

Fog equation

$$f = \frac{\text{end} - z}{\text{end} - \text{start}}$$

z is the distance in eye coordinates from origin to fragment being fogged.

Screen-space view

A 3D rendered scene showing a blue and red jet flying in a foggy environment. The jet is positioned in the center of the frame, and the fog is a light gray color, gradually obscuring the background as the jet moves further away from the viewer.

Command manipulation window

```
GLfloat color[4] = { 0.70 , 0.70 , 0.70 , 1.00 };  
glFogfv(GL_FOG_COLOR, color);  
glFogf(GL_FOG_START, 0.77 );  
glFogf(GL_FOG_END, 3.00 );  
glFogi(GL_FOG_MODE, GL_LINEAR);
```

Click on the arguments and move the mouse to modify values.

Fog in OpenGL (2)

```
GLuint fogMode[] = { GL_EXP, GL_EXP2, GL_LINEAR };
```

```
glFogi(GL_FOG_MODE, g_fogMode[g_fogfilter]); // Fog Mode
glFogfv(GL_FOG_COLOR, g_fogColor); // Set Fog Color
glFogf(GL_FOG_DENSITY, 0.1f); // How Dense Will The Fog Be
glHint(GL_FOG_HINT, GL_DONT_CARE); // Fog Hint Value
glFogf(GL_FOG_START, 1.0f); // Fog Start Depth
glFogf(GL_FOG_END, 5.0f); // Fog End Depth
glEnable(GL_FOG); // Enables GL_FOG
```

Play windows AVI in OpenGL

- *Open AVI file*
- *Grab a frame from AVI video*
- *Change the frame format to bitmap*
- *Create a texture from the bitmap*

DEMO