

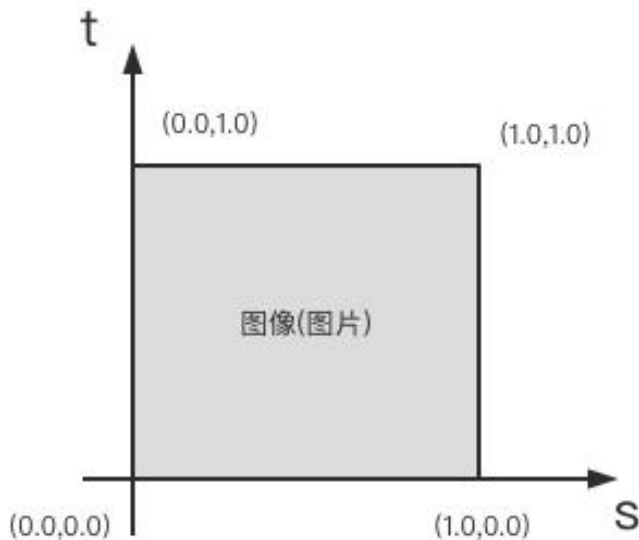


给图形添加背景图



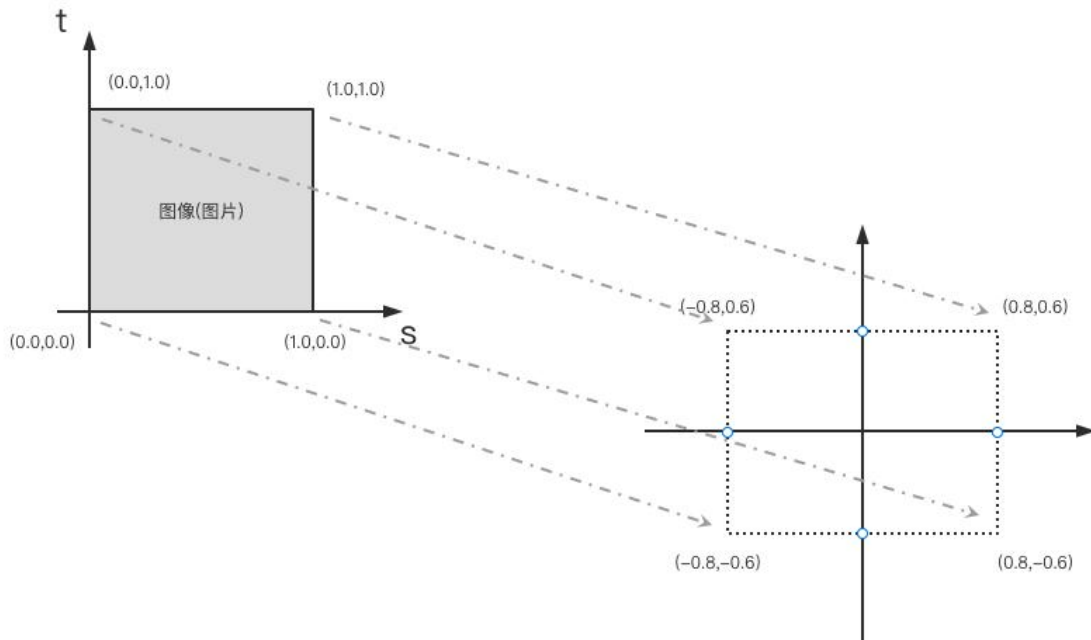
1.2 介绍纹理坐标

◆ 纹理坐标 也称为 **st坐标**。如下所示



1.2 介绍纹理坐标

- ◆ 在webgl里需要通过纹理坐标和图形顶点坐标的映射关系来确定贴图。



1.3 创建纹理对象

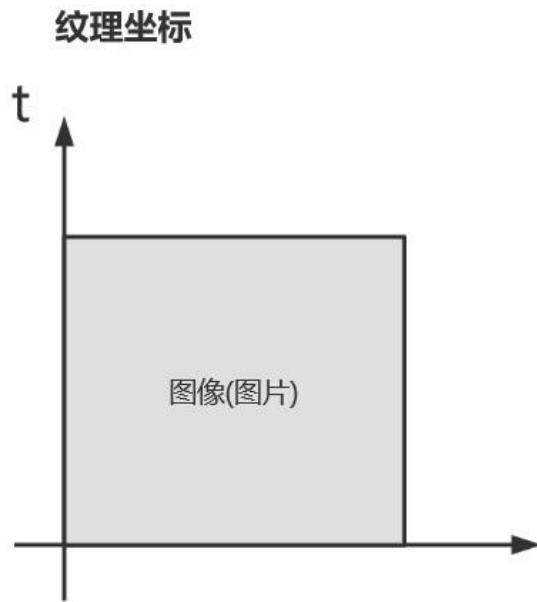
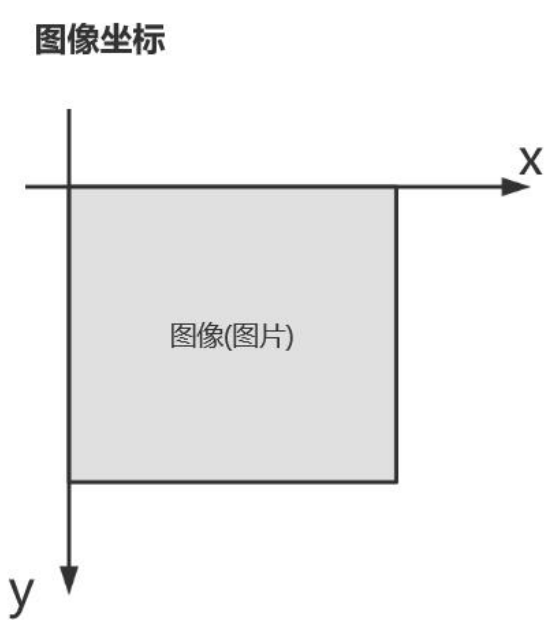
- ◆ 纹理对象主要用于存储纹理图像数据。

```
const texture = gl.createTexture();
```

- ◆ 可以通过 `gl.deleteTexture(texture)` 来删除纹理对象。

1.4 进行Y轴翻转

```
gl.pixelStorei(gl.UNPACK_FLIP_Y_WEBGL, 1);
```



1.5 开启(激活)纹理单元

```
gl.activeTexture(gl.TEXTURE0);
```

- ◆ WebGL 是通过纹理单元来管理纹理对象，每个纹理单元管理一张纹理图像。

1.6 gl.bindTexture(type, texture)

◆ type 参数有以下两种：

gl.TEXTURE_2D: 二维纹理

gl.TEXTURE_CUBE_MAP: 立方体纹理

◆ texture: 纹理对象

1.7 gl.texParamteri(type, pname, param)

◆ type 参数同上

pname:

纹理参数有四个选项

gl.TEXTURE_MAG_FILTER 放大

gl.TEXTURE_MIN_FILTER 缩小

gl.TEXTURE_WRAP_S 横向（水平填充）

gl.TEXTURE_WRAP_T 纵向（垂直填充）

1.7 gl.texParamteri(type, pname, param)

◆ param

赋值给
gl.TEXTURE_MAG_FILTER
和
gl.TEXTURE_MIN_FILTER

gl.NEAREST 使用像素颜色值

gl.LINEAR 使用四周的加权平均值

1.7 gl.texParamteri(type, pname, param)

◆ param

赋值给
gl.TEXTURE_WRAP_S
和
gl.TEXTURE_WRAP_T

gl.REPEAT 平铺重复

gl.MIRRORED_REPEAT 镜像对称

gl.CLAMP_TO_EDGE 边缘延伸

1.8 gl.texImage2D(type, level, internalformat, format,dataType, image)

◆ type 同上

◆ level 为0即可

1.8 gl.texImage2D(type, level, internalformat,format, dataType, image)

internalformat
图像的内部格式

gl.RGB

gl.RGBA

gl.ALPHA

gl.LUMINANCE 使用物体表面的
红绿蓝 分量的加权平均值来计算

gl.LUMINANCE_ALPHA

1.8 gl.texImage2D(type, level, internalformat, format,dataType, image)

- ◆ format 纹理的内部格式，必须和 internalformat 相同

1.8 gl.texImage2D(type, level, internalformat, format,dataType, image)

dataType

纹理数据的数据类型

gl.UNSIGNED_BYTE

gl.UNSIGNED_SHORT_5_6_5

gl.UNSIGNED_SHORT_4_4_4_4

gl.UNSIGNED_SHORT_5_5_5_1

1.8 gl.texImage2D(type, level, internalformat, format,dataType, image)

◆ image 图片对象

1.11 vec4 texture2D(sampler2D sampler, vec2 coord)

◆ sampler 纹理单元编号

◆ coord 纹理坐标

2. 流程

