自动语义标注研究内容

# 根据三维模型的结构来分析语义信息

提出一种算法来抽取并分析三维模型中的视觉和语义信息。这其中包括：分析并识别出三维文档中包含的视觉信息和特性。针对三维模型文档的信息特性来推测出同类的语义。

# 通过分析模型转换的二维图像并比较模型的相似度

通过两个三维模型或者整个三维场景的图像识别比较算法，计算出图像之间的相似度，从而得出模型之间的异同点，找出相似的地方。然后使用已有的语义标注信息，进行自动的三维模型自动语义标注。

## 模型转二维图像比较比较方法

1. 通过前台使用Javascript实现算法以及HTML Canvas分析二维图像的RGB & Alpha值以及亮度值。

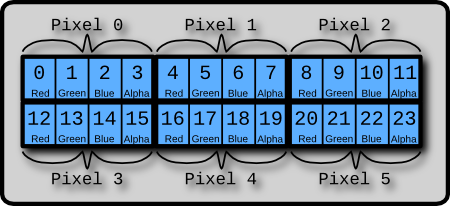


图1. 图像的像素颜色值定义

1. 图像比较方法使用像素扫描方式来对比两个像素结构中的颜色值是否相等。
2. 每次循环若像素不相同则

* 生成相差的像素点。
* 相同值点叠加1

图2. 模型转二维图像比较结果

# 

图3.详细比较结果（相似度，处理时间）

**总结：通过上述的比较算法。虽然结果是近似性的方法但可以因为基于前台处理，所以效率比较高。为服务器端减少了处理压力以及网络传输流量问题。解决了同时比较大量图像数据。提高了三维模型的自动语义标注工作的效率。**

# 构建原型系统，可以与整个3d建模平台整合并且自动对模型分类，聚类。

构建出原型系统。按照上述所说的思想和算法，搭建基于三维模型场景的语义标注的原型系统Leap Studio。该系统中包括三维模型文档的信息和特性提取。通过模型的比较特性（如颜色，材料，种类。。。）来对它进行自动语义标注。系统通过将新建的三维模型转换为canvas图像，与数据库中的样本模型进行图形比较找出异同点。从而为新模型自动生成语义标注。

# 两种语义标注方法比较。

|  |  |
| --- | --- |
| 图像识别自动语义标注 | 比较模型相似的自动语义标注 |
| D:\method1.jpg | D:\method2.jpg |
| 方法1 | **方法2** |

图4. 两种模型语义标注比较

总结：通过以上的工作流程图，可看出两种基于三维模型的自动语义标注方法不同之处。虽然从实现方法是比较独立的两种方法。但实际语义标注系统中是可以**结合使用**的。

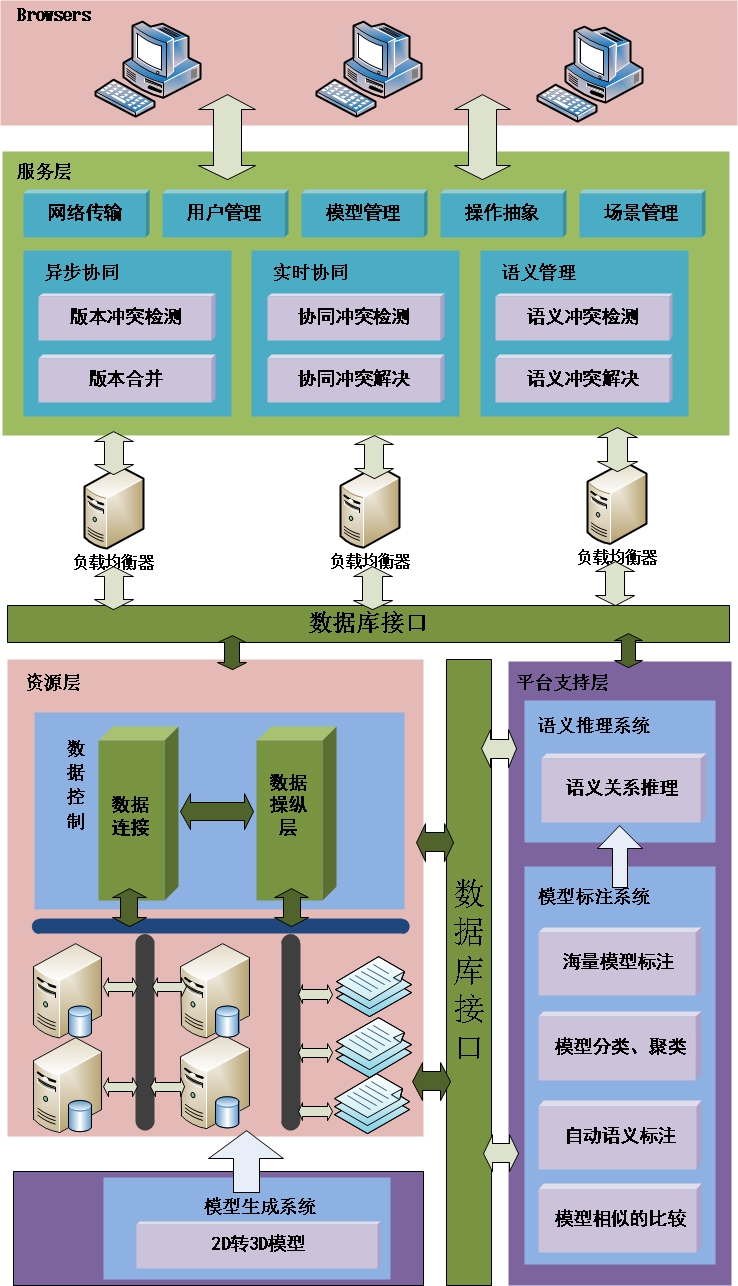
例如当我们需要将一个三维模型转换的二维图像与数据库的其他三维模型做比较的时候。假设三维模型数据库存放大量成千上万个模型数据。显然如果将一个三维模型与整个数据库里面的所有模型进行比较，为了同步他们之间的语义标注信息。这样的效率会特别慢，没有达到要解决海量数据的意义。

为了解决这样的问题，我们完全可以结合以上两种语义标注方法使用：

**步骤一：**首先要为导入的三维模型进行多视角截图，然后通过**方法1**从多个模型生成二维图像进行图像识别方法。并得到语义标注信息（见图方法1）。

**步骤二：**当三维模型拥有了语义标注信息之后。可对其进行分类，聚类模型操作。从**方法2** 可看到比较模型相似的自动语义标注方法的输入资源是从**方法1**图像识别自动语义标注方法得到的结果。之后通过访问数据库，在海量模型数据中找出类似同样的模型数据类型。得到更为相关性的模型数据。然后进行原三维模型与其他同类模型的相似度比较。再次进行语义标注，同步相似度比较高的模型之间的语义信息。

# 本语义标注系统与整个三维模型系统的关系。



**三维模型语义标注系统**

图5. 整个三维模型系统的架构图

从图5中可看到本语义标注系统与整个三维模型系统之间的关系。本系统是置于整套系统的支持层。当大量的三维模型通过建模平台建出来之后，在数据库里面未定义或者未聚类，分类。所以没有达到实际应用，比较难管理和使用。在整个三维模型系统的架构图中可看到三维模型先存储在公用数据库里面。然后通过平台支持层提供的API接口来对模型定义，添加语义标注以及对模型分类，聚类。

**注明**: **（数据库接口作用）**这里图5中数据库接口就是将访问数据库层的操作，方法都封装起来。通过访问接口来获取或添加数据。这样确保数据库访问的安全性，封装性。而且这样的设计架构容易对外开放并且扩展。实现外面子系统不需考虑或依赖原来数据库的结构。