第一章 绪论

1.1 课题背景与意义

随着VR/AR的发展，越来越多的人接触到虚拟现实技术和增强现实技术并且应用到各个领域，因此，我们思考将VR/AR技术应用到web平台上，现如今，webVR/webAR逐渐进入广大用户的视野，由于webVR/AR无需下载应用就能体验，能够快速的捕捉客户。众所周知，虽然许多VR/AR技术可以应用到多种平台，但是比如VR游戏，VR玩家和非VR玩家是基本不能实现共享游戏的，这就是VR玩家稀少比较局限的主要原因，所以webVR/AR的一个很关键的优势就是，具有能够跨越移动端、网站和VR/AR平台来达到众多用户体验的能力，这就使得webVR/AR可以让VR/AR中的用户与非VR/AR用户进行连接，使webVR/AR应用能够大范围的普及，进而影响更多的用户，得到更好的用户体验。同时，webVR/AR技术可以让内容不再受到平台、硬件的约束，再加上现阶段很多的浏览器，例如谷歌chrome、火狐firefox等这样的浏览器已经积累的大量的潜在用户，这也为webVR/AR浏览器市场奠定了坚实的用户基础。此外webVR/AR技术大大降低了开发者进入VR/AR领域的门槛，促使大量内容的涌进。

而几个月前W3C已经公开表示，正式将API更名为webXR，相关机构正在制定完整版规格，现在官方把它叫做”webXR Device API”。实际上VR和AR是两种不同的技术，不过二者也有很多重叠的地方：都是与设备有关，安装许多传感器，可以追踪动作、位置、方向。然而这两者最明显的不同就是AR是将模拟层叠加在现有的世界之上，而VR是创造全新的模拟现实，这样二者使用的技术重叠、开发者处理的例子就会碰到相同的挑战，于是W3C VR团队决定开发一个API能够同时处理这两个技术，它就是webXR。去年谷歌发布消息称，从今年一季度开始，webXR标准就会开始试用，这样开发者可以在稳定版的谷歌上使用实验框架，但是由于webXR还在发展初期，目前webXR还不太稳定，虽然可以进行简单的开发，但是还是会有很多的限制，因此仍然需要不断的改进和优化，webXR的出现，使得开发者瞄准所有设备就会更容易，而且它还支持真正的本地AR渲染技术。技术的发展是为了让开发者能够更加轻松更加便捷的开发，若是从前，如果想学习XR，就必须理解3D模型，研究游戏引擎（比如unity3D）还有编程技术，但是webXR能流行起来，很多问题都能得到缓解，这样开发者可以用现有的框架开发，不需要从零开始。

1.2 国内外研究现状

WebVR将提供一种“神奇的窗口”。虚拟现实之所以能够有如此大的吸引力，主要就是因为其能够给用户带来不同凡响的体验，虚拟现实开始涌现更多的“创造性”体验。

实际上，苹果safari是目前唯一一个还不能支持webVR/AR的浏览器，这意味着为了能够让webVR/AR的内容可以在safari上快速反应，开发者必须要自己添加能够模仿webVR API提供信息的polyfill规范，虽然polyfill可能仍然不能适用于一些浏览器（例如安卓版Edge），但是大多数情况下还是能给到用户不错的体验。当你的浏览器接受到webVR内容的时候，你就可以与3D实现交互，这样内容就可以对你的手机的方位的移动作出反应，除此之外，在屏幕的右下角会有个“enter VR”按钮可以让你切换到VR模式，然后当你连接VR头显就可以实现完全的网页沉浸，WebVR的应用范围很广，这就是我们认为它是VR行业最强大分支之一的原因。而对于webAR来说，相信大家都有看过《奇异博士》中的任意门吧，现在使用webAR技术就能实现在现实中。随着iphonex的发售，AR逐渐走进我们的生活，而随着iphone和安卓手机对webAR的支持，webAR正逐步融入我们的生活中。

国内市场中，很多的信息技术公司都在开始研究webVR/AR技术，并不断的应用于不同的app中比如微信、微博等，同时也会在webVR/AR中陆续加入手势识别、人脸识别等特性。相信不久的将来，webVR/AR一定会在国内市场大放异彩。

1.3 研究目标与工作内容

目标：本课题主要讨论webVR/AR设计，并且做一个webAR的实验范例

内容：本课题是在web环境下开发VR/AR，主要是利用aframe框架、ar.js、artoolkit在webstorm中进行webVR/AR的开发，导入3D模型，调取摄像头，识别marker，并实现虚拟场景和现实场景的切换，最后实现交互。

1.4 本章小结

本章主要是对webVR/AR技术进行一个简单的介绍，包括课题研究的背景和意义、国内外研究现状、研究目标以及工作内容，对webVR/AR有个初步的认识,在此基础上，对webAR进行深入的研究。

第二章 技术支持原理

2.1 A-Frame框架

A-Frame可以从纯HTML文件开发而不需要安装任何东西。是一个用于构建虚拟现实体验的Web框架，使用HTML和实体组件制作WebVR/AR，适用于Vive，Rift，Daydream，GearVR，桌面，A-Frame是构建虚拟现实（VR）体验的Web框架。 A-Frame最初来自Mozilla，被开发成为开发VR内容的简单而强大的方式。 作为一个独立的开源项目，A-Frame已经发展成为最大和最受欢迎的VR社区之一。

A-Frame基于HTML的顶层，使其开始变得简单。 但A-Frame不仅仅是一个3D场景图或标记语言， 核心是一个强大的实体组件框架，为three.js提供了一个声明式的，可扩展的和可组合的结构。

A-Frame支持大多数VR耳机，如Vive，Rift，Windows混合现实，Daydream，GearVR，Cardboard，甚至可以用于增强现实。 尽管A-Frame支持整个频谱，但A-Frame的目标是定义完全身临其境的交互式VR体验，超越基本的360°内容，充分利用位置跟踪和控制器。只用使用一行 HTML，包括：canvas、场景、渲染器、渲染循环、摄像机以及 raycaster。然后，我们可以通过使用添加子元素的方式来为场景添加对象，无需构建。

2.1.1 特性

（1）虚拟现实变得简单

只需放入<script>标签和<a-scene>即可。 A-Frame将处理3D样板、VR设置和默认控件。 不用开发者安装，没有构建步骤。

（2）声明性HTML

HTML易于阅读，理解和复制粘贴。 基于HTML之上，A-Frame适用于所有人：网页开发者，VR爱好者，艺术家，设计师，教育者，制作者，孩子们。

（3）跨平台VR

为Vive，Rift，Windows Mixed Reality，Daydream，GearVR和Cardboard构建VR应用程序，并支持所有相应的控制器。即使没有耳机或控制器，A-Frame仍可在标准台式机和智能手机上使用。

（4）实体 - 组件体系结构

A-Frame是一个功能强大的three.js框架，提供了一个声明式的，可组合的，可重用的实体组件结构。 HTML只是冰山一角; 开发人员可以无限制地访问JavaScript，DOM API，three.js，WebVR和WebGL。

（5）性能

A-Frame在WebVR的基础上进行了优化。当A-Frame使用DOM时，其元素不会触及浏览器布局引擎。而在一次requestAnimationFrame调用下，3D对象会更新全部在内存中完成，开销很小。

（6）工具

由于Web是基于HTML的概念构建的，因此A-Frame与大多数库，框架和工具（包括React，Preact，Vue.js，d3.js，Ember.js，jQuery）兼容。

（7）Visual Inspector

A-Frame提供了一个方便的内置可视3D检查器。 打开任何A-Frame场景，按下<ctrl> + <alt> + i，然后四处飞行以在引擎盖下可以查看。

（8）组件

使用A-Frame的核心组件（如几何图形，材质，灯光，动画，模型，raycasters，阴影，位置音频，文本以及Vive / Touch / Windows Motion / Daydream / GearVR / Cardboard控件）运行。 甚至获得更多组件比如粒子系统，物理，多用户，海洋，山脉，语音识别，动作捕捉，远距传物，超级手和增强现实等。

2.2 ar.js

AR.js是一个可以在github上高效地完成网络上的增强现实的一个库。它有几个特点第一是非常快，即使在手机上也能有效运行，即使是几年前的手机上也可达到60 fps，第二是基于网络，这是一个纯粹的网络解决方案，因此不需要安装。并且是一个基于three.js + jsartoolkit5的完整JavaScript，第三是开源，它是完全开源的，免费的，第四是它适用于任何带有webgl和webrtc的手机，目的是让人们容易做增强现实（AR），可以在当今手机上轻松体验，并且可以使用Web技术轻松设计。 AR.js项目旨在帮助这些人。 所以现在，任何拥有现代手机的人都可以免费享受开放源代码的AR，并且无需安装。由此可见，不必等待每个人都购买支持AR的设备（例如Tango，Moverio等）用自己口袋里的手机就可以实现。AR.js基于标准，可在任何使用WebGL和WebRTC的手机上工作。 它适用于Android和Windows Mobile。 不幸的是，它目前不适用于iOS设备。 Safari不支持WebRTC，但Apple正在开发它。

2.3 threex-artoolkit

ARToolkit，是一个开源的AR(增强现实)SDK。threex.artookit是可轻松处理artoolkit的three.js扩展。threex.artoolkit由3个类组成：

THREEx.ArToolkitSource：这是要进行位置跟踪分析的图像，它可以是网络摄像机，视频甚至图像；

THREEx.ArToolkitContext：这是主引擎，它实际上会找到标记位置在图像源中；

THREEx.ArMarkerControls：这是控制标记的位置，它使用经典的three.js控件API，会确保将开发者的内容放置在标记上方。

2.3 webGL

WebGL（全写Web Graphics Library）是一种3D绘图协议，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，这样Web开发人员就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。WebGL是基于OpenGL ES的低级3D图形API的跨平台，免版税的Web标准，通过HTML5 Canvas元素暴露给ECMAScript。 熟悉OpenGL ES 2.0的开发人员将使用GLSL将WebGL识别为基于Shader的API，其构造在语义上与基础OpenGL ES API的构造相似。 它与OpenGL ES规范保持着非常接近的地位，并为开发人员所期望的诸如JavaScript之类的内存管理语言提供了一些优惠。 WebGL 1.0公开了OpenGL ES 2.0功能集; WebGL 2.0公开了OpenGL ES 3.0 API。

WebGL将无插件3D带入网络，直接在浏览器中实施。 主要浏览器厂商Apple（Safari），Google（Chrome），Microsoft（Edge）和Mozilla（Firefox）是WebGL工作组的成员。

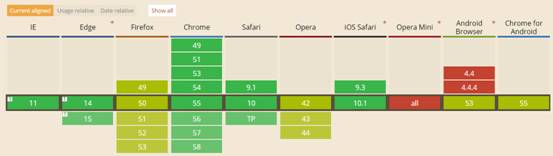


图2-3-1 webGL兼容性

2.4 webRTC

WebRTC，名称源自网页实时通信（Web Real-Time Communication）的缩写，是一个支持网页浏览器进行实时语音对话或视频对话的技术，WebRTC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。

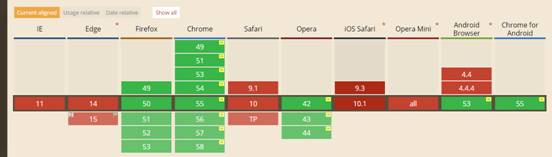


图2-4-1 webRTC兼容性

2.5 three.js

Three.js就是用javascript来写3D程序。three.js是JavaScript编写的WebGL第三方库。Three.js 是一款运行在浏览器中的 3D 引擎，你可以用它创建各种三维场景，包括了摄影机、光影、材质等各种对象。

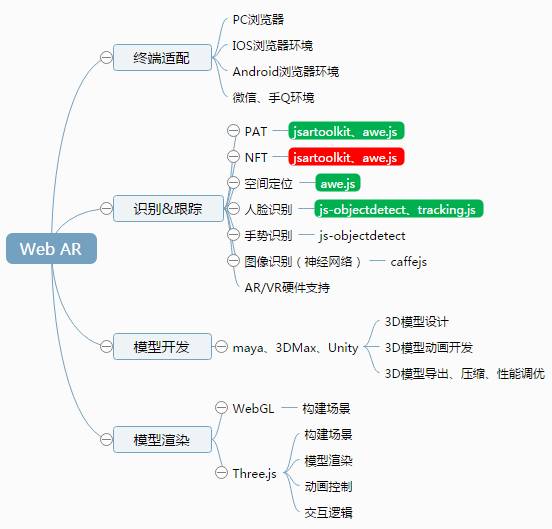


图2-5-1 three.js结构图

第三章 模型建立

3.1建模工具

由于本课题建模所需要的格式是obj、mtl以及mtl所能摄取到的纹理材质格式jpg、jpeg、png，OBJ文件是Wavefront公司为它的一套基于工作站的3D建模和动画软件"Advanced Visualizer"开发的一种文件格式，因此对模型的要求很高，从中选出了两个适合的工具进行对比。

3.1.1 3dsmax

3D Studio Max，常简称为3d Max [1]  或3ds MAX，是Discreet公司开发的（后被Autodesk公司合并）基于PC系统的三维动画渲染和制作软件。拥有性价比高，使用者多，便于交流，上手容易的优势。3dsmax导出obj方法：依次点击文件--导出--导出外部文件格式--保存类型为obj格式。

优点：性价比高；使用者多，便于交流；上手容易。

缺点：导出的模型材质，纹理等贴图容易丢失，同时会产生很多附加文件，不利于模型的载入。

3.1.2 artec扫描仪

这种新一代计量装置配备了高精度扫描自动温度稳定功能。Artec Space Spider是一款面向CAD用户和工程师的新型增强型精密仪器，是一款基于蓝光技术的高分辨率3D扫描仪。它非常适合以高分辨率捕捉小物体或大型工业物体的复杂细节，具有稳定的精确度和鲜艳的色彩。

扫描仪能够呈现复杂的几何形状，锋利的边缘和细小的肋骨，使我们的技术与众不同。它是一款理想的工业三维扫描仪，可用于高分辨率捕捉诸如注塑零件，印刷电路板，钥匙，硬币甚至人耳等物体，然后将最终3D模型导出到CAD软件中。

Space Spider在逆向工程，质量控制，产品设计和制造等领域提供几乎无限的可能性。

这款新一代工业三维扫描仪最初专为国际空间站的规格而开发，具有强大的温度稳定性和高等级的电子设备，不仅可以使扫描仪在三分钟内达到最高精度，而且可确保数据采集的长期可重复性，设备的准确率不受环境条件波动的影响。

在这款功能强大的三维扫描仪中，该技术经过精心研制，确保扫描质量达到最佳状态，并确保设备的稳定性。

优点：

（1）达到精度要求，携带方便，为解决中小型物体提供了良好的方案。

（2）分辨率高。

（3）扫描速度、质量、便携式令人震撼。

（4）使用简单。

（5）用时短。

（6）高分辨率、高精度，高测量工具。

缺点：价格昂贵。

3.2 建模过程

（1）在电脑上安装Artec Studio，并将Artec扫描仪与电脑连接（通过USB），等待驱动安装完毕，在计算机上运行安装好的Artec Studio，新建项目，并且点击左侧菜单栏的“扫描”进入扫描界面，根据玛尼石的形态复杂特征，选择“几何 + 纹理”选项。其他选项没有特殊要求默认设置。



图3-2-1 Artec Space Spider

（2）将被扫描的玛尼石正面朝上放置在准备好的旋转台，调整位置，将Artec扫描仪对准玛尼石与旋转台形成45°夹角。启动开始按钮，同时启动旋转台，观察扫描成像区，可以看到扫描的部分，以及左边的距离-信号图表，控制距离在绿色区间，扫描一周即可，扫描完毕后，关闭按钮，暂停扫描。

（3）将玛尼石背面朝上放置在旋转台上，重复（2）的操作，扫描完毕。

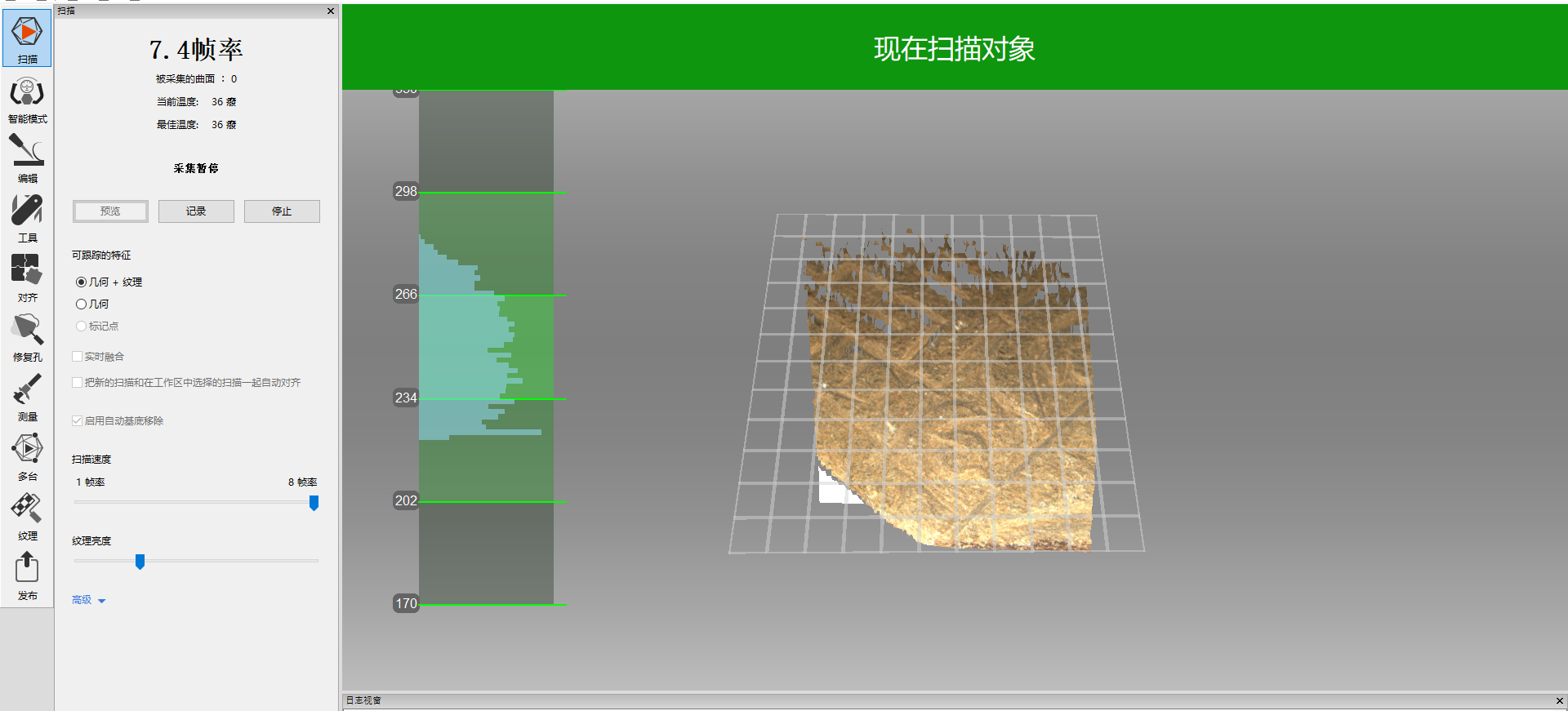


图3-2-2 Artec Studio扫描界面

（4）将获得的两组扫描信息进行后期处理。首先是对齐，选中右边的扫描数据，点击“对齐”，然后选中扫描数据，点击“自动对齐”。若“自动对齐”不能完全将不同角度的扫描信息对齐，则可以点击其下面的“对齐”按钮，手动微调至对齐即可。

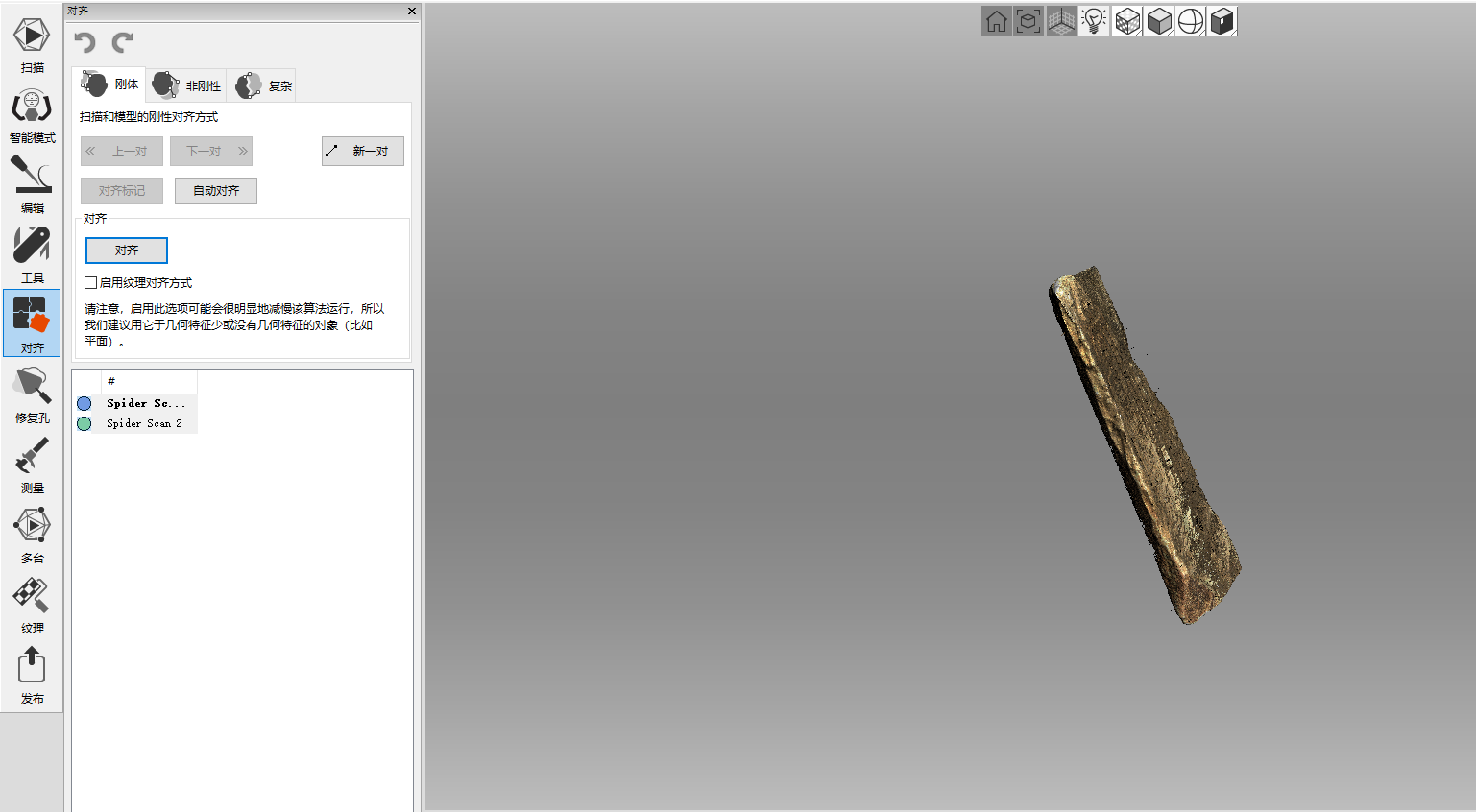


图3-2-3 Artec Studio对齐界面

（5）接着对模型进行修整。点击“工具”，进入工具模块，依次选择“整体配准”、“删除离群噪点”、“光顺融合”选项。

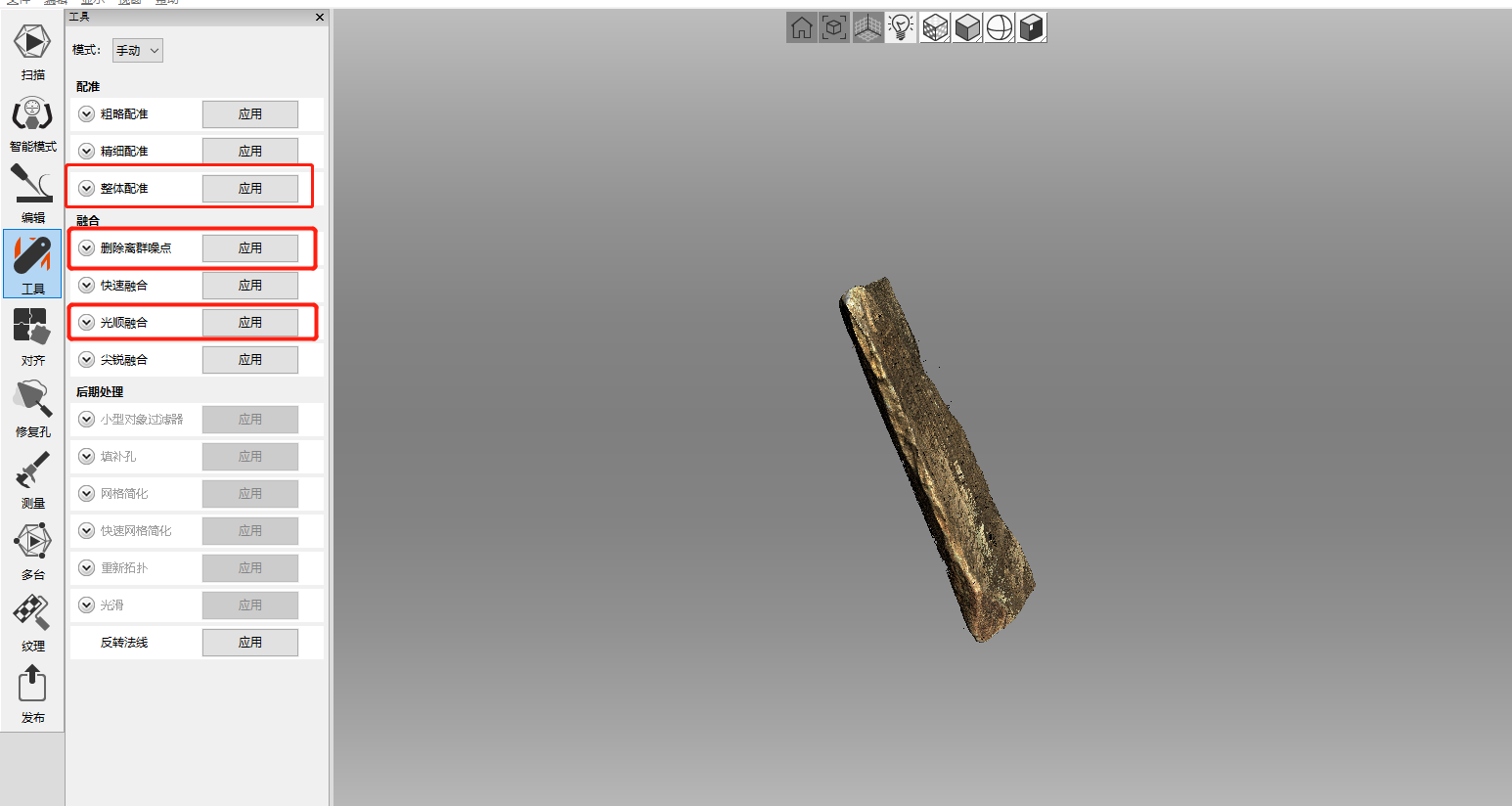


图3-2-4 Artec Studio工具界面

（6）导出纹理和材质。选择贴图“导出”选项，下面则勾选“启用纹理正常化”和“修复缺少的纹理”，根据具体实际情况来选择分辨率，这里选择的是4096\*4096。

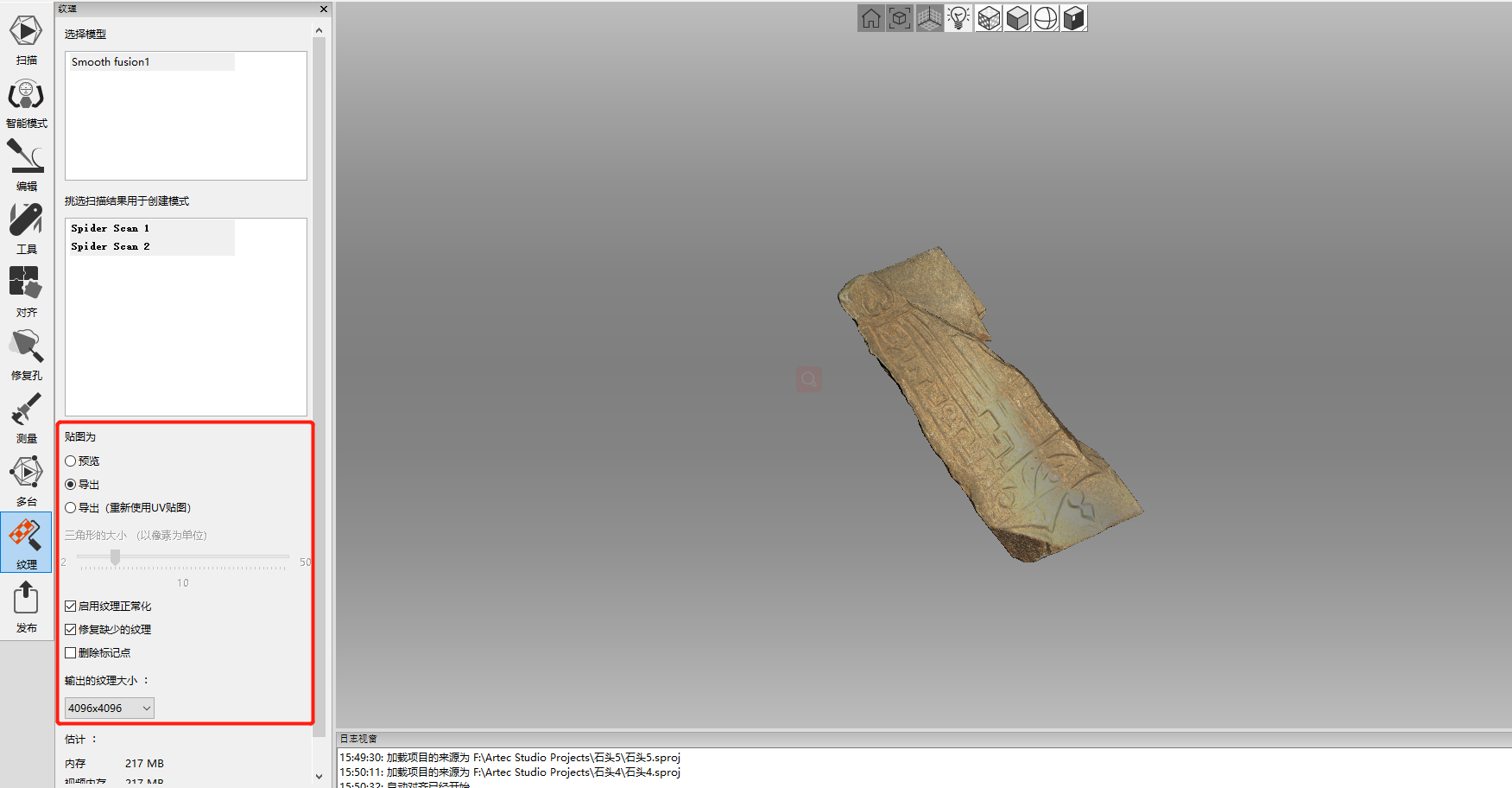


图3-2-5 Artec Studio纹理导出界面

（7）导出模型。点击菜单栏“文件”，点击“导出”，保存为需要的obj格式，保存后查看导出的模型有三个文件：“.obj”，“.mtl”以及“.png”格式。

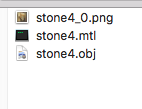


图2-4-6 Artec Studio模型导出与结果文件

第四章 详细设计

4.1 设计工具

4.1.1 WebStorm

WebStorm 是jetbrains公司旗下一款JavaScript 开发工具。WebStorm为开发者提供使用现代JavaScript生态系统的全部功能，享受智能代码完成，即时错误检测，强大的JavaScript导航和重构，TypeScript，样式表语言和最流行的框架。在IDE中轻松调试客户端和Node.js应用程序 - 在源代码中放置断点，探索调用堆栈和变量，设置手表并使用交互式控制台。充分利用linters，构建工具，测试运行器，REST客户端以及更多工具，这些工具都与IDE深度集成。 但是任何时候开发者需要终端，它也可以作为一个IDE工具窗口。在WebStorm中使用Karma，Mocha，量角器和Jest运行和调试测试。 立即在编辑器中查看测试状态，或在方便的树状视图中快速跳转到测试。使用简单的统一用户界面来处理Git，GitHub，Mercurial和其他VCS。 在IDE中提交文件，查看更改并解决与视觉差异/合并工具的冲突。WebStorm提供了强大的内置工具，用于调试，测试和跟踪客户端和Node.js应用程序。 通过将最低配置和集成到IDE中，WebStorm可以轻松完成这些任务。WebStorm与用于Web开发的常用命令行工具集成，为开发者提供高效，简化的开发体验，而无需使用命令行。

特性：

（1）智能编码协助

WebStorm为开发者带来了针对JavaScript和编译成JavaScript语言，Node.js，HTML和CSS的智能编码辅助。 享受代码完成，强大的导航功能，即时错误检测以及所有这些语言的重构。

（2）现代框架

WebStorm为Angular，React，Vue.js和Meteor提供高级编码帮助。 能够享受React Native，PhoneGap，Cordova和Ionic的移动开发支持，并使用Node.js为服务器端开发。 所有在一个IDE。

（3）智能编辑器

IDE会分析开发者的项目，为所有支持的语言提供最佳的代码完成结果。 数百个内置检测报告会在开发者键入内容并提供快速修复选项时报告任何可能的问题。

（4）导航和搜索

WebStorm可帮助开发者更有效地解决代码，并节省处理大型项目时的时间。 只需点击一下即可跳转到方法，函数或变量定义，或搜索用法。

（5）调试

WebStorm为开发者的客户端代码（适用于Chrome）和Node.js应用程序提供了一个内置的调试器。 放置断点，浏览代码并评估表达式，所有这些都不会离开IDE。

（6）测试

WebStorm与Karma测试运行器，Mocha，Jest和Protractor集成，轻松进行测试。 在IDE内部运行和调试测试，以良好的可视格式查看结果，并导航到测试代码。

（7）追踪和分析

WebStorm具有spy-js功能，这是一款内置工具，可帮助开发者追踪JavaScript代码。 探索文件如何与函数调用进行连接，并有效识别任何可能的瓶颈。

（8）代码质量工具

除了数百个WebStorm自己的检测外，它还可以针对开发者的代码运行ESLint，JSCS，TSLint，Stylelint，JSHint或JSLint，并在编辑器中即时突出显示任何问题。

（9）项目模板

使用受欢迎的项目模板（如Express或Web入门工具包）从欢迎屏幕启动新项目，并通过与Yeoman集成来访问更多的项目生成器。

4.1.2 安卓手机

AR技术的目标是在屏幕上把虚拟世界套在现实世界并进行互动。所有安卓手机都能体验到WebVR/AR平台，由于iphone并没有对webAR开放权限，无法获得本地摄像头，iPhone手机如果实现AR功能，需要通过摄像头或者搭载新的传感器，对所拍摄的物体进行定位，并通过Slam算法在手机上实现出来，这需要等待好几年甚至更久。

4.1.3 firefox

去年浏览器出现了个漏洞，这个漏洞存在于Microsoft Edge浏览器、老版本的Google Chrome浏览器（CVE-2017-5033）以及基于Webkit的浏览器（如Apple Safari，CVE-2017-2419）中。为了安全策略，不得不修复这个漏洞，只能访问https协议的本地网页，由于本课题要调取本地摄像头，所用的是http协议，但是无法在这些浏览器中打开，而firefox是唯一能够访问的浏览器，因此本课题决定用firefox进行开发。

4.2 设计思路及原理

4.2.1设计原理

AR 的优势在于把目之所及的现实场景变成了背景，并将现实世界和数字世界无缝连接，而webAR虽然是在开发初期，但是非常的有前景，AR的实现主要有两种方式：光学透视式 (Optical see-through) 和视频透视式 (Video see-through)。目前，市面上的头戴式设备通常采用 2 种方式中的 1 种或 2 种都采用，而手持设备（手机、平板等）通常采用视频透视式。而目前，web端想要实现AR技术，主要是靠视频透视式技术。要想实现AR技术需要三步：识别、追踪和渲染。最核心的就是识别和追踪。

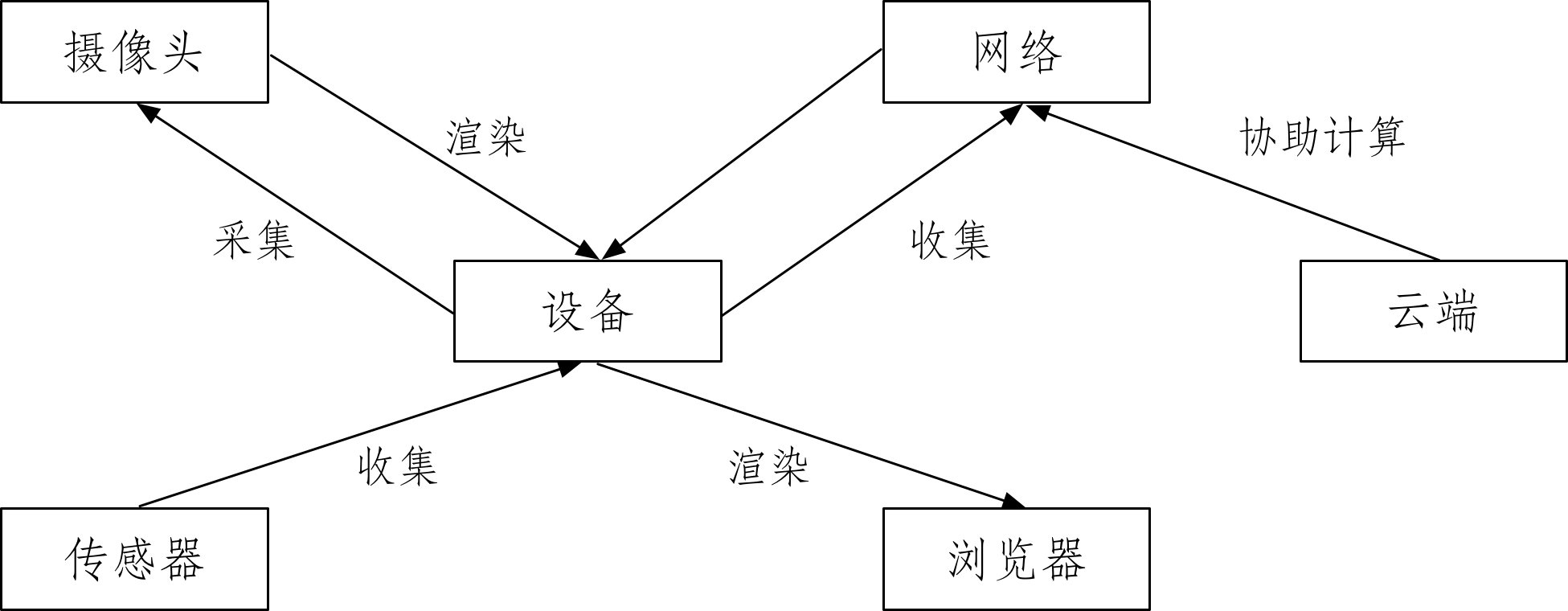


图4-2-1 webAR流程图

当然目前还没有涉及到云端。webVR规范中的部分API对webAR也同样适用，本课题将用aframe框架进行开发，这是一个封装了一个three.js的库。

4.2.2webRTC获取视频流

首先AR要识别，那就要用到 WebRTC 技术。WebRTC（Web 实时通信，Web Real-Time Communication），顾名思义是一个支持网页浏览器进行实时语音对话或视频对话的技术。它其中有个很重要的 API：getUserMedia() 可以实时获取摄像头的视频流，这是视频透视式的 AR 实现的前提。

4.2.3 aframe渲染与交互

A-Frame 是 Mozilla 团队在 2015 年开源的一款做 WebVR 的框架，但日前 A-Frame 团队发布的 aframe-xr 其中包括了一些 Web AR 组件。一开始我们也说过 VR 和 AR 中有部分实现是重合的，所以用 A-Frame 的各种组件可以让你用很少的代码构建出 AR 所需要的 3D 立体世界。aframe框架对three.js进行了封装和优化，因此只需要用一些标签和组件就能完成webAR的设计。Three.js 直接提供了 Raycaster 类提供实现 ray casting 算法。其实现原理很简单，就是摄像头（这里的摄像头不是指手机的摄像头，而是你渲染时的 Camera）视作视点，与你在屏幕上触碰的点坐标连成一条射线，看这条射线与你视图中哪些物体相交。事实上在实现 AR 的时候，识别追踪和渲染交互是同时进行的。

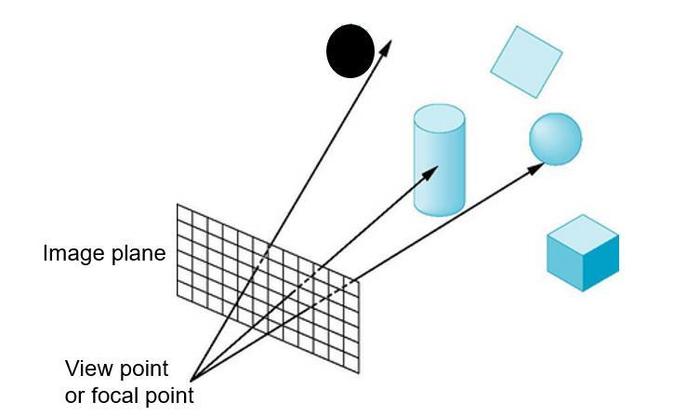


图4-2-2 Ray casting 算法

4.3 设计过程

4.3.1下载js库

A-Frame非常易于使用，为了确保AR.js能够很好地运行。 因此，结合A-Frame和AR.js，每个开发者都能很容易在网络上创建AR内容。在github上有很多开源的js库，下载aframe.js和ar.js，本课题下载的是aframe.min.js和aframe-ar.js，如图所示：



图4-3-1 下载的js库

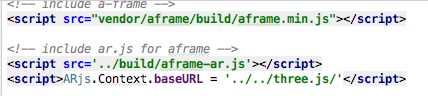


图4-3-2 加载js库

4.3.2 创建场景

场景由<a-scene>元素表示。 场景是全局根对象，所有实体都包含在场景中。场景从Entity类继承，因此它继承了它的所有属性，方法，附加组件的能力以及等待其所有子节点（例如<a-assets>和<a-entity>）的行为。 在开始渲染循环之前加载。<a-scene>为我们处理所有three.js和WebVR样板文件：

设置画布，渲染器，渲染循环；

默认相机和灯；

设置webvr-polyfill，VREffect；

添加UI以输入调用WebVR API的VR；

而这些同样适用于AR，要包含AR.js，需要包含aframe-ar.js。 然后在<a-scene>中初始化ar.js。



图4-3-3 添加场景

Embedded是嵌入，从画布中移除全屏样式。而这里是嵌入ar.js，只有在场景中嵌入ar.js，才能实现AR的效果，然后设置trackingMethod（跟踪方法）为最好的。

4.3.3 导入模型

这里需要导入obj model，obj-model组件使用Wavefront.OBJ文件和.MTL文件加载3D模型和材质。我们可以通过指向指定.OBJ和.MTL文件路径的资源来加载.OBJ模型。

Obj:选择器指向指向.OBJ文件的<a-asset-item>或指向.OBJ文件的内联路径。

Mtl:选择器指向一个.MTL文件或一个.MTL文件的内联路径的<a-asset-item>。



图4-3-4 将obj和mtl文件放入资源



图4-3-5 加载模型

A-Frame通过<a-entity>元素表示一个实体。 如实体组件系统模式中所定义的，实体是占位符对象，我们将组件插入组件以向其提供外观，行为和功能。 在A-Frame中，实体固有地与位置，旋转和缩放组件相连。

4.3.4 添加锚点

创建一个锚点来附加你的增强现实(AR)，<a-anchor>锚可用于创建指向另一个文档的链接（通过 href 属性），或者创建文档内的书签。hit-testing-enabled=‘true’是启用命中测试为“真”。



图4-3-6 添加锚点

4.3.4 获取摄像头

相机组件定义用户从哪个角度观看场景。 照相机通常与控制组件配对，以允许输入设备移动和旋转照相机。 我们可以通过修改相机实体的位置和旋转来更改视口。当退出VR时，相机会在进入VR之前将其旋转恢复到旋转状态。 这是当我们退出VR时，相机的旋转恢复到正常的桌面屏幕，同理AR也是如此。开发者告诉A-Frame你需要arjs来控制摄像机。 为此，只需添加一个<a-camera>，如图所示：



图4-3-7 添加相机

这里添加的是静态相机。

4.3.5 marker 图像标记

首先，上传自己的图像至标记生成器并生成一个模式文件pattern-marker.patt。 其次，下载训练有素的标记。 第三，打印标记。 这里不建议使用电子设备显示标记，电子设备设计频闪问题会影响标记的识别。本课题选用默认的标记“hiro”。



图4-3-8 标记图像

4.3.6 虚拟场景切换

这里添加一个天空盒<a-sky>，天空盒原始图像为场景添加背景色或360°图像。 天空盒是一个大的球体，其颜色或纹理映射到内部。为了使呈现效果是无缝的，图片应该是等距的。



图4-3-8 将图片放入资源库



图4-3-9 添加天空盒

将图片放入天空盒后通过radius属性调整半径。

4.3.7 交互

首先是添加文字，mixin提供了一种组合和重用常用组件属性集的方法。使用<a-mixin>元素进行定义，并放置在<a-assets>中。 Mixins应该设置一个id，当一个实体将该id设置为其mixin属性时，该实体将会吸收所有mixin的属性。可以创建mixin来定义可复用的组件集合<a-mixin>，与使用 <a-entity> 为场景添加一个对象不同，我们使用 <a-mixin> 来创建可复用的体素，使用它们就像使用预设实体一样。如果由多个mixin和/或实体定义，多属性组件的属性将会合并。在这里用来添加文字到资源库，设置文字居中

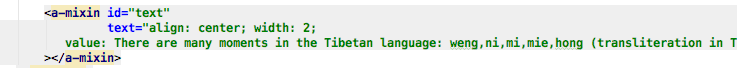


图4-3-10 文字内容添加到资源库

用<a-text>标签是包装文本组件。可以添加文字，调整位置、方向以及每行的字数。



图4-3-11 添加文字

其次是本课题要展示的是玛尼石，希望能够全方位的观察玛尼石，因此要设置旋转。我们可以在A-Frame中添加动画，并将<a-animation>元素作为实体的子元素进行动画处理。

为了定义一个围绕z轴的实体需要20秒，我们可以抵消它的位置并为父实体的旋转制作动画。 该动画从围绕0度的z轴的初始旋转开始，并绕360度旋转。 它定义为20000毫秒的持续时间，在动画的每个循环中保持最终值，并无限循环。

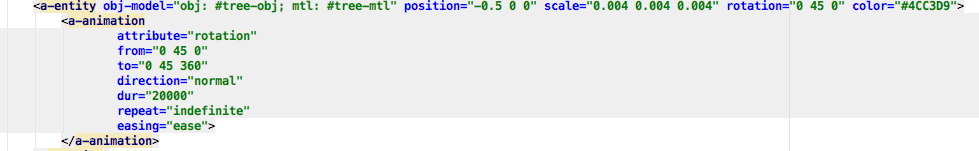


图-4-3-12 添加动画

attribute：属性为动画。 要指定组件属性，这里设置旋转。

from：开始值。

to：结束值，必须指定。

direction：动画的方向（from和to）之间。可选值有alternate, alternateReverse, normal, reverse。

dur：动画的持续时间（毫秒）。

repeat：重复计数或不确定。

easing：动画缓和功能，有很多可供选择。

4.4 案例展示

（1）代码展示

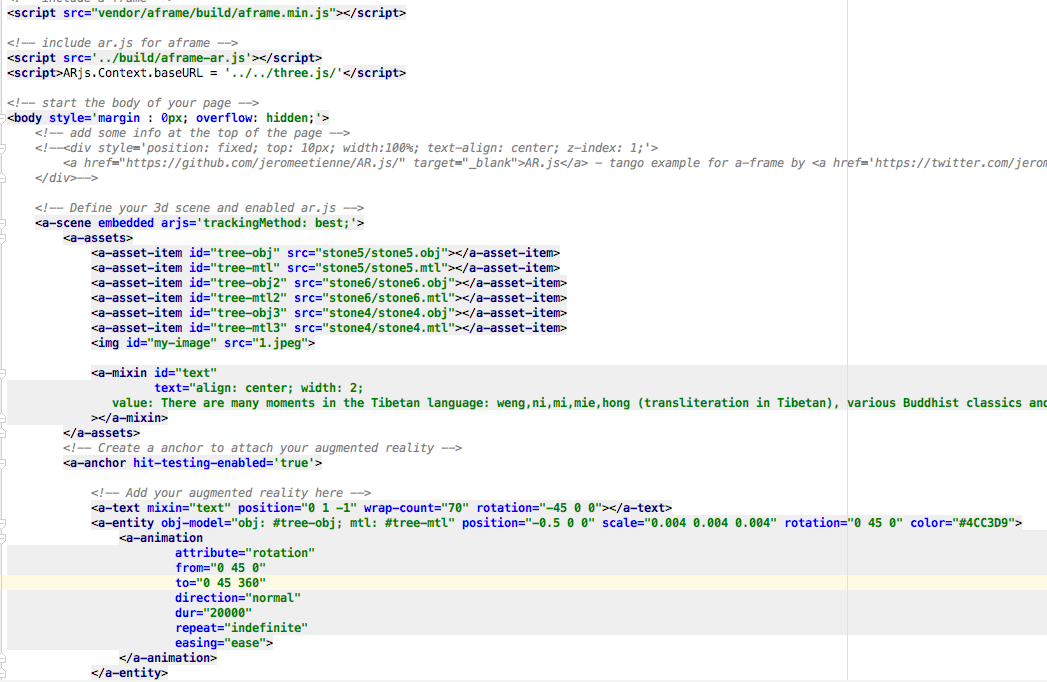


图4-4-1 代码展示1

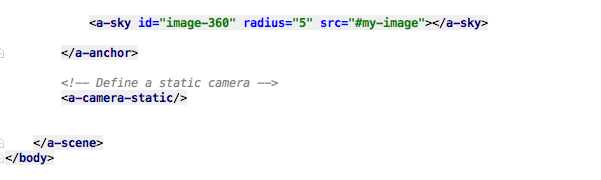


图4-4-2 代码展示2

（2）案例展示（PC端）

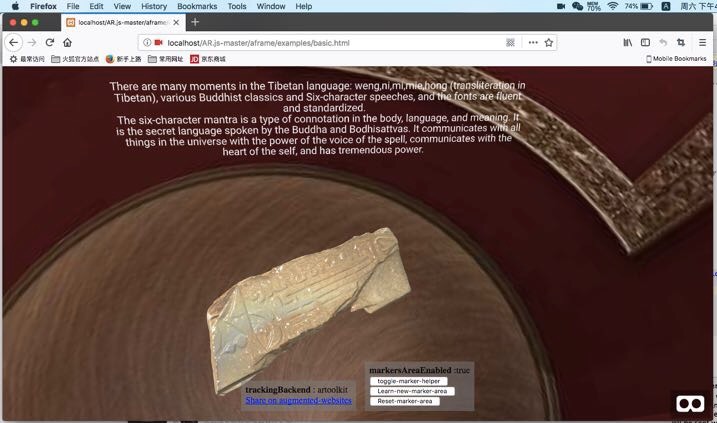


图4-4-3 4号石头PC端1

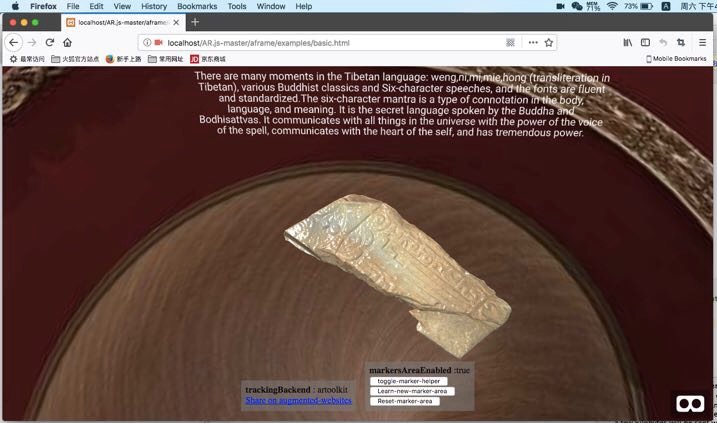


图4-4-4 4号石头PC端2

（3）案例展示（手机端）

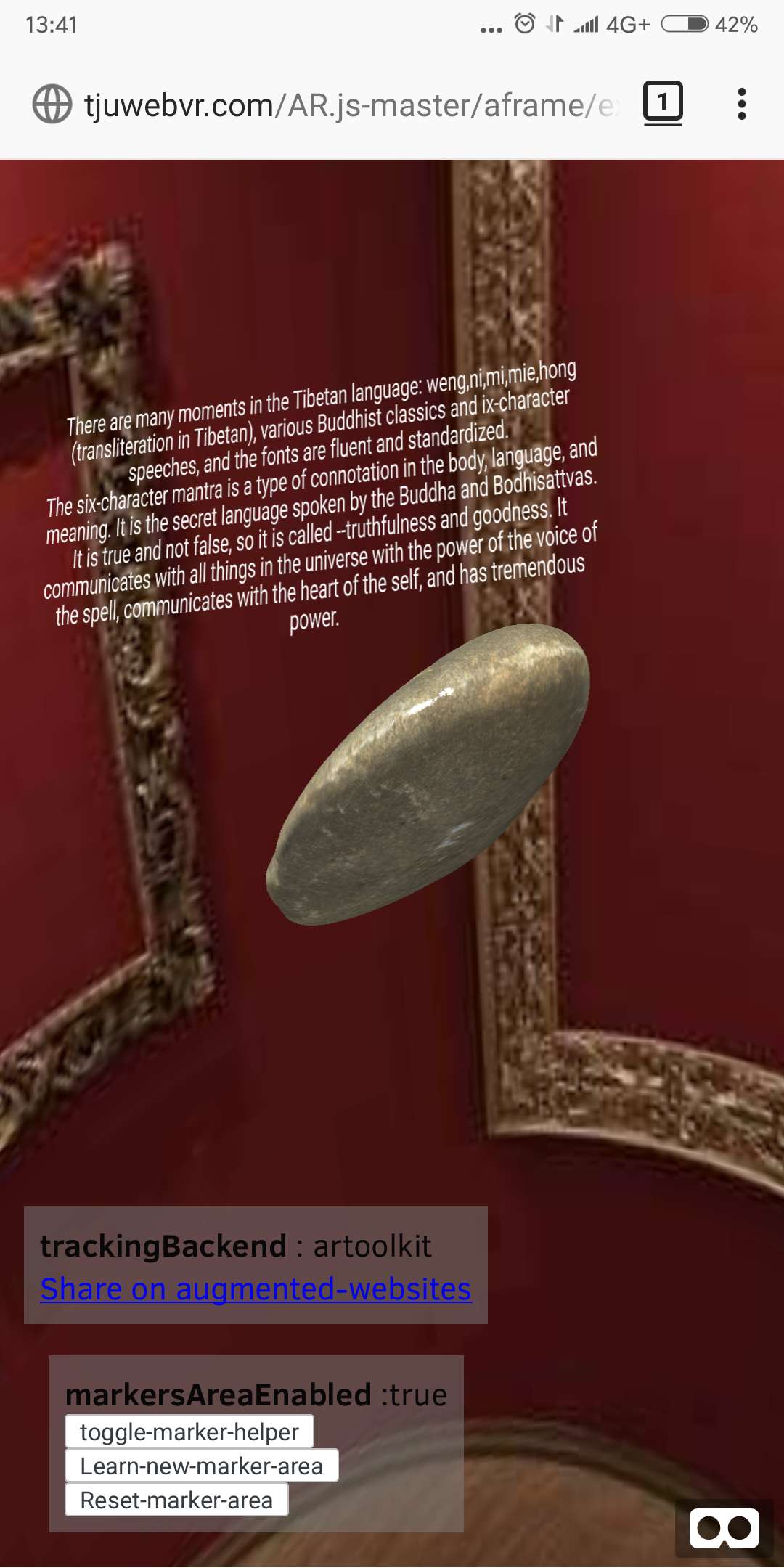


图4-4-5 5号石头手机端

第五章 案例应用

5.1 5G、物联网与移动化WebAR应用

在国家的大力推广下，我们将迎来了5G时代在国内的商业试点，而其技术正在不断的成熟。不同于4G时代，这次将会融合了大数据、移动互联网、人工智能、VR、AR等技术，可见5G时代不再只是狭隘的通信领域范围，更是通过与大数据、移动互联网、人工智能、VR、AR等技术的结合，在各个企业、民用领域能够深入的融合，这也预示着我们将迎来新一波的信息技术时代，是对VR、AR的良好接纳。

VR技术是人工制造的交互环境，而AR技术在VR技术的基础之上，VR、AR技术的到来使得用户不再只是通过视觉的效果所感知，它是通过全方位的视觉、听觉、触觉等感知，实现了人与人、人与自然、人与物之间的交互，信息量更大了，更立体了，也更真实，这都是符合人类的应用习惯的。而物联网技术使得产生人工环境的信息书记来源更加普世、真实和有效率。这使得我们的生活更加的智能化和信息化，这也是我们所追求的，而物联网技术和AR技术的结合，无疑为这种新的交互方式奠定了技术基础。AR技术的普及应用，也将会改变人们目前已有的观察、获取外部信息的方式以及交互习惯，通过这两个技术的结合，也使得AR技术更加的有实用性，而不是纯粹的好玩有趣，而是将这些运用到我们的实际生活中。

而基于webAR技术，特别是基于智能移动终端的webAR技术，也在慢慢的得到普及，这种不需要下载任何app的方式，只需要通过网页浏览器就能打开AR应用，给广大用户提供了便利，况且webAR还可以应用在我们熟悉的社交应用上，比如微信、微博、今日头条等，实现了快速的传播，同时也为实现AR技术的跨平台以及大规模的传播奠定了基础，同时也提供了新的研究方向。

手机端的webAR无疑是webAR技术的重中之重，在手机端，我们只需在社交应用中以超链接的形式通过微信朋友圈、微博、微信公众号等传播，加快对用户的影响力。用户只需在浏览器中点开网址，并将摄像头对准标记marker图像，就可以看到3D效果，给人视觉的享受。

5.2 webAR在H5新媒体营销的应用

众所周知，传统的H5都是局限在屏幕上进行点击操作，而当webAR加入到了H5中，就会将互动空间从屏幕拓展到了整个现实世界，并且每个打开H5的用户都能够获得自己独特的用户体验，使得用户更加的有沉浸感和新鲜感。能够让用户参与其中，提升了用户的参与感。由此可见，webAR的出现，能够更好的满足了新时代用户彰显独特个性的需求，无疑是为传统的H5营销注入了新的能量。

不同于传统H5的普通点击和滑动等操作，webAR可以实现触觉上的感受，给用户带来真实感和沉浸感，让H5具有更强的互动性和可玩性。同时本案例中将虚拟场景与真实场景的融合，给用户不一样的体验。

从大趋势上来看，webAR会给H5带来全新的互动展示形式，摆脱传统的形式，带领H5走上全新的时代。

5.3 数字化文化遗产保护应用

webAR技术在文化遗产的应用还属于早期阶段，很多博物馆以及历史文化遗产的保护也都开始涉足webAR技术，可以通过数字采集、复原再现、展示传播等手段，给文化遗产的数字化保护增加了很多的可能性，能够激发出体验者强烈的在场感和参与感，让文化遗产焕发新的生机。目前有很多的博物馆都开展了AR的展示，比如旧金山博物馆兵马俑展、敦煌莫高窟AR展览、百度AR博物馆、纽约MOMA博物馆等等。

本案例呈现的也是一个博物馆的展示，将制作好的玛尼石模型加载到webAR中，通过手机摄像头对标记marker图像的识别扫描，在网页中呈现玛尼石，并且有相关石头的简介。这位文化遗产的保护以及数字化处理都开辟了新的途径，让体验者对文化遗产充满了兴趣，也能体验到文化遗产背后的历史情境。

第六章 案例优势与局限性

6.1 优势

由于去年Pokémon Go的出现，掀起了AR的浪潮，摆脱了昂贵的头显硬件，使用手机便能体验AR技术。从AR设备、本地app再到现在的webAR技术，这是一个市场化普及的过程。不同AR类型的技术特性决定了对软硬件的需求。这也就凸显了webAR技术的优势。它的适应性更强，开放程度更高，更符合当前面向多种用户体验习惯。

（1）多用户性

与以往的AR不一样，以往的AR只能是AR玩家才能体验，用户体验比较少，而webAR技术的传播，是能够连接AR玩家和非AR玩家之间的重要桥梁，能够让更多的用户轻松、便捷的体验到AR，使得AR技术能够得到广泛的传播和更好的普及。

（2）成本低

webAR技术不需要头显设备，不需要专门硬件，不需要下载app，只需要一个简单的aframe框架，便可以进行开发。

（3）降低门槛

webVR与webAR是两种不同的技术，不过二者也有许多重叠的地方：它们都与穿戴在脸上的设备有关，都安装许多传感器，可以追踪动作、位置、方向。二者最明显的不同就是webVR创造全新的模拟现实，而AR将模拟层叠加在现有世界之上。尽管如此，二者使用的重叠技术、开发者处理的范例经常会碰到相同的挑战，于是W3C VR团队决定开发一个API，同时处理两种技术，它就是WebXR。webXR将webVR与webAR都包装在一起，只需要一个库，便可以开发，即使你对代码一无所知，但是也能轻松的上手。

6.2 局限性

由于体验设备的硬件需求更低，所以webAR对于其他AR技术模式，仍然有许多的问题。

（1）本地应用的兼容性不够高

目前由于苹果手机的权限问题，webAR还不能在苹果手机上使用，因此只能在安卓手机上使用，一定程度上减少了用户体验，不能大范围的传播和推广。

（2）识别局限

目前我们只能做到通过标记marker图像来识别，在一定程度上来说是很局限也很不方便的。

（3）模型要求高

只有特定格式的模型才能加载到webAR中，目前还并不能加载别的格式的模型。

（4）交互少

与webVR的交互不同，webAR可开发的交互目前还太少，只能是实现一些动画效果，旋转，移动等等，能让用户的体验感比较少。

第七章 总结与展望

7.1 总结

通过本次课题的研究，我基本掌握了webVR/AR技术的开发方法，最后呈现效果还不错。总的来说webVR/AR技术相对于其他方式呈现VR/AR具有很大的前瞻性和便捷性，不仅是对前端技术有了很深的了解，同时也对VR/AR技术有了不一样的体会和感受，随着信息时代的发展，技术不是越来越复杂，而是将复杂的技术用简单的方式让更多的人能够了解，体验和使用，webVR/AR就是这么一个简单的方式，连接着VR/AR开发者和非VR/AR开发者，从而让更多的人能够参与其中，这也是本课题研究的意义。

7.2 展望

webVR/AR技术的出现不仅是一个时代的开始，更是预示着越来越多的用户能够参与其中，而近日，亚马逊的Sumerian正式上线，致力于帮助开发者用最少的代码来构建webVR/AR，Sumerian是一个基于web的编辑器，使用者不需要拥有专业的知识，只需要将建筑物等3D对象拖放到场景中，或者导入来自第三方资源库的3D对象，这个编辑器可以在任何web浏览器中运行，能够支持ARkit和ARCore。如果开发者具备一定的编程素质，Sumerian还会提供可编程API和Sumerian命令行界面进行更加深入的定制场景，并且编写复杂的应用程序逻辑。这个平台的出现对于webVR/AR内容有着比较大的意义。