

## 第20讲 | 如何在游戏中载入UI和菜单?

2018-07-10 蔡能

从0开始学游戏开发

[进入课程 >](#)



今天我们要在游戏中载入 UI 和菜单，在开始之前，我们先调整一下，我们之前讲过的游戏代码部分的内容。

首先我们需要更改游戏的背景图片，使之看起来更像是一款打飞机的游戏，而不是最早之前我们随使用的一幅山水图。我们先将游戏背景修改为正常的游戏背景，并且贴上飞机图像。



这里，我想到一个问题，之前有人留言问我，程序员需不需要有美术功底。我在这里说一下我的看法。如果你只是要做一个程序员，那我可以告诉你，不需要。但是，如果你不是只想做一个“码农”，你想有更多的发展，那各方面的知识，比如策划、美术，你都需要去了解。

## UI 的两种呈现形式

言归正传，我们需要在这个游戏画面上面，加一系列的内容来代表 UI 的呈现。UI 的呈现有两种形式：

第一种就是美术画好 UI 图片，然后直接贴图，用鼠标控制一系列的点击和按下操作；

另外一种是自己画 UI 界面，比如画一个框，并且在框里面写上文字。

这两种方式各有利弊。

就是，增加了游戏安装包的大小，毕竟 UI 是一幅图，只要是图就会有一定的体积，就会增加安装包的大小。

如果是程序员自己绘制的 UI 界面，**好处**就是主程序体积变得稍大一点，而游戏安装包不会变大。但是这样的**缺点**也很明显，就是程序员的工作量会增加很多。而且当游戏需要迭代的时候，或者界面需要更新的时候，程序员需要重新绘制或者重新编写 UI 源代码，大大增加了工作量。

我们现在是自己来开发，那我就讲一讲程序员绘制 UI 的方法。

我们通过模拟按钮的方式来摆放 UI 界面。**首先，我们要在 UI 界面上摆放一系列字符，我们要实现的效果是，只要使用鼠标点击到这个字符，就会变换字符的内容。**这个过程，我们会用到鼠标操作、绘制矩形、字体和字符绘制相关的知识，下面我就来具体给你讲。

## 鼠标操作

我们先来看一下鼠标操作的知识。在 Pygame 中，鼠标操作用到的模块是 `pygame.mouse`。在这个模块里面，点击事件的函数是 `get_pressed`。假如有返回按钮 1、按钮 2 和按钮 3 等等很多按钮，我们随便选一个点，假如说选了按钮 1，那代码可以这么写：

复制代码

```
1 pygame.mouse.get_pressed()[0]
```

这条语句不需要在事件语句里面操作，写在别的地方也可以，而且鼠标的操作在循环里一直是实时监测的。

## 绘制矩形

随后，我们要绘制矩形。**绘制矩形的目的是为了模拟一个按钮。**矩形绘制的代码是 `Rect`，但是我们需要绘制在一个 `surface` 上，这个 `surface` 需要新建，然而在 `pygame` 中，如果使用 `pygame.surface.surface` 初始化一个 `surface` 的话，不能指定位置，`x` 值是从 0 开始的。

还需要将这个子图层进行拷贝。所以，我们的代码看起来是这个样子的。

[📄 复制代码](#)

```
1 the_rect = Rect(200, 100, 150, 40)
2 block_surface = screen.subsurface(the_rect).copy()
```

首先第一行代码是建立一个矩形，分别是左侧起始值是 200，顶部从 100 开始，宽度 150，长度 40。随后，使用 screen 这个图层来建立一个子图层，子图层的大小按照 the\_rect 这个矩形大小来建立。随后的一个 copy 函数，是将子图层进行拷贝，在后续的使用中，不会出现锁定图层的情况。

## 绘制字体和字符

之后我们要开始编写文字处理的代码，字体我们要用到 pygame.font 模块，我们先初始化一个字体，这个字体在安装 pygame 游戏库的时候就包含在了 pygame 里面，我们直接就可以拿来使用。现在我们初始化字体，并且将字体大小调整到 25：

[📄 复制代码](#)


```
1 fnt = pygame.font.Font('freesansbold.ttf',25)
```

其中，freesansbold.ttf 是 pygame 安装的时候默认存在的 ttf 字体文件，随后，我们在第二个参数设置为大小 25。

我们拿到了 fnt 对象，然后使用这个对象调用 render 函数。这其实就是渲染，将文字渲染在屏幕，并且形成一个文字图层，函数原型是这样的：

[📄 复制代码](#)


```
1 Font.render(text, antialias, color, background=None)
```

 复制代码

```
1 tsurf = fnt.render(text, True, (255,255,255))
```


我们将颜色设置为白色，所以是 (255, 255, 255) 。

tsurf 是 render 返回的一个文字的图层 (surface)，我们之后要按照这个图层，来确定它的矩形框。

 复制代码

```
1 trect = tsurf.get_rect()
```

随后我们需要将文字摆在这个 trect 矩形框的中央，所以我们要进一步将 trect 确定在中央的位置，计算完中央的坐标值并且赋值过去。

 复制代码

```
1 trect.center = ((block_surface.get_width()/2),(block_surface.get_height()/2))
```

我们将最开始的复制的子图层的宽度和高度除以 2，就得到了中心点的位置。

最后我们要做的就是封装的函数内将 blit 部分包含进去，现在我们来看一下完整的包装函数代码。

 复制代码

```
1 def text_out(text):
2     fnt = pygame.font.Font('freesansbold.ttf',25)
3     tsurf = fnt.render(text, True, (255,255,255))
4     trect = tsurf.get_rect()
5     trect.center = ((block_surface.get_width()/2),(block_surface.get_height()/2))
6     block_surface.blit(tsurf, trect)
```



我们看到，在函数返回后，我们 blit 了 block\_surface 这个被修改的图层。

随后我们在游戏的大循环里面，需要判断鼠标的点击事件，我们之前所定义的矩形，代码是这样：

复制代码

```
1 the_rect = Rect(200, 100, 150, 40)
```

所以这 x 的起始位置和结束位置是 200 和 350，y 轴的起始位置和结束位置是 200 和 240，为什么 y 轴也是 200 开始呢？因为起始点 200 既是 x 轴开始的点也是 y 轴开始的点。我们在代码里面这么判断鼠标的点：

复制代码

```
1 txt = "Pause"
2 x, y = pygame.mouse.get_pos()
3     if pygame.mouse.get_pressed()[0]:
4         if x >=200 and x <= 350 and y >= 200 and y <= 240:
5             txt = "Clicked"
```

我们将 txt 定义为 Pause 字符串，并且判断是不是鼠标左键点击的。如果是的话，判断是不是在 x 轴和 y 轴进行点击，如果不是的话，就将 txt 改为 Clicked。

随后我们绘制按钮框，并且将按钮框背景设置为绿色，然后输出文字，并且绘制。

复制代码

```
1 screen.blit(block_surface, (200, 200))
2     block_surface.fill([0,20,0])
3     text_out(txt)
```

将 block\_surface 这个图层绘制在 (200,200) 的坐标点，并且将之涂为绿色，最后调用 text\_out 函数，由于 text\_out 里面已经编写了 blit 函数，所以不需要再次 blit 了。



我们看到了一个绿色的按钮放置在屏幕上，并且有一个白色的 Pause 字样放在按钮上，如果是鼠标左键点击在这个按钮上，就会变成 Clicked 字样。

到这里，你可能会问了，为什么没有解释怎么输出中文呢？

在这种情况下，输出中文有两种解决方案。

第一种是比较底层的方案，就是**根据中文进行点阵绘制**，这需要很底层的代码操作，效率也不太高，所以这种方案我们不作讨论。

第二种就是**改变字体**，我们可以在初始化字体的时候，下载一些网上的中文字体先进行尝试，或者我们可以使用系统字体。在使用系统字体的时候，我们可以使用 SysFont 来初始化字体。

另外，还需要修改一个地方。我们需要在 Python 代码最开始的地方添加编码方式，并且将中文文字前面添加 u 字样来告诉解释器是 Unicode 的。

[复制代码](#)

```
1 # encoding: utf-8
2 txt = u" 暂停 "
```

我们来看一下效果：



至此，中文的输出也已经完成了。



作为菜单这些高级操作，比如点击出现下级菜单、隐藏菜单这些动态效果，可以使用图片的方式来制作菜单，并且进行载入。如果使用程序来编写菜单的效果，工作量就太大了。而如果是图片形式的话，只需要载入并且控制鼠标点击和绘制子菜单就可以了。

## 小结

今天我们学习了 UI 部分的编写以及文字的输出、鼠标的移动和抓取鼠标按键的实现，我来总结一下今天的内容。

1. 不管何种类型的游戏引擎，鼠标的操作基本有 3 个值需要控制，左键、中键和右键。
2. 按钮可以用常规的方法来绘制。如果想做出更好的效果，比如画框，可以在框里面再画一个小框，看起来像是有凸出感觉的样子。这就需要你平时多注意观察一些细节，然后去分析如何用我们讲过的简单的操作来实现这些内容。
3. 在 2D 游戏中，很多游戏引擎都不支持中文的输出。如果要输出中文，假如你的引擎支持，那你可以使用系统字体或者其他中文字体；如果引擎不支持，可以使用一个一个点阵绘制的方式在屏幕上绘制中文。最后一种方式，也就是比较极端的方式，那就是使用图片来直接贴上中文字符，这种方式直接粗暴，但是图片资源量太大，而且如果你要在游戏中进行网络聊天，这里面其实还是没有从根本上解决中文输出的问题。

现在给你留一个小问题。

如果让你在上述的代码中，将按钮变成菜单，也就是点击按钮，就在下方出下一个下拉框，你会如何实现？

欢迎留言说出你的看法。我在下一节的挑战中等你！

# 从0开始学游戏开发

你的游戏开发入门第一课

蔡能

原网易游戏引擎架构师  
资深游戏底层技术专家



新版升级：点击「 请朋友读」，10位好友免费读，邀请订阅更有**现金**奖励。

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 第19讲 | 热点剖析（四）：安卓端和苹果端游戏开发有什么区别？

下一篇 第21讲 | 如何用鼠标和键盘来操作游戏？

## 精选留言 (4)

写留言



野山门

2018-07-12

1

> 这里，我想到一个问题，之前有人留言问我，程序员需不需要有美术功底。我在这里说一下我的看法。如果你只是要做一个程序员，那我可以告诉你，不需要。但是，如果你不是只想做一个“码农”，你想有更多的发展，那各方面的知识，比如策划、美术，你都需要去了解。

...

展开



青何 | ...

2018-07-31

按钮是黑色的啊？



下载APP



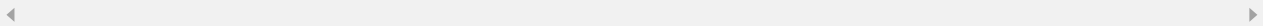
wusiration

2018-07-17



我会设置一个标志位来记录是否被点击，如果被点击，标志位置为1，渲染下拉菜单；如果未点击，则将菜单出现的位置重新渲染一次。

作者回复: 说得有点简单，最主要要将绘制内容放在类里独立调用



云学

2018-07-13



这些坐标计算工作量实际中是不是很大？看起来好像不太灵活

展开 ∨