Triangle rasterization
Which pixels should be used to approximate a triangle?
Triangle rasterization issues
Who should fill in shared edge?

Which pixels should be used to approximate a triangle?
Which pixels should be used to approximate a triangle?

Who should fill in shared edge?
Which pixels should be used to approximate a triangle?

Use Midpoint Algorithm for edges and fill in?
Which pixels should be used to approximate a triangle?

Use an approach based on barycentric coordinates.
We can interpolate attributes using barycentric coordinates

\[ \mathbf{c} = \alpha \mathbf{c}_0 + \beta \mathbf{c}_1 + \gamma \mathbf{c}_2 \]

**Gouraud shading**

(Gouraud, 1971)

http://jtibble.dyndns.org/graphics/eecs487/eecs487.html
Triangle rasterization algorithm

for all \( x \) do 
  for all \( y \) do 
    compute \((\alpha, \beta, \gamma)\) for \((x,y)\) 
    if \((\alpha \in [0, 1] \text{ and } \beta \in [0, 1] \text{ and } \gamma \in [0, 1])\) then 
      \[ c = \alpha c_0 + \beta c_1 + \gamma c_2 \] 
      drawpixel\((x,y)\) with color \( c \)
Triangle rasterization algorithm

for all $x$ do
  for all $y$ do
    compute $(\alpha, \beta, \gamma)$ for $(x, y)$
    if $(\alpha \in [0, 1]$ and $\beta \in [0, 1]$ and $\gamma \in [0, 1])$ then
      $c = \alpha c_0 + \beta c_1 + \gamma c_2$
      drawpixel($x, y$) with color $c$
Triangle rasterization algorithm

use a bounding rectangle

for \( x \) in \([x_{\text{min}}, x_{\text{max}}]\)
for \( y \) in \([y_{\text{min}}, y_{\text{max}}]\)
    compute \((\alpha, \beta, \gamma)\) for \((x, y)\)
    if \((\alpha \in [0, 1] \text{ and } \beta \in [0, 1] \text{ and } \gamma \in [0, 1])\) then
        \(c = \alpha c_0 + \beta c_1 + \gamma c_2\)
        drawpixel\((x, y)\) with color \(c\)
Triangle rasterization algorithm

for \( x \) in \([x_{\text{min}}, x_{\text{max}}]\)
  for \( y \) in \([y_{\text{min}}, y_{\text{max}}]\)
    \[
    \begin{align*}
    \alpha &= \frac{f_{bc}(x, y)}{f_{bc}(x_a, y_a)} \\
    \beta &= \frac{f_{ca}(x, y)}{f_{ca}(x_b, y_b)} \\
    \gamma &= \frac{f_{ab}(x, y)}{f_{ab}(x_c, y_c)}
    \end{align*}
    \]
    if \((\alpha \in [0, 1] \text{ and } \beta \in [0, 1] \text{ and } \gamma \in [0, 1])\) then
      \[
      c = \alpha c_0 + \beta c_1 + \gamma c_2
      \]
      drawpixel(x, y) with color \( c \)
Triangle rasterization algorithm

Optimizations?

\[
\begin{align*}
\text{for } x \text{ in } [x_{\text{min}}, x_{\text{max}}] \\
&\quad \text{for } y \text{ in } [y_{\text{min}}, y_{\text{max}}] \\
&\quad \quad \alpha = \frac{f_{bc}(x, y)}{f_{bc}(x_a, y_a)} \\
&\quad \quad \beta = \frac{f_{ca}(x, y)}{f_{ca}(x_b, y_b)} \\
&\quad \quad \gamma = \frac{f_{ab}(x, y)}{f_{ab}(x_c, y_c)} \\
&\quad \quad \text{if } (\alpha \in [0, 1] \text{ and } \beta \in [0, 1] \text{ and } \gamma \in [0, 1]) \text{ then} \\
&\quad \quad \quad c = \alpha c_0 + \beta c_1 + \gamma c_2 \\
&\quad \quad \text{drawpixel}(x,y) \text{ with color } c
\end{align*}
\]
Triangle rasterization algorithm

Optimizations?

\[
\begin{aligned}
&\text{for } x \text{ in } [x_{\text{min}}, x_{\text{max}}] \\
&\quad \text{for } y \text{ in } [y_{\text{min}}, y_{\text{max}}] \\
&\quad \quad \alpha = f_{bc}(x, y) / f_{bc}(x_a, y_a) \\
&\quad \quad \beta = f_{ca}(x, y) / f_{ca}(x_b, y_b) \\
&\quad \quad \gamma = f_{ab}(x, y) / f_{ab}(x_c, y_c) \\
&\quad \text{if } (\alpha \geq 0 \text{ and } \beta \geq 0 \text{ and } \gamma \geq 0) \text{then} \\
&\quad \quad c = \alpha c_0 + \beta c_1 + \gamma c_2 \\
&\quad \quad \text{drawpixel}(x, y) \text{ with color } c \\
\end{aligned}
\]

make computation of bary. coords. incremental
color can also be computed incrementally
don’t need to check upper bound
Triangle rasterization algorithm

dealing with shared triangle edges

\[
\begin{align*}
\text{for } x \text{ in } [x_{\text{min}}, x_{\text{max}}] \\
\text{for } y \text{ in } [y_{\text{min}}, y_{\text{max}}] \\
\alpha &= f_{bc}(x, y)/f_{bc}(x_a, y_a) \\
\beta &= f_{ac}(x, y)/f_{ac}(x_b, y_b) \\
\gamma &= f_{ab}(x, y)/f_{ab}(x_c, y_c) \\
\text{if } (\alpha \geq 0 \text{ and } \beta \geq 0 \text{ and } \gamma \geq 0) \text{ then} \\
\text{if } (\alpha > 0 \text{ or } f_{bc}(a)f_{bc}(r) > 0) \text{ and } \\
(\beta > 0 \text{ or } f_{ca}(b)f_{ca}(r) > 0) \text{ and } \\
(\gamma > 0 \text{ or } f_{ab}(c)f_{ab}(r) > 0) \text{ then} \\
c &= \alpha c_0 + \beta c_1 + \gamma c_2 \\
\text{drawpixel}(x,y) \text{ with color c}
\end{align*}
\]