Colour, displays and image processing

Daniel Chatfield

December 2, 2014

From Supervision 2

1. Derive the conditions necessary for two Bézier curves to join with:

   Consider two bezier curves $p(t)$ and $q(t)$, defined by control points $(p_0, p_1, ..., p_n)$ and $(q_0, q_1, ..., q_n)$ respectively.

   (a) just C0-continuity

   $C0$ continuity can be achieved by setting $p(1) = q(0)$ giving:

   $$q_0 = p_n$$

   (b) C1-continuity

   $C1$ continuity can be achieved with the additional constraint that $p'(1) = q'(0)$ giving:

   $$q_1 - q_0 = p_n - P_{n-1}$$

   Which when combined with the above constraint gives:

   $$q_1 = 2p_n - p_{n-1}$$

   (c) C2-continuity

   For $C2$ continuity we must also have that $p''(1) = q''(0)$, giving:

   $$q_2 - 2q_1 + q_0 = p_n - 2P_{n-1} + p_{n-2}$$

   Which when combined with the above constraint gives:

   $$q_2 = p_{n-2} + 4(p_n - p_{n-1})$$

2. What would be difficult about getting three Bézier curves to join in sequence with C2-continuity at the two joins?

Control points would be dependant on two other curves?

3. For a cylinder of radius 2, with endpoints $(1, 2, 3)$ and $(2, 4, 5)$, show how to calculate:

   (a) an axis-aligned bounding box

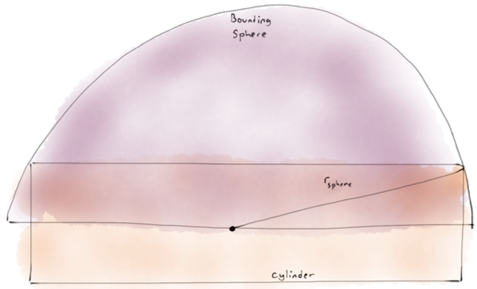
   Let m, n, and o represent the x, y, and z axes respectively.

   Let A and B be the endpoints such that $A_M <= B_M$.

   Let $\delta = \frac{\sqrt{(A_n - B_n)^2 + (A_o - B_o)^2}}{(A_m - B_m)^2 + (A_n - B_n)^2 + (A_o - B_o)^2}$

   $m_{min} = A_m - \delta \times r$
   $m_{max} = B_m + \delta \times r$

   (b) a bounding sphere

   The bounding sphere has midpoint that is half way between the two endpoints: $mid = (1.5, 3, 4)$

   The radius is the distance from the midpoint to a point on the edge of the cylinder.

   One such point is $(2, 4 + \sqrt{2}, 5 - \sqrt{2})$.

   Giving: $r = \sqrt{(0.5)^2 + (1 + \sqrt{2})^2 + (1 - \sqrt{2})^2} = 2.5$

4. Break down the following (2D!) lines into a BSP-tree, splitting them if necessary:

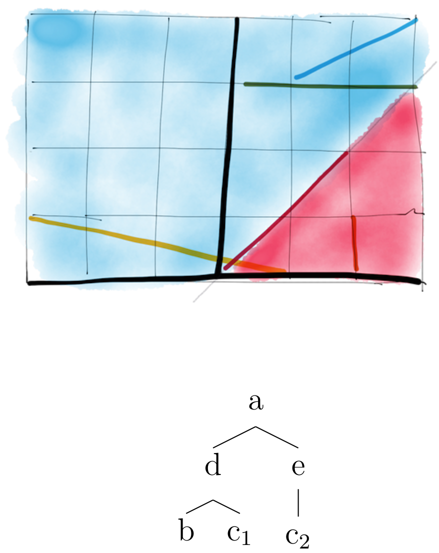
   a $(0, 0) - (2, 2)$

   b $(3, 4) - (1, 3)$

   c $(1, 0) - (-3, 1)$

   d $(0, 3) - (3, 3)$

   e $(2, 0) - (2, 1)$



5. (a) Compare the two methods of doing 3D clipping in terms of efficiency

   Not sure which is more efficient - both require clipping against 6 axes.

   (b) How would using bounding volumes improve efficiency of these methods?

   Computing and storing bounding volumes for each object in the scene makes it much easier to quickly determine that two objects don't intersect.

6. Describe a complete algorithm to do 3D polygon scan conversion, including details of clipping, projection, and the underlying 2D polygon scan conversion algorithm.

```
class Screen():
  def __init__(self, x=1920, y =1080):
    self.x = x
    self.y = y
  def get_pixels(self):
    for x in range(0, self.x):
      for y in range(0, self.y):
        yield (x,y)

class Polygon():
  get_pixels():
    # projects the polygon to 2D and runs 2D polygon scan-
       conversion
```

```
for (x,y) in screen.get_pixels():
  color[x][y] = background_color
  depth[x][y] = infinity

for polygon in scene:
  for (x,y) in polygon.get_pixels():
    if polygon.depth(x,y) < depth[x][y]:
      depth[x][y] = z
      color[x][y] = polygon.color(x,y)
```

7. Describe how you would form a good approximation to a cylinder from Bézier patches. Draw the patches and their control points and give the coordinates of the control points.

Not sure why you can't just use two bezier patches, one for each side of the cylinder (although this doesn't do the top).

8. Given the following sixteen points, calculate the first eight of the next patch joining it as t increases so that the join has continuity $C1$. Here the points are listed with $s = 0$, $t = 0$ on the bottom left, with s increasing upwards and t increasing to the right:

```
(-0.2, 3.4,  0.3)    (1.0, 3.1, 0.2)    (2.0, 2.6, -0.2)    (3.1, 2.8,  0.2)
( 0.0, 1.2,  0.4)    (1.2, 2.0, 1.2)    (1.4, 1.9, -0.2)    (2.7, 1.8,  0.2)
( 0.2, 1.0, -0.2)    (1.1, 0.8, 0.5)    (1.4, 1.0,  0.0)    (3.1, 1.1, -0.2)
( 0.0, 0.0,  0.0)    (1.0, 0.0, 0.5)    (2.0, 0.2,  0.4)    (2.7, 0.0, -0.2)
```

```
(3.1, 2.8,  0.2)    (4.2,  3.0,  0.6)
(2.7, 1.8,  0.2)    (4.0,  1.7,  0.6)
(3.1, 1.1, -0.2)    (4.8,  1.2, -0.4)
(2.7, 0.0, -0.2)    (3.4, -0.2, -0.8)
```

Colour, displays and image processing

Warmup Questions

1. Compare and contrast the use of LCDs and electrophoretic displays for screens in portable devices.

LCDs are more common in mobile devices as they can produce a wide variety of colours and are thus suitable for the rich interface on mobile phones and tablets.

Electrophoretic displays are typically used in e-book readers as it is easier to read for extended periods of time from an electrophoretic display than an LCD display. They also have the benefit that power is only required to change the display which means the e-book reader can have a long battery life as changing the display is not a frequent operation.

LCDs have a faster refresh rate than e-ink displays and a higher pixel density, and due to the backlight can be seen in darkness (although some e-ink displays do now have backlights).

2. Compare the rendering in some different pieces of printed material. Use a magnifying glass to explore the resolution, colours and patterns used.

Longer Questions

1. Explain the use of each of the following colour spaces:

   (a) RGB

   The standard RGB colour space was created for use on monitors and the internet. Since displays use red, green, and blue LEDs to create light it suits this application well.

   (b) XYZ

   The first attempt to produce a colour space based on measurements of human colour perception and it is the basis for almost all other colour spaces.

   (c) HSL

   Used by artists as it is more natural to think about a colour in terms of hue and saturation rather than additive or subtractive colour components.

   (d) LUV

   It is used in computer graphics when dealing with coloured lights. Additive mixtures of different colours will not fall along a line in the LUV colour space unless the mixtures have the same lightness.

2. Explain the difference between additive colour (RGB) and subtractive colour (CMY). Where is each used and why is it used there?

Additive colour is good when dealing with light as that is how combinations of light sources behave. Subtractive colour is used when dealing with the colour of an object as this works by absorbing light.

3. Compare the two methods of *Error Diffusion* described in the notes, with the aid of a sample image.