**中图分类号：TP31**

**论文编号：**



学术硕士学位论文

**交通事故预测的机器学习方法**

**研究与应用**

作者姓名 任红雷

学科专业 软 件 工 程

指导教师 宋友 教授

培养院系 软件学院

**Research and Application on the Machine Learning Method of Traffic Accident Prediction**

A Dissertation Submitted for the Degree of Master

**Candidate：Honglei Ren**

**Supervisor：Prof. Song You**

School of Software Engineering

Beihang University, Beijing, China

**中图分类号：TP31**

**论文编号：**

硕 士 学 位 论 文

交通事故预测的机器学习方法

研究与应用

作者姓名 任红雷 申请学位级别 学术硕士

指导教师姓名 宋友 职 称 教授

学科专业 软件工程 研究方向 软件工程与管理

学习时间自 2015年 09月 01日 起至 2018年03月31日止

论文提交日期 2018年 01月 10日 论文答辩日期 2018年 01月10日

学位授予单位 北京航空航天大学 学位授予日期 年 月 日

关于学位论文的独创性声明

本人郑重声明：所呈交的论文是本人在指导教师指导下独立进行研究工作所取得的成果，论文中有关资料和数据是实事求是的。尽我所知，除文中已经加以标注和致谢外，本论文不包含其他人已经发表或撰写的研究成果，也不包含本人或他人为获得北京航空航天大学或其它教育机构的学位或学历证书而使用过的材料。与我一同工作的同志对研究所做的任何贡献均已在论文中作出了明确的说明。

若有不实之处，本人愿意承担相关法律责任。

学位论文作者签名： 日期： 年 月 日

学位论文使用授权书

本人完全同意北京航空航天大学有权使用本学位论文（包括但不限于其印刷版和电子版），使用方式包括但不限于：保留学位论文，按规定向国家有关部门（机构）送交学位论文，以学术交流为目的赠送和交换学位论文，允许学位论文被查阅、借阅和复印，将学位论文的全部或部分内容编入有关数据库进行检索，采用影印、缩印或其他复制手段保存学位论文。

保密学位论文在解密后的使用授权同上。

学位论文作者签名： 日期： 年 月 日

指导教师签名： 日期： 年 月 日

**摘 要**

随着现代交通的发展和城市化趋势，交通领域的问题愈发凸显，在一些大城市如：北京，上海，广州等，交通事故越来越多，据统计2016年中国交通事故约50万起，因交通事故而死亡的人数约25万人。此外，国务院在第十二个五年规划纲要中明确提出要将交通事故致死率减少36%。这些证明了交通安全问题已变成一个不可忽视的问题，为了减少交通事故以及相关的生命和财产损失，对交通事故准确的实时预测研究很有必要。

同时，随着大数据时代的到来，传统的交通研究也开始从基于统计规律的研究转变为基于交通大数据的研究。越来越多的异质类型的数据如交通流探测器产生的：浮动车GPS数据、线圈车流量数据，地铁、公交的刷卡数据，基于互联网和社交媒体产生的LBS数据。这些不同的数据源每天产生着大量的数据，为研究和和解决交通问题提供了无限的可能。而近年来兴起的机器学习和深度学习技术，也为我们分析和解决交通问题提供了新思路和有效的手段。因此，采用基于城市交通大数据的方法，对实时城市交通事故预测的机器学习方法进行研究很有必要。本课题研究在分析当前需求与问题的基础上，主要研究内容包括：

1. 设计一种具有较高准确率和召回率的交通事故预测的机器学习方法

利用相关性分析找出了交通事故时空模式，以及相关影响因素。并在此基础上，设计了一种能够较精准的预测城市区域交通事故的机器学习方法。

1. 基于预测性能对影响交通事故的因素进行了分析和排序

单个的交通事故较为随机，通过统计方法分析出影响交通事故的重要因素，对改善城市交通管理策略、以及进行更好的事故预测至关重要。基于格兰杰因果系数，通过比较不同特征和因素组合的情况下的预测性能，对影响交通事故的主要因素进行了重要性排序。

1. 设计异质交通大数据的存储策略和架构

根据不同类型的交通数据（出租车轨迹数据、交通事故数据、天气、空气质量等）的特点和数据量，设计了较为通用和高效的存储策略和架构。

1. 原型系统的开发与验证

通过建立原型系统，使用前述所提出的交通事故预测方法、框架、存储方案，基于 B/S(浏览器/服务器)架构，设计了一种较为通用的、具有良好交互性的，实时交通事故可视化、及事故预测平台，方便交通管理者和决策者进行实时的交通决策。本课题从系统功能和运行效率两方面验证工作的有效性，为一般性的交通事故预测平台提供参考依据。

本课题通过设计具有较高准确率的交通事故预测方法，以及相应的设计出在线的实时事故预测可视化平台。为城市的交通事故避免和决策的提供了方法和策略的支持和方法参考。

关键词：交通事故预测，机器学习方法，异质大数据，影响因素分析

**Abstract**

With the development of modern traffic and the trend of urbanization, the problems in the transportation field have become increasingly prominent. In some big cities such as Beijing, Shanghai and Guangzhou, there are more traffic accidents. According to statistics, about 500,000 traffic accidents in China in 2016 About 250,000 people died as a result of traffic accidents. In addition, the State Council explicitly proposed in the 12th Five-Year Plan to reduce the fatality rate of traffic accidents by 36%. These prove that the problem of traffic safety has become a problem that can not be ignored. In order to reduce the traffic accidents and the related loss of life and property, it is necessary to study the real-time prediction of traffic accidents accurately.

At the same time, with the advent of the era of big data, traditional traffic research has also begun to shift from research based on statistical laws to research based on big data of traffic. More and more heterogeneous types of data are generated by traffic detectors: floating car GPS data, coil car flow data, subway, bus swipe data, LBS data generated from the Internet and social media. These different data sources generate a large amount of data every day, providing endless possibilities for studying and solving traffic problems. In recent years, the rise of machine learning and deep learning technology, but also provide us with new ideas and effective means to analyze and solve traffic problems. Therefore, it is necessary to study the machine learning method of real-time urban traffic accident forecasting based on the method of urban traffic big data. Based on the analysis of current needs and problems, the main research contents include:

1) Design a machine learning method for traffic accident prediction with high accuracy and recall

Using the correlation analysis to find out the traffic accident spatio-temporal pattern, as well as related influencing factors. Based on this, a machine learning method that can predict traffic accidents in urban areas more precisely is designed.

2) Based on the predictive performance of the factors that affect traffic accidents were analyzed and sorted

Individual traffic accidents are more random. Analyzing the important factors influencing traffic accidents through statistical methods is very important to improve the urban traffic management strategy and make better accident prediction. Based on the Granger causality coefficient, the main factors influencing traffic accidents are sorted by importance by comparing the predictive performance of different features and combinations of factors.

3) Design heterogeneous transportation big data storage strategy

According to the characteristics and data volume of different types of traffic data (taxi track data, traffic accident data, weather, air quality, etc.), a more general and efficient data storage strategy and architecture is designed.

4) Prototype system development and verification.

Through the establishment of a prototype system, the traffic accident forecasting method, the framework and the storage scheme proposed in the foregoing are used to design a generic and good interactive visualization system for real-time traffic accidents based on the B / S (browser / server) architecture , And accident prediction platform to facilitate traffic managers and policy makers to make real-time traffic decisions. This subject verifies the validity of the work from the aspects of system function and operation efficiency, and provides reference for the general traffic accident forecasting platform.

The subject of this paper is to design a traffic accident prediction method with high accuracy and to design a visualization platform of real-time accident prediction online. For the city's traffic accident prevention and decision-making methods and strategies to provide support and methods for reference.

**Key words**: Traffic Accident Prediction, Machine Learning Method, Heterogeneous and Big data, Analysis of Influencing Factor

**目 录**

[第一章 绪论 1](#_Toc464427415)

[1.1 研究背景 1](#_Toc464427416)

[1.1.1 研究背景 1](#_Toc464427417)

[1.1.2 研究意义 2](#_Toc464427418)

[1.2 国内外研究现状 2](#_Toc464427419)

[1.2.1 基于3D的实时协同建模平台框架的研究现状 3](#_Toc464427420)

[1.2.2 协同3D场景建模研究现状 3](#_Toc464427421)

[1.2.3 基于HTML5的协同应用研究现状 4](#_Toc464427422)

[1.3 研究内容 4](#_Toc464427423)

[1.3.1 建立一种通用的实时协同3D建模平台架构 4](#_Toc464427424)

[1.3.2 设计一种去冗余的3D场景存储方案 5](#_Toc464427425)

[1.3.3 研究基于HTML5的协同建模应用 5](#_Toc464427426)

[1.3.4 原型系统的开发与验证 5](#_Toc464427427)

[1.4 论文组织结构 5](#_Toc464427428)

[第二章 Web3D实时协同建模平台框架 7](#_Toc464427429)

[2.1 场景分析与问题定义 7](#_Toc464427430)

[2.2 以零件库为中心的通用实时协同框架 9](#_Toc464427431)

[2.2.1 设计思路 9](#_Toc464427432)

[2.2.2 总体框架 10](#_Toc464427433)

[2.3 应用模式分析 12](#_Toc464427434)

[2.4 基于零件库的工作流程 13](#_Toc464427435)

[2.5 本章小结 14](#_Toc464427436)

[第三章 协同3D场景数据的存储 15](#_Toc464427437)

[3.1 协同3D场景数据分析 15](#_Toc464427438)

[3.2 抽象场景数据结构 16](#_Toc464427439)

[3.3 抽象场景数据文件 17](#_Toc464427440)

[3.4 联合存储方案 19](#_Toc464427441)

[3.5 本章小结 19](#_Toc464427442)

[第四章 基于HTML5的协同建模 21](#_Toc464427443)

[4.1 基于WebGL的三维场景构建 21](#_Toc464427444)

[4.1.1 场景构建方法 21](#_Toc464427445)

[4.1.2 与传统插件法对比 22](#_Toc464427446)

[4.2 基于WebSocket的协同通信 24](#_Toc464427447)

[4.2.1 协同通信方法 24](#_Toc464427448)

[4.2.2 与传统连接法对比 26](#_Toc464427449)

[4.3 基于WebStorage的数据存储 27](#_Toc464427450)

[4.3.1 数据存储方法 27](#_Toc464427451)

[4.3.2 与传统Cookie对比 28](#_Toc464427452)

[4.4 本章小结 29](#_Toc464427453)

[第五章 原型系统的实现与应用 30](#_Toc464427454)

[5.1 应用场景与需求分析 30](#_Toc464427455)

[5.1.1 应用场景 30](#_Toc464427456)

[5.1.2 需求分析 30](#_Toc464427457)

[5.2 系统的设计与实现 31](#_Toc464427458)

[5.2.1 总体设计 31](#_Toc464427459)

[5.2.2 详细设计 33](#_Toc464427460)

[5.2.3 功能模块实现 33](#_Toc464427461)

[5.3 系统讨论 36](#_Toc464427462)

[5.4 本章小结 38](#_Toc464427463)

[第六章 总结与展望 40](#_Toc464427464)

[6.1 工作总结 40](#_Toc464427465)

[6.2 后期展望 40](#_Toc464427466)

[参考文献](#_Toc464427467)

[攻读硕士学位期间取得的学术成果](#_Toc464427468)

[致谢](#_Toc464427469)

**图 目 录**

[图 1 Web3D实时协同建模平台问题分析图 8](#_Toc464427615)

[图 2基于零件库的协同建模架构 10](#_Toc464427616)

[图 3应用模式示意图 13](#_Toc464427617)

[图 4基于零件库的协同建模平台工作流程 14](#_Toc464427618)

[图 5 抽象场景数据结构示意图 17](#_Toc464427619)

[图 6抽象场景文件格式样例 18](#_Toc464427620)

[图 7 3D场景联合存储方案示意图 19](#_Toc464427621)

[图 8 Threejs有优异的3D显示效果 21](#_Toc464427622)

[图 9 协同物理连接与逻辑连接示意图 26](#_Toc464427623)

[图 10 WebStorage通过键值对形式进行存储 28](#_Toc464427624)

[图 11交互式结构设计业务流程图 31](#_Toc464427625)

[图 12 原型系统技术架构图 31](#_Toc464427626)

[图 13 三维协同设计平台功能设计图 32](#_Toc464427627)

[图 14 三维协同建模平台界面设计 36](#_Toc464427628)

[图 15多人协同建模 37](#_Toc464427629)

**表 目 录**

[表 1 协同3D场景数据分类表 15](#_Toc464427640)

[表 2各主流WebGL框架对模型的支持 22](#_Toc464427641)

[表 3 WebGL与传统方法对比 23](#_Toc464427642)

[表 4 协同操作指令集 24](#_Toc464427643)

[表 5 WebSocket与传统方法对比 27](#_Toc464427644)

[表 6 WebStorage与传统方法对比 28](#_Toc464427645)

[表 7 原型系统的功能实现情况与测试 37](#_Toc464427646)

[表 8 Web3D实时协同建模平台与传统研究对比 38](#_Toc464427647)

1. 绪论
   1. 研究背景
      1. 研究背景

在现代社会,城市化的快速发展带来了车辆的增长和繁荣。然而，这却带来了很多亟待政府解决的问题，如交通拥堵。在大数据和深度学习的帮助下，尽管高性能的实时交通流预测已经使得人们可以通过选择相对不拥堵的路线，来避免交通堵塞。而另外一个重要的问题，交通事故，尽管已经造成了巨大的经济损失和人员伤亡，却一直没能被很好的理解和解决。根据世界卫生组织在2015年发布的《全球道路安全现状报告》，每年大约有125万人死于交通事故。因此，对于我们的安全来讲，交通事故预防的研究显得重要而且急迫。

同时随着大数据时代的到来，传统的交通研究也开始从基于统计规律的研究转变为基于交通大数据的研究。越来越多的异质类型的数据如交通流探测器产生的：浮动车GPS数据、线圈车流量数据，地铁、公交的刷卡数据，基于互联网和社交媒体产生的LBS数据。这些不同的数据源每天产生着大量的数据，为研究和和解决交通问题提供了无限的可能。而近年来兴起的机器学习和深度学习技术，也为我们分析和解决交通问题提供了新思路和有效的手段。

在预防交通事故的解决方案中，建立高效而准确的交通事故系统预测无疑是很有希望的一种。因为通过预测系统给出的危险区域信息能够被方便地传递给驾驶员，并且他们可以通过选择不太危险的区域来避免交通事故。然而想要达到准确的事故预测却十分困难，因为许多相关因素都会影响交通事故。举例而言，像下雨和下雪这样的坏天气条件能够限制道路的可见能力和道路通行能力从而导致更多的事故。另外，交通事故的发生率在一天中的不同时间段是变化的。尽管许多研究都关注于识别与交通事故相关联的关键因素，但是他们并不能动态和定量地给出交通事故风险的信息。

* + 1. 研究意义

从学术研究领域看，本研究课题属于城市计算领域下的智能交通子领域。由于交通事故造成了大量的生命和财产损失，而据世界卫生组织统计，中国每年有超过25万人死于交通事故危害。对交通事故的实时预测研究，将可以使得交通管理者们实时调整安全资源的配置、实时发布道路安全信息、使得驾驶员和行人可以选择安全性更高的道路和区域出行，避免高风险的事故区域，从而改善道路的安全性，避免生命和财产损失。

本课题在一方面可以帮助交警部门科学地和有效地安排他们的警力，另一方面可以很容易地应用于现实世界，并通过以选择更安全的区域的方式，来帮助人们尽可能地避免交通事故。

* 1. 国内外研究现状
     1. 交通事故严重性的影响因素分析

在目前，国内外已经在道路交通事故严重程度影响因素分析领域开展了广泛研究，其中以经典统计模型的相关应用最为广泛。其中Chang采用了逻辑斯蒂模型对2000年台湾单一车辆道路事故对比了摩托车驾驶员和非摩托车驾驶员的死亡率的影响因子，发现摩托车驾驶员的死亡率是非摩托车驾驶员的3倍多，区别两者的主要因素在于安全带是否使用、行驶速度，与受伤严重性相关的因素包括道路类型、车辆类型。Malyshkina 采用了多项式逻辑斯蒂回归模型对正规设计的道路和设计的例外道路分别进行了事故严重性和事故频率的估计，对比结果显示例外设计的道路并在事故严重性和频率的结果上与正规设计的道路并没有统计的显著差异。Huang 采用了贝叶斯层次二项式逻辑斯蒂回归模型来识别对信号灯路口事故中人员和车辆分别的损伤严重程度的影响因素，结果显示时间、路口的光线照明情况对人员的伤亡严重性有较大影响。而车辆的大小、重量等对车辆在事故中的受损程度有直接影响。国内学者李世民等利用累积Logit模型研究了交叉口交通事故严重程度与转弯车辆、用地性质等因素的关系。马壮林等分别利用累积Logit模型和灰色模型研究了道路设计因素对隧道交通事故严重程度的影响。侯树展等利用主成分分析法研究了交通流特征和高速公路事故严重程度之间的联系。

绝大多数的统计回归模型都依赖于其自身的假设和预定义的变量间的潜在依赖关系。一旦这些假设和依赖关系被违反，这些模型将产生错误的事故严重性估计。考虑到经典统计模型的缺陷和不足，近年来有学者开始尝试将非参数、人工智能方法应用于交通事故严重程度影响因素分析。Mussone等提出了对传统模型的替代性的方法——人工神经网络法，该模型对米兰不同情况下交通路口发生事故的严重性做了计算和分析，结果显示路口的复杂性对高交通事故死亡率有着决定性作用。Chang 提出了分类回归树模型，该模型不需要目标变量和预测变量间的潜在依赖关系，通过对台北交通事故数据的分析，该模型表明对交通事故严重性影响最大的因素是交通工具的类型，行人、骑摩托车、自行车的驾驶者往往是在交通事故中受伤最严重的群体。Juan de Oña 采用了贝叶斯网络模型来对西班牙1536起乡间的交通事故严重程度进行分类，贝叶斯网络模型也无需依赖前提假设，并且能够利用图形化的表征对有着相互组件依赖的辅助系统进行刻画，其采用了18个变量构建了三个不同的网络来对三种不同的事故严重性程度（轻微受伤、严重受伤、死亡）进行分类。其结果显示对于死亡或者严重受伤类型的交通事故，有很大影响的因子有事故类型、司机年龄、道路光线情况、事故受伤人数。研究表明, 非参数模型在实证研究中具有更优的统计拟合度和泛化能力。

* + 1. 交通事故热点区域发现

目前绝大多数识别交通事故聚类的方法主要分为两类：第一类是对一条道路简单输出其是否有聚类的假设检验结果。其代表性的方法有：k-方法[21, 22]和最近邻方法[23, 24]。第二类是可以在一段道路或者路网中识别出精确地聚类区域或者位置，例如核密度估计方法(KDE)[25]或者危险指标方法[26]等。

关于道路交通事故热点，有关文献并没有统一而明确的定义。Hauer 在论文中指出部分学者根据交通事故发生概率按位置给出热点排名，而另外一些学者根据事故发生的频率，每公里道路上的历史事故数量[53]。传统的道路事故热点分析主要集中在道路片段或者特定连接位置[54]，而区域范围的热点以及该区域事故引起的风险分散被人忽略。自20世纪70年代开始，统计模型就被应用于交通事故分析，而早期模型的主要缺陷在于假设事故的发生服从正态分布[55-57]。随后一些学者为了改正这项缺陷，而采用泊松对数线性回归来对事故在空间和时间的随机性进行刻画[58]。而很多其他的学者例如，Hauer和Persaud[59], Miaou[60], Shanker[61], Maher 和 Summersgill[62], Abdel-Aty和Radwan[63]，都采用负二项回归模型。以上提到的模型都考虑了在给定时间段内的报告的事故发生数量，而对于该时段内事故的位置特征则被当作常量。

随后的研究主要基于核密度估计(KDE)的方法来对事故热点区域进行聚类，Sabel 首先采用了核密度估计聚类分析技术来自动的识别交通事故的热点区域，该研究结合了蒙特卡洛模拟技术，能够快速的识别统计显著的聚类[25]。其他学者也陆续采用核密度估计方法提出了改进的方法或者应用[27]。Plug应用空间、时间以及时间-空间结合的技术对西澳大利亚不同尺度的交通事故模式进行了分析，采用蜘蛛图来识别按天和按星期的时间尺度上的交通事故模式，用核密度估计的方法来探究事故的空间结构模式，采用Comap来识别事故的时空间的相互影响[64]。Xie 和Yan之后考虑到KDE长久以来一直被用作事故热点的可视化工具，因为缺乏对其统计推断结果的定量评估，因此结合了NetKDE（一种被证明在网络结构的事故分析中常用的工具）[65]，与100米范围内的局部莫兰指数(一种度量空间相关性的指标)[66]。

* + 1. 交通事故起因识别

在早期交通安全研究方面，许多研究都致力于识别导致交通事故的关键条件或特定的交通模式。例如，Oh提出了破坏性的交通流是导致撞车的导火索(Ohetal.2001)。基于线圈检测器的数据和撞车事故数据，他们发现交通事故发生前的5分钟速度标准差是撞车事故的一个有效指示器。Abdel-Aty则将交通模式的空间变化考虑在内，并且发现事故发生地点上游5分钟的平均占有率和下游车速的5分钟变异系数是交通事故的标志(Abdel-Atyetal.2004)。Golob则通过研究发现平均车流量，流量变化，速度中位数和速度变化是撞车事故的良好标志(GolobandRecker2004)。尽管已有各种各样的交通事故指示因子提出，但是因为很多因素都与交通事故有着复杂的关联，因此它们不能满足精准预测交通事故的需要。

* + 1. 实时交通事故预测

随着机器学习的发展，已有许多的研究开始关注实时交通事故预测。Lv基于欧氏距离选择特征变量并且利用k近邻方法来预测交通事故(Lv,Tang,andZhao2009)。Hossain提出了基于贝叶斯网络的方法来对高速公路的车辆事故进行实时预测(HossainandMuromachi2012)。Park收集了首尔高速公路的交通事故大数据并且构建了一个基于k-均值聚类分析和逻辑斯蒂回归方法的预测工作流程(Park,Kim,andHa2016)。最近，Chen使用日本的行人移动数据并且构建了堆叠降噪编码机来实时的推测交通事故风险(Chenetal.2016)。

然而他们都忽视了交通事故的时间特征，并且也没有意识到其他与交通事故相关联因素的重要性，比如交通流，天气条件和空气质量等。这种忽视会带来预测准确率上的巨大差异，尤其是在城市和国家层面的事故预测中。

* 1. 研究内容
     1. 设计一种具有较高准确率和召回率的交通事故预测的机器学习方法

利用相关性分析找出了交通事故时空模式，以及相关影响因素。并在此基础上，设计了一种能够较精准的预测城市区域交通事故的机器学习方法。

* + 1. 基于预测性能对影响交通事故的因素进行了分析和排序

单个的交通事故较为随机，通过统计方法分析出影响交通事故的重要因素，对改善城市交通管理策略、以及进行更好的事故预测至关重要。基于格兰杰因果系数，通过比较不同特征和因素组合的情况下的预测性能，对影响交通事故的主要因素进行了重要性排序。

* + 1. 设计异质交通大数据的存储策略和架构

根据不同类型的交通数据（出租车轨迹数据、交通事故数据、天气、空气质量等）的特点和数据量，设计了较为通用和高效的存储策略和架构。

* + 1. 原型系统的开发与验证

通过建立原型系统，使用前述所提出的交通事故预测方法、框架、存储方案，基于 B/S(浏览器/服务器)架构，设计了一种较为通用的、具有良好交互性的，实时交通事故可视化、及事故预测平台，方便交通管理者和决策者进行实时的交通决策。本课题从系统功能和运行效率两方面验证工作的有效性，为一般性的交通事故预测平台提供参考依据。

* 1. 论文组织结构

本文共分为六个部分，组织结构如下：

**第一章 绪论**。在该部分中阐述了研究背景，提出了本课题将集中解决的主要任务，并对现有情况的国内外现状进行了分析。

**第二章 交通事故预测的机器学习方法**。在该部分中，详细讲解了交通事故预测方法的异质数据来源、预处理过程、机器学习模型、参数选取、训练过程。之后对本课题提出的基于深度循环神经网络方法的预测性能，与其他方法包括传统机器学习方法与深度学习方法进行了对比。

**第三章 交通事故影响因素的重要性排序**。在该部分中，通过格兰杰因果分析的方法，即通过比较不同单独因素或者组合因素的预测准确率，本课题对影响交通事故的若干因素进行了重要性排序。

**第四章 异质交通大数据的存储策略与架构设计**。该部分讲述了本课题如何根据不同类型的交通数据的特点和数据量，设计较为通用和高效的存储策略和存储架构。

**第五章 原型系统的实现与应用**。该部分讲解了开发的交通事故可视化与事故预测原型系统，对其设计、功能及效率进行了一定的分析。

**第六章 总结和展望**。本部分总结本课题的研究内容，研究工作的局限性，并对未来的工作方向进行了展望。

1. Web3D实时协同建模平台框架

协同环境下的3D建模框架是建模平台的核心。为了提出多人协同环境下的通用3D 建模框架，具有可复用性，需要针对实际的应用场景，分析实现实际情景下的多人协同建模时，可能需要满足的需求和遇到的挑战，并有针对性地设计框架的整体方案。因此，本章首先对多人协同环境下的3D建模场景进行分析并定义问题。通过对实际问题的具体分析，获得需求和主要需要解决的问题。接着，提出框架的整体方案，利用架构图的方式作整体展示，并加以说明解释。最后，针对整体的框架设计，提出架构应用的模式分析，明确了整个框架的应用方式和功能拓展点。

* 1. 场景分析与问题定义

随着互联网应用的逐渐兴起，许多传统行业正面临着互联网带来的机遇和挑战。一方面，随时随地互联使很多传统单机进行的工作可以以更方便、更快捷、更高效的方式来进行；另一方面，随着网络带宽的增加和计算机技术的发展，一些利用新型计算机技术而产生的新应用也应用而生。而计算机图形学的发展正是其中重要的一块，其与计算机网络互联的结合正在产生新的应用需求与机遇。

在一些传统设计领域，如工业设计，家居装潢，关卡设计等行业，互联网的兴起给它们带来了新的活力，通过多人协同进行设计，设计的效率和满意度得到了提高，而计算机图形学的发展也为这类面向协同的设计提供了技术的平台；而一些已经应用了协同与互联的相关领域，如远程会议，远程教育等，也能够在计算机图形学和网络带宽的发展下获得新的应用模式。通过引入虚拟现实技术，能够使远程会议和远程教育的过程更加生动形象，通过3D技术传递更多的信息将有助于提高协同工作的效率。而伴随着计算机图形学发展起来的计算机游戏与数字艺术行业，更是得益于网络互联的兴起，使人们通过计算机娱乐的方式得到了更有创意和丰富的体验。

为确定问题边界，我们将本文所讨论的Web3D实时协同建模平台定义为基于Web的设计平台，使用三维模型作为建模素材，支持多领域多地域的用户，根据需求对模型实时协同编辑，开展协同建模与概念设计。核心是如何利用Web3D支持这样的多人协同建模功能，以及如何提供更好的用户体验。基于这两点，我们考虑的问题主要从以下五个角度进行考虑：如何支持3D建模功能，如何支持实时协同，如何面向协同群体中不同的角色，如何利用计算机辅助一些自动化工作，以及如何设计出具有良好系统特性的系统。如图1所示。



图 1 Web3D实时协同建模平台问题分析图

1）支持3D建模。作为建模平台，首先要能在Web上提供基本的3D建模功能。考虑的问题包括如何使用Web3D技术在网页端建立一个三维建模场景；如何在网页上与这个场景进行交互用户应该如何操作；当场景建立以后如何将三维模型导入到场景中并显示；以及如何对大量的模型进行统一的管理。

2）支持实时协同。协同建模平台的第二个核心是能够实现多人的实时协同。考虑问题包括如何使多人场景同步，所见即所得；如何保证实时性，使得延迟尽可能低，用户体验良好；如何去解决多人协同操作中发生的协同冲突问题；如何对于多人数据副本保持一致性；以及如何设计协同交互的交互方式。

3）针对不同用户。在一个协同合作任务中，参与的会有来自不同专业不同地域的人们，如何有针对性的安排每个人工作，需要进行统一管理。这里会包括对于用户数据的管理，以及如何去分配不同用户的操作权限。

4）自动化工作。计算机支持的协同操作应该能最大限度地发挥计算机的优势来辅助人们的工作，于是一些自动化的工作可以交由计算机完成。比如对建模中的约束进行计算判断，对数据进行版本控制等。

5）良好的系统特性。设计一个通用的实时协同平台应该具有良好的系统性能。由于系统面对的用户很可能不是专业的3D建模人员，因此系统要有比较好的易用性，方便上手和操作；为了将平台应用于各种领域，需要能方便的订制各种业务；另外平台需要有良好的可扩展性，来适应各种不同的需求。

* 1. 以零件库为中心的通用实时协同框架
     1. 设计思路

工业设计是一个较传统的领域。可以看到，在大多数软件已经逐步走向云化、移动化的今天，工业设计软件还是固守在纯桌面软件领域，基本以套装软件的形式存在，即便是像达索、欧特克这样的世界工业设计软件的领导者，在互联网和移动互联网领域做出的尝试也是极其有限的。一方面，从技术角度看，工业设计软件的互联网化是有非常高的技术难度的；另一方面，工业设计软件的互联网化的商业模式也不够成熟。

现有的研究都提出了自己的协同框架构建思路，但几乎都是从零开始构建，不仅成本较高，而且系统复杂，导致应用并不广泛。调研发现目前实际应用中更多的是存在各式各样的零件库系统，这些系统的特点是用户可以分享自己的零件，可以在平台上进行零件的上传下载点评等，商业模式是零件库中的零件分为免费付费，有偿使用零件。典型例子如Traceparts，拥有全球最大使用最广泛的数字3D 零件库，1亿多个3D数字零件模型，支持CATIA， SolidWorks and Spatial，AutoCAD， Inventor等主流3D建模软件格式，所有的零件库模型开放免费下载。拥有各国标准零件库GB国标库，DIN 德国工业标准库， ISO 国际标准化组织，数百家制造商零件库，如西门子，SMC，施耐德，研华，FAG，NSK，空中客车，阿尔斯通，SKF 等数百家知名制造商零件库；3DSource零件库是近年来非常流行的零件库产品，提供超过150万个标准件、常用件、厂商件三维模型下载。该零件库采用服务端/客户端模式，可以在线使用和离线使用。客户端支持主流的三维CAD软件，包括：PRO/E；UG NX；CATIA ；SolidWorks ；Inventor；CAXA实体设计。

可以看到成熟的零件库系统本身就是宝贵的资源，本文提出了一种基于零件库的实时协同建模框架，可以建立在现有零件库系统之上，不仅降低成本，而且符合商业发展趋势，目前已经有一些实际的零件库系统开始增加协同建模功能，更是验证了这一设计思路。

* + 1. 总体框架

为了满足新时代对于面向协同的3D 建模平台所提出的需求，适应新的应用场景，具备强大的通用性，本课题基于设计思路，提出了一种基于零件库的多人协同建模框架，如图2所示。

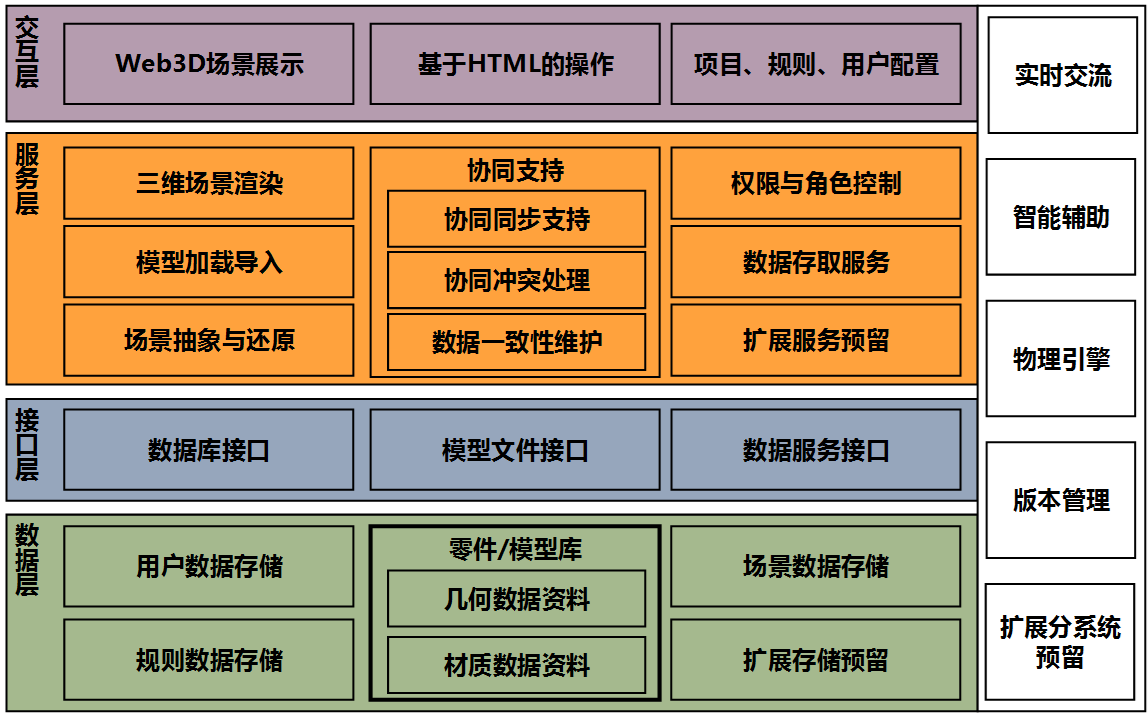


图 2基于零件库的协同建模架构

整个架构从上到下可分为交互层，用于支持交互功能的服务层，对数据调用进行统一控制的接口层，以及管理所有数据的数据层。同时还包括若干分系统。下面对每个部分别加以详细分析。

1. 交互层

系统的前台是基于网页的3D展示与操作环境。通过建立基于网页3D的轻量级用户界面，系统用户可以轻松地登录系统并开始建模操作。而同时为了照顾用户可能并不是专业的3D建模专家，系统将提供一套易于交互的建模操作方式，而反映在系统中则将是能够方便进行操作和易上手的操作界面。无需安装和配置的客户端以及易于操作的交互方式，将帮助更多的用户能够运用这个系统进行基于3D的建模的操作，扩大协同工作的适应面。

Web3D展示主要是利用基于WebGL的三维构建，WebGL（全写Web Graphics Library）是一种3D绘图标准，这种绘图技术标准允许把JavaScript和OpenGL ES 2.0结合在一起，通过增加OpenGL ES 2.0的一个JavaScript绑定，WebGL可以为HTML5 Canvas提供硬件3D加速渲染，可以借助系统显卡来在浏览器里更流畅地展示3D场景和模型了，还能创建复杂的导航和数据视觉化。显然，WebGL技术标准免去了开发网页专用渲染插件的麻烦，可被用于创建具有复杂3D结构的网站页面。

交互层还包括对于用户、规则、项目的一些基本配置，这些都是协同任务的重要参数，需要提供相应的页面供管理人员进行设置。

1. 服务层

服务层将平台所需解决的大部分问题整合为服务模块，为交互层提供支持。三维展示方面，需要为场景提供渲染支持，对三维模型进行加载与导入，以及对整个三维场景的抽象加以传输或保存，对抽象的场景进行还原。协同支持方面，对协同的同步工作提供实现，处理协同过程中的冲突问题，以及保证多人协同的数据一致性问题。其他模块还包括对权限与角色的控制，数据的存取读入等。

3D建模操作需要大量的模型支持，并且需要建立良好的模型扩展机制。模型的来源主要有两种渠道，一是为特定建模软件进行模型定制，通过对模型属性和特性的定制，利用常用的3D建模软件建立模型，并导出成基本模型以供其他建模或图形引擎使用。二是对现有模型进行解析和抽取，获取其中我们需要的部分，并提取模型数据存储到模型数据库中。这种方式获取模型方便而灵活，能够最大程度地利用已有的丰富模型资源，但是需要掌握对于特定模型的解析与抽取技术。在本课题中，我们选用Threejs与Obj格式相结合，来实现模型的渲染与加载。

1. 接口层

为了更加高效灵活地管理应用的相关数据，不同类型的数据我们使用不同的方式进行存储。对于三维模型数据使用云进行存储，方便用户共享与下载；用数据库记录用户信息等条目数据；以文件形式保存的数据则使用典型的文件系统进行管理。我们为每一类数据的读写提供不同的接入方法，作为数据接入层供支撑层调用。通过提供统一的接口，可以进行数据持久化等工作。

1. 数据层

系统使用的全部数据将统一在数据层进行管理，不同的数据将使用不同的存储方式。由于系统是基于现有的零件库系统，因此模型的几何资料与材质资料将由零件库系统进行管理。对于用户的信息以及协同任务中的约束规则，这些信息更适合统一存放在数据库中。另外对于整个场景的保存我们使用场景文件进行存储，对应的有场景数据文件的管理。

1. 分系统

就像任何一个版本管理软件一样，一旦涉及到多个用户对于同一场景进行修改、更新、提交，便会产生版本控制问题。关于版本冲突管理，目前已有很多成熟的研究，而本架构中的场景管理与版本控制模块，便是要结合多人协同环境3D建模的相关应用，合理地选择并实现一种版本控制机制，保持用户场景数据的一致性。

一般的协同建模会考虑到一些物理约束，于是物理引擎也是系统分系统的一部分。物理引擎通过为刚性物体赋予真实的物理属性的方式来计算运动、旋转和碰撞反映。好的物理引擎允许有复杂的机械装置，像球形关节、轮子、气缸或者铰链。

大多数协同应用都会有实时交流模块，考虑到现有市面上已经有很多成熟的实时交流工具如QQ等，建立独立的模块不仅重复劳动增加成本，而且效果一般，于是将这一模块作为分系统集成到系统总体框架之中。

* 1. 应用模式分析

多人协同环境下的3D建模，旨在利用网络环境带来的互联优势，提供一种全新的3D场景协同作业模式。相比于传统的单机3D场景建模，这种作业模式强调协作性，轻量级，实时性，以及多适用面。其带来的优势主要有以下几点：

1. 轻量级3D建模作业，无需专业的3D建模软件操作知识，甚至无需环境的配置。
2. 针对轻量级应用，如房间设计、花园装饰、小型关卡设计、电子商务布置等。
3. 可实现多人协同对同一场景进行编辑，不同领域的领域专家可以实时协同工作。

为了能够快速实现这样的协同3D应用，并解决实现过程中的种种问题，需要有一套完整的面向协同的3D建模平台。而实际应用实现人员则可以对平台进行相应的定制，通过对功能的裁剪、模型的控制、人员权限的控制来进行特定应用的实现。而基于零件库的设计正好能达到这一目的，只需要更换不同的零件库系统，就能将系统应用与不同的业务场景，而且这一设计将零件提供者和产品设计者的工作分离，提高了合作效率，如图3所示。

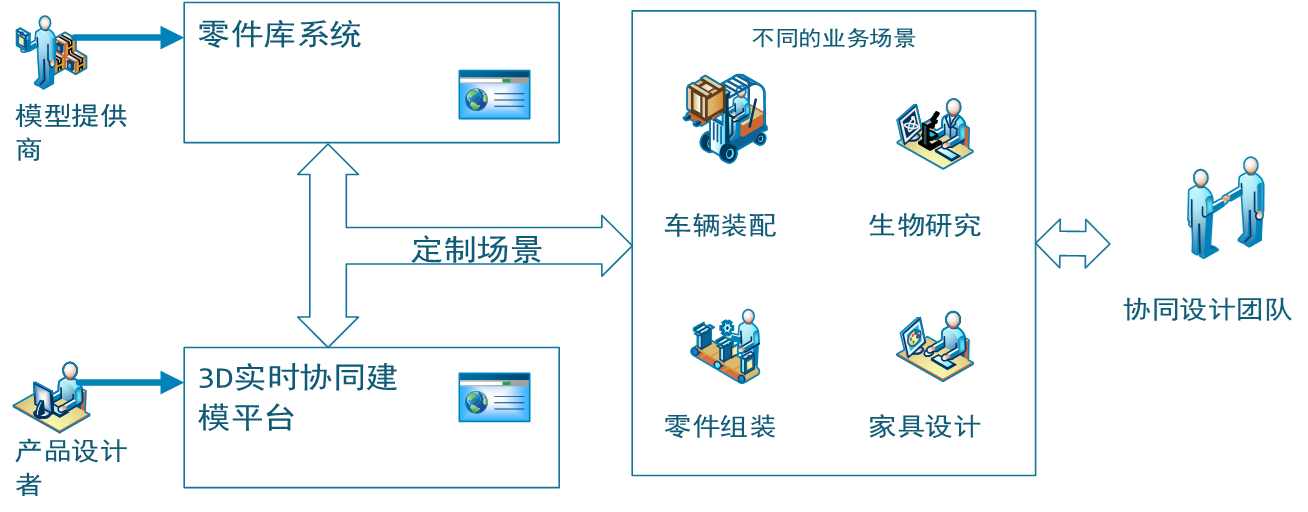


图 3应用模式示意图

本课题所提出的面向协同的3D建模平台，包括3D模型管理、场景协同交互、用户权限控制等多个功能模块和可配置模块。通过建立这样的平台，具体应用的设计者便可以通过定制化来实现多种应用系统。定制化的过程通过模型定制、功能定制、人员权限的定制来实现不同应用系统的差异化，从而形成包括远程教育系统、虚拟会议室、协同设计、娱乐游戏活动等多种基于网页的协同3D应用系统，提供给具体的协同用户进行使用。

基于成熟的零件库系统，在3D模型的管理方面，利用网络带来的便捷性，网络上海量的3D模型数据将有可能被系统所利用，作为协同3D建模的模型提供源。当然，这要求系统能够对更多的3D模型进行识别和利用；另一方面，在云计算兴起的今天，3D模型的供应商完全可以通过云的方式进行提供，按需索取，定制生成，通过服务等方式与系统无缝集合，从而为系统的应用者提供更大的可定制性[27]。

* 1. 基于零件库的工作流程

由于框架基于现有的成熟零件库，于是可以设计一种工作流程，将零件库与协同建模平台紧密结合在一起。三维建模人员用专业的3D模型设计工具3ds Max 、Maya等生成组件模型文件后，上传至组件库经简单的配置就能供3D场景模块调用。我们将组件库以云存储的方式保存至云端，方便多人进行上传与共享。于是，只要扩充组件库就可以实现更加丰富的设计，而更换组件库就能进行不同领域的产品设计，如汽车制造、家居设计等。如图4所示。

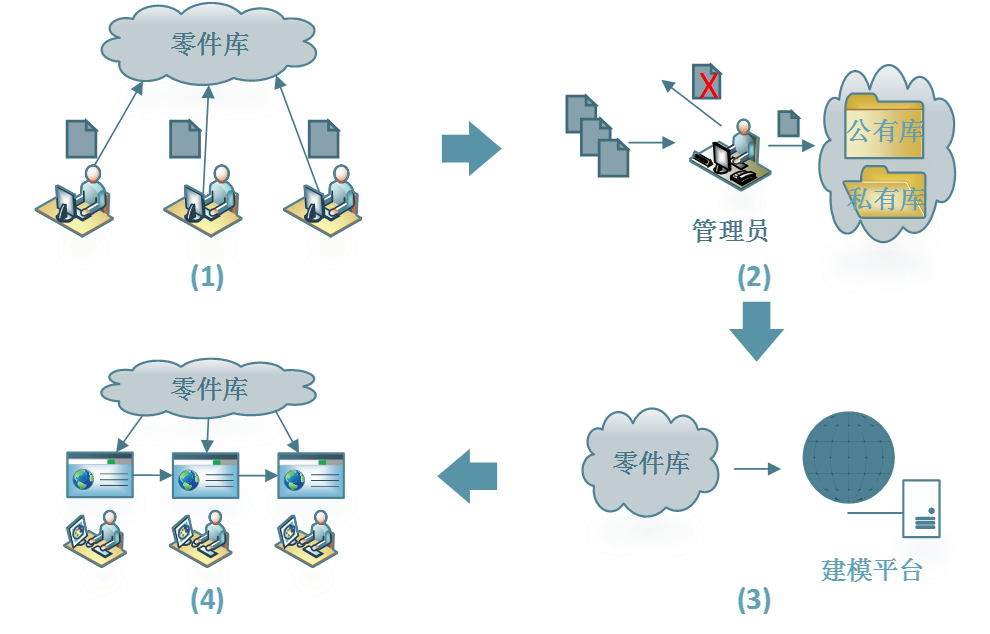


图 4基于零件库的协同建模平台工作流程

例如开发一个家居装修协同设计系统时，可以提供组件扩展功能给家具提供商，家具提供商定期将自己的最新产品制作成3D模型上传至模型库，设计师或普通用户就能利用海量家具通过任意拖动摆放调整来完成新家的设计方案。

* 1. 本章小结

本章提出了本课题中多人协同环境下3D建模框架的整体概况。本章首先从应用场景出发，通过分析典型的应用场景，得出在新环境下对于多人协同3D建模系统的需求和系统需要解决的主要问题。随后针对这些问题，提出了框架的总览，分层次介绍了每个部分的主要功能和解决的问题。然后对本架构的应用模式进行了分析，并总结了本框架和研究的主要应用方式和拓展点，最后根据提出的协同框架，设计了一种基于零件库的协同建模系统工作流程[36]。

1. 协同3D场景数据的存储

在本章中将讨论对协同3D场景数据的存储工作。基于现有的对于协同3D场景数据的传输研究，本章首先对面向协同环境的3D场景数据进行归纳，总结出在本框架下实现多人协同的数据类型和表示方式，而后针对这些数据，抽象提取出一种合适的抽象数据结构，用于协同环境下3D场景数据的核心描述，并设计相应的文件格式。最后提出一种联合存储方案，实现去冗余的场景数据存储。

* 1. 协同3D场景数据分析

在协同的建模环境中，不同的操作者之间需要进行数据的交换，同时操作者（客户端）和操作中心（服务器）之间也需要进行数据交换。在3D建模的环境下，一个协同3D场景的数据主要由两大部分构成：一是主要的场景建模数据，二是在建模与协同过程中的其他信息，如版本、操作、属性等，这需要有一种比较良好的数据定义来组织。

现有的研究提出了抽象场景的方案。抽象场景的结构基于原场景的场景图结构，但是为了减少原场景图的数据量，又要尽可能地保留其原有的结构化信息，主要做了如下两步：分割出场景图中数据量较大的节点，比如几何节点和材质节点中的纹理节点，并将他们单独存储成文件。几何节点一般包含了成千上万个坐标点的集合，而纹理节点则一般是对二进制图片数据的封装。该些节点在场景建模的过程中几乎不可能被改变，所以我们可以将该些节点从原场景中分割。二是协同过程中使用的约束规则和用户数据等数据，这些数据不是专属于某一个特定场景的，因此也可以进行抽象分离。

考虑一般的存储情况，对于一些协同任务中的动态信息，存储时是可以对其进行舍弃的，不影响整个3D场景的还原。基于以上的分析，我们对协同3D场景数据进行两方面分类，如下表1所示。

表 1 协同3D场景数据分类表

|  |  |  |
| --- | --- | --- |
| 数据描述 | 是否专属场景 | 是否需要保存 |
| 场景树状结构 | 是 | 是 |
| 模型几何数据 | 否 | 是 |
| 模型材质数据 | 否 | 是 |
| 模型三维信息 | 是 | 是 |
| 模型附加数据 | 是 | 是 |
| 模型约束信息 | 是 | 是 |
| 模型约束规则 | 否 | 是 |
| 操作指令数据 | 是 | 否 |
| 场景用户信息 | 是 | 是 |
| 用户数据描述 | 否 | 是 |
| 场景状态信息 | 是 | 是 |

因此，对于某个特定的协同3D场景，需要保存的专属数据有：场景的树状组织结构，即场景由哪些零件组成，各模型间的组织关系；各模型的三维信息，包括三维坐标数据、三维转角数据、三维缩放数据，这些数据可以用一个特定矩阵来表示；各模型的附加数据，这类数据主要由不同的业务决定，如给模型附加文字标注、附加计算参数等；各模型的约束信息，用于记录模型或模型间需要遵循的模型约束规则；场景用户信息，记录了参与该设计任务的用户成员；场景状态信息则包括场景名称，文件描述，版本信息等场景的各类标记信息。

对于这些专属数据，我们统称为抽象场景数据。对于分割出来的，不属于特定场景而是单独存储的文件称之为资源。几何节点对应的文件叫做几何资源，纹理节点对应的文件叫做纹理资源。这些资源文件并不同抽象场景一起存储到数据文件中，而是单独存储。

* 1. 抽象场景数据结构

三维场景的一般组成结构是场景由许多组合零件组成，而组合零件又由更小的组合零件和不可分割的小模型构成，以此类推，形成整个场景树。一般来说，我们把场景中的没有子结构，即不能继续分解的最小元素称为元模型。元模型一般是一个非常基本的几何结构，如球体，圆柱体，长方体等，或是一组由点面组成的片元。

对于抽象场景数据，基于现有的传输数据研究，设计一种树状数据结构如下图5所示。

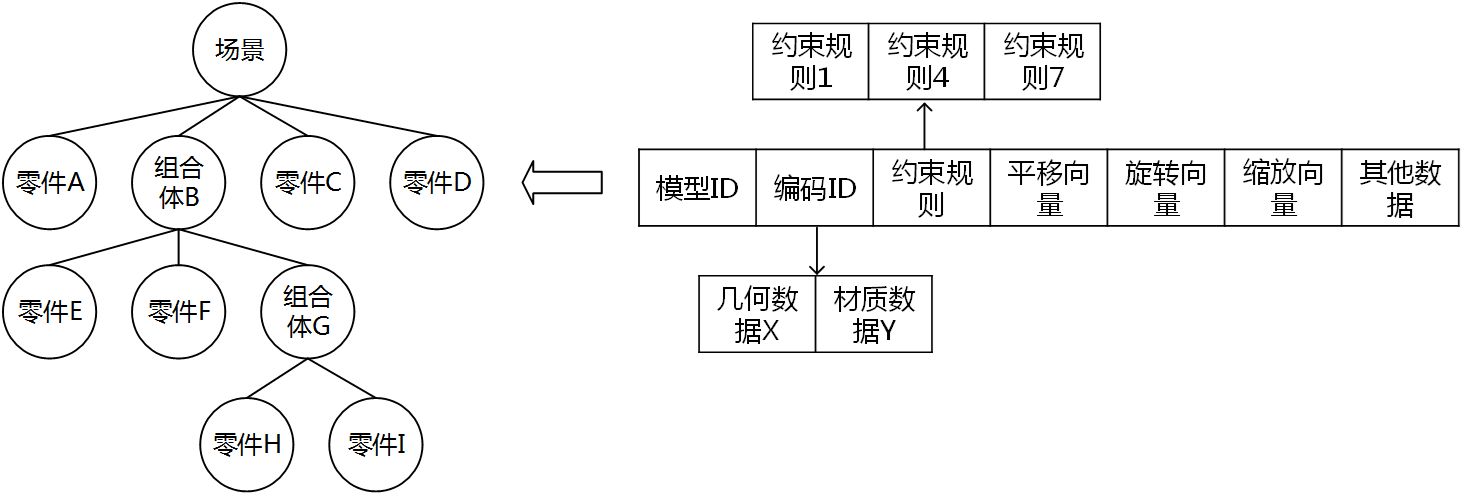


图 5 抽象场景数据结构示意图

我们用树状结构来表示场景的几何信息。但作为抽象结构，我们只维持这个树状结构，也就是说具体的数据内容被省去。简单理解，就是在树状结构中仅保存组成场景的模型或组合模型的标识（ID）。此外，还包括对于模型或组合模型的变换，如移动，放缩，旋转等信息。这些几何变换信息与具体模型或组合模型信息一一对应。数据根节点下，包含多个模型或组合模型。这些模型或组合模型也是概念结构，当然我们只包含对次概念结构的一个引用，也即一个ID的指向。这样，只需要维护一个包含ID的树状结构即可完成对整个场景的数据描述。另一方面，数据信息中包含了对这些模型或组合模型的变换信息。每当一个模型或组合模型被包含到当前场景中时，需要对其进行合适的变换，如移动位置，放缩大小，旋转变换等，以放置与当前场景中的合适部分，并成为当前的场景的组成部分。而这些在建模时变换信息需要被保存下来以帮助在复原场景的时候对场景组成进行合适的变换。在进行实际场景的展示时，可以通过这种层次结构一层层地递归寻找模型和组合模型的概念结构，直到全部的模型概念结构。而后通过概念结构中的实际模型ID（当场景概念结构的类型是模型时，此部分数据信息储存的实际是一个模型ID，用于对实际场景数据的映射）一层层地构造出含全部场景几何数据的可操作或展示的场景。

* 1. 抽象场景数据文件

针对上述的数据结构，我们设计了一种用于存储抽象场景数据的文件格式。该文件格式由标准的json格式的字符串构成，大致形式如图6所示。



图 6抽象场景文件格式样例

JSON(JavaScript Object Notation) 是一种轻量级的数据交换格式。它基于ECMAScript的一个子集。 JSON采用完全独立于语言的文本格式，但是也使用了类似于C语言家族的习惯，这些特性使JSON成为理想的数据交换语言，易于人阅读和编写，同时也易于机器解析和生成。有很多成熟的库能对json数据进行封装与解析，十分方便。

JSON建构于两种结构，一是“名称/值”对的集合。不同的语言中，它被理解为对象（object），记录（record），结构（struct），字典（dictionary），哈希表（hash table），有键列表（keyed list），或者关联数组 （associative array）。二是值的有序列表，在大部分语言中，它被理解为数组（array）。

本文设计的抽象场景文件将抽象场景组织为对象，对应的json数据可以分为三部分。第一部分是场景总体信息，用一些字段进行描述。如Title字段表示的是场景的标题，Edition字段表示的是场景的版本，Count字段记录的是构成场景的元模型总数量，Users字段记录的是用户数据；第二部分是元模型数据。每个元模型字段我们用s字母开头作为标识，并按顺序进行编号。每个元模型存储了三维数据共九个数字，模型的id，模型在模型库中的索引，模型所需要遵循的约束规则索引，以及模型的附加数据；第三部分是关系数据，用于标识元模型间的组合关系。每一组组合关系我们用c字母开头作为标识，并按顺序进行编号。每组关系记录了组合模型的id，以及其对应的模型集合。通过一组组关系数据便可对树状数据进行表示。

* 1. 联合存储方案

上述工作提到，对于3D场景中每个场景独有的数据，我们提取出来作为抽象场景数据，而其他的数据则分别存储。这样可以减少数据存储的冗余性。对于任意一个协同3D场景，需要对各类数据进行抽取，分别存放，也需要能够将各种数据组合起来还原成3D场景。因此本文提出一种联合存储方案，实现这一存储和还原过程，如图7所示。

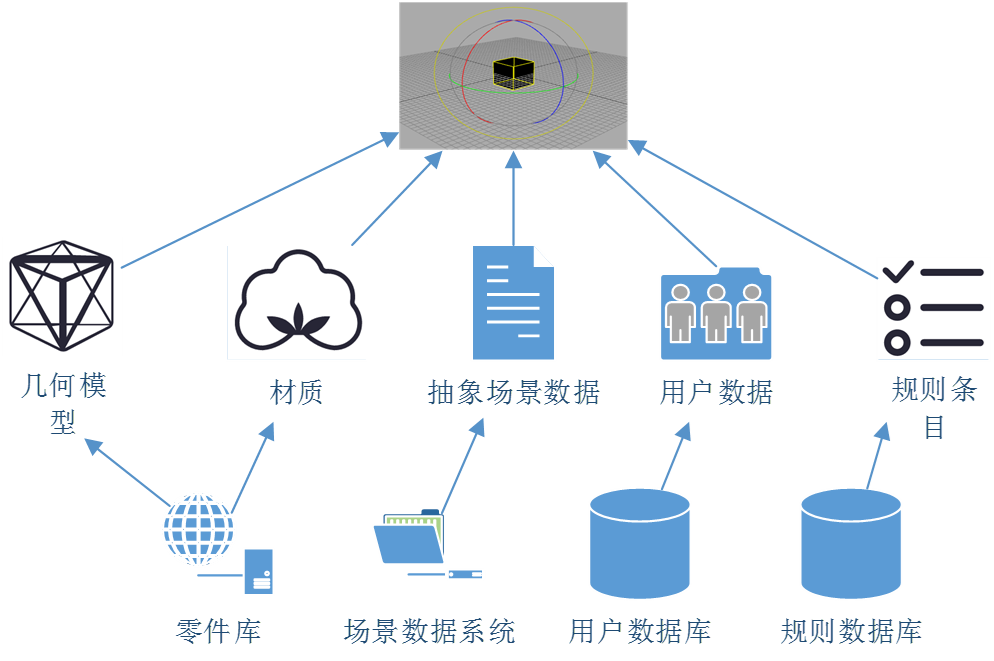


图 7 3D场景联合存储方案示意图

存储过程是对全部数据的一个拆解过程。将3D场景数据抽象成抽象场景数据文件存放在对应的文件系统里，用户数据和规则数据由管理员进行配置，存放于单独的数据库中，而模型数据是由零件库系统单独进行组织管理。

还原过程则描述为如下五个步骤，使用各种数据联合起来可以实现3D场景的还原：

1. 取得抽象场景数据文件，提取基本的场景信息；
2. 根据文件中的用户索引，从用户数据库中提取用户信息；
3. 根据文件中每个元模型的模型索引，从零件库中获得模型的几何资料和材质资料，并加载导入协同平台；
4. 根据文件中每个元模型的三维数据，将其应用到每一个元模型，并根据文件中的组合关系对元模型进行组合；
5. 将文件中的规则数据、附加数据导入系统中。
   1. 本章小结

本章首先介绍了在协同3D建模过程中所需要的数据，为建立用于保存的抽象场景结构提出了具体需求。然后通过具体描述抽象场景结构，详细讲述了结构中各个部分如何满足这些需求并提供了足够的数据和功能支持。而后通过设计对应的文件格式来对数据结构进行实现。最后，通过上述基础提出了一种联合存储方案，详细描述了对于Web3D实时协同建模平台场景的存储和还原方法。

1. 基于HTML5的协同建模

目前已经有一些协同应用使用了HTML5。本章总结现有研究成果，对HTML5技术在协同建模中的应用方法进行统一论述，并与传统的实现方法进行对比。分别是基于WebGL的三维场景构建，基于WebSocket的协同通信，和基于WebStorage的数据存储。

* 1. 基于WebGL的三维场景构建
     1. 场景构建方法

为了进行3D产品设计，我们需要在用户界面上创建一个三维设计场景作为工作空间，HTML5提供的WebGL特性能很好地支持在浏览器中展示3D图形内容。WebGL（Web图形库）是一个交互式3D图形库，提供API能在任何Web浏览器不使用插件地利用JavaScript[38]。不过，WebGL提供的大量原生API对于大部分开发者来说都是十分晦涩难懂难以使用的，好在目前已经出现了很多优秀的基于WebGL的第三方开源库，如Three.js、Babylon.js、SceneJS、x3dom等。

Three.js 是一个开源的JavaScript 3D引擎，该项目的目标是创建一个低复杂、轻量级的3D库，用最简单、直观的方式封装WebGL中的常用方法。目前由一个强大的贡献团队维护[39]。其显示效果如下图所示。

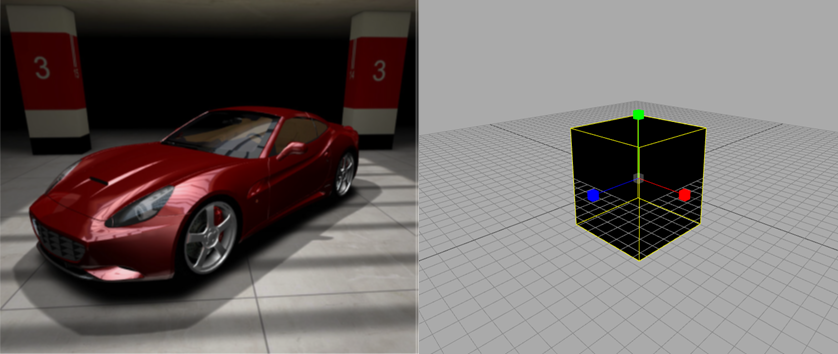


图 8 Threejs有优异的3D显示效果

PhiloGL 是由Sencha 实验室开发的一个新的WebGL开源框架，提供了强大的API，可帮助开发者轻松开发WebGL并整合到Web应用中，实现数据可视化，代码。创作和游戏开发。

Babylon.js是一个强大的3D游戏引擎，由Microsoft的员工David Cathue主导开发。和Three.js相比，three.js更倾向于动画，而Babylon.js则更适合游戏开发。

SceneJS是一个开源的JavaScript 3D引擎，特别适合需要高精度细节的模型需求，比如工程学和医学上常用的高精度模型。

X3DOM是一个实验性的开源框架，它有一个原创的X3D元素，可以通过X3D插件、Flash或WebGL渲染。

CopperLicht 是一个“商业级别的WebGL 3D引擎和编辑器”，可以免费使用，但是要想获得未压缩的完整版带支持文档的源码和其他服务，则需要购买授权。

上述每个框架都有各自的特点，开发者可以根据自己的需求选择最适合自己的框架。我们使用Three.js作为3D引擎，一是因为它是目前最受欢迎的WebGL框架，二是它最符合建模平台的需求，三是它对于不同的模型文件有着最好的支持，如下表2所示。

表 2各主流WebGL框架对模型的支持

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | *X3dom* | *SceneJS* | *Babylon.js* | *Three.js* |
| STL | X | X | O | O |
| OBJ | X | O | O | O |
| glTF | X | X | O | O |
| Babylon | X | X | O | O |
| Collada | X | X | X | O |
| MD2 | X | O | X | O |
| PLY | X | X | X | O |
| 3DS | X | O | X | X |
| X3D | O | X | X | X |

可以看到Three.js覆盖面最为广泛，具有良好的适用性。使用Three.js与动画库Tween.js来建立三维工作空间，以支持用户进行3D组件装配。通过将Three.js的WebGLRenderer与HTML5的Canvas绑定，就可以基于Three.js生成三维设计场景并在浏览器中渲染出三维图像。而利用Three.js的Loader加载三维模型文件，就能将设计所需要的组件的添加到设计场景中。

对于工作空间中的三维组件，通过修改其position、rotation、scale参数能直接改变其对应的三维空间形态，这些参数都是三维的向量；我们也可以修改matrix参数来进行更为复杂的变换，这是一个包含了全部三维信息的四阶矩阵。我们所支持的各项三维组件装配操作都是基于对上述参数的计算修改。

* + 1. 与传统插件法对比

早期的Web3D技术非常不成熟，比如Java Applet实现的简单交互式三维动画，不仅需要下载一个巨大的支持环境，而且画面非常粗糙，性能也很差。究其原因，就在于早期的这些交互式三维动画在做图形渲染时，并没用直接利用到图形硬件本身所带的加速功能，也就是，即使安装了性能很高的显卡，对于Web页面的交互式三维动画的显卡也起不了什么作用。

后来，Adobe的Flash Player浏览器插件几乎在一夜之间，占据了Web交互式三维动画的半壁江山。和Java Applet 不同的是，它直接利用操作系统提供的图形应用程序接口，来调用图形硬件的加速功能，实现了高性能的图形渲染。这么一来，交互式三维动画几乎可以瞬时加载完成，而且对于相当复杂的交互和三维动画而言，它都表现不俗。

可是，这样的解决方法也有不少的问题。首先，它是通过浏览器插件来实现的，这就意味着对于每种不同的操作系统和浏览器的组合，都需要下载特顶板的插件，对于手持设备上运行的比较特殊的操作系统的操作系统和浏览器，就可能没有对应版本的插件；其次，对于操作系统上的图形应用程序接口的调用，它并不是遵循一个公开的标准。比如，在Windows平台上，Flash是通过调用DirectX来实现，而Mac OS上则是调用AGL。

这两点不足，很大程度上限制了Web交互式三维动画的应用范围。微软推出的Silverlight，本质上也无法解决这个问题。因此，开发一种无需浏览器插件支持、通过统一的图形应用程序接口调用来实现Web交互式三维动画图形渲染的技术，就变得很有必要，而这正是WebGL的革命意义所在。将WebGL与传统的Web3D实现方法对比情况如表3所示。

表 3 WebGL与传统方法对比

|  |  |  |  |
| --- | --- | --- | --- |
| 方案 | 需要插件 | 使用硬件加速 | 统一接口调用 |
| Flash(DirectX) | 是 | 是 | 否 |
| Silverlight | 是 | 是 | 否 |
| Threejs(WebGL) | 否 | 是 | 是 |

尽管HTML5中引入了canvas标签，这个标签本身支持Web交互式三维动画的制作，但是，如果没有WebGL的支持，它并不能调用图形硬件加速功能。因此，此前虽然很多浏览器支持canvas标记，但由于性能不够好，无法得到广泛的应用。可是，一旦加入了WebGL的支持，它的面貌就焕然一新了。WebGL完美地解决了现有的Web交互式三维动画的两个问题：第一，它通过HTML脚本本身实现Web交互式三维动画的制作，无需任何浏览器插件支持；第二，它利用底层的图形硬件加速功能进行图形渲染，是通过统一的、标准的、跨平台的OpenGL接口实现的。这意味着，以后不需要通过任何浏览器插件，仅仅用HTML和JavaScript，就可以作出性能丝毫不亚于现在用Flash、Silverlight等做出来的Web交互式三维动画，而且在任何平台式都能以同样的方式运作。

* 1. 基于WebSocket的协同通信
     1. 协同通信方法

协同功能是多人能够同时进行拼装，而且实时看到相同的画面。为了实现多人设计工作的实时同步，让每个用户工作空间中的内容始终保持一致，我们采用如下的方法实现数据同步：当某一用户进行了某一操作后，其他用户端都“模仿”相同的操作。具体可表述为如下过程：第一步，浏览器端对用户的操作进行立即响应，修改本地数据与更新画面显示，并将这一操作转化成指令发送给服务器；第二步，服务器接收指令进行处理，然后发送给其他用户的浏览器端；第三步，其他浏览器端接收到指令后使用指令中的数据进行“模仿”，在本地执行同样的操作。任一用户的任一操作都会在所有用户的工作空间中得到响应，如此便可达到实时同步的目的，同时用户的操作会得到即时的反馈，保证了用户体验。

在上述同步过程中，需要将设计过程中的各种用户操作和操作参数封装成操作指令，称之为“指令化”[40]。我们用字符串拼接的方式把各参数字段连接成为一个长字符串，由发送方序列化为json格式进行传输，由接收端负责解析处理。除了能直接支持基本的增删改等三维组件操作之外，通过组合使用这些基础指令，也能对复杂的三维操作进行同步。例如某用户对一个多组件的组合体进行复制时，可以先使用多条ADD指令在其他用户端创建出子组件，然后通过COM指令将子组件进行组合，最后利用MOD指令对组合完成的组合体进行位置转向等三维参数的同步。

表 4 协同操作指令集

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **指令码** | **发送方** | **接收方** | **指令数据** | **指令功能** |
| ADD | Browser | Browser | {groupID,partID,partCode} | 添加组件 |
| DEL | Browser | Browser | {groupID,partID} | 删除组件 |
| MOD | Browser | Browser | {groupID,partID,position,rotation,scale,[optional data]} | 修改组件 |
| COM | Browser | Browser | {groupID,partID,idList} | 组合组件 |
| LOC | Browser | Browser | {groupID,partID,flag} | 锁定组件 |
| UPD | Browser  (host) | Server | {groupID,sceneData,lockData} | 更新数据 |
| CHE | Server | Browser  (not host) | {sceneData,lockData} | 数据校验 |
| CLO | Server | Browser  (not host) | {} | 关闭协同 |

框架支持用户独立进行设计工作，也允许用户在设计的任何阶段开放协同环境以邀请其他用户加入到自己的工作中。开放协同环境的用户我们称之为Host，其他参与用户称之为Participant。按照我们的同步方法，数据将会以复制模式分布在每个用户的浏览器中，服务器端也会保留一份数据。因此为了维护数据的一致性，我们始终以Host的工作空间数据作为基准，Host会在数据发生改变的时候使用Update指令更新服务器端的数据，服务器则定期发送Check指令供Participant进行数据校验，以修正协同过程中可能出现的数据错误。另外，当Host开放协同环境时，会发送一次Update指令将当前数据发送至服务器；当新的用户加入协同环境时，服务器会发送一次Check指令给新用户令其初始化为当前的协同状态。

虽然所有用户的工作空间始终保持一致，但每个用户是拥有自己独立的任意的观察视角的，画面的显示由各浏览器利用工作空间数据自行绘制，并不需要服务器进行计算。服务器主要作为“信使”，与所有浏览器端频繁地通信来保证同步过程。我们借助于HTML5中的WebSocket特性，能提供更加稳定有效的通信支持。WebSocket规范引入了WebSocket JavaScript接口，它定义了一个全双工连接，信息可以在客户端和服务器之间发送。WebSocket标准简化了在双向网络通信和连接管理的复杂性。这个技术允许实时同步多用户。其连接示意图如图9所示。

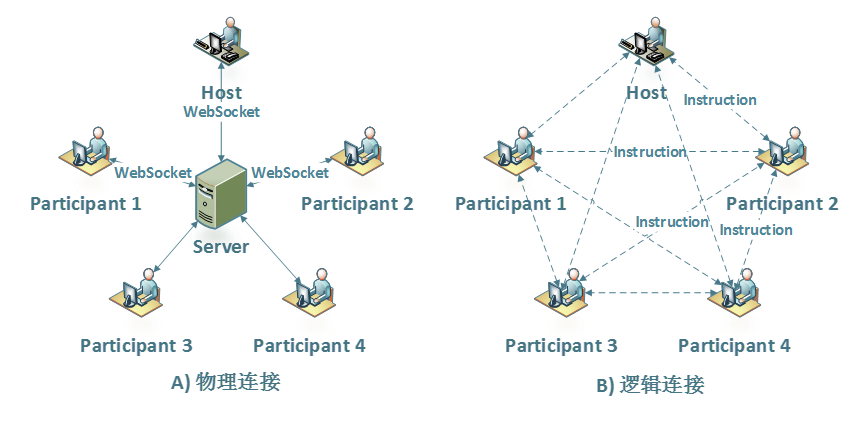


图 9 协同物理连接与逻辑连接示意图

通过上图我们可以看到，虽然协同工作的物理连接是星型的，但逻辑连接是网状点对点的协同。

* + 1. 与传统连接法对比

最开始的方式是轮询。轮询(Http Polling)是指不管服务器端有没有更新，客户端(通常是指浏览器)都定时的发送请求进行查询，轮询的结果可能是服务器端有新的更新过来，也可能什么也没有，只是返回个空的信息。不管结果如何，客户端处理完后到下一个定时时间点将继续下一轮的轮询。

这种方式明显的缺点是存在浪费，占用带宽较多。于是在这个基础上又有了长轮询技术。长轮询(Long Polling)的服务其客户端是不做轮询的，客户端在发起一次请求后立即挂起，一直到服务器端有更新的时候，服务器才会主动推送信息到客户端。 在服务器端有更新并推送信息过来之前这个周期内，客户端不会有新的多余的请求发生，服务器端对此客户端也啥都不用干，只保留最基本的连接信息，一旦服务器有更新将推送给客户端，客户端将相应的做出处理，处理完后再重新发起下一轮请求。可见，长轮询的特点是一是服务器端会阻塞请求直到有数据传递或超时才返回；二是客户端响应处理函数会在处理完服务器返回的信息后，再次发出请求，重新建立连接；三是当客户端处理接收的数据、重新建立连接时，服务器端可能有新的数据到达；这些信息会被服务器端保存直到客户端重新建立连接，客户端会一次把当前服务器端所有的信息取回。

这两种轮询方式都是被动的，半双工性质。WebSocket协议是HTML5定义的一种新协议，它实现了浏览器与服务器全双工通信(full-duplex)。通过浏览器发出websocket连线请求，然后服务器发出回应，建立一个联系的通道。对比效果如下表5所示。

表 5 WebSocket与传统方法对比

|  |  |  |  |
| --- | --- | --- | --- |
| 方案 | 占用带宽 | 推送类型 | 双工性 |
| Http Polling | 多 | 被动 | 半双工 |
| Long Polling | 中 | 被动 | 半双工 |
| WebSocket | 少 | 主动 | 全双工 |

websocket不仅节约了header的问题（websocket的head信息只有短短的2个字节）。更加重要的是是通信的稳定性，轮询在遇到网络问题之后，想要在不刷新页面的情况下恢复通信，非常困难，而websocket中提供了onclose函数来处理断开网络后的情况，这为我们与服务器的通信提供了可靠的保障。

* 1. 基于WebStorage的数据存储
     1. 数据存储方法

HTML5本地存储规范中，定义了2个重要的API：WebStorage和本地数据库WebSQLDatabase。WebStorage提供了两种存储类型API接口：sessionStorage和localStorage，sessionStorage的生命周期在会话期间内有效，而localStorage就存储在本地，并且数据是永久的，除非用户或程序对其执行删除操作。目前所有的主流浏览器都在一定的程度上支持HTML5的WebStorage特性。

localStorage主要作用时将数据保存在客户端中，而一般的客户端是指用户的计算机。在安全性方面，localStorage是域内安全的，即localStorage是基于域的，任何在该域内的所有页面，都可以访问localStorage数据。但是存在一个问题，就是各个浏览器厂商之间的数据是独立的，也就是说，如果在Firefox中的localStorage存储的数组，在Chrome中就无法读取。同样，由于localStorage数据是保存在用户的设备中的，因此在同一个应用程序在不同设备上保存的数据是不同的。

sessionStroage主要是将数据保存到会话中，其原理和服务器语言的session功能类似。sessionStorage存储的生命周期只保存在存储它的当前窗口或由当前窗口新建的新窗口，直到相关联的标签页关闭。

WebStorage以键值对形式对数据进行存储，如图10所示。这样的设计也非常符合本文的存储研究工作，可以很好地对json格式数据进行存取，其本身也包含对json格式的封装与解析功能。

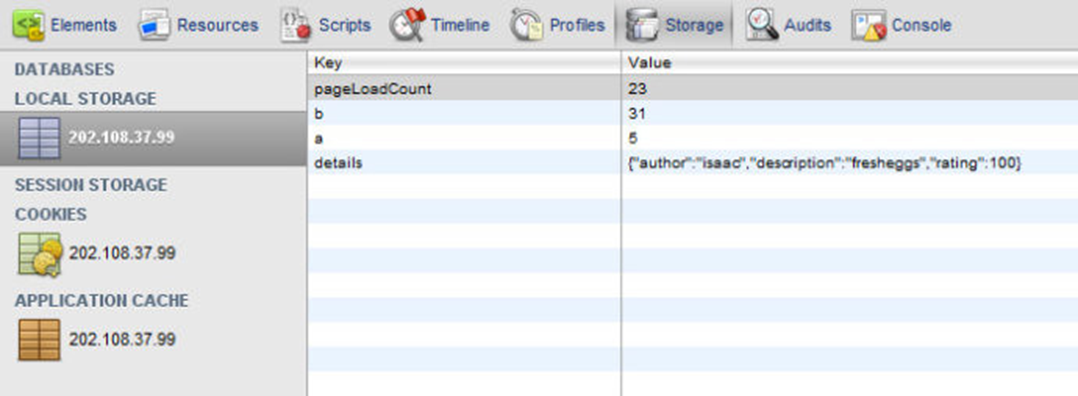


图 10 WebStorage通过键值对形式进行存储

* + 1. 与传统Cookie对比

WebStorage是HTML5中本地存储的解决方案之一，在HTML5的WebStorage概念引入之前浏览器兼容的本地存储方案只有使用cookie。而cookie的缺陷是非常明显的：

1）数据大小：作为存储容器，cookie的大小限制在4KB左右，尤其对于现在复杂的业务逻辑需求，4KB的容量只能简单存储一些配置字段。

2）安全性问题：由于在HTTP请求中的cookie是明文传递的（HTTPS不是），带来一定的安全性问题。

3.）网络负担：cookie会被附加在每个HTTP请求中，在HttpRequest 和HttpResponse的header中都是要被传输的，无形中增加了一些不必要的流量损失。

WebStorage是HTML新增的本地存储解决方案之一，但并不是为了取代cookie而制定的标准，cookie作为HTTP协议的一部分用来处理客户端和服务器通信是不可或缺的，session正是依赖于实现的客户端状态保持。WebStorage的意图在于解决本来不应该cookie做，却不得不用cookie的本地存储。两者对比如表6所示。

表 6 WebStorage与传统方法对比

|  |  |  |  |
| --- | --- | --- | --- |
| 方案 | 限制大小 | 请求时携带 | 永久性 |
| Cookie | 4KB | 是 | 否 |
| WebStorage | 5MB | 否 | 是 |

WebStorage提供两种类型的API：localStorage和sessionStorage，localStorage在本地永久性存储数据，除非显式将其删除或清空，sessionStorage存储的数据只在会话期间有效，关闭浏览器则自动删除。两个对象都有共同的API。

* 1. 本章小结

本章主要阐述了如何应用HTML5技术解决协同建模平台中的问题，详细介绍了方法与原理，并与传统的技术进行了比较。利用WebGL特性能建立三维协同场景，利用WebSocket能实现实时协同通信，利用WebStorage可以进行客户端数据存储。可以看出使用HTML5技术能实现良好的性能。在下一章原型系统中我们也将使用这些技术。

1. 原型系统的实现与应用

基于上述的研究工作，本课题开发了一个实际工程项目，同时作为原型系统来验证框架设计、场景存储、HTML5应用等工作的有效性。本章首先将介绍该工程项目的应用场景与业务需求，然后介绍该项目的总体设计与详细实现，最后通过功能和效率两方面来检验论文的研究工作。

* 1. 应用场景与需求分析
     1. 应用场景

项目来源于中科院某单位，希望建立一套卫星结构交互式设计系统，紧紧围绕空间科学任务概念设计阶段的交互需求，快速形成卫星及有效载荷的安装布局，实现平台与载荷的构型设计。同时为了支持科学家团队能实时协同地进行设计工作，希望系统能采用B/S结构，支持多人同时看见相同的设计场景。

在该项目基础上进行改进，开发一个基于Web3D的协同设计平台，支持多样化零件的选择，进行不同的协同拼装任务。

* + 1. 需求分析

结合通用的实时协同建模平台业务需求，提取核心业务，并将逻辑梳理成下述流程图11所示。以科学家小组为例，多名科学家将登录系统，由一名用户建立协同环境，其他用户将加入该协同任务。设计小组利用设计页面提供的常用结构模型，通过鼠标组装，完成卫星的概念设计。同时，也可在屏幕上通过精确输入每个结构的尺寸和位置，调整卫星的尺寸和位置，完成卫星结构的设计。最后，结构数据可保存至平台。通过交互式方式使用户能够快速生成简单、抽象的载荷和零件特征模型，达到快速设计的目的。将模型进行变更便可切换业务场景。



图 11交互式结构设计业务流程图

* 1. 系统的设计与实现
     1. 总体设计

本课题实现系统采用B/S架构，使用第二章提出的协同建模框架，具体所采用的技术如图12所示。

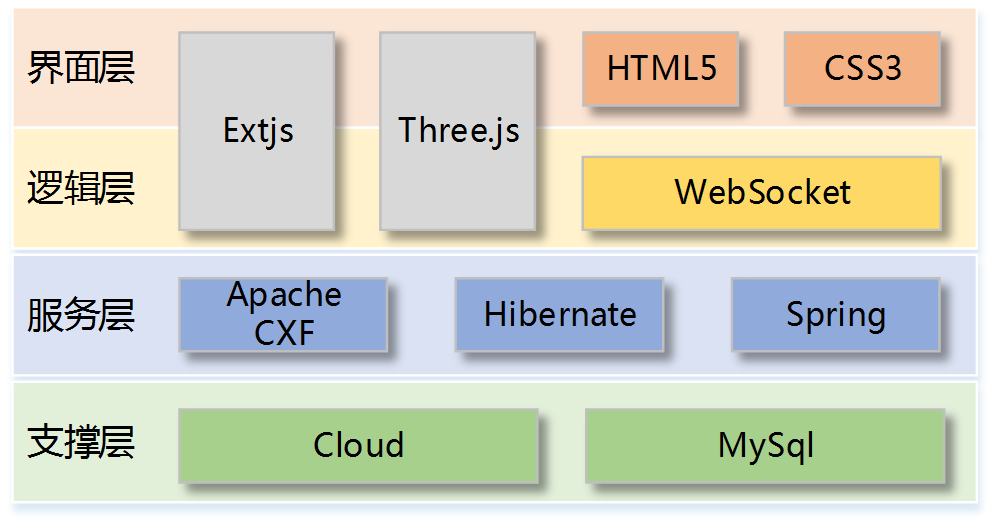


图 12 原型系统技术架构图

系统的技术架构图分为四个层次，分别为界面层、逻辑层、服务层、数据支撑层。

界面层：界面的主体部分由ExtJS丰富的界面和组件绘制完成，系统主要的业务功能部分即三维设计和分析部分采用了HTML5和three.js第三方JS库来实现。整个界面的主题和色调采用ExtJS的主题库以及CSS3来实现。

逻辑层：系统的前端采用了ExtJS的MVC架构，逻辑层采用ExtJS框架的控制器和数据模型来实现对界面的控制，其中也运用了Ajax技术、Http协议和HTML5的SessionStorage等技术。系统使用SessionStorage存储系统的全局变量，用于在不同页面之间传递信息，使用Ajax技术实现异步调用，运用Http协议向后端的服务层发送请求。

服务层：主要运用了Apache CXF、Spring、等技术来实现，Spring框架用来搭建整个后端框架，支持控制翻转和面向切面编程，运用Apache CXF技术将系统的功能封装成为Web Service和RESTful风格的接口，供前端逻辑控制层访问并返回消息给逻辑控制层。

数据支撑层：主要包括MySQL技术和云存储接口两部分，MySQL用于建立关系型数据库存储数据，云存储接口通过调用云存储接口实现文件的存取。

系统的功能架构如图13所示。

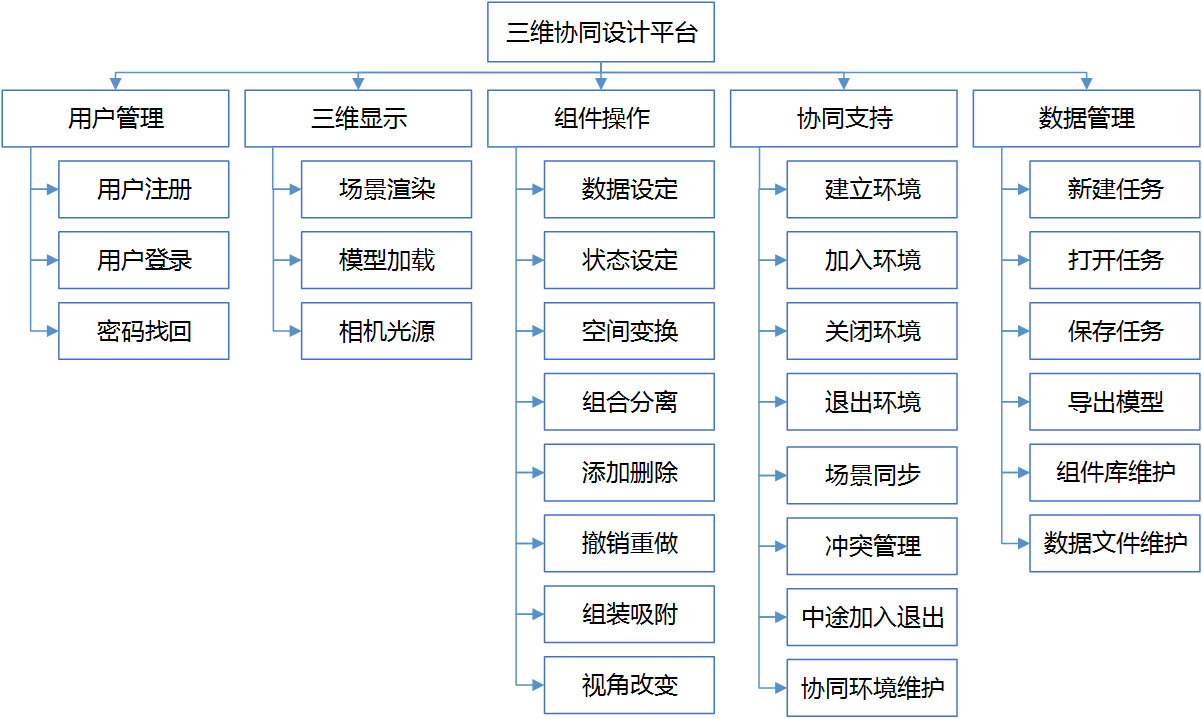


图 13 三维协同设计平台功能设计图

平台总共分为用户管理、三维显示、组件操作、协同支持、数据管理等五大块功能。用户管理主要进行用户相关功能如登录、注册等；三维显示功能主要包括三维场景的构建，图形显示相关部分；组件操作提供了丰富的对组件的操作功能，也是系统的主要交互手段；协同支持部分是系统提供的各种协同工作相关功能；数据管理则对数据部分进行存储与读取管理。

* + 1. 详细设计



该部分待完成。

* + 1. 功能模块实现

1）用户登录功能。登录模块使用的是LeanCloud的登录接口，代码逻辑在index.html中，具体细节参照LeanCloud的文档。

2）设计场景的平移、旋转与缩放。逻辑代码位于SceneManage.js中，调用了threejs库的OrbitControl.js。方法为使用如下代码：

orbitControl = new THREE.OrbitControls(camera, container);

将场景绑定，便实现了鼠标的平移旋转缩放功能。OrbitControl提供了一个'change'监听事件，当鼠标移动了场景视角时就会触发，可以在里面实现一些刷新事件等。

3）模型库中模型的添加与替换功能。模型库是一个容器，为ModulePanel.js。每一种类别的模型都是一个独立的view，集中放在modulepanel中。对于每一种类别的模型，以BasicComponentGridPanel.js为例，这个view会关联模型图片的位置resources/images/basicLoad/以及view的store即BasicLoadStore，而store又会与model相对应。BasicLoadStore会查询resources/data/BasicLoad.json中的数据，关联索引和名字。因此若想添加或替换模型，只要修改这个json文件即可。另外图片、obj和mtl文件应该与模型同名并放于resources中正确的文件夹。

4）模型库模型加载功能。双击模型库中一张图片，逻辑被ModuleController.js中响应。创建一个Structure对象并调用其初始化方法。注意模型加载完成是需要时间的，需要模型加载完成才能进行的逻辑要用回调函数的方法进行实现。

5）鼠标拾取模型功能。鼠标拾取模型是一个把鼠标二维坐标转换为场景三维坐标的过程。原理是鼠标点击页面上一点时，会从这一点向三维场景里发出一条射线，被射线击中的对象都会被捕捉到。代码逻辑位于StructManage.js中的addMouseControl里。具体如下：

//射线法选中物体，注意各种传入的参数

var raycaster = new THREE.Raycaster();

var projector = new THREE.Projector();

var directionVector = new THREE.Vector3(((event.clientX - Struct.container.getBoundingClientRect().left) / Struct.container.width ) \* 2 - 1, -((event.clientY - Struct.container.getBoundingClientRect().top) / Struct.container.height ) \* 2 + 1, 1);

projector.unprojectVector(directionVector, Struct.camera);

raycaster.set(Struct.camera.position,directionVector.sub(Struct.camera.position).normalize());

然后调用下面这条语句，就能获得一个射线击中对象的列表。

raycaster.intersectObjects(Struct.scene.children, true));

遍历列表寻找被击中的模型。

6）新建场景功能。代码逻辑位于SceneManage.js中的resetScene函数中，共四步：一是重置视角，二是调用每个组件对象的删除函数，三是清空存储，四是刷新渲染（每次场景有更改都要刷新场景）

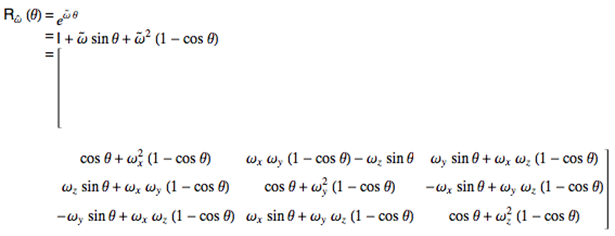
7）导出场景功能。为了弹出等待窗口而不让页面卡住设置延迟。代码逻辑位于ButtonController.js中的btnExportClick函数中，导出的数据使用storageManage.exportObj获得。为了了解exportObj的工作原理需要熟悉obj文件格式。导出数据时先把所有的组合体解散变成单一的组件，然后获得每一个组件的数据，完成后恢复组合体。而为了将所有数据合在一起需要设置偏移量来保证索引的顺序正确。

8）平移与旋转与缩放功能。组件的平移旋转缩放使用的是第三方库TransformControl，代码位于StructManage中的addTransformControl函数，创建对象并绑定相机等就可以实现操作轴效果。这里有两个设置，一是为了防止组件移动越界，设置了监听函数保证越界后的纠正。二是拖动轴会和旋转场景这个动作重合，为了在拖动轴时不旋转场景，设置了一个标记transformFlag，并修改了OrbitControls库，旋转场景动作时先检查标记，若此刻在拖动轴则屏蔽旋转动作。

9）组合与分离功能。组合与分离的逻辑位于StructManage.js中的combinaOperate与combineByID函数中，利用Structure对象的initByCombina方法。会使用到全局坐标与局部坐标的相互转换。

10）复制功能逻辑位于StructManage.js的structCopy函数中。注意复制组合体时需要先复制出组合体中的每一个个体，然后再把他们组合。组合需要等待模型全部加载完成才能进行，因此需要使用回调函数。凡是涉及模型加载的都要注意异步的问题。

11）吸附功能。吸附功能的操作是先后点击两个组件的两个面，然后这两个面会粘贴到一起。我们重新定义问题，即如何通过旋转移动组件，使得两个组件的面吸附到一起，重点在于求解旋转移动组件的矩阵。通过Threejs，我们能够获得先后点击的两个面的两个点击点的信息，包括位置与法向量，最终当面重合的时候这两个点应该是重合的，而且法向量反向平行。重合点容易，问题在于解决法向量反向平行。于是问题转换为现在有两个向量，求一个矩阵，使得一个向量通过矩阵变换后与另外一个向量反向平行。这里将使用罗德里格旋转公式。已知单位向量[Inline3](http://images.cnitblog.com/blog/477176/201302/15164324-7a2cd186fda44852bff992caed574b74.gif) ， 将它旋转θ角。由罗德里格旋转公式，可知对应的旋转矩阵[Inline5](http://images.cnitblog.com/blog/477176/201302/15164326-1211440157be4633918d2e17d8a571f6.gif) ：

[](http://images.cnitblog.com/blog/477176/201302/15164342-44d9364dcbf740b0bd2605307ce3a3ad.png)

其中I是3x3的单位矩阵，omega^~ 是叉乘中的反对称矩阵r。根据旋转前后的两个向量值，先求出旋转角度和旋转轴，然后用罗德里格旋转公式即可求出对应的旋转矩阵。

* 1. 系统讨论

使用支持WebGL的浏览器才能登录设计环境。设计环境允许用户以任意视角对模型进行观察。用户从组件库中选取所需要的组件添加到场景中，通过常用的三维变换如空间位置平移、空间角度旋转、模型缩放等，或直接输入设计参数，以及其他的装配操作如吸附连接、对齐等，就能利用基本的卫星零部件构建出一颗卫星的概念模型。

同一时刻不同用户将看到一致的界面。虽然工作空间是一致的，但不同的用户可以拥有不同的观察视角，可以选择不同的组件进行操作。被其他用户正在操作的组件是不可使用的，这些组件将呈现为透明状态。任意用户对场景的任意操作都将导致所有用户的界面同步进行变化。

当组件长时间被某一用户占用时，会发生饥饿问题。我们希望通过某种动态显示的方法让正在占用的用户能感知到其他用户的等待。因此，当等待的用户点击那些被锁定的组件时，都让组件的占有者界面中该组件短暂地高亮显示，以提醒其尽快完成对组件的操作占用。

最终实现的系统如图14所示。



图 14 三维协同建模平台界面设计

主要区域为设计区，用户可以通过鼠标拖拽等方式对场景模型进行操作。右侧为模型库区域，显示基本的元模型作为建模设计的基础零件。上方的功能区则为建模过程提供强大的支持功能。

多名用户登录系统进行协同设计，能同时看见相同的设计场景，其效果如图15所示。



图 15多人协同建模

根据前面讨论的需求，目前系统的功能实现情况和功能性测试结果如表7所示。

表 7 原型系统的功能实现情况与测试

|  |  |  |  |
| --- | --- | --- | --- |
| 功能项 | 实现情况 | 测试效率 | 未来发展方向 |
| 模型导入 | 支持任意Obj模型的导入 | 对于太庞大的模型，会存在若干秒的延迟 | 采用统一的模型标准 |
| 基于Web的建模操作 | 使用HTML5和Threejs的无插件界面，支持各种鼠标键盘操作 | 在主流最新浏览器没有问题，但在老版IE浏览器中会出现显示问题 | 更为专业的建模和协同界面 |
| 实时同步效率 | 使用指令转发和数据复制进行同步 | 在用户数小于5，模型数小于20，局域网条件下能做到实时同步 | 增加更多用户和更多模型的测试。 |
| 协同冲突 | 实现了基于锁的协同操作 | 没有发生协同错误 | 采用无锁的冲突处理方法可以提高用户体验 |
| 协同数据一致性 | 测试了用户中途加入、用户中途退出、关闭协同状态等特殊情况 | 没有发生协同错误，资源得到了及时释放 | 加上数据校验以应对更多的未知错误情况 |

由上表可以看出，本课题所提出的面向协同的Web3D建模框架能够很好的支持整个平台的搭建和实现。各方面的功能均基本正常，性能表现良好。尽管有些功能尚处在初期开发阶段，实现方式较为简单，但通过验证其可以很好和系统进行集成，相关研究和开发有进一步深化的可能。

针对其他文献中有关协同3D或网络3D的相关应用和尝试，经过总结整理，现列出如下所示的结果比较如表8所示。

表 8 Web3D实时协同建模平台与传统研究对比

|  |  |  |  |
| --- | --- | --- | --- |
|  | 基于网页插件的3D协同平台 | Web3D实时协同建模平台 | 基于客户端的协同平台 |
| 操作平台 | 网页插件，需要安装，轻量级 | 无插件网页，无任何配置 | 客户端，需要安装和配置 |
| 性能效率 | 较差，取决于插件实现方式 | 可以接受，随着HTML5的浏览器性能的提升有进一步改进空间 | 很好，有成熟的图形渲染工具 |
| 数据传输方式 | 以完整场景为主，兼以用户操作为辅 | 以抽象场景为主 | 模型直接传输 |
| 通用性 | 差 | 相关应用已见雏形，在分布式和基于网页的3D展示方面有广阔的前景 | 很好，在传统设计领域非常成熟 |

通过对比可以发现，本课题提出的面向协同的Web3D建模平台，能够在克服前人通过插件方式实现网页3D的各类弊端的基础上，提供接近传统3D建模的性能和扩展性。相信在分布式计算和设备实时互联的新时代背景下，本课题提出的实现方式将有着更大发展空间与可用性。

* 1. 本章小结

本章通过分析功能需求，实际建立Web3D协同建模平台，并详细介绍了系统的技术框架、功能模块设计和细节设计。随后对实现的系统进行了功能性和效率两方面的考察，确定了验证的平台达到了预期的功能完备性和效率可行性，从一定程度上证明了本课题研究的可行性与有效性。最后通过和其他协同3D平台实现方式的比较，展示了本框架的优势和可扩展性。

1. 总结与展望
   1. 工作总结

本课题面向Web3D协同建模平台，通过分析实际应用场景，获取新形势下计算机可视操纵所面对的新需求和困难，在总结前人的研究成果基础上，有针对性地提出了面向协同的3D建模平台整体框架，并对3D场景存储方法和HTML5应用行了深入讨论。通过实际的项目实现，验证了整体框架的功能完备性和有效性。

本文的主要工作如下所述：

提出了一种Web3D实时协同建模平台框架。通过对业务需求的梳理分析，提出了一种基于零件库的协同建模平台框架，并描述了该框架的应用模式。同时设计了一种在该协同框架下的工作流程。

研究了协同3D场景数据的存储。通过对场景数据的梳理，提出了一种抽象场景数据结构，并设计了相应的文件存储格式，提出了联合存储的协同方案。

研究了基于HTML5的协同建模。总结了现有的HTML5技术在协同建模中的应用方法，并将其与传统方法进行对比。

原型系统的实现与应用。开发了一个原型系统验证工作的有效性，并按照软件工程的思路分别介绍了需求分析、总体设计与详细设计部分，给出相应的讨论。

* 1. 后期展望

本文主要针对面向协同的3D建模的整体框架设计，以及其核心3D场景存储设计的研究，为相关的应用和研究搭建了一个完整可用的平台。而3D协同作为一个有着很大研究价值和挑战的课题，在本课题的基础上，仍有许多需要解决与研究的问题：

1）模型的标准化与语义管理。为了使3D建模平台具有足够的模型支持，需要对不同建模工具、不同数据格式的模型进行一定的标准化操作，使之具有一定的通用性，同时在其他模块的使用过程中，如冲突处理等能够提供标准化的格式支持。此外，为了使建模操作具有一定的实际意义和智能指导功能，需要对模型进行更为自然的语义管理，使模型的管理能够以更为专业的方式，从而在建模过程中能够以自然语义的方式进行智能提示和自然语义冲突检测。

2）协同过程中的冲突处理的深化研究。协同伴随着冲突，对于协同冲突的解决一直是相关研究的重点所在，而对于3D协同的研究则开启了一个全新的领域。在本文中，介绍的冲突处理相对简单，同时对几何冲突和语义冲突没有进行进一步的扩展。为了保证协同建模的有效性和一致性，需要在这方面做进一步的研究。

3）需要有更多的应用和实际项目来验证框架并改进研究的方向。相对而言，目前面向协同的3D建模领域的相关研究和应用还较少，并没有形成一个系统的研究框架。因此，在3D技术以及网络互联技术不断发展的今天，需要有更多的应用和研究在此基础上继续探索，从而能够在这个领域形成系统的研究框架，改变研究内容相对分散的局面。同时也希望此领域的应用能够被更多人所关注并获得更大的发展空间。

参考文献

[1] Shen W, Hao Q, Li W. Computer supported collaborative design: Retrospective and perspective[J]. Computers in Industry, 2008, 59(9):855-862.

[2] Pengyu Du, et al. "A Real-Time Collaborative Framework for 3D Design based on HTML5." IEEE 20th International Conference on Computer Supported Cooperative Work in Design, 2016.

[3] 王成睿. Web3D虚拟现实平台及吊舱推进器虚拟操控的研发[D]. 集美大学, 2014.

[4] 王钢. 基于AutoCAD三维造型实时协同设计的研究[D]. 西南交通大学, 2011.

[5] 王丽君. 基于Web模式的3D装箱系统可视化关键技术及应用研究[D]. 大连海事大学, 2015.

[6] 赵瑞. 面向协同办公管理系统的3D可视化服务设计与实现[D]. 北京邮电大学, 2015.

[7] 李遇. 网页二维矢量图形协同绘制技术研究与实现[D]. 浙江工业大学, 2011.

[8] Tian F, Wang H, Zhang F, et al. Co-Creative Pen Toolkit: A Pen-based 3D Toolkit for Children Cooperatly Designing Virtual Environment[C]. Proceedings of the 10th International Conference on Computer Supported Cooperative Work in Design, (CSCWD 2006). IEEE, 2006: 1-5.

[9] Wang L, Guo J W, Yang C L, et al. O3D-Based Game Learning Environments for Cultural Heritage Online Education[M]. Entertainment for Education. Digital Techniques and Systems. Springer Berlin Heidelberg, 2010:417-428.

[10] Ieronutti L, Chittaro L. Employing virtual humans for education and training in X3D/VRML worlds[J]. Computers & Education, 2007, 49(1): 93-109.

[11] Tay F E H, Roy A. CyberCAD: a collaborative approach in 3D-CAD technology in a multimedia-supported environment[J]. Computers in Industry, 2003, 52(2):127-145.

[12] Nam T J, Wright D. The development and evaluation of Syco3D: a real-time collaborative 3D CAD system[J]. Design Studies, 2001, 22(6):557-582.

[13] 陈铮, 龚建华. 一个基于Java3D的协同虚拟环境工作平台[J]. 计算机应用研究, 2005, 22(8):136-139.

[14] 冯开平, 潘光洋. 基于Web3D的灯饰协同设计定制系统[J]. 图学学报, 2012, 33(3):56-60.

[15] Behr J, Jung Y, Keil J, et al. A scalable architecture for the HTML5/X3D integration model X3DOM.[C]// 3d Technologies for the World Wide Web, Proceedings of the, International Conference on Web 3d Technology, Web3d 2010, Los Angeles, California, July. 2010:185-194.

[16] Sons K, Klein F, Rubinstein D, et al. XML3D: interactive 3D graphics for the web[J]. 2010:175-184.

[17] Kolbe T H. Representing and Exchanging 3D City Models with CityGML[M]// 3D Geo-Information Sciences. 2008:15-31.

[18] Tian F, Wang H, Zhang F, et al. Co-CreativePen Toolkit: A Pen-based 3D Toolkit for Children Cooperatly Designing Virtual Environment[C]// International Conference on CSCW in Design, CSCWD 2006, May 3-5, 2006, Southeast University, Nanjing, China. 2006:360-364.

[19] Xiong H, Sun S. A Distributed Collaborative Product Customization System Based on Web3D[C]// International Conference on Computer Supported Cooperative Work in Design, CSCWD 2007, April 26-28, 2007, Melbourne, Australia. 2007:926-930.

[20] Wang L, Guo J, Yang C, et al. O3D-Based Game Learning Environments for Cultural Heritage Online Education[C]// Entertainment for Education, and, International Conference on E-Learning and Games. Springer-Verlag, 2010:417-428.

[21] Ieronutti L, Chittaro L. Employing virtual humans for education and training in X3D/VRML worlds[J]. Computers & Education, 2007, 49(1):93-109.

[22] Begole J, Rosson M B, Shaffer C A. Flexible collaboration transparency: supporting worker independence in replicated application-sharing systems[J]. ACM Transactions on Computer-Human Interaction, 1999, 6(2):95-132.

[23] Chu C H, Chan Y H, Wu P H. 3D streaming based on multi-LOD models for networked collaborative design[J]. Computers in Industry, 2008, 59(9):863-872.

[24] Chu C H, Wu P H, Hsu Y C. Multi-agent collaborative 3D design with geometric model at different levels of detail[J]. Robotics and Computer-Integrated Manufacturing, 2009, 25(2):334-347.

[25] 汪涵. 面向协同的3D场景概念建模研究[D]. 上海交通大学, 2014.

[26] 王珍海. 基于Web的协同3D场景建模系统的研究[D]. 上海交通大学, 2015.

[27] 余飞. 基于HTML5的图形图像协同处理技术研究与实现[D]. 长江大学, 2015.

[28]Baloian N, Gutierrez F, Zurita G, An Architecture for Developing Distributed Collaborative Applications using HTML5. Computer Supported Cooperative Work in Design (CSCWD), 2013 IEEE 17th International Conference on. IEEE, 2013:213 - 220.

[29]Katayama S Y, Goda T, Shiramatsu S, et al, A Fast Synchronization Mechanism for Collaborative Web Applications Based on HTML5. Proceedings of the 2013 14th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing. IEEE Computer Society, 2013:663 - 668.

[30]周湘吉. 基于三维CAD的网络协同设计系统的研究与应用[D]. 湖南大学, 2014.

[31]周峰, 王兆其, 刘金刚. 多人实时协同设计在虚拟家装系统中的研究与实现[J]. 微计算机信息, 2011, 27(1):271-273.

[32]李威龙, 徐奕奕. 基于WEB 3D技术的服装协同设计平台研究[C]. 广西计算机学会2012年学术年会论文集. 2012.

[33]潘丹丹, 孙文磊, 赵群. 基于Web的SolidWorks网络协同设计系统的设计与实现[J]. 机床与液压, 2013, 41(1):107-110.

[34]张博, 徐浩, 贾金原. 基于透明适配法的虚拟家居在线协同设计系统[J]. 系统仿真学报, 2013, 25(7):62-65.

[35]Xiong H, Sun S. A distributed collaborative product customization system based on Web3D[C]. Proceedings of the 11th International Conference on Computer Supported Cooperative Work in Design, (CSCWD 2007). IEEE, 2007: 926-930.

[36] Chen M, Cai W, Lu S, et al. Web 3D Model Library System Based on Cloud Platform[M]// Proceedings of the 2012 International Conference on Information Technology and Software Engineering. Springer Berlin Heidelberg, 2013:353-360.

[37] Song Y, Wei W, Deng L, et al. 3D-colladesign: A real-time collaborative system for web 3D design[C]// IEEE, International Conference on Computer Supported Cooperative Work in Design. IEEE, 2015.

[38] C Marion, J Jomier. Real-time Collaborative Scientific WebGL Visualization with WebSocket[J]. 2012.

[39] Nyamsuren P, Lee S H, Hwang H T, et al. A Web-based collaborative framework for facilitating decision making on a 3D design developing process[J]. Journal of Computational Design & Engineering, 2015, 31.

[40] Cheng Y, He F, Cai X, et al. A group Undo/Redo method in 3D collaborative modeling systems with performance evaluation[J]. Journal of Network & Computer Applications, 2013, 36(6):1512–1522.

[41]李晓梅. 实时协同设计技术的研究[D]. 燕山大学, 2010.

[42]莫剑冬. 基于Web的三维协同设计模型及其关键技术[J]. 计算机辅助工程, 2013, 22(3):53-56.

[43]徐立云, 叶盛, 许雄飞,等. 支持Web协同的轻量化三维设计框架的研究[J]. 中国工程机械学报, 2010, 08(4):404-408.

[44]Nyamsuren P, Lee S H, Hwang H T, et al. A web-based collaborative framework for facilitating decision making on a 3D design developing process[J]. Journal of Computational Design & Engineering, 2015, 31(3):148–156.

[45] Yong J H. A framework for seamless computer supported cooperative work in three-dimensional design[C]// International Conference on Computer Supported Cooperative Work in Design. 2005:638-642 Vol. 1.

攻读硕士学位期间取得的学术成果

学术论文：

[1]. Pengyu Du, et al. "A Real-Time Collaborative Framework for 3D Design based on HTML5." IEEE 20th International Conference on Computer Supported Cooperative Work in Design, 2016.

[2]. Wentao Wei, You Song, Li Deng, Pengyu Du, et al. "3D-CollaDesign：A Real-time Collaborative System for 3D Design based on Web." IEEE 19th International Conference on Computer Supported Cooperative Work in Design, 2015.

[3]. Ye Zhang, You Song, Li Deng, Pengyu Du, et al. " A cooperative design model based on vertex-colored graph." IEEE 19th International Conference on Computer Supported Cooperative Work in Design, 2015.

专利发明：

[1]. 宋友, 杜鹏宇等. 一种基于颜色、计算机视觉和网格划分的人机交互方法, 专利号2013104628072. CN 103529939 B. 2016

致谢

经过两年多研究生阶段的学习，值此论文付梓之际，特向指导和帮助过我的老师、 同学、朋友及关心支持我的家人表示感谢。

首先感谢我的导师宋友教授，感谢宋老师在我攻读学位期间一直给予悉心的指导和照顾。因为他我才能接触到这个课题，从而完全踏入了一个新的领域。从论文的选题到具体问题的分析研究，以及毕业设计论文的写作，宋老师都给我提出了很多宝贵的指导意见。

其次感谢实验室的魏文涛、王玥、聂殿辉等同学们，以及杨思宇、张沛、朱弘军、马欣欣等学弟妹们，他们在课题研究过程中给了我很多支持和帮助，并参与了全部的测试，他们的建议也激发了我很多的灵感。

同时，感谢我的室友们，一路走来有他们相伴，毕设期间的工作也得到了他们的支持。感谢他们陪我走过了研究过程中最艰难的时光。

感谢软件学院的各位老师帮助我掌握了坚实的专业知识。

感谢一直关心我、支持我的父母，正是他们深沉的爱使我积极向上，乐观进取，认真的学习，快乐的生活。感觉我的女朋友给予我真挚的爱与长久的陪伴。感谢每一位关爱我和帮助过我的人。

最后，谨向百忙之中抽出宝贵的时间评审本论文的各位专家致以诚挚的谢意。