渐变三角形

这里我们先定义一个h，用来表示每个顶点的颜色浓度

$$C_h = (R_c * h, G_c * h, B_c * h)$$

当 h = 0的时候就是黑色，否则就是其本身的颜色。

对于颜色渐变，其实也可以做的很简单，跟之前一样，用线性的变换。

```
x01 = Interpolate(y0, x0, y1, x1)
h01 = Interpolate(y0, h0, y1, h1)

x12 = Interpolate(y1, x1, y2, x2)
h12 = Interpolate(y1, h1, y2, h2)

x02 = Interpolate(y0, x0, y2, x2)
h02 = Interpolate(y0, h0, y2, h2)
```

然后 h 的取值计算也是跟 x 类似

```
remove_last(x01)
x012 = x01 + x12

remove_last(h01)
h012 = h01 + h12

m = x02.length/2
if x02[m] < x012[m]{
  x_left = x02
  x_right = x012

  h_left = h02
  h_right = h012
} else {
  x_left = x012
  x_right = x02

  h_left = h012
  h_right = h02
}
```

对于图中的 h_segment 可以这样算：

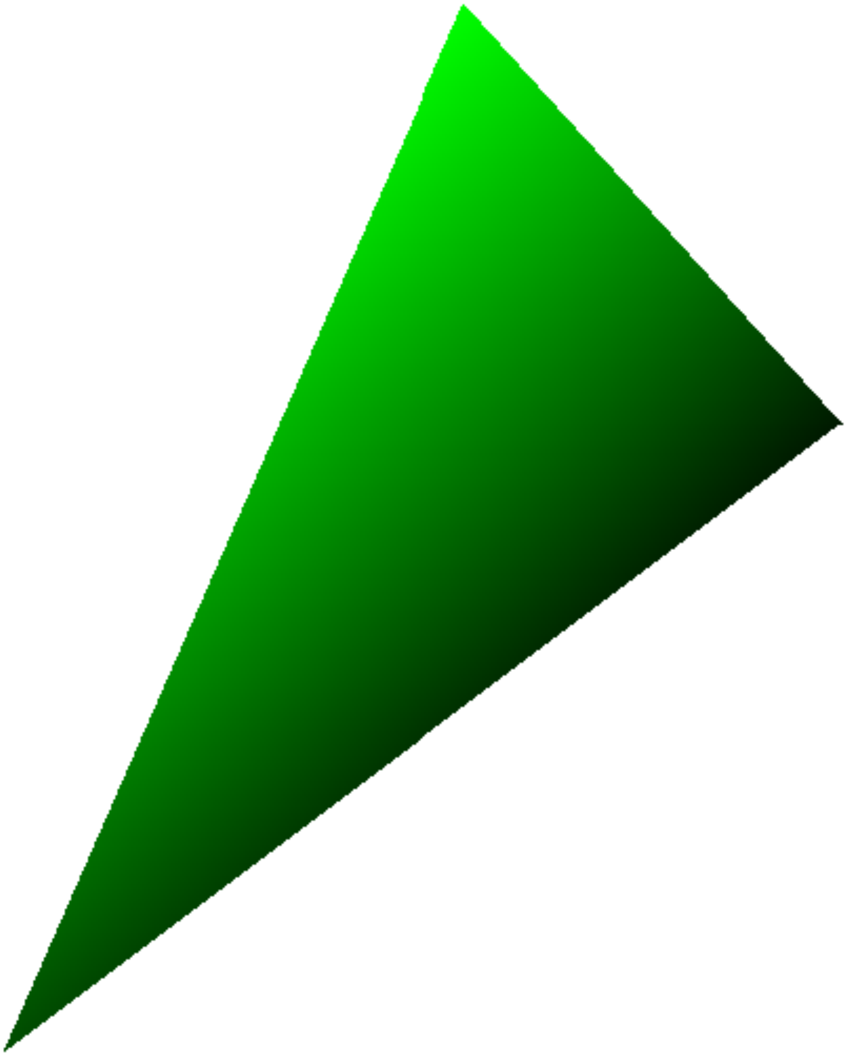
```
h_segment = Interpolate(x_left[y-y0], h_left[y-y0], x_right[x-x0], h_right[y-y0]
```

所以画渐变三角形可以这样来做:

```
DrawShadedTriangle(P0, P1, P2, color){
  # sort the points so that y0 ≤ y1 ≤ y2
  if y1 < y0 { swap(p1, p0) }
  if y2 < y0 { swap(p2, p0) }
  if y2 < y1 { swap(p2, p1) }

  # compute the x coordinates of the triangle edges
  x01 = Interpolate(y0, x0, y1, x1)
  h01 = Interpolate(y0, h0, y1, h1)

  x12 = Interpolate(y1, x1, y2, x2)
  h12 = Interpolate(y1, h1, y2, h2)

  x02 = Interpolate(y0, x0, y2, x2)
  h02 = Interpolate(y0, h0, y2, h2)

  # concatenate the short sides
  remove_last(x01)
  x012 = x01 + x12

  remove_last(h01)
  h012 = h01 + h12

  # determine which is left and which is right
  m = x02.length/2
  if x02[m] < x012[m]{
    x_left = x02
    x_right = x012

    h_left = h02
    h_right = h012
  } else {
    x_left = x012
    x_right = x02

    h_left = h012
    h_right = h02
  }


  # draw the horizontal segments
  for y = y0 to y2 {
    x_l = x_left[y - y0]
    x_r = x_right[y - y0]
    h_segment = Interpolate(x_left[y-y0], h_left[y-y0], x_right[x-x0],
h_right[y-y0])
    for x = x_l to x_r{
      shaded_color = color * h_segment[x - xl]
      canvas.putPixel(x, y, shaded_color)
    }
  }
}
```

结果

看效果：



代码链接