# 基于WebGL的虚拟现实眼镜的实现

## #Realization of Virtual Reality Glasses based on WebGL

## 摘要

1981年，还是数学家的费诺·文奇（Venor Vinge）在他的传世名作《真名实姓》中描述了一个虚拟世界，黑客们以巫师的名义行走。这个世界外的世界就是VR（虚拟现实，Virtual Reality），由计算机模拟的，沉浸式全景多媒体，复写创造感官体验，包括虚拟的视觉、味觉、触觉、声音等等。 自然世界大致是连续的，计算机模拟的数据是离散的，经过多年来的努力，来自美国的Oculus VR公司创造出了Oculus Rift头盔显示器，它将图像分别射入左右眼，并通过头部动作追踪使视角随用户的操作改变，让人如入实景。 Oculus Rift被众人捧之为民用虚拟现实领域里程碑之作，然而成本并不低廉，以至于它被发布了两年仍未上市。在谷歌法国巴黎部门的两位工程师大卫·科兹(David Coz)和达米安·亨利(Damien Henry)利用了谷歌著名的“20%时间”规定，花了6个月的时间打造出来了CardBoard创意，意在将智能手机变成一个虚拟现实的原型设备，这个看起来非常寒碜的再生纸板盒却成了2014年谷歌I/O大会上最令人惊喜的产品，现在你可以用几十元人民币就可以在淘宝购买到以cardboard为原型的工程塑料制品。 本文提出的基于WebGL的虚拟现实的实现将CardBoard的成本再一次降低，使得只要使用了支持WebGL浏览器的智能设备都可以体验。本产品实现了全景场景展示，利用输入设备（如重力感应、鼠标等）实现动作追踪功能，并利用光线投射实现用户对物品的选定和场景切换。最后本产品基于JavaScript的原型机制对外开放出API，并利用该API制作了三个简单的应用，分别是全景图展示、电影选座以及简单的增强现实实现。 ***关键词:*** 虚拟现实;WebGL;JavaScript

## Abstract

In 1981, as a Mathematician Venor Vinge described a virtual world in his famous book True Names, where hackers can walk around as wizards. This world out of the true world is VR, Virtual Reality. A computer simulated muti-media device which shows an immersion panoramic view and duplicates your sensory experience, virtual visions, tastes, tactile sensations, sounds etc. The natual world is roughly continuous, and computer simulations of the data are discrete.After years of hard work, Oculus VR, a company from the United States, created a helmet monitor named Oculus Rift. The device projects two different images to the eyes, and uses head motion tracking to change the visual angle following the user's operation to make a visual world. Oculus Rift was held as a milestone for the field of civil vitual reality, yet the cost is not cheap, that it was released for two years and not marketed.David Coz and Damien Henry are two engineers works in the Google Paris Department. By the provision of Google's famous "20% time", they spent 6 months to built out the idea of CardBoard , which was intended to use samrtphone as a vitual reality prototype. This looks very shabby recycled cardboard box is the most amazing product on 2014 Google I/O Conference.And now you can buy a plastic one in Taobao with a few dozens of yuan. The virtual reality based on WebGL proposed in this paper will reduce the cost of CardBoard again.It lets everyone who holds a device with a webGL supported web browser experience VR. The product realizes a panoramic scene play, using input devices, such as gravity sensor, a mouse etc. to achieve motion tracking function, and uses ray casting for user picking items and changing scenes. Finally, using JavaScript's prototype mechanism, the product exposes API for APPs. In this paper, three simple application are made. Panorama, a gallery shows panoramas. Movie, an application to choose the best seat in a cinema and NoodleFace, a simple realization of augmented reality. ***Keywords:*** Vitual Reality;WebGL;JavaScript

## 绪论

### 课题研究的背景及其意义

虚拟现实(Virtual Reality，简称VR，又译作灵境、幻真)是近年来出现的高新技术，也称灵境技术或人工环境。虚拟现实是利用电脑模拟产生一个三维空间的虚拟世界，提供使用者关于视觉、听觉、触觉等感官的模拟，让使用者如同身历其境一般，可以及时、没有限制地观察三度空间内的事物。 北京时间2015年1月22日凌晨微软(Microsoft)公司发布了旗下的第一款全息眼镜HoloLens如图1.1,当然这不是世界上的第一款以虚拟现实为技术的设备，遗憾的是这些设备很笨重，长时间佩戴会给用户带来眩晕的感觉，而且最重要的，他们都价格不菲，这些问题直接导致虚拟现实这一技术仅能停滞于概念上。



虚拟现实技术演变发展史大体上可以分为四个阶段，包括有声音、形状、动作的模拟是蕴含虚拟现实思想的第一阶段（1963年以前）、虚拟现实萌芽为第二阶段（1963年～1972年）、虚拟现实概念的产生和理论初步形成为第四阶段（1990~2004年）。随着虚拟现实的逐步完善，该技术也逐渐从科幻片走进人们的生活，如图1.2是谷歌全球趋势对虚拟现实技术从2014年1月至2016年2月的统计及预测。



WebGL是一种3D绘图标准，这种标准允许把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3D加速渲染，这样Web开发人员就可以借助系统显卡来在浏览器里流畅的展示3D场景了，还能创建复杂的导航和数据视觉。WebGL技术标准免去了开发网页专用渲染插件的麻烦，可用于具有复杂3D结构的网站页面，甚至可以用来设计3D网页游戏等等。

虚拟现实因为能给人带来沉浸似的感受所以能很好的利用于医疗、教育、娱乐、军事航天、室内设计、房产开发、工业仿真、应急推演等等方面，WebGL作为浏览器一级的3D渲染标准有着轻量、易移植的特点，若能很好的接合这两项技术，会为移动互联网行业带来一次强大的技术革新。

### 国内外发展现状

#### 国内发展状况

我国虚拟现实现状和一些发达国家相比还有比较大的距离，随着计算机图形学、计算机系统工程等技术的高速发展，虚拟现实技术已经的到了相当的重视，引起我国各界人士的兴趣和关注，国内许多研究机构与高校也都在进行虚拟现实领域进行了研究并取得了一些不错的研究成果。

北京航空航天大学计算机系是我国最早进行虚拟现实的研究的单位之一，其虚拟现实与可视化新技术研究室继承了分布式的虚拟环境，可以实现三维动态数据库、虚拟现实演示环境用于飞行员训练的虚拟现实系统、虚拟现实应用系统的开发平台等，在虚拟现实环境中物体的物理特性的表示与处理上取得了相当可观的进展。

故宫文化资产数字化应用研究所出的“故宫VR”以太和殿为中心，构建了故宫内建筑群体的现状，利用超高清晰的虚拟现实技术自由地、多角度地表现传统文化、展示传统文化的魅力。

#### 国外发展状况

美国是虚拟现实技术研究的发源地，其在该领域的贡献可追溯到上世纪60年代。最初的研究主要集中于美国军方对飞行驾驶员与宇航员的模拟训练。随着冷战结束，这些技术逐步转为民用。施乐公司研究中心在虚拟现实领域主要从事建立未来办公室研究。波音公司的波音777运输机采用全无纸化设计，利用虚拟现实实现模板的设计并将其显示在加工件上，工人利用此模板简化了加工过程。2014年在美国公告牌音乐颁奖礼上，迈克尔·杰克逊的遗产委员会斥巨资使天王“活生生”地站在了舞台之上，引起观众惊叫连连。

在欧洲，英国在分布式并行处理、辅助设备设计和应用方面相比领先。英国ARRL公司关于远地呈现的研究实验，主要包括虚拟现实重构问题。他们的产品还包括建筑和科学可视化计算。其他的国家如荷兰、德国、瑞典也积极进行了虚拟现实的研究和应用。

日本在虚拟现实技术的发展在世界相关领域的研究中同样有着举足轻重的地位，它在建立大规模虚拟现实知识库和虚拟现实游戏方面作出了很大成就。在东京技术学院精密与智能实验室研究了一个用于建立三维模型的人性化界面，称为SpmAR NEC公司开发了一种虚拟现实系统，用代用手来处理CAD中的三维形体模型。通过数据手套把对模型的处理与操作者的手联系起来。

### 本文的研究内容和结构安排

本文主要利用双目视觉实现虚拟现实界面，经过阅读大量国内外文献，结合图形学知识，利用WebGL实现了以CardBoard为原型的虚拟现实眼镜的设计。

本文的主要分为六章，具体安排如下：

1. 第一章为绪论，主要阐述本文研究的意义和背景，并根据需要了解了一些国内外发展的状况。
2. 第二章介绍本产品开发时的开发环境、运用到的相关技术的简介。
3. 第三章详细介绍本产品所使用的算法，以及实现。
4. 第四章系统分析，对系统的功能进行系统分析，从经济、技术、使用三个方面分析可行性。
5. 第五章系统整体设计，对系统的所有功能进行详细的整体设计，包含功能设计和接口设计。并对各个模块的具体细节进行设计。
6. 第六章系统实现，主要讲解了本系统功能的具体实现和界面部分。
7. 第七章系统测试，介绍了整个系统的测试方法以及最终测试结果，包括单元测试、压力测试和UI测试。
8. 第八章是结论，是全文的小结和对后期的工作的展望。

## 相关技术及工具简介

本章对产品的开发环境、使用的编程语言、相关技术以及开发模式的介绍。 ###开发环境简介 ####Yeoman简介 Yeoman是一个强健的工具、库和工作流程的组合，帮助开发人员快速创建出漂亮且引人入胜的网页程序，其功能如下：

* 快速创建骨架应用程序,使用可自定义的模板、AMD以及其它工具轻松地创建新项目的骨架。
* 自动完善脚本，所有脚本都会自动针对jshint运行，从而确保它们遵循语言的最佳实践。
* 内建预览服务器，开发者不需要启动自己的HTTP服务器，内建的服务器一条命令就可以启动。
* 非常棒的图像优化，Yeoman使用OptPNG和JPEGTran对所有图像进行优化从而减少用户在下载资源时使用的时间。
* 可用PhantomJS单元测试并自动为开发者提供测试内容骨架。

#### Bower简介

Bower 是一个针对Web开发的包管理器。该工具主要用来帮助用户轻松安装CSS、JavaScript、图像等相关包，并管理这些包之间的依赖。其特点如下：

* 它会为你节省寻找客户端的依赖关系的时间。
* 脱机工作，Bower会在用户主目录下创建一个.bower的文件夹，这个文件夹会下载所有的资源、并安装一个软件包使它们可以离线使用。
* 通过bower.json可以很容易地展现客户端的依赖关系。
* 让升级变得简单。假设某个库的新版本发布了一个重要的安全修补程序，为了安装新版本，只需要运行一个命令，bower会自动更新所有有关新版本的依赖关系。

#### Gulp.js简介

Gulp.js是一种基于流，代码优先于配置的新一代构建工具，其特性如下：

* 使用方便，可以简化任务。
* 构建快速，采用流式操作，可以减少IO操作，更快地构建项目。
* 插件高质，Gulp.js有严格的插件指导策略，确保插件完成简单高质的工作。
* 易于学习，Gulp.js的API较少，掌握起来不费吹灰之力。

#### Git简介

Git是一个开源的分布式控制系统，用以有效、高速的处理从很小到非常大的项目版本管理。它起初是Linus Torvalds为了帮助管理Linux内核开发而开发的一个开放源码的版本控制软件。相比于其它版本管理，Git的优势在于：

* 分支更快、更容易。
* 支持离线工作，可在本地开发稍后提交至服务器。
* Git的提交是原子的，且为整个项目范围。
* Git的每一个工作树都包含一个具有完整项目历史的仓库。

### 编程语言的介绍

#### JavaScript简介

JavaScript一种直译式脚本语言，是一种动态类型、弱类型、基于原型的语言，内置支持类型。它的解释器被称为JavaScript引擎，为浏览器的一部分，广泛用于客户端的脚本语言，最早是在HTML（标准通用标记语言下的一个应用）网页上使用，用来给HTML网页增加动态功能。 在1995年时，由Netscape公司的Brendan Eich，在网景导航者浏览器上首次设计实现而成。因为Netscape与Sun合作，Netscape管理层希望它外观看起来像Java，因此取名为JavaScript。但实际上它的语法风格与Self及Scheme较为接近。 为了取得技术优势，微软推出了JScript，CEnvi推出ScriptEase，与JavaScript同样可在浏览器上运行。为了统一规格，因为JavaScript兼容于ECMA标准，因此也称为ECMAScript。

### 相关技术介绍

#### WebGL和THREE.js简介

WebGL是一项利用JavaScript API呈现3D电脑图形的技术，有别于过往需要加装浏览器插件，通过WebGL技术，只要编写网页代码就可以实现3D图像的展示。WebGL基于OpenGL ES 2.0，使用HTML5 Canvas，并允许利用部分文档对象模型接口，可利用部分JavaScript实现自动存储器管理。WebGL标准尚在发展中，由非盈利的Khronos Group管理。

THREE.js是用JavaScript编写的WebGL第三方库提供了很多的三维显示功能，是一款运行于浏览器的3D引擎，利用它可以创建各种三维场景，包括摄像机、灯光、材质等各种对象，其代码托管于github上。

#### jQuery简介

JQuery是继prototype之后又一个优秀的Javascript库。它是轻量级的js库，兼容CSS3，还兼容各种浏览器，jQuery2.0及后续版本将不再支持IE6/7/8浏览器。jQuery使用户能更方便地处理HTML（标准通用标记语言下的一个应用）、events、实现动画效果，并且方便地为网站提供AJAX交互。jQuery还有一个比较大的优势是，它的文档说明很全，而且各种应用也说得很详细，同时还有许多成熟的插件可供选择。jQuery能够使用户的html页面保持代码和html内容分离，也就是说，不用再在html里面插入一堆js来调用命令了，只需要定义id即可。

#### WebRTC简介

WebRTC（Web Real-Time Communication）实现了基于网页的视频会议，标准是WHATWG 协议，目的是通过浏览器提供简单的javascript就可以达到实时通讯（Real-Time Communications (RTC)）能力。WebRTC项目的最终目的主要是让Web开发者能够基于浏览器（Chrome...）轻易快捷开发出丰富的实时多媒体应用，而无需下载安装任何插件，Web开发者也无需关注多媒体的数字信号处理过程，只需编写简单的Javascript程序即可实现，W3C等组织正在制定Javascript 标准API，目前是WebRTC 1.0版本，Draft状态；另外WebRTC还希望能够建立一个多互联网浏览器间健壮的实时通信的平台，形成开发者与浏览器厂商良好的生态环境。同时，Google也希望和致力于让WebRTC的技术成为HTML5标准之一。本文第三个示例应用（NoodleFace）使用了此技术。

### 服务端环境介绍

#### Raspberry Pi 2

Raspberry Pi是一款针对电脑业余爱好者、教师、小学生以及小型企业等用户的迷你电脑，预装Linux系统，体积仅信用卡大小，搭载ARM架构处理器，运算性能和智能手机相仿。目前为止，通过开发者社区的共同努力，Raspberry Pi 电脑已经可以运行 WebKit、LibreOffice、Scratch、Pixman、XBMC/Kodi、libav、PyPy、Raspbian、Ubuntu 等众多开源系统和程序。

Raspberry Pi 2 的基本配置如下：

* 900MHz 主频四核 ARM Cortex-A7 处理器（博通 BCM2836）
* 两块 1G LPDDR2 SDRAM （Micron）
* 两个全尺寸 USB 接口（可用于供电）
* 一个 microUSB 接口
* 一个全尺寸 HDMI 接口
* 3.5 毫米音频接口
* RJ45 以太网线接口

#### Koa

Koa是基于Node.js平台的下一代web开发框架它由 Express 原班人马打造的，致力于成为一个更小、更富有表现力、更健壮的 Web 框架。使用 koa 编写 web 应用，通过组合不同的 generator，可以免除重复繁琐的回调函数嵌套，并极大地提升错误处理的效率。koa 不在内核方法中绑定任何中间件，它仅仅提供了一个轻量优雅的函数库，使得编写 Web 应用变得得心应手。

### 开发模式介绍

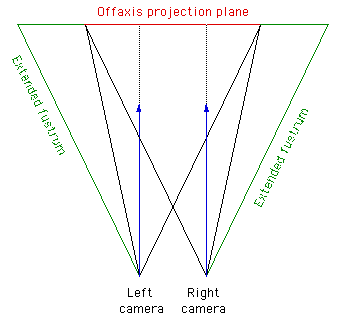
本文使用增量模型，在增量模型中，软件被作为一系列的增量构件来设计、实现、集成和测试，每一个构件是由多种相互作用的模块所形成的提供特定功能的代码片段构成. 增量模型在各个阶段并不交付一个可运行的完整产品，而是交付满足客户需求的一个子集的可运行产品。整个产品被分解成若干个构件，开发人员逐个构件地交付产品，这样做的好处是软件开发可以较好地适应变化，客户可以不断地看到所开发的软件，从而降低开发风险。

## 相关的算法和实现

### 双目立体图像对的生成算法

有些3D引擎提供了必要的工具来创建正确的立体图像，其他的引擎需要适当的几何变换来生成立体图像对，更多渲染包只提供简单的透视投影。本文使用的算法适用于所有能够实现透视投影的引擎。

创建立体图像对的选择涉及到两眼间距隔开的两个位置同时绘制左右两眼的视图。眼睛沿着平行的两个向量，从投射到每个眼睛的光束形成的视锥是不对称的。本方案通过扩展视锥使得每个眼睛的视锥对称并在渲染图像之后将扩展的地方裁剪出来，如图2所示。



离轴平行立体成像原理

剩下的就是计算只需要给出合适的眼间距和焦距，渲染图往往比原图更大一些，这样裁剪之后才能和原图一样大偏移量的大小取决于给定的焦距大小需要裁剪掉的大小如公式1所示。

e w

delta =

2 fo tan(a / 2)

w是图像宽度，fo是焦距，e是眼间距，a是预测光圈大小，为了得到准确的光圈大小，还要在图像裁切之后修改为实际值如公式2。

(w + delta) tan(a/2)

a' = 2 atan

w

### 用户坐标变换算法

智能手机的方向传感器作为本产品的输入设备，将传感器获取的离散信息处理成方向信息并利用此信息计算对应的用户方向来生成虚拟图像是非常重要的，接下来先介绍以一下智能手机的传感器。

地球坐标系使用的是“东、北、上”三个方向作为坐标系，而对于智能手机也利用的此标准，x轴是当屏幕面对用户由屏幕中央指向右手边的方向，y轴由屏幕中央指向上方，而z轴则垂直屏幕指向用户，如图3所示。

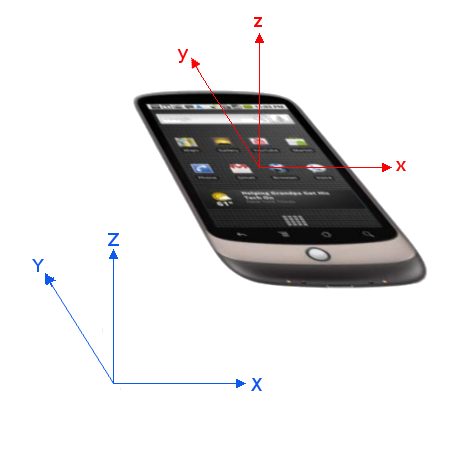
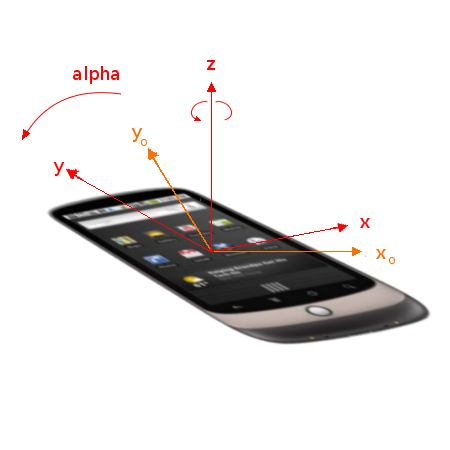
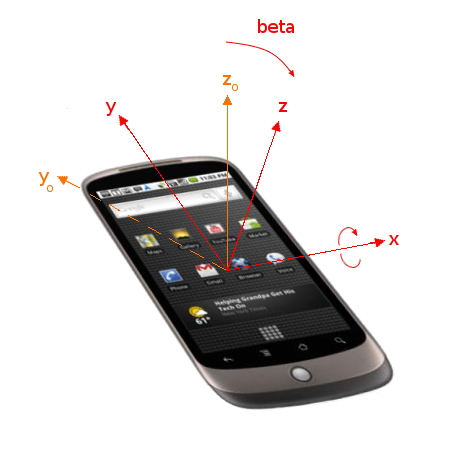
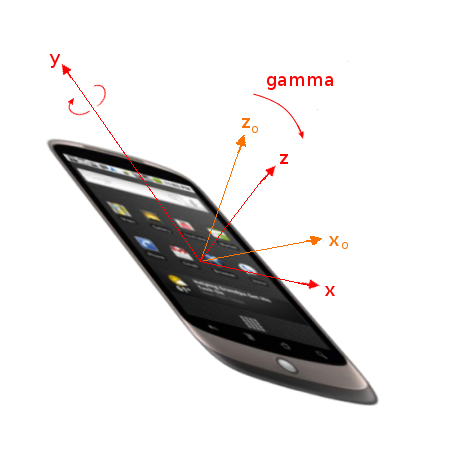


图3 智能手机传感器的坐标系

旋转设备，传感器会向浏览器传回4个值，alpha为设备绕z轴旋转的偏移量，如图4，大小为0至360；beta是设备绕x轴旋转的偏移量，如图5，取值区间为-180180；gamma则是绕y轴旋转的角度，如图6，区间为-9090；absolute在传感器运行时始终为false。

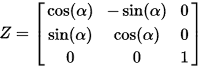
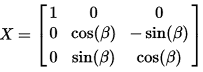
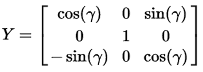
  

在佩戴CardBoard的时候，用户面对着手机，手机横放。根据智能手机坐标系，用户观看的向量为v，其值如公式3

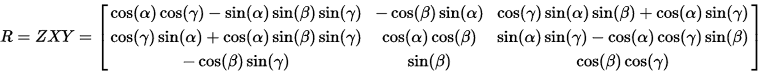


公式3

相应的旋转z、x、y轴获得的变换矩阵如公式4、公式5、公式6。

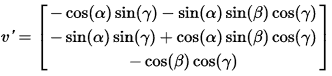
  

可得R为全部的变换公式，如公式7。



公式7

经过变换获得的v'如公式8.



公式8

由此可得用户的偏移角度，如公式9.



公式9

## 系统分析

本章将详细介绍基于WebGL的虚拟现实眼镜的详细设计，由于本产品采用增量开发模式，接下来每一小节将分别介绍系统核心、全景图浏览应用、电影选座应用和NoodleFace应用的详细设计。 ###需求分析 系统核心的作用是实现双目立体图像的算法以及用户坐标变换算法，同时创造对外接口，实现其它模块接入；Panorama利用核心接口浏览全景图；Movie利用核心接口和网络接口创建电影院，并能够通过改变相机坐标渲染出用户选座之后的位置；NoodleFace利用摄像头实现增强现实效果。从功能的角度上对系统进行分化，分析结果如表1。

| 模块| 功能 | 功能点 | 功能描述 | |-----|------|-------|---------| | 核心 | 总控制器 | 全局相机 | 系统所有应用的显示 | | | | 全局场景 | 系统所有应用容器 | | | | 全局控制器 | 针对有方向感易器的设备使用方向感应器，其它设备使用鼠标 | | | | 全屏控制 | 用户点击窗口全屏 | | | | 自适应功能 | 窗口发生改变时自动重绘画面 | | | 总容器 | 可选择对象容器 | 存放选择和点击的对象 | | | | 代替鼠标功能 | 类似鼠标，当用户选择某一对象进行反馈 | | | 主界面 | 应用接口 | 存在新应用时显示图标 | | | | 穹顶 | 创造一个可视的背景 | | | 总路由 | 全局路由 | 负责应用跳转 | | | 其它 | 全局灯光 | 全局的照明 | | | | 立方体 | 生成一个立方体 | | | | 平面 | 生成一个平面 | | | | 热点 | 生成一个可选择热点 | | | | 图片 | 生成一张显示图片的平面 | | | | 文字 | 生成一行字 | | Panorama | 应用界面 | 下一张 | 切换下一张图 | | | | 上一张 | 切换上一张图 | | | | 回主界面 | 回到主界面 | | | 路由 | 资源路由器 | 显示资源URL地址 | | Movie | 应用界面 | 座位 | 显示座位号 | | | | 屏幕 | 显示屏幕位置 | | | 爬虫 | 抓取座位信息 | 抓取影院座位信息 | | NoodleFace | 应用界面 | 左眼 | 左眼显示内容 | | | | 右眼 | 右眼显示内容 |

### 可行性分析

#### 技术及开发的可行性

本产品使用语言为JavaScript相较于其它语言学习成本较低运行起来也很轻量，完全可以同时胜任前后端开发工作。基于CardBoard的web应用虽然不多但是移动应用的数量是可观的，可以从参考一些安卓应用进行开发。THREE.js经过一年的发展已经较为稳定，相关的文档也陆续完善。WebGL也成为现代浏览器的标配，确保本产品的兼容性。

#### 经济可行性

产品后端为node.js服务器，可以兼容于任何已知的硬件设备，前端仅需要现代浏览器，成本可谓十分低廉。

### 系统核心的业务流程分析

用户打开浏览器，系统将总容器添加到全局场景中，总路由通过查看URL锚点，通过锚点确定在总容器中添加的是哪一个应用，若锚点为空，或不存在该应用接口则将主界面添加到场景。在主界面用户选择任一图标，系统将主界面移出总容器并将对应应用界面移入容器，同时更新锚点。整体流程如图？

### 编码规范

作为脚本语言，JavaScript的弊端是在没有良好的编码规范下，代码会比较难懂，造成调试困难甚至变量重名，因此在此规定编码规范。

* 每行必须添加分号。
* 字符串除JSON格式之外使用单引号。
* 在同一行使用左大括号。
* 常量名全部使用大写，类名首字母大写后面的每一词首字母大写（大驼峰式），对象名首字母小写后面每一词首字母大写（小驼峰式）。
* 代码块需要用匿名函数包裹以避免污染全局作用域。
* 代码块前面添加严格模式声明。
* Tab缩进为4个空格。

## 系统设计

本章介绍系统的详细设计，包括各模块的类以及类图、接口设计和路由设计。 ###功能设计 ####系统功能模块图 本产品的模块设计较为简便，用户通过本产品选择浏览全景图、电影选座和增强现实功能。功能框图如图？所示。

系统核心通过主路由器判断将哪一个应用放入容器，系统核心模块的相关类的类图如图？所示。

#### 主界面设计

主界面有应用列表，用户通过选择应用切换场景。相关类的类图如图？所示。

#### 全景图浏览设计

Panorama实现全景图浏览，用户选择前进后退切换图片，同时通知路由更换地址。相关类的类图如图？所示。

#### 电影选座应用设计

Movie实现电影选座，用户选择座位，改变相机位置，如图？所示。

#### 增强现实应用设计

NoodleFace实现增强现实效果，如图？所示。

### 接口设计

系统通过读script目录中的manifest.js文件获取应用信息，关于manifest.js文件中各字段的设计如表？所示

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 类型 | 说明 | 示例 |
| title | string | 应用名称 | 'Panorama' |
| icon | string | 应用图标地址 | '../images/apps/panorama.jpg' |
| url | string | 路由地址 | 'panorama' |
| face | string | 应用界面类名 | 'PanoFace' |

Movie（电影选座应用）需要抓取豆瓣网的座位信息，为了防止网站更改，本文的爬虫先将信息缓存到根目录movie.json文件下，movie.json文件中各字段设计如表？所示

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 类型 | 说明 | 示例 |
| seats | array | 入口 |  |
| isSeat | boolean | 确定是座位还是过道 | true |
| seated | boolean | 是否已被选走 | true |
| num | string | 显示行数：列数 | 7:15 |

#### 路由设计

系统由前端路由器控制，前端路由器通过捕捉浏览器hashchange事件判断当前应用界面，具体的路由设计按照#!/app/应用名/资源名设计，如图？

## 系统实现

本章主要介绍系统各主要功能模块的界面设计，以及具体功能模块的设计和实现的过程。 ###主界面 当URL不存在锚点时，系统跳转到主界面，用户通过主界面选择图标进入不同应用。如图？。

### 全景图浏览界面

当URL为#!/apps/panorama/时，会渲染全景图中的第一张图片，选择左右切换图片。如图？。

### 电影选座应用界面

当URL为#!/apps/movie/时，会进入一个虚拟的影院，选择任一座位，相机位置就会被移到座位上。如图？。

### 增强现实应用界面

当URL为#!/apps/noodleface/时，会访问设备相机，将左右眼的图像添加设备相机拍摄的图片。如图？。

#### 系统测试