# 三维数据格式介绍

## 定义

三维数据，顾名思义，三维空间的信息表达，描述的是在XYZ三个维度上的空间信息。区别传统的二维数据，三维数据需要考虑在Z轴方向上的具体内容。

## 类型与格式介绍

### 2.1点云数据

点云是某个坐标系下的点的数据集。点包含了丰富的信息，包括三维坐标X，Y，Z、颜色、分类值、强度值、时间等等。点云格式一般有以下几种：las、laz、pts、ply、ptx、xyz等。最常使用的点云数据是las格式。

### **2.2人工模型**

人工模型一般由专业建模人员使用建模软件如3DMax、Maya等进行建模后导出的数据。常用格式一般有：obj、dae、3ds、fbx等。较常使用的是obj与fbx。

### **2.3倾斜摄影数据**

将一组从不同的角度拍摄的高重叠数据照片作为输入源，使用三维自动建模软件，输出的高分辨率的带有真实纹理的三角网格模型，即是倾斜摄影数据。可导出osgb、obj、3mx等数据格式。一般最常用的是osgb格式。

### 2.4 BIM数据

BIM一般指建筑信息模型。建筑信息模型（Building Information Modeling）是建筑学、工程学及土木工程的新工具。常用格式一般有：ifc、rvt、dgn、stp等。

### **2.5地形数据**

常用格式有tif、geotiff等。

## **内容与解析**

### **3.1点云格式与解析**

#### 3.1.1 las

Las文件用来包含LiDAR点数据记录。Las格式为二进制数据，包含一个头文件区，变长记录区和点记录区。

具体格式介绍可参考：<https://max.book118.com/html/2017/0612/113958010.shtm。>

常用解析库有liblas、laszip等。

Liblas：[https://github.com/libLAS/libLAS。](https://github.com/libLAS/libLAS。Laszip)

[Laszip](https://github.com/libLAS/libLAS。Laszip) : [https://github.com/LASzip/LASzip。](https://github.com/LASzip/LASzip。两个都是开源c++库。)

[两个都是开源c++库。](https://github.com/LASzip/LASzip。两个都是开源c++库。)

### **3.2模型与解析**

#### **3.2.1 OBJ**

Obj格式作为模型数据的表达，可以说是最简单也是最自由的人工模型数据。一份完整的obj数据一般由.obj + .mtl + 纹理贴图组成。Obj数据记录顶点、纹理、法线、面索引信息以及每个部件所使用的材质名称。Mtl文件是obj的材质信息文件，记录每个部件的材质信息，其中包含透明度、光照信息、及材质所使用的纹理图片相对路径。纹理贴图即为图片文件，一般较常使用的格式为jpg、png、tga等。

Obj格式介绍：<https://blog.csdn.net/qq_26900671/article/details/81739977。>

Mtl格式介绍：<https://www.cnblogs.com/ch06src/p/3431751.html。>

**OBJ文件：**

# Blender v2.81 (sub 16) OBJ File: ''

# www.blender.org

mtllib test.mtl

o Cube

v 1.000000 1.000000 -1.000000

...

vt 0.625000 0.500000

...

vn 0.0000 1.0000 0.0000

...

usemtl Material

s off

f 1/1/1 5/2/1 7/3/1 3/4/1.

...

**v**: 表示本行指定一个包含X、Y、Z坐标值顶点

**vt**: 表示本行指定一个U、V值纹理坐标

**vn**: 表示本行指定一个包含X、Y、Z的法向量

**g**: 表示组名，指定从此行之后到下一个以g开头的行之间的所有元素结合到一起

**f**: 表示本行指定一个三角面，一般通过顶点、纹理及法向的索引来定义

**usemtl**: 此关键字后参数为材质名称

**mtllib**: 此关键字指定了obj文件所使用的材质库文件(mtl文件)的文件名称

**mtl文件：**

# Blender MTL File: 'None'

# Material Count: 1

newmtl Material

Ns 323.999994

Ka 1.000000 1.000000 1.000000

Kd 0.800000 0.800000 0.800000

Ks 0.500000 0.500000 0.500000

Ke 0.000000 0.000000 0.000000

Ni 1.450000

d 1.000000

illum 2

**newmtl** 后跟材质的名称（一个MTL文件可以含有多个材质）

**Ka**: 材质的环境光（ambient color）

**Kd**: 散射光（diffuse color）

**Ks**: 镜面光（specular color）

**Ke**: 放射光（emissive color）

**Ns**: 材质的光亮度

**d**，**Tr** 均可用于定义材质的Alpha透明度

**sharpness**： 材质的锐度（sharpness）

**illum**： 照明度（illumination）后面可接0~10范围内的数字参数

**map\_Ka**、**map\_Kd**、**map\_Ks**材质包含的环境，散射和镜面贴图路径

由于obj的格式为文本格式，所以obj的解析既可以自己编写，也可以使用开源库，比较简单的解析库有tinyobjloader : <https://github.com/tinyobjloader/tinyobjloader。>(由于比较简单，所以对于obj这种自由度比较高的格式兼容度比较低，容易出现错误或者信息缺失等情况)。

一般比较完整的解析库可以使用osg或者assimp。其中assimp的读取速度相对osg更快。

OpenSceneGraph : <https://github.com/openscenegraph/OpenSceneGraph>

Asset Importer : <https://github.com/assimp/assimp>

#### **3.2.2 FBX**

FBX是由AutoDesk提供的模型文件格式，在Unity于UE4引擎中被广泛使用。FBX格式为二进制文件格式。AutoDesk公司提供FBX SDK解析FBX模型。

FBX SDK：<https://www.autodesk.com/developer-network/platform-technologies/fbx-sdk-2019-0>

FBX模型结构：https://blog.csdn.net/jxw167/article/details/81630899

另外在osg与assimp开源库中也有内置相关的解析插件，在编译开源库时可以配置对应的内容，即可使用。

### **3.3 倾斜模型与解析**

#### **3.3.1 OSGB**

Osgb数据是二进制文件。osgb数据一般情况下生成出来都是自带LOD，自带空间分块，所以获取到的倾斜数据一般都是很多个Tile\_XXX\_XXX类似文件夹组成。osgb数据的解析库为osg开源库。Osgb数据使用的一般是节点式存储，节点间存在父子关系拓扑。Osgb数据的一般节点组成如图：



如上图所示，PageLOD节点存储着节点信息以及子节点的文件名等信息。Osg::Geometry节点存储网格数据，包括顶点坐标、纹理坐标、面索引，一般的倾斜摄影数据不带法线信息。Osg::StateSet节点存储材质纹理，有的数据StateSet节点也会挂接在osg::Geometry节点下。StateSet节点可以获取到材质信息（osg::Material）和纹理信息（osg::Texture）。Osg::Material存储光照信息及透明度。Osg::Texture存储纹理图片数据及采样器参数。

### **3.4 BIM数据与解析**

#### **3.4.1 IFC**

IFC是一个数据交换标准, 用于不同系统交换和共享数据。IFC格式数据一般由EXPRESS语言编写，属文本格式，但对于人工读取非常不友好。EXPRESS语言是由STEP（Standard for the Exchange of Product Model Data）开发的概念性语言，STEP是计算机可读的用于交换和描述产品制造信息的标准，正式代号为ISO 10303 EXPRESS是用于面向对象的信息描述语言，处于STEP中基础与核心的地位。它将IFC模型构描述为：规范的类，与类相关联的属性，类的约束，以及类与其他约束之间的关系。然而，EXPRESS语言仅适合软件的读写，在人工读取上存在一定的困难。因此，EXPRESS-G视图以图表式的表达被发展起来，并得到了普遍应用。

参考：<https://www.cnblogs.com/herd/p/6434220.html。>

基于C++的IFC格式的解析库：

Ifcopenshell 第三方开源库，效率极低

第三方开源库，效率极低

ifc文件

Ifcengine 第三方商业库，无源码，效率高

第三方商业库，无源码，效率高

第三方商业库，无源码，效率高

第三方商业库，无源码，效率高

Ifcplusplus 第三方开源库，效率适中,有缺陷

第三方开源库，效率适中,有缺陷

IFC数据非常鲜明的特点就是能很好的存储属性信息。目前所提供的IFC数据仅有材质信息，没有贴图信息。

### **3.5 地形数据与解析**

#### **3.5.1 TIFF**