Raster Graphics

Victor Eijkhout

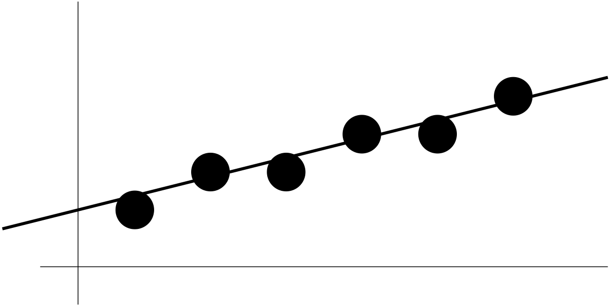
Notes for CS 594 – Fall 2004

From mathematics to pixels

- Shapes and curves described mathematically (bezier)
- Screen has pixels
- different arithmetic
- rounding behaviour
- Vector graphics vs Bitmap, Raster

Line drawing

- Symmetry: limit to slope ≤ 1



- one pixel on per column

Incremental drawing

- Line $y = mx + B$, slope $m = \delta y/\delta x$.
- Pixels: $\delta x \equiv 1$, so $\delta y = m$

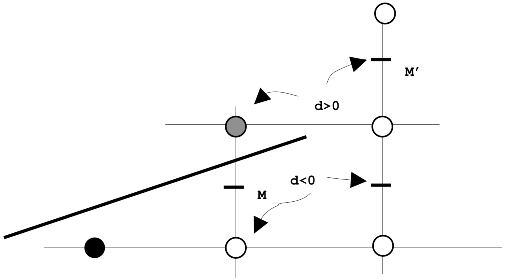
$$y_{i+1} = y_i + \delta y.$$

- implementation

  let x_0, y_0 and m be given, then
  for $i = 0 \ldots n-1$
  $\quad \texttt{WritePixel}(x_i, \mathrm{Round}(y_i))$
  $\quad x_{i+1} = x_i + 1$
  $\quad y_{i+1} = y_i + m$
- roundoff, cost

Midpoint algorithm

- Given ‘on’ pixel, choices are 1 right, 2 right-and-up



- Write
$$y = \frac{dy}{dx}x + B, \qquad F(x, y) = ax + by + c = 0.$$
then $a = dy$, $b = -dx$, $c = B$
- derive dx, dy from the end points

Midpoint location

- Does the midpoint M lie above or under the line?
- use $F(\cdot,\cdot)$: evaluate the 'decision value' of the midpoint:

$$d = F(x_p + 1, y_p + 1/2).$$

- The two cases to consider then are

$$\begin{array}{ll} d < 0: & M \text{ lies over the line, so we take } y_{p+1} = y_p; \\ d \geq 0: & M \text{ lies under the line, so we take } y_{p+1} = y_p + 1. \end{array}$$

Use of d

- Use d instead of midpoint:

$$d' = F(x_{p+1} + 1, y_{p+1} + 1/2).$$

- Two cases:

$$\begin{array}{rll} d' = & a(x_{p+1}+1) + b(y_{p+1}+1/2) + c = & \\ d < 0: & = a(x_p+2) + b(y_p+1/2) & = d + a = d + dy \\ d \geq 0: & = a(x_p+2) + b(y_p+3/2) + c & = d + a + b = d + dy - dx \end{array}$$

- Update d with dy or $dy - dx$ depending on whether it's negative or non-negative.

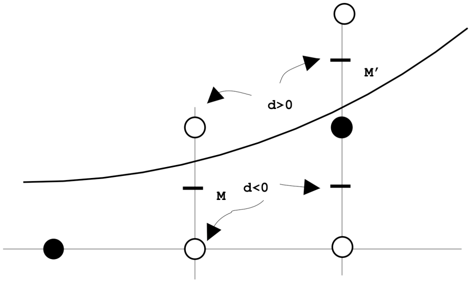
Final refinement

- Start off

$$d_0 = F(x_0+1, y_0+1/2) = F(x_0, y_0)+a+b/2 = 0+dy-dx/2.$$

- Get rid of the division by 2:
  $\tilde{F}(x, y) = 2F(x, y)$;
  update d with $2dy$ and $2(dy - dx)$ in the two cases.
- Digital Differential Analyzers (DDA)

Circle drawing



- Circle:

$$F(x, y) = x^2 + y^2 - R^2,$$

- decision value in the midpoint M is

$$d = F(x_p + 1, y_p + 1/2) = x^2 + 2x + y^2 + y + 5/4.$$

- Cases

$d < 0$: M lies in the circle, so we take $y_{p+1} = y_p$;
$d \geq 0$: M lies outside the circle, so we take $y_{p+1} = y_p + 1$.

- Updating:

$$
\begin{array}{rll}
d' = & F(x_{p+1}+1, y_{p+1}+1/2) = & \\
d < 0: & = x^2+4x+y^2+y+4\,1/4 & = d+2x+3 \\
d \geq 0: & = x^2+4x+y^2+3y+6\,1/4 & = d+2(x+y)+5
\end{array}
$$

- Construct $2x$,$2y$ by shift

Cubics

Stepwise computation

- Cubic function $f(t) = at^3 + bt^2 + ct + d$
- Strategy: compute the value $f(t+\delta)$ by updating

$$f(t+\delta) = f(t) + \Delta f(t).$$

- (alternatives: Horner, midpoint)
- Difference:

$$\begin{array}{rcl} \Delta f(t) & = & f(t+\delta) - f(t) \\ & = & a(3t^2\delta + 3t\delta^2 + \delta^3) + b(2t\delta + \delta^2) + c\delta \\ & = & 3a\delta t^2 + (3a\delta^2 + 2b\delta)t + a\delta^3 + b\delta^2 + c\delta \end{array}$$

- Quadratic term left

- Define

$$\begin{aligned}\Delta^2 f(t) &= \Delta f(t+\delta) - \Delta f(t)\\ &= 3a\delta(2t\delta+\delta^2) + (3a\delta^2+3b\delta)\delta\\ &= 6a\delta^2 t + 6a\delta^3 + 2b\delta^2\end{aligned}$$

- Third difference: $\Delta^3 f(t) = \Delta^2 f(t+\delta) - \Delta^2 f(t) = 6a\delta^2$
- Together: compute $f_{n+1} \equiv f((n+1)\delta)$ by

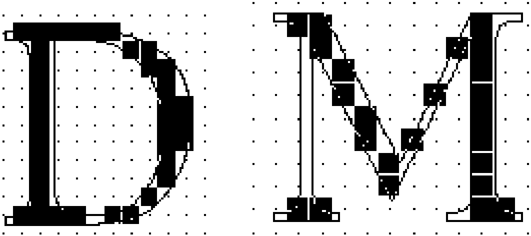
$$\Delta^3 f_0 = 6a\delta^2, \quad \Delta^2 f_0 = 6a\delta^3 + 2b\delta^2, \quad \Delta f_0 = a\delta^3 + b\delta^2 + c\delta$$

and computing by update

$$f_{n+1} = f_n + \Delta f_n, \quad \Delta f_{n+1} = \Delta f_n + \Delta^2 f_n, \quad \Delta^2 f_{n+1} = \Delta^2 f_n + \Delta^3 f_0$$

Rasterizing type

Type is tricky: lots of features in small objects
everyone immediately sees when it's wrong



Every dog must have his day.

A stitch in time saves nine.

Haste makes waste.

Waste not want not.

Variety is the spice of life.

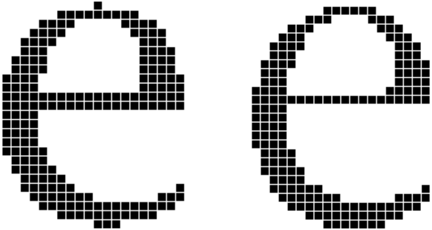
Absence makes the heart grow fonder.

Beauty is as beauty does.

Loose lips sink ships.

Badly rasterized characters

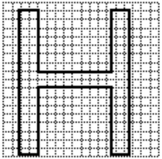
Obvious algorithm: pixel on if center in the contour



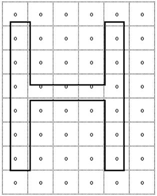
- Problems with curves tangent to $n + 1/2$ lines
- Different scalings, different raster
- Variable placement

Scaling and rasterizing

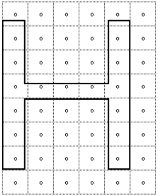
Original character is on internal raster:



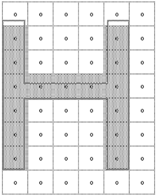
Scale to target raster:



Round to target raster:



Set pixels:



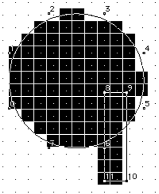
Scaling vs design size

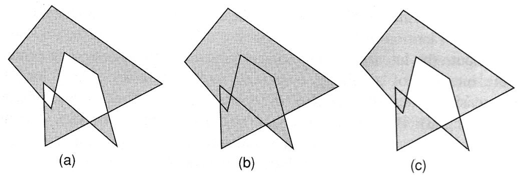
Ten point type is different from magnified five-point type.

- Scaling is a compromise
- Different design sizes
- Adobe Multiple Master

Filling in

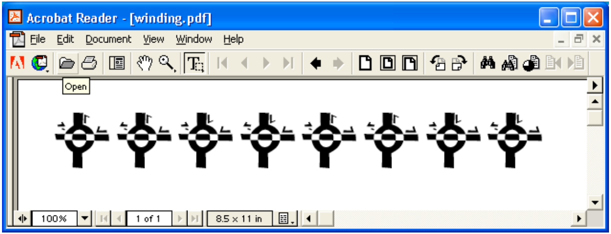
- Precisely what does ‘pixel lies within the contour’ mean?
- Complications: letters with ‘bowls’; multiple contours

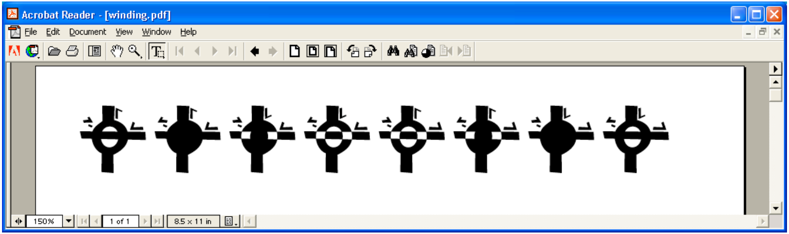




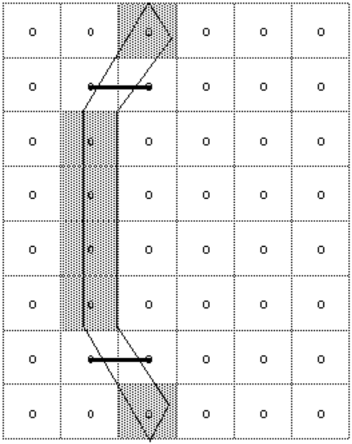
- Winding rules

Winding rules

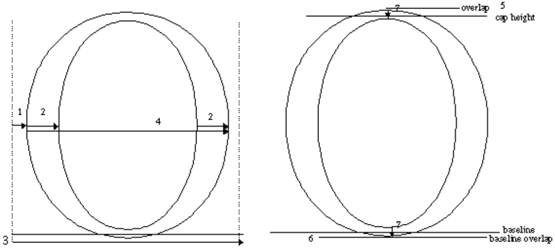




Dropouts



Hinting / instructing



- Small programs per font / character
- Give constraints on placement, relations, distance