

基于浏览器的机械产品虚拟装配展示软件系统设计与实现

韩欣雨, 张应中

(大连理工大学, 辽宁 大连 116023)
✉xinyudlut@163.com; zhangyz@dlut.edu.cn



摘 要: 随着互联网技术的发展, 在浏览器网页上展示和操作机械产品装配模型有了非常广泛的应用。针对应用需求, 采用新兴的WebGL(Web图形库), 提出了基于浏览器的机械产品虚拟装配展示方法。通过加载服务端STL(立体光刻)三角网格装配模型, 对其装配约束关系自动重构, 基于WebGL在浏览器上实现装配零部件导航, 基于装配约束关系的装配爆炸图自动生成和展示, 并以一级减速器为例进行展示系统实验, 验证了方法的可行性。

关键词: WebGL; 三维装配; 装配展示; 装配约束

中图分类号: TP393 **文献标识码:** A

Design and Implementation of a Browser-based Virtual Assembly Display Software System for Mechanical Products

HAN Xinyu, ZHANG Yingzhong

(Dalian University of Technology, Dalian 116023, China)
✉xinyudlut@163.com; zhangyz@dlut.edu.cn

Abstract: With the development of Internet technology, it is widely used to display and operate mechanical product assembly models on web browser. For application needs, this paper proposes a browser-based virtual assembly display method for mechanical products by using emerging WebGL (Web Graphics Library) technology. By loading the server-side STL (Stereolithography) triangular mesh assembly model, the assembly constraint relationship is automatically reconstructed. Assembly component navigation is realized on the browser based on WebGL. Assembly explosion diagram based on assembly constraint relationship is automatically generated and displayed. The feasibility of the method is verified by the experiment of display system with the first-stage reducer as an example.

Keywords: WebGL; three-dimensional assembly; assembly display; assembly constraints

1 引言(Introduction)

机械产品一般由众多零部件组成, 新产品的推广、用户的产品体验、产品的维护和维修等应用都需要产品装配的三维展示, 例如, 产品的高质量渲染、装配爆炸、装配拆卸等。传统的基于计算机辅助设计(Computer Aided Design, CAD)模型的装配展示需要安装复杂的CAD软件系统, 操作复杂, 成本高昂。随着计算机技术的发展, 互联网已经在各个行业得到广泛的应用, 如果仅仅通过浏览器就可以展示和操作复杂的机械产品, 用户在客户端便可体会到定制产品逼真的3D效果, 将会促进用户与产品的交互, 大大方便和促进产

品三维数字化展示和互联网+技术的应用。

近年来, HTML5融合WebGL为浏览器提供了一个硬件3D加速渲染功能, 从而借助系统显卡在浏览器里流畅地展示3D场景和模型, 可以创建复杂的导航和数据可视化^[1-2], 为浏览器展示和操作复杂机械产品三维装配提供了支撑。为此, 国内外学术和产业界开展了大量的相关研究, 戴秀海等^[3]采用WebGL中的Three.js开发了一个镀膜机展示平台, 但只能提供产品的三维渲染显示; 姚震等^[4]也采用Three.js(3D引擎)开展了机械产品三维展示的研究, 主要完成了场景的配置和三维模型的导入; SUN等^[5]基于WebGL技术开发了一个轻量级

的跨平台铸造Web3D系统。实际上, WebGL技术在建筑、地理信息系统取得了更成功的应用, ZHOU等^[6]基于WebGL开放一个开放式建筑管理信息跨平台在线可视化系统。WebGL为浏览器显示高质量三维模型提供支撑, 但是对于由众多零部件组成的机械产品开展更深入的应用, 需要有产品的装配模型, 包括零部件的层次组成、零部件之间的装配约束关系和各种物理约束关系等。一个复杂机械产品的数字化装配模型, 包括几何模型十分复杂, 需要功能强大的CAD造型设计软件来完成, 但不同的CAD软件都采用各自的模型存储格式, 一般提供较低级的边界几何信息。因此需要对CAD软件完成的装配模型进行重构, 构建出产品的装配模型, 才能更好地开展各种深入的应用。

本文针对机械产品三维展示需求和特点, 基于WebGL技术研究了一个基于浏览器的机械产品装配三维展示方法, 通过对CAD软件提供的装配模型STL几何数据的导入和装配约束关系的自动识别, 重构出原产品装配模型, 在浏览器上实现了机械产品装配的三维虚拟展示, 包括模型的材质渲染、装配爆炸、零件拆装、零件导航等功能, 从而使用户在客户端便可体会到定制产品逼真的3D效果。

2 系统架构设计(System architecture design)

2.1 应用需求分析

系统的创建用于实现在浏览器上展示和操作服务端的机械产品模型, 目前应用主要需要实现以下功能:

(1) 产品三维模型展示

三维模型展示是指在浏览器上无插件加载服务端的产品三维模型, 对产品进行真实感光照渲染、材质渲染和各种旋转、缩放变换操作等。此外还需要对模型进行各种交互操作, 例如, 零件的选择、零件选择导航操作等。因此, 浏览器要能够加载产品几何模型, 具备渲染、光照和视点变换的场景功能。

(2) 产品虚拟装配展示

产品装配是将众多零部件通过装配约束方式结合在一起, 对产品虚拟装配展示的主要方式有装配爆炸、装配拆卸等, 需要模拟真实零部件状况, 零件移动和旋转时的干涉检查是必需的。

2.2 系统总体架构

本文根据上述应用需求分析, 基于WebGL技术, 提出如图1所示系统总体架构。

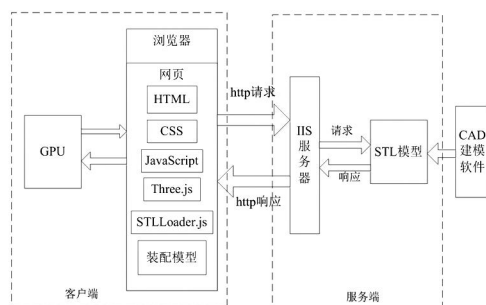


图1 系统总体架构

Fig.1 Overall system architecture

系统架构分为如下两大部分:

(1) 浏览器客户端

在浏览器客户端采用WebGL的第三方库Three.js创建三维展示场景, 加载产品三维模型文件, 并为后续装配展示提供基础。浏览器客户端主要包括HTML5、层叠样式表(Cascading Style Sheets, CSS)、JavaScript、Three.js及重构的产品装配模型, 基于JavaScript语言及Three.js在VS Code(Visual Studio Code)开发平台实现。

在浏览器客户端基于JavaScript中的树形组件tree及raycaster射线构建产品零部件导航界面, 并通过Three.js中的相机控件实现对产品的移动、旋转、拆装等交互操作。

(2) 产品模型服务端

要展示的机械产品装配模型通常放在服务端。产品装配模型十分复杂, 一般通过CAD建模软件设计完成, 不同CAD建模软件以不同的装配模型格式存储。Three.js提供一个加载STL文件功能, 而STL格式文件是将零件几何模型以三角面的形式进行描述, 即一个STL文件是由一组无序的三角面组成的。STL格式在工业界得到广泛的应用, 主流CAD建模软件都能将几何模型生成STL格式文件。本文将要展示的产品装配模型按照STL格式存储在服务端, 客户端通过资源定位器就可以加载到浏览器上展示。

2.3 基于浏览器的三维场景

本文通过WebGL的第三方库Three.js构建虚拟装配展示系统, 包括渲染器、场景、相机、光源等的设置^[7]。

(1) 渲染器设置

采用Three.js中的WebGLRenderer渲染器进行场景渲染, 它通过HTML5中的<canvas>标签直接内嵌于网页中进行渲染, 并能够利用GPU硬件进行加速, 达到提高渲染性能的效果。

(2) 场景设置

场景是一个3D空间, 用来放置物体、相机及灯光。在Three.js中, 场景用THREE.Scene表示。

(3) 相机设置

为了保证机械产品模型展示的真实性, 采用透视相机(PerspectiveCamera)。它采用的是透视投影, 视景体为锥台形区域, 投射结果与几何体角度和距离有关, 会产生“近大远小”的效果。

同时, 为了保证加载模型能够居中渲染, 需要借助相机对象的LookAt()方法实现, 即借助包围盒BOX3计算模型几何中心在Three.js坐标中的世界坐标, 并将该坐标作为LookAt()的参数。

(4) 光源设置

为了保证较好的渲染效果, 便于后续模型放大、缩小、旋转时更好地显示模型, 需要多种光源配合, 因此采用两种点光源(PointLight)和环境光源(AmbientLight)配合的方法。

2.4 产品三维装配模型创建

Three.js中包含多种二维及三维网格对象, 提供创建简

单几何体及高级几何体的方法,可以直接创建三维模型,但是相对机械产品复杂模型,其功能是远远不够的,必须借助CAD造型软件系统。另一方面,Three.js支持加载STL、OBJ、FBX等多种格式的三维模型文件,可以直接加载外部模型,目前主流的CAD造型软件都能生成STL格式文件。因此,本文基于三维建模软件SolidWorks对机械产品装配模型进行建模,并导出STL格式的文件。

3 产品三维装配展示模型(3D assembly display model of the product)

3.1 产品三维装配模型设计

要实现上述产品装配展示的应用需求,在客户端必须要构建一个产品三维装配模型。产品装配模型是对装配零部件信息的描述,主要包括装配体单元组成的层次关系表达、装配零件、部件的装配定位及约束关系表达、各种装配特征的表达等。本文采用如图2所示的装配模型,产品装配由零件、子装配通过装配约束关系形成的一个层次装配结构树。零件模型由零件名称、零件属性和一组来自STL文件的三角面表示的几何边界来描述。

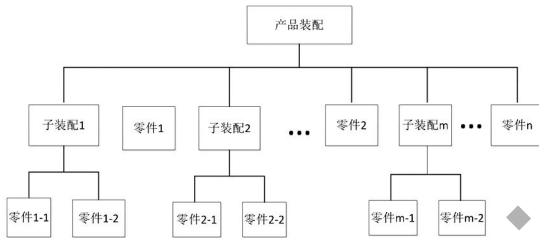


图2 产品层次装配模型示意图

Fig.2 Diagram of the product hierarchical assembly model

零件在装配中的位置和姿态是通过装配约束实现的,即装配零件之间存在一个装配约束关系。本文采用面向对象技术定义的部分装配约束关系如图3所示。“装配约束”类是装配约束关系的基类,下面派生出“同轴约束”类、“共面约束”类、“相切约束”类等,“共面约束”类又可以派生出“同向共面”类和“反向共面”类。在“装配约束”类中设立“参考零件”对象、“被约束零件”对象、“参考约束面”对象、“被约束面”对象和“约束参数”对象等属性。

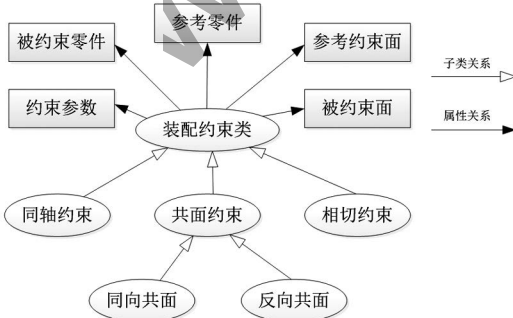


图3 装配约束关系

Fig.3 Assembly constraint relationship

3.2 产品三维装配模型重构

由于CAD装配模型转换为STL格式文件后,原有的装配

模型信息和边界模型几何信息丢失,转换为一组三角面集,要构造上述产品三维装配模型需要从这些三角面模型中进行重构。自动重构过程如图4所示。

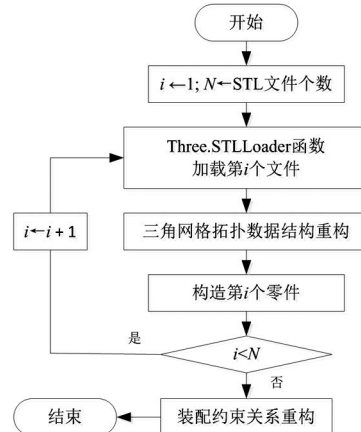


图4 产品三维装配模型重构流程图

Fig.4 Reconstruction flow chart of product 3D assembly model

重构过程主要有三个步骤:一是加载STL格式三维模型文件,使用Three.js中的STLLoader对象将STL三角面文件转入系统场景中,同时对文件进行MeshLambertMaterial材质设置,完成对模型文件的加载;二是完成对加载STL三角面的拓扑数据结构的重构,方便对三角面进行各种几何和查询操作,形成一个零件几何边界;三是在所有STL文件加载、零件构造完成后,通过零件之间的位置和姿态识别零件之间的装配约束关系。第二个和第三个步骤将在下节介绍。

图5是一个减速器产品三维装配模型通过本文方法重构后装配模型实例。

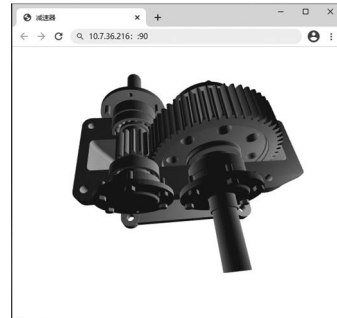


图5 减速器产品三维装配模型加载重构后实例

Fig.5 3D assembly model example for the reducer product after loading and reconstruction

4 产品装配约束关系自动重构(Automatic reconstruction of product assembly constraint relationship)

4.1 三角网格拓扑数据结构重构

如前面所述,STL文件是一组独立三角面的集合。一个三角面由三个顶点和一个面法矢量组成,相邻三角面没有任何拓扑关系,并且存在大量重复顶点,因此非常有必要开展三角网格拓扑数据结构重构。

拓扑重构主要包括两个部分工作:一是进行冗余顶点剔除;二是创建三角网格拓扑数据结构,即构建三角网格

面与相邻面之间、三角面与边之间、边与顶点之间的拓扑关系。基于该拓扑关系可以实现对三角面进行邻域搜索和几何操作。

目前关于三角网格拓扑数据结构已经开展了大量的研究,典型的拓扑数据结构是半边结构。文献[8]提出一个采用半边编码的紧凑三角网格拓扑数据结构,在该数据结构中,三角网格的几何信息通过顶点和三角面描述,拓扑信息隐含在半边信息中,半边信息编码为一个无符号长整型数,消耗内存少。

本文采用JavaScript编程,基于文献[8]给出的重构算法,实现三角网格拓扑数据重构。从零件三角网格边界中,已知一个三角面,就可以很快地遍历所有其他三角面;如果已知一条边,就可遍历三角面的所有其他的边;不需要条件判断完成一环邻域直接查询,为后续装配约束关系重构打下了基础。

4.2 装配约束关系自动识别

完成STL零件模型的拓扑重构后,即可得到一个由三角面组成的零件几何边界,但以前的零件装配信息丢失,需要根据零件的几何位置和姿态,对装配约束关系自动识别,重构其装配约束关系。装配约束关系自动识别主要步骤如下:

(1)两个零件包围盒检查

零件包围盒是一个包容零件三角面几何边界的最小六面体,可以通过对零件三角面的三维顶点求最大最小获取。设零件A的最小包围盒为 A_{\min} ,零件B的最小包围盒为 B_{\min} ,则如果 $A_{\min} \cap B_{\min} = 0$,表明零件A和零件B可能有交集,即可能存在装配约束关系,转到下一步进行检测。

(2)零件三角面的面面检测

对存在边界交集的两个零件开展三角面的面面检测。检测基本方法是任取零件A的一个三角面,与零件B上所有三角面进行面接触检测,主要步骤如下:

Step 1: 计算两个三角面夹角。由于每个三角面都保有面法向量,采用夹角公式计算 $\varphi = \arccos(\overrightarrow{N_A} \cdot \overrightarrow{N_B})$,如果夹角值 $fabs(\varphi) < \varepsilon$,表明两个三角面是同向平行;如果 $fabs(\varphi - \pi) < \varepsilon$,表明两个三角面是反向平行(ε 是设定的最小阈值)。

Step 2: 计算两个平行三角面距离。如果距离 $dist < \varepsilon$,表明两个三角面相接触,再通过三角面拓扑关系进行邻域搜索;如果还存在一个相邻三角面相接触,表明两个零件存在共面约束,再进一步判断是同向共面、反向共面还是同轴关系。

(3)同轴关系判定

如果相邻三角面存在共面约束,同时计算与相邻面之间的两面夹角。由于三角网格在近似表示曲面时存在一个舍入误差 ε ,当误差 ε 足够小时,此时则代表平面。此外,假设二面角存在一个最大值 γ ,当大于这个值时,则两邻接面是不同的几何面^[9-10]。因此,两邻接面关系主要有如表1所示三种情况。

表1 两邻接面关系判断

Tab.1 Judgment of the relationship between two adjacent surfaces

二面角值	两邻接三角面关系
$\varphi \leq \varepsilon$	共面
$\varepsilon \leq \varphi \leq \gamma$	同为曲面一部分
$\varphi > \gamma$	不同的几何面

根据表1推断出一个三角面与三个邻接面的关系,存在两个以上的三角面共面时,即可确定为在一个平面上,然后通过递归邻域搜索属于同一平面的其他三角面,从而确定整个平面。

当一个三角面的三个邻接面存在至少一个属于曲面部分,则将它们作为曲面的候选三角面,然后比较它们的尺寸,当它们的尺寸大小基本相同时,通过判断圆柱轴线方向的方法来判断是否为曲面,即将其假设为圆柱表面的一部分,求得圆柱体轴线的方向,再检查圆柱轴线与下一个三角面法向量的垂直度,从而识别圆柱表面。识别圆柱表面后,同样通过递归邻域搜索到同属于圆柱表面的其他三角面,进而确定两圆柱的轴线位置,计算两零件圆柱面轴线夹角 α ,当 $\alpha < \varepsilon$ 或 $(\alpha - \pi) < \varepsilon$ 时,此时两零件同轴。

5 产品三维装配展示(3D assembly display of the product)

基于上述工作,本文在谷歌浏览器上开展了机械产品三维虚拟装配展示实验,由于篇幅所限,下面仅介绍两项产品装配展示方法。

5.1 产品零部件导航

为了方便用户更好地在浏览器界面下查找装配模型中的各个零件,在浏览器左侧基于JavaScript中的树形组件tree构建产品装配结构树,如果选择一个零件名称,相应地在装配展示区,该零件高亮显示。

在产品装配模型中,每个零件几何边界是通过加载服务端的STL文件生成的,加载的STL文件名就是构造装配模型中的零件名。获取了装配树中点击的零件名称,就可以通过装配模型搜索到对应零件的几何边界。如图6所示,在左侧结构树中选择减速器箱座时,相应地在装配展示区,减速器箱座高亮显示。

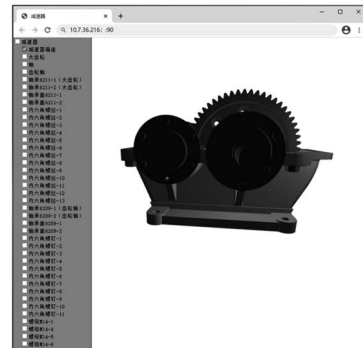


图6 装配零件导航示意图

Fig.6 Illustration for assembly parts navigation
相应地在装配展示区,鼠标在模型中移动时,鼠标光标

所接触的零件也会高亮显示。其基本原理是在鼠标移动光标位置处构造一条raycaster射线,计算鼠标所经过的3D空间的对象是否被射线击中,从而返回被击中的零件对象,完成零件的选中。拾取流程包括三个步骤:

Step 1: 坐标转换。将所选零件的屏幕坐标转为WebGL标准设备坐标。

Step 2: 射线计算。通过选中零件位置与相机参数来计算射线位置。

Step 3: 射线交叉计算。通过intersectObjects()计算出与射线相交的网格模型,实现对展示区零件的选中。

5.2 产品装配爆炸图展示

复杂的机械产品通常包含大量零部件,为了揭示和分析其内部结构,表达零部件间的相对空间位置关系,需要将装配体中的零部件按照指定的次序、方向和距离相互分离,这种视图称为爆炸图或爆炸视图。爆炸图可清晰地展示装配体中所有零部件之间的相互配合、连接关系等。

目前关于装配爆炸图的生成有大量的研究,主要有自动生成方法和交互手工生成方法。本文在上述装配约束关系重构的基础上,实现了产品装配爆炸图自动生成及展示,主要实现步骤如下:

Step 1: 沿坐标轴方向设立六个爆炸方向: $-x, x, -y, y, -z, z$ 。

Step 2: 搜索装配约束关系模型。对具有反向共面约束的零件,按照其约束参考面法向收集到不同爆炸方向设定的爆炸序列中。例如,零件A反向共面约束到零件B上,A是约束零件,B是参考零件,B上共面的面为参考面,如果该面法向是x轴正向,则零件A被收集到正x轴方向爆炸排序队列中。

Step 3: 对六个爆炸方向的爆炸排序队列重新进行排序。每个排序零件都设置一个初始值为零的约束度,如果该零件被约束关系每引入一次参考零件,则该零件的约束度加1;约束度最小,同时离垂直坐标平面距离最近的零件先进入排序队列。

Step 4: 对六个爆炸排序队列里零件爆炸平移变换。计算装配包围盒,确定爆炸距离;按照队列先进先出原则,按照计算设定的爆炸距离平移变换爆炸零件。

图7给出一个在谷歌浏览器下生成的减速器装配爆炸图。

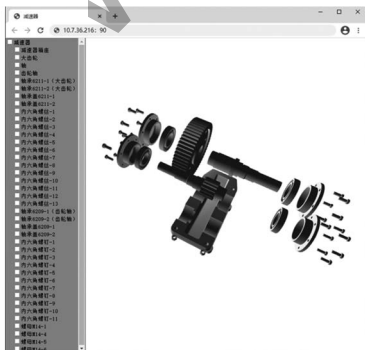


图7 减速器产品装配爆炸图

Fig.7 Explosion view of the reducer product assembly

6 结论(Conclusion)

通过互联网技术将服务端的机械产品三维CAD数字化模型在客户端浏览器上进行装配展示,对促进产品的数字化推广和服务具有重要意义。本文基于WebGL技术,开展了机械产品虚拟装配展示技术的研究,对装配展示技术架构进行了设计,采用第三方库Three.js创建虚拟展示场景,对装配模型的STL文件进行装配模型及装配约束关系的自动重构,初步实现了一个基于客户端浏览器的机械产品装配展示,包括装配模型零件导航及交互、装配爆炸图展示等功能。实验表明,本文提出的方法是可行的,基于WebGL技术可以构建功能强大的装配模型虚拟展示平台,使得用户在浏览器中无须安装插件即可浏览和操作产品装配数字模型,大大促进了产品三维数字化展示和互联网+技术的应用。

参考文献(References)

- [1] 邵宝仓.基于WebGL的矿山数字化应用平台的设计与实现[D].天津:天津大学,2018.
- [2] 宁静.基于WebGL实物交互技术及其实现的研究[D].武汉:华中科技大学,2014.
- [3] 戴秀海,姚震,仲梁维,等.基于Three.js的镀膜机展示平台的设计[J].软件工程,2019,22(02):26-28.
- [4] 姚震,仲梁维,陈彩凤.基于Three.js的机械产品展示研究[J].软件导刊,2018,17(05):187-190.
- [5] SUN F, ZHANG Z C, LIAO D M, et al. A lightweight and cross-platform Web3D system for casting process based on virtual reality technology using WebGL[J]. The International Journal of Advanced Manufacturing Technology, 2015, 80(5):801-816.
- [6] ZHOU X P, WANG J, GUO M, et al. Cross-platform online visualization system for open BIM based on WebGL[J]. Multimedia Tools and Applications, 2019, 78(20):28575-28590.
- [7] DIRKSEN J. Three.js开发指南[M].李鹏程,译.北京:机械工业出版社,2015:23-60.
- [8] 张应中,谢馥香,罗晓芳,等.采用半边编码的三角网格拓扑数据结构[J].计算机辅助设计与图形学学报,2016,28(02):328-334.
- [9] SUNIL V B, PANDE S S. Automatic recognition of features from freeform surface CAD models[J]. Computer-Aided Design, 2008, 40(4):502-517.
- [10] LANGEDAHN L T, OLEKSANDR S, IVANNA B, et al. Extracting shape features from a surface mesh using geometric reasoning[J]. Procedia CIRP, 2020(93):544-549.

作者简介:

韩欣雨(1996-),女,硕士生.研究领域:基于WebGL机械产品虚拟装配展示。

张应中(1961-),男,博士,教授.研究领域:产品智能几何建模与特征建模,工程语义信息知识表示与处理,可持续(绿色)设计与制造。