

# Yet More Parameter Attributes--Curve, OpenGL, Fill

CS116A

Chris Pollett

Sep22, 2004.

# Introduction

- Curve Attributes
- OpenGL Point Attribute Functions
- OpenGL Line Attribute Functions
- Fill-area attributes

# Curve Attributes

What attributes are available is similar to the line case:

- color
- width
- dot-dash patterns
- pen and brush options

However, different techniques are sometimes used for rendering curves

# Curve Display Techniques

To display circles, conics, etc. of varying thicknesses:

- Can use horizontal or vertical spans of pixels as in the line case. This can be used circles, conics, etc.
- Can fill in area between two parallel curves. This works well for circles, but have to be careful when do for ellipses.
- Pixel mask and pen techniques the work same as in line case.

# OpenGL Point Attribute Functions

We now begin to discuss how various state attributes are specified in OpenGL. We begin with points.

- To specify the size of a point in OpenGL we use the function:

```
glPointSize(size);
```

# OpenGL Line Attribute Functions

- The command for drawing a line with a pattern is `glLineStipple(repeatFactor, pattern);`
- `repeatFactor` is the number of times each pixel in the pattern should be repeated before going to the next pixel.
- The pattern is a 16 bit hex number Ex: 00FF.
- Finally have to use `glEnable(GL_LINE_STIPPLE);/glDisable(GL_LINE_STIPPLE);` to enable/disable the line pattern mode.

# Example code fragment

```
glClear(GL_COLOR_BUFFER_BIT);
glEnable(GL_LINE_STIPPLE);
glColor3f(1.0, 0.0, 0.0);
glBegin(GL_LINES);
    glLineStipple(1, 0x000F);
    glLineWidth(5.0);
    glVertex2f(100.0, 50.0);
    glVertex2f(40.0, 200.0);
glFlush();
glEnd();
glDisable(GL_LINE_STIPPLE);
```

# More Line Effects.

- Can also have lines that smoothly transition from one color to another color.

- Here's how:

```
glShadeModel(GL_SMOOTH);  
glBegin(GL_LINES);  
    glColor3f(0.0, 0.0, 1.0);  
    glVertex2i(50, 50);  
    glColor3f(1.0, 0.0, 0.0);  
    glVertex2i(250, 250);  
glEnd();
```

- The shade model `GL_FLAT` would cause the line to be drawn with the second endpoint color.



# Fill-areas

Many graphics packages limit fill regions to polygons as can describe with a set of linear equations. Paint programs on the other hand allow more complicated fill regions circles, ellipses and things with curved boundaries.

There are two common fill techniques:

- scan horizontally and figure out which line segments are in our region
- start at some interior position and spread to the boundary.

# Fill Area Attributes: Fill Style

- A basic fill-area attribute is the display style of the interior of the figure.
  - Possible types include: hollow, solid, patterned.
- Patterns are described by specifying a mask that is to be applied to the display area.
- Some systems allow one to specify an initial position from which to start applying the mask.
- If the fill pattern is rectangular then the process of filling the region is called *tiling* and the fill pattern is called a *tiling pattern*.

# Implementing Fill Styles

- Fix a starting point for pattern (xp, yp).
- When at a point on a scan line that is in the fill area figure out which pixel position in the pattern we should be at. If it is on then draw that pixel, else don't.

# Color-Blended Fill Regions

It is also possible to combine a fill pattern with the background color in various ways:

- A pattern can be combined with background colors using a *transparency factor*. This could be used to say how much of the background should be mixed with the object color.
- Another technique is to mix the fill pattern with background according to some logical operations. For instance, AND, OR, XOR.

# Soft-Fill or Tint-Fill

- Used to soften fill colors at object borders that have been blurred to antialias the edge.
- Done by letting **F** be the foreground color and **B** be the background color. Would fill with color  $\mathbf{P} = t\mathbf{F} + (1 - t)\mathbf{B}$ , where  $t$  is between 0 and 1.
- This technique can be generalized to blend more than one color.