如何找全景图

1. 概念描述

本项目是使用的模块叫做Pannellumjs，它是使用主流的3D全景模Threejs拓展而出的，他是一个可以免费引用修改的js模块，而我们使用的是我之前自行进行二次修改，去掉了不必要的一些图标和广告，界面更简洁，更简单的一个版本。

它的主要功能是使用图片资源在网页建立3D场景

Pannellum支持四种处理全景图片的方式：equirectangular,partial,cubic,multi-resolution

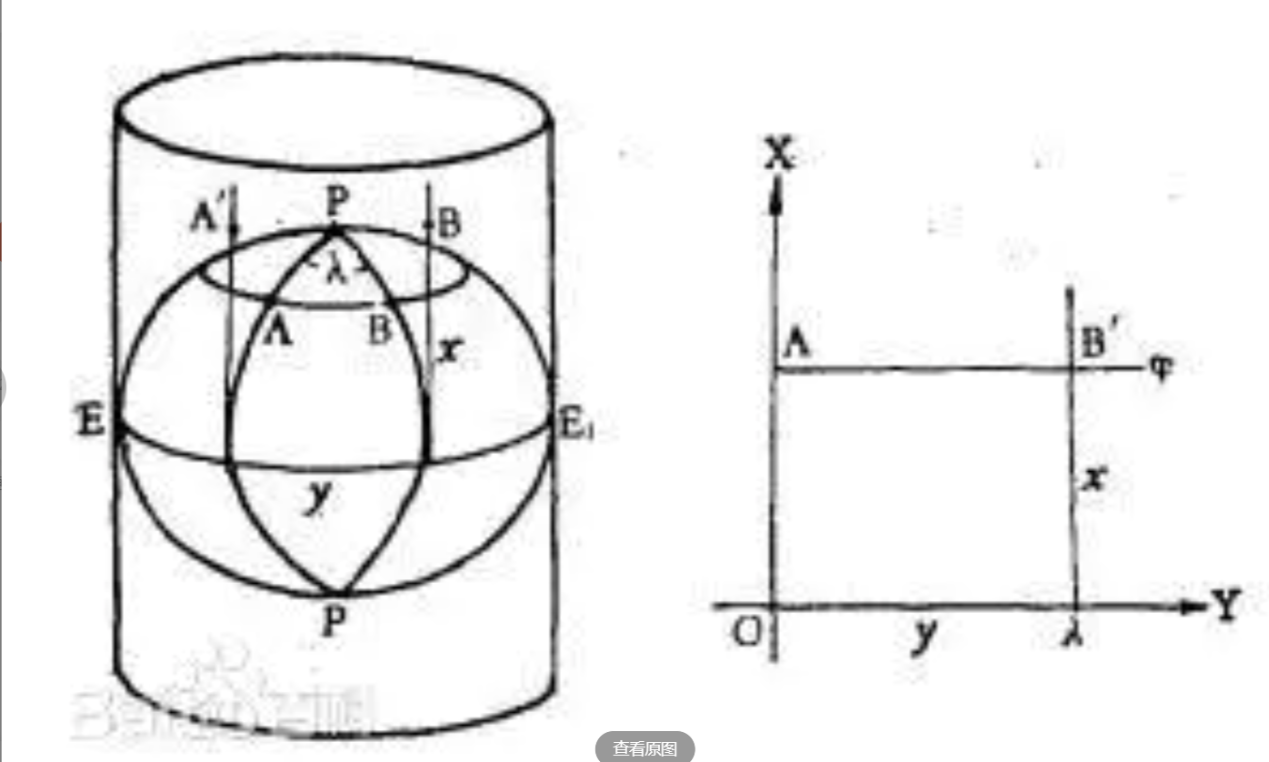
其中第一和第二个是针对单张全景图片的处理方式，第三和第四是针对拆分全景图片的处理方式，接下来分别进行说明：

只要看第一种的描述即可，后面的不用了解也行

1、equirectangular：圆柱投影，百度百科解释如下：

假想一个圆柱与地球相切或相割，以圆柱面作为投影面，将球面上的经纬线投影到圆柱面上，在正常位置的圆柱投影中，圆柱面展平后纬线为平行直线，经线也是平行直线，而且与纬线直交。

圆柱投影的变形仅随纬度而变化，即在同纬线上各点的变形相同而与经度无关。



圆柱投影

简单来说就是把一个球体的表面使用圆柱体投影到圆柱体的侧面上，这个过程的逆过程就变成了一张全景图投影到球体上，这样就能够实现全方位的全景体验，这也使得全景图跟普通的"大图"不一样，全景图还需要包含上下两个方向的图像信息，具体表现就是全景图有“顶”和“底”的

全景图

而如果只是使用普通的大图，就会出现图像的边缘无法重合以及顶部底部没有图像的问题：

关于圆柱体投影的更多资料读者可以参考[维基百科](https://link.jianshu.com?t=https:/en.wikipedia.org/wiki/Equirectangular_projection" \t "_blank)和Pannellum的[圆柱体投影算法](https://link.jianshu.com?t=https:/github.com/mpetroff/pannellum/blob/master/doc/pannellum-equirectangular-projection-reference.pdf" \t "_blank)。

使用方法：在页面中嵌入

<iframe width="480" height="390" allowfullscreen style="border-style:none" src="../src/standalone/pannellum.htm?panorama=../../examples/school.jpg&amp;title=全景图" ></iframe>

其中panorama参数是需要显示的全景图片路径，这里还设置了title为“全景图”，显示效果就是上面的全景图效果，独立页面显示全景图就在页面后添加viewer：

pannellum.htm

由于Pannellum使用闭包的方式向外提供接口，因此就可以将pannellum作为一个类来操作，viewer（arg1，arg2）作为构造函数来构造全景显示器，arg1就是页面中用于展示全景图的<div>，arg2是一系列参数的集合，定义了全景的各项显示参数，上图定义了在id为‘container’的<div>中显示school全景图，全景图投影类型为圆柱体投影（equirectangular）。

下面的描述可以不用看，直接跳到举例即可

2、partial 局部圆柱体投影

在理解partial投影之前，需要知道Pannellum的投影坐标系，坐标系很简单，一图以概括：

投影坐标系

圆圈中的直线代表视角方向，水平方向用yaw值表示，取值为180度到-180度，竖直方向用pitch值表示，取值为90度到-90度，yaw和pitch两个值共同表示整个投影坐标系。在实际使用中可以通过setYaw(),getYaw(),setPitch(),getPitch()等方法将视角移动到指定的坐标处，也可以获取当前视角对应的坐标位置。接下来需要注意的是Pannellum提供了setPitchBounds（number[]）方法和setYawBounds（number[]）方法，这两个方法的作用是限制视角的移动范围，例如如果想将视角限制在水平yaw值0~140的范围内，设置如下：

设置水平视角范围

限制水平视角

同理在竖直方向上只需要setPitchBounds就可以了。

Pannellum除了可以限制视角的移动范围，还可以限制全景图的覆盖范围，通过参数haov、vaov、vOffset来设置，haov（horizontal angle of view）表示水平展示的图片的角度大小，vaov（vertical angle of view）表示竖直方向的角度大小，vOffset表示竖直方向的偏移度，上图以助理解：

设置角度

实际效果图

可以看到跟setXXBounds方式不同，该方法直接设置了全景图片整个所占的面积和角度，vOffset设置为30，整个图像向上偏移了30度。需要注意的是只能在构造方法中设置这三个参数，这三个参数只对“equirectangular”类型的图片有效，且设置后无法改变。这样的投影方式就是partial投影。可见圆柱体投影就是partial投影设置haov=360，vaov=180，vOffset=0的结果。

3、cubic 立方体投影

立方体投影是针对拆分全景图片的投影方式，即将整个投影空间看作一个立方体，具有前后左右上下6个面，分别将各个拆分全景图片渲染在对应的各个面上，在感官上也能够达到柱面投影的效果。在使用cubic投影方式处理全景图片的时候需要配置json文件，在json中配置图片路径、标题、作者、标注点等信息，首先根据上文的拆分后的全景图配置json文件：

然后在创建全景图的时候将配置文件设置为cube.json：

cube.json

嵌入页面：

<iframe width="480" height="390" allowfullscreen style="border-style:none;" src="../src/standalone/pannellum.htm?config=../../examples/cube.json"></iframe>

独立页面：

独立展示页面

注意：使用的各个面的图片必须为正方形且大小相等！且需要按照特定的顺序传入图片。

Pannellum传入图片的顺序是：前、右、后、左、上、下。如果不按照顺序传入相对应面的图片就会出现错位的情况。设置好后的效果如下：

官方示例图

关于CubeMap的更多资料读者感兴趣的话可以查看[维基百科](https://link.jianshu.com?t=https:/en.wikipedia.org/wiki/Cube_mapping" \t "_blank)。

从上图可以看到使用cubic和使用equirectangular的方式效果差不多，cubic的投影坐标系跟equirectangular坐标系是一致的，也可以通过设置视角坐标来操作，但是cubic要求图片的顺序、大小保证一致以及对各个图片之间的重合度要求更高，使用起来更加复杂，那么既然效果差不多为什么还要使用这种方式呢？

首先，该方式有利于粗加工或未加工的图片展示为全景图片，因为cubic方式可以直接采用源图片来进行投影拼接成全景图，所以只要能够在用相机拍摄图片的时候固定好各个方向镜头的角度，拍摄得到的图片只需要经过粗加工（例如裁切成正方形）就可以直接用来展示，免去了再将所有图片合成为一张大的全景图的过程。

其次，cubic方式比equirectangular方式更加灵活，如果全景图的某一个区域出现了“坏点”，“黑洞”，或者图像撕裂等问题，equirectangular的全景图就需要重新合成一张新的全景图，或者对照着圆柱体投影来修图，过程很麻烦，而对于cubic方式来说就只需要替换掉对应面上的图片即可。

cubic的另一个作用就是能够提供多清晰度全景图片，这就是Pannellum的第四种方式：multi-resolution。

4、multi-resolution

顾名思义，就是可以提供不同清晰度的全景图片，Pannellum能够根据缩放的程度（改变Hfov）来显示不同清晰度的照片，示例图如下：

错误!未指定文件名。

multires Test

示例图中可以看出根据缩放的层级来分别加载不同清晰度的图片，在使用中可以通过鼠标滚轮、点击“+”按钮、手机双指缩放等操作来改变缩放层级，在Pannellum里表现为改变Hfov（Horizontal field of view）参数，取值为50~120，默认值为100。

构建multi-resolution的全景图片方式和cubic方式相同，不同的是type为multires，并且需要改变json配置文件：

myltires.json

在“multires”节点中，将每一个最小的图片看做瓦片（tile），立方体的每一个面都是由若干个瓦片组成，接下来分别说明multires节点的参数。

basePath设置一个公用路径，后面的path只需要写相对于basePath的相对路径即可，这里需要注意的是path的值，path的值是一个格式化的字符串，其中 %l 代表缩放等级，%s 代表立方体的面 ，%x 和 %y 代表了该瓦片在这一个面中的位置，其文件结构如图：

当缩放等级为1时，就会在“1”文件夹中将各个面对应的图片投影到对应的立方体面上（对应%s），然后根据每一个瓦片的坐标来确定瓦片在每一个面中的位置，这里由于坐标是（0,0），且每一面只有一个瓦片，所以是铺满整面，多个瓦片的效果可以看上面的实例。

fallbackPath的作用就是当某些浏览器不支持multires功能的时候能够回退到fallbackPath指定的文件夹下加载默认的全景图片。

extension指定瓦片的图片格式，不需要加前面的 “.” 符号。

tileResolution指定瓦片的像素，单位为px。

maxLevel指定最多支持几个缩放等级。

cubeResolution指定立方体面的总像素，单位为px。

如此就能够根据不同的缩放等级来显示不同的清晰度的图像，这样的好处是如果要加载一张4k的全景图，不需要一次性就将整个全景图都加载进来，可以先加载一个缩放等级低的全景，然后当使用者进行缩放查看细节的时候再加载清晰度更高的图像，这样就可以明显提高加载速度，避免因为图片过大使得加载时间过长和不必要的流量浪费。不足之处就是需要为一张全景图额外准备不同清晰度的图片，增加了图片处理的工作量，也增加了图片存储的空间占用。（[全景组件Pannellum解析与实践 - 简书 (jianshu.com)](https://www.jianshu.com/p/fdbcb551d75b)）

1. 什么样的全景图可以用在本项目中

强烈建议使用Equirectangular图或被转换成的Equirectangular图，举例三个



推荐理由就是方便制作，而且好找

如果特别希望我使用其他格式进行制作，那么请找好对应的资源

1. 去哪找

百度搜索，拍照（我并不确定手机相机的全景拍照功能是否可用），或者找到某种方式合成

孙达明2022.4.14

主页[tyza66 (洮羱芝闇) (github.com)](https://github.com/tyza66)