



逻辑教育  
Logic education

# Hello 视觉全训班

OpenGL

OpenGL ES

GPUImage

Metal



## 视觉全训班. 大长腿功能

@ CC老师

全力以赴·非同凡“想”

课程研发:CC老师

课程授课:CC老师

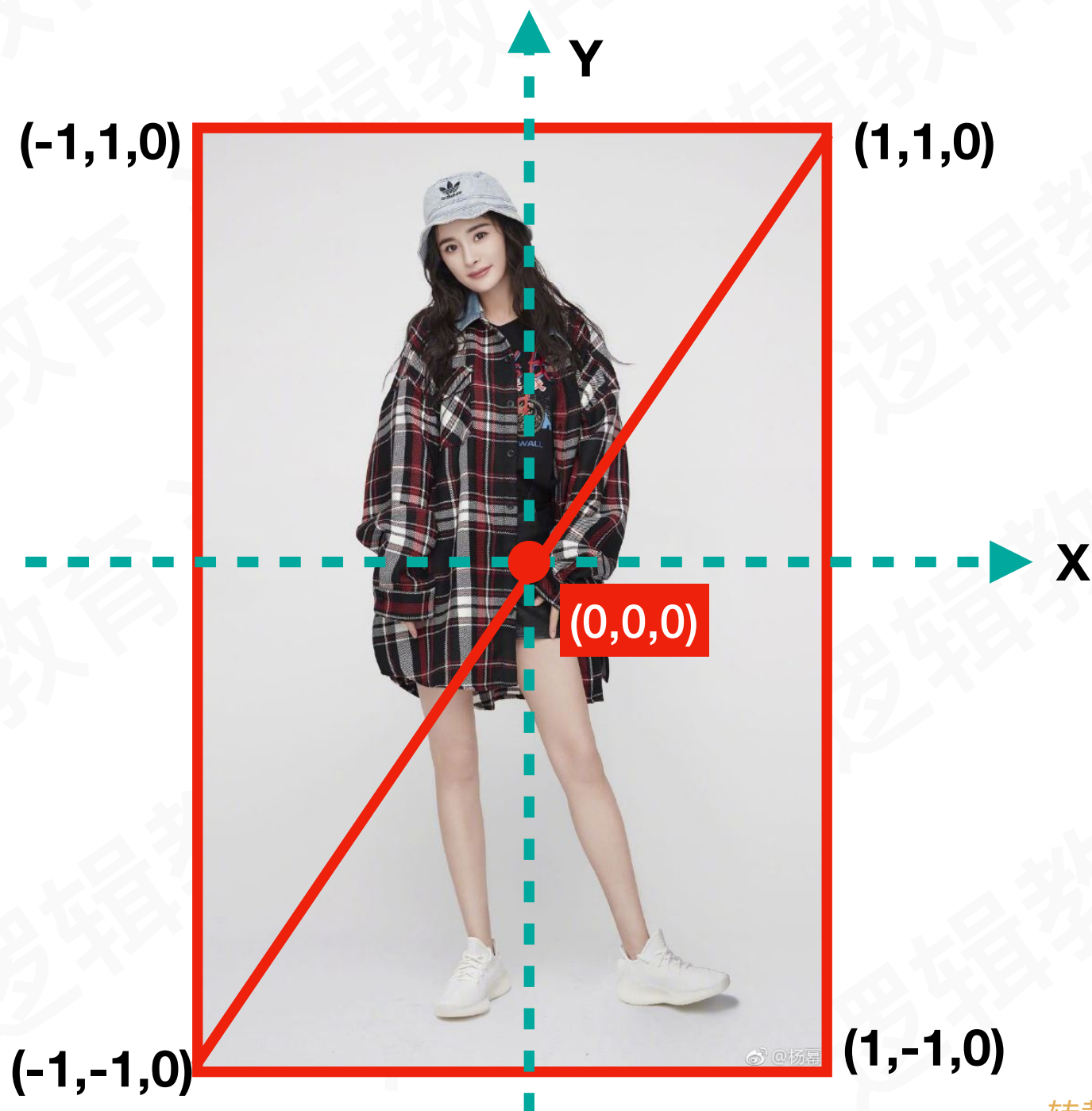
转载需注明出处,不得用于商业用途.已申请版权保护



逻辑教育  
Logic education

# OpenGL ES 实战篇 — 腿部拉长功能实现

@顶点坐标



课程研发:CC老师  
课程授课:CC老师

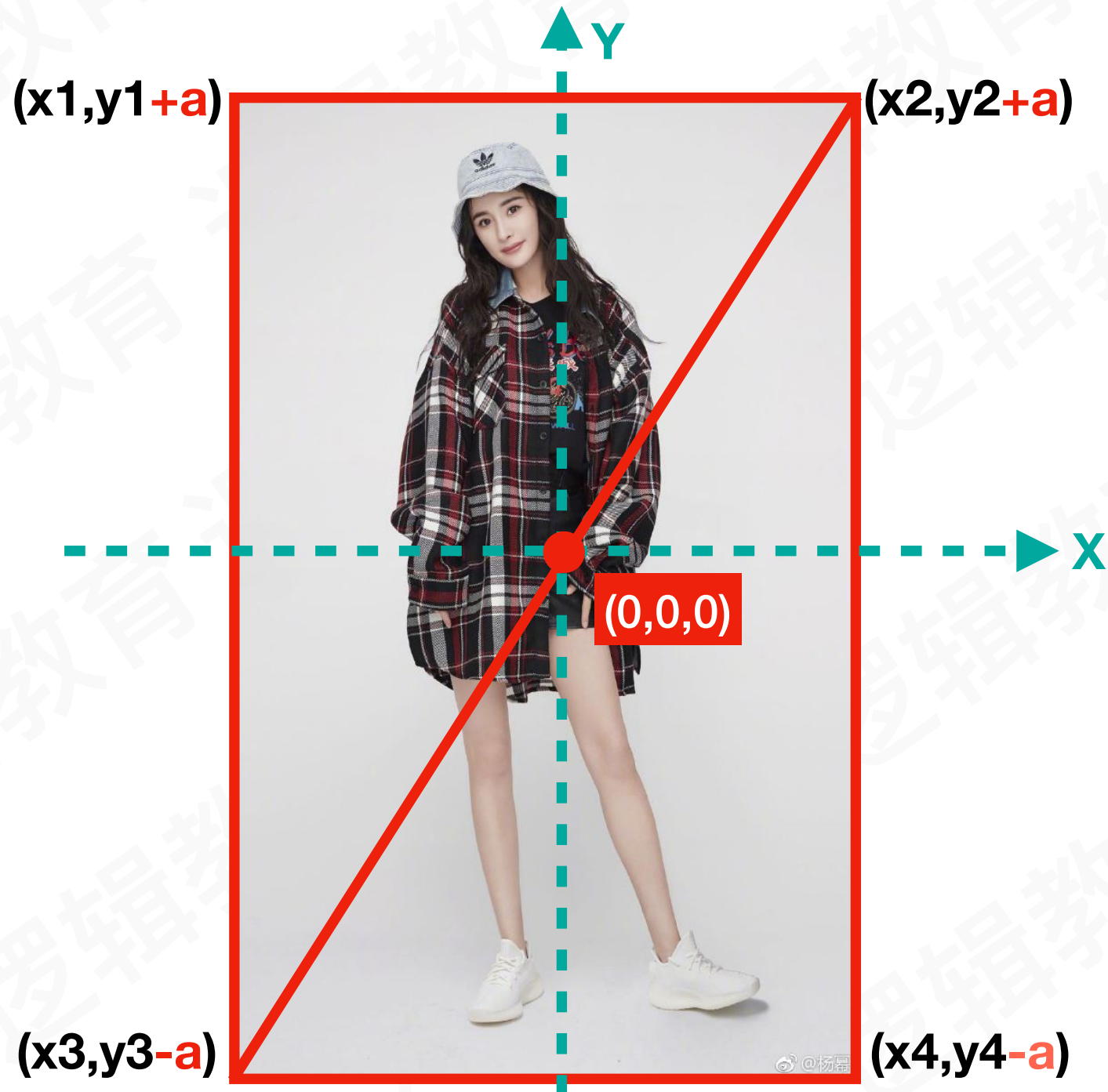
转载需注明出处,不得用于商业用途.已申请版权保护



逻辑教育  
Logic education

## OpenGL ES 实战篇 — 腿部拉长功能实现

@顶点坐标



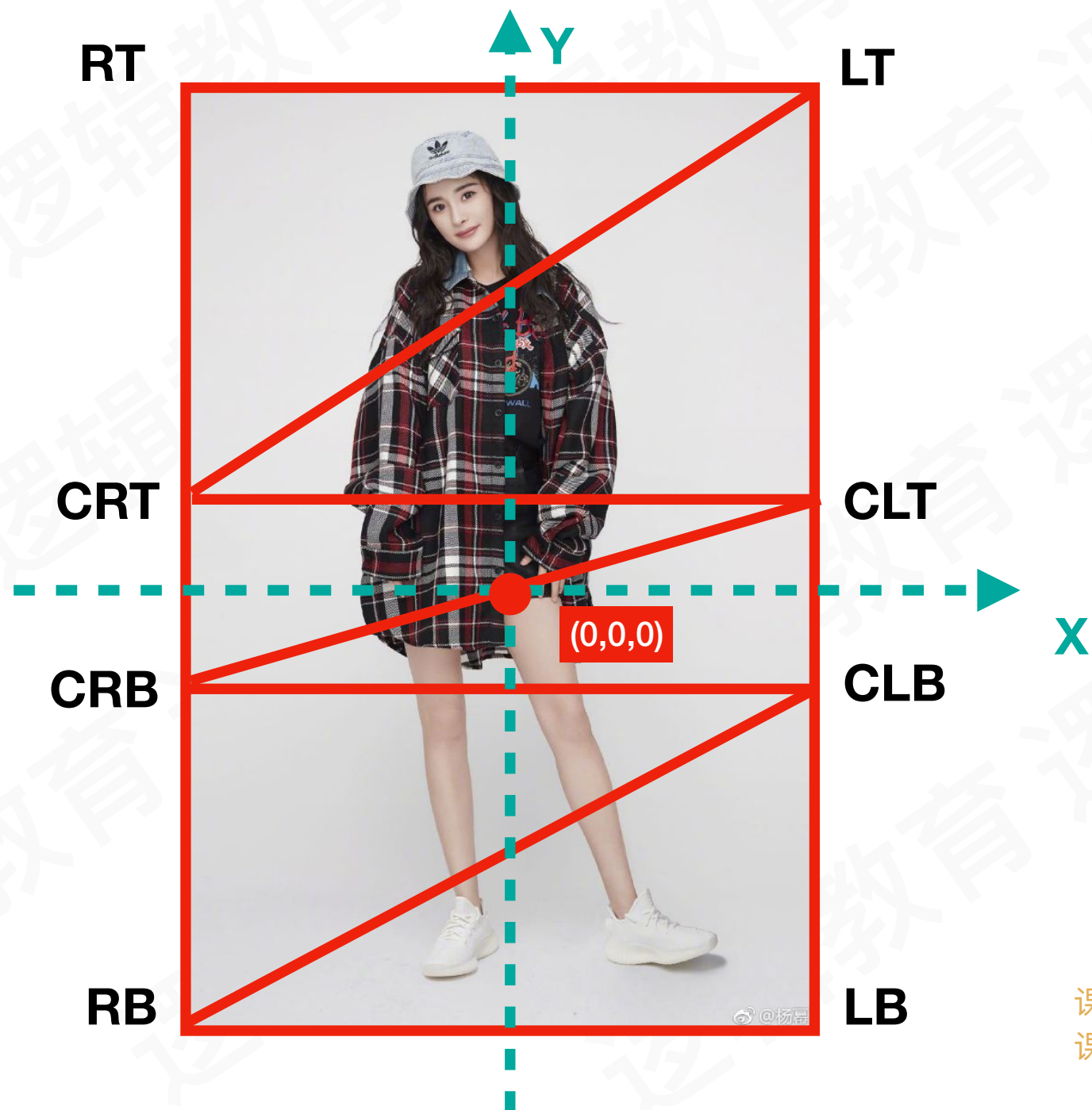
课程研发:CC老师  
课程授课:CC老师

转载需注明出处,不得用于商业用途.已申请版权保护



逻辑教育  
Logic education

# OpenGL ES 实战篇 — 腿部拉长功能实现 @顶点坐标



课程研发:CC老师  
课程授课:CC老师

转载需注明出处,不得用于商业用途.已申请版权保护





## OpenGL ES 实战篇 — 腿部拉长顶点以及纹理相关距离计算



```
newHeight = (self.currentBottom - self.currentTop) * ((sender.value) + 0.5);  
newHeight = (0.75 - 0.25) * (0.48+0.5)  
newHeight = (0.5) * (0.98) = 0.49;
```

currentTop = 0.25f

currentBottom = 0.75f

课程研发:CC老师  
课程授课:CC老师

@param size 原始纹理尺寸  
@param startY 中间区域的开始纵坐标位置 0~1  
@param endY 中间区域的结束纵坐标位置 0~1  
@param newHeight 新的中间区域的高度

## OpenGL ES 实战篇 — 腿部拉长顶点以及纹理相关距离计算



- 已知原始纹理Size : 690 × 1035 ;
- 已知LongLegView.Size: 65 × 606
- 已知默认textureHeight : 0.7
- 已知currentTextureWidth = 默认textureHeight \* radio = 0.8;

$$\text{拉伸后的宽高比ratio} = \frac{\text{纹理的height}}{\text{纹理的width}} \times \frac{\text{view的height}}{\text{view的width}}$$

拉伸后的宽高比: ratio = 0.87;

```
CGFloat textureWidth = self.currentTextureWidth;  
CGFloat textureHeight = textureWidth * ratio;
```

课程研发:CC老师  
课程授课:CC老师

@param size 原始纹理尺寸  
@param startY 中间区域的开始纵坐标位置 0~1  
@param endY 中间区域的结束纵坐标位置 0~1  
@param newHeight 新的中间区域的高度

## OpenGL ES 实战篇 — 腿部拉长顶点以及纹理相关距离计算



拉伸量  $\text{delta} = (\text{newHeight} - (\text{endY} - \text{startY})) \times \text{textureHeight}$

判断  $\text{delta} + \text{textureHeight} \geq 1$  则需要对纹理高度以及delta进行优化;  

```
if (textureHeight + delta >= 1) {  
    delta = 1 - textureHeight;  
    newHeight = delta / textureHeight + (endY - startY);  
}
```

拉伸点开始的位置

$\text{startYCoord} = \text{textureHeight} - 2 * \text{textureHeight} * \text{startY};$   
 $= 0.7 - 2 * 0.7 * 0.25 = 0.35$   
= (纹理原始高度 - 原来0.7的一半在往上挪0.25个比例位置)

拉伸点结束的位置

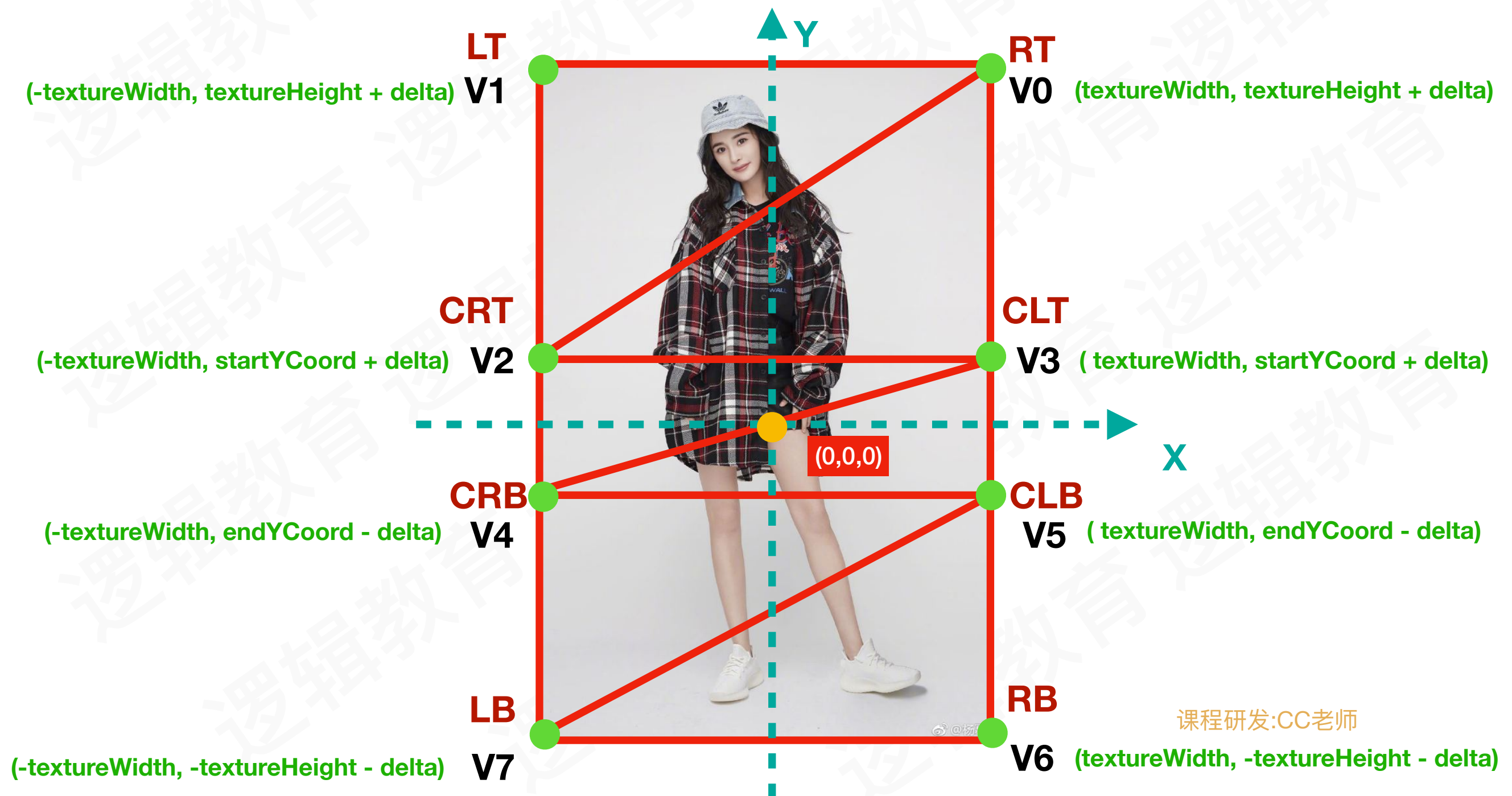
$\text{endYCoord} = \text{textureHeight} - 2 * \text{textureHeight} * \text{endY};$   
 $= 0.7 - 2 * 0.7 * 0.75 = -0.35$   
= (纹理原始高度 - 原始纹理0.7高度一半再往下挪0.75个比例的位置)

课程研发:CC老师  
课程授课:CC老师



delta: 图片整体拉伸的长度;  
startYCoord : 中间部分拉伸的头部位置  
endYCoord: 中间部分拉伸的底部位置

# OpenGL ES 实战篇 — 腿部拉长功能实现 @顶点坐标



课程研发:CC老师





逻辑教育  
Logic education

## OpenGL ES 实战篇 — 腿部拉长功能实现

@纹理坐标



课程研发:CC老师  
课程授课:CC老师

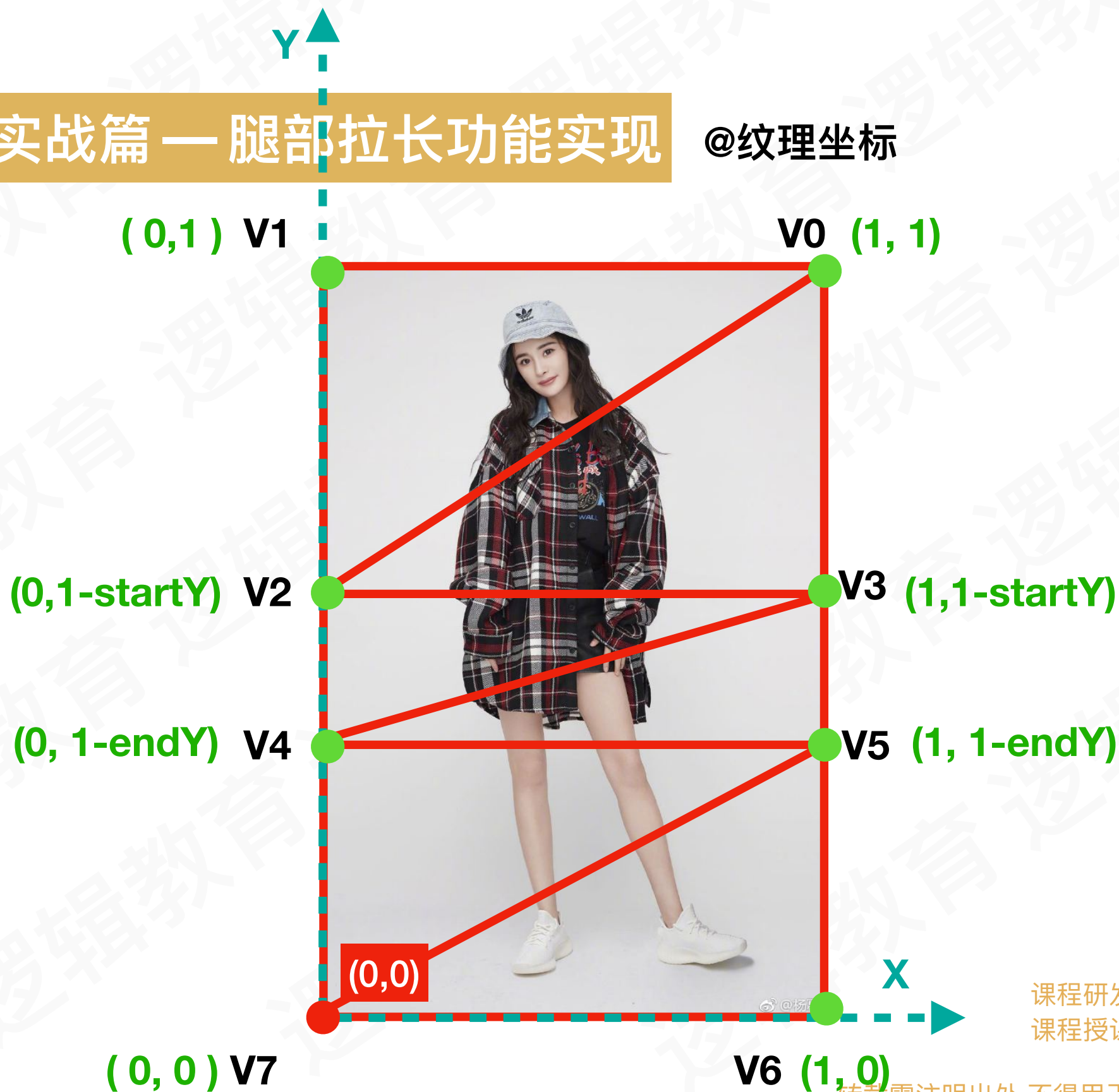
转载需注明出处,不得用于商业用途.已申请版权保护



逻辑教育  
Logic education

# OpenGL ES 实战篇 — 腿部拉长功能实现

@纹理坐标



课程研发:CC老师  
课程授课:CC老师

转载需注明出处,不得用于商业用途.已申请版权保护



逻辑教育  
Logic education

## OpenGL ES 实战篇 — 腿部拉长功能

# 用什么方式来实现拉伸?

课程研发:CC老师  
课程授课:CC老师

转载需注明出处,不得用于商业用途.已申请版权保护



逻辑教育  
Logic education

## OpenGL ES 实战篇 — 腿部拉长功能

# 如何实现重复拉伸?

课程研发:CC老师  
课程授课:CC老师

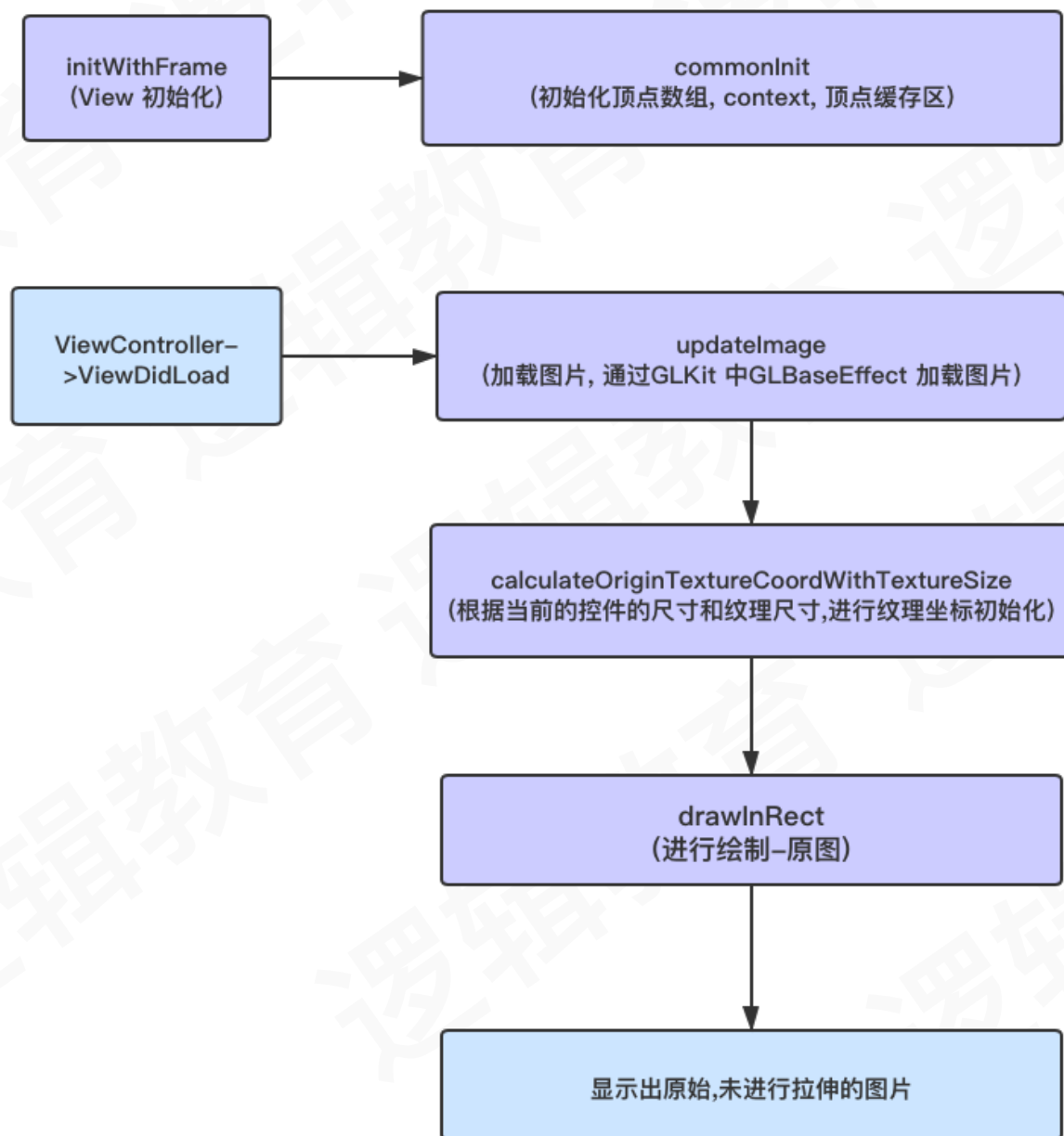
转载需注明出处,不得用于商业用途.已申请版权保护





# OpenGL ES 实战篇 — 腿部拉长功能[第一次加载原始图片]

大长腿功能: 关于LongLegView 加载第一次加载图片的过程分析



课程研发:CC老师  
课程授课:CC老师



## OpenGL ES 实战篇 — 腿部拉长功能[滑块调整]

viewController 中调整上下滑块触发的相关方法调用

actionPanBottom/actionPanTop  
(滑动上下拉伸滑块)

判断LongLegView 是否发生修改, 如果发生修改, 则调用updateTexture 方法. 用最新的纹理. 同时修改slider Value = 0.5

修改滑块位置以及约束信息

计算移动后的currentTop 和 currentBottom 位置. 进行拉伸调整;

上下滑块位置发生了变换;

课程研发:CC老师  
课程授课:CC老师



## OpenGL ES 实战篇 — 腿部拉长功能[图片拉伸过程]

LongLegView 进行Slider 拉伸调整方法调用过程

sliderValueDidChange  
(当Slider发送调整时)

stretchingFromStartY  
(将区域拉伸或压缩为某个高度)

calculateOriginTextureCoordWithTextureSize  
(根据当前控件的尺寸和纹理的尺寸, 计算初始纹理坐标)

drawInRect  
(绘制)

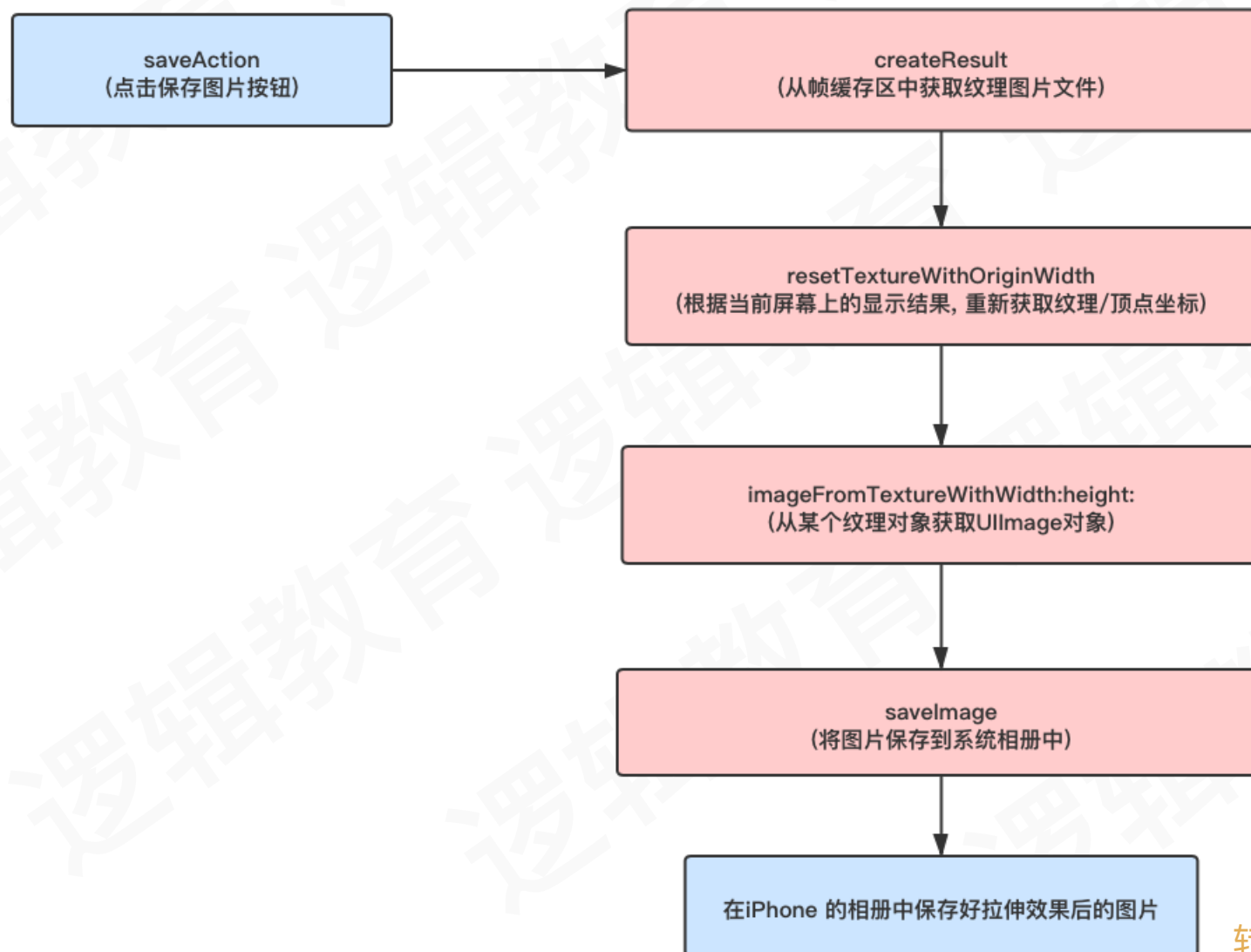
显示拉伸后的图片

课程研发:CC老师  
课程授课:CC老师



# OpenGL ES 实战篇 — 腿部拉长功能[图片保存到相册]

LongLegView 将拉伸后效果的图片保存到系统相册中;



课程研发:CC老师  
课程授课:CC老师





逻辑教育  
Logic education



GPUImage

Metal

OpenGL ES

OpenGL

see you next time ~

@ CC老师

全力以赴·非同凡”想”

课程研发:CC老师

课程授课:CC老师

转载需注明出处,不得用于商业用途.已申请版权保护