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。

目前相关部门已经对外宣布将webVR/AR的API更改为webXR,但是正在不断的完善，现在官方把它叫做”webXR Device API”。其实webVR和webAR是两个不一样的技术，不过这两个技术也有很多一样的地方，比如他们都需要头显设备，通过头显设备可以感知到用户的方向、位置以及感知。就像电影《头号玩家》一样，男主角穿上感应装备以后能感受到触觉、听觉、视觉以及痛觉，这样就能真实的感受到虚拟现实世界里所呈现的立体世界。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。2017年google对外宣布消息称，从2018年一季度开始，webXR这个标准就会开始大范围使用，这样开发者可以在稳定版的谷歌浏览器上使用实验框架，但是由于webXR还在发展初步阶段，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局限性，因此仍然需要不断的改进和优化，webXR的出现，让开发者能更加容易轻松的进行开发，让开发者从迷茫到清晰的认识，而且webXR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，如果是以前，如果想学习XR，就必须首先理解3D模型，然后再研究游戏引擎（比如unity3D）还有编程技术，但是webXR如果能流行起来，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，不需要从头开始。这也是让XR用户和非XR用户都能够利用webXR技术来进行开发，这就会让开发会更加的简单和方便。

1.2 国内外研究现状

WebVR将提供给用户一个3D世界的通道。虚拟现实技术（VR）之所以能够有这么大的吸引力，首要还是因为其能够给用户带来不一样的体验，这使得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与webVR/AR类似的接口来提供信息，因此就有了polyfill规范，虽然polyfill可能仍然不能适用于一些浏览器（例如安卓版的Edge等），但是大多数情况下还是能给到用户不错的体验。当webVR技术在浏览器中呈现的时候，就可以展示3D立体内容，同时手机端就能接受到信息并且做出反应。通过观察可以发现在浏览器的右下角有个“enter VR”的按钮，用户可以通过点击这个按钮进入双目沉浸模式，当用户将手机与头显设备连接的时候，就可以实现网页的全面沉浸，给用户带来了更加完美的沉浸感，这就是大家都认为它是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的发售出行，以及iphone和安卓手机对webAR技术的支持，webAR技术正逐步完善、改进并慢慢的融入我们的生活中。

在国内市场中，很多的信息技术公司都在开始研究webVR/AR技术，并把这两个技术不断的应用于不同的社交app中，比如wechat、新浪微博等社交应用，同时也会在webVR/AR中陆续加入更好的识别等功能。而我们也在致力于向微信公众号中实现webVR/AR，目前由于微信浏览器还不能支持webVR/AR,因此只能是点击网址跳转至特定的浏览器，而该技术由于微信公众号的接口权限问题目前还无法做到，相信不久的将来，webVR/AR一定会在微信公众号以及国内市场展现它的魅力。

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，制作一个玛尼石博物馆的展示，能够全方位的观察玛尼石，技术方面主要是利用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进行深入的学习和研究。

第二章 技术支持原理

2.1 A-Frame框架

A-Frame是一个只要用一些web标签和组件就能构建webVR和webAR的并且不需要下载任何插件的框架。使用HTML标签和实体组件来制作WebVR/AR，适用于HTC Vive，Oculus Rift，Daydream，GearVR，桌面等，A-Frame是一个构建虚拟现实（VR）3D效果的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，更有甚者，A-Frame已经能够支持简单的webAR。

A-Frame基于HTML的最上方，让webVR/AR开始变得简单。但是A-Frame不仅仅能够实现3D立体渲染，同时它的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布局。

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。更有甚者，A-Frame实现了使用户能够完全沉浸其中的交互方式，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，开发者可以为场景添加对象，不需要重新构建。

2.1.1 特性

（1）虚拟现实变得简单

只需要放入<script>标签和<a-scene>标签便可。 A-Frame能处理3D样板、VR设置和默认控件。不需要开发者安装，并且没有构建步骤。

（2）声明性HTML

HTML标签能够比较容易让开发者阅读，并且可以复制粘贴和理解。 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。这使得VR/AR不再局限于开发者。

（3）跨平台VR

为HTV Vive，Oculus 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、可组合的和可重用的实体组件结构。HTML只是其中的一部分，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等）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上高效率地完成网络上的增强现实（AR）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项目旨在帮助这些开发者或用户。 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。当然目前还只能在安卓手机上使用。然而不幸的是，AR.js目前不适用于iOS。例如Safari尚不支持WebRTC，但Apple正在努力开发它。

2.3 threex-artoolkit

ARToolkit，是一个开源的适用于AR(增强现实)的SDK。threex.artookit是可轻松处理artoolkit的three.js扩展。threex.artoolkit由以下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。

2.4 webGL

WebGL（全称Web Graphics Library）是一种3D绘图协议，这种3D绘图技术标准允许把JavaScript和OpenGL ES 2.0组合在一起，通过增加对OpenGL ES 2.0的JavaScript连接，WebGL可以为HTML5的场景画布提供硬件3D立体效果，实现加快渲染，这样开发者就能够在浏览器留直观的清晰的展示3D场景和模型。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。了解OpenGL ES 2.0的开发者可以通过使用GLSL将WebGL识别为在Shader基础上的API，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3D带入网络，直接在浏览器中展现。主要浏览器厂商Apple（Safari），Google（Chrome），Microsoft（Edge）和Mozilla（Firefox），这些都是WebGL工作组的成员。兼容性如图所示：

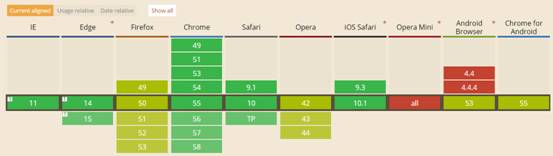


图2-4-1 webGL兼容性

2.5 webRTC

WebRTC，也叫网页实时通信（全称Web Real-Time Communication），是一个支持网页浏览器进行实时语音或者视频对话的技术，WebRTC实现了基于网页的视频会议，它的标准是WHATWG 协议，目的是通过web浏览器提供简单的javascript以便达到实时通讯（Real-Time Communications (RTC)）的能力。兼容性如图所示：

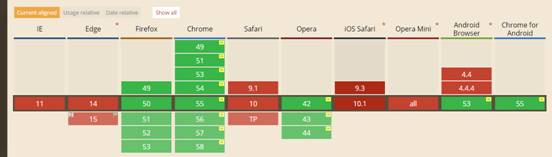


图2-5-1 webRTC兼容性

2.6 three.js

Three.js就是用javascript来写3D程序的一个库。同时three.js也是用JavaScript编写的WebGL第三方库。Three.js 是一个运行在浏览器中的 3D 库，开发者可以用它建立各类三维场景，包括了照相机、光线、材质等各种对象。Three.js包含于a-frame框架中。Three.js结构图如图所示：

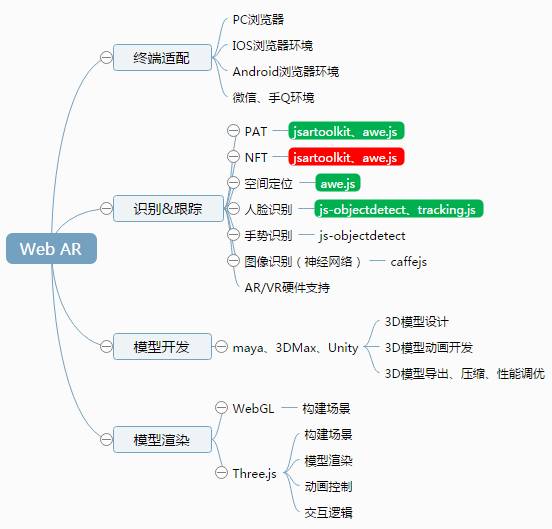


图2-6-1 three.js结构图

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（通常简称为3d Max或3dsMAX），是一个应用于window电脑系统的可以进行三维动画渲染和制作的软件。现在被广泛应用于医学、动画、游戏、汽车行业等领域。具有性价比高，使用者多，便于交流，上手轻易的优势。3dsmax导出obj格式文件的方法：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

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。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。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在汽车零部件设计，控制质量，产品设计和制造等领域提供了很多的可能性。

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。其他选项如果没有特殊要求为默认设置。如图所示：



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。

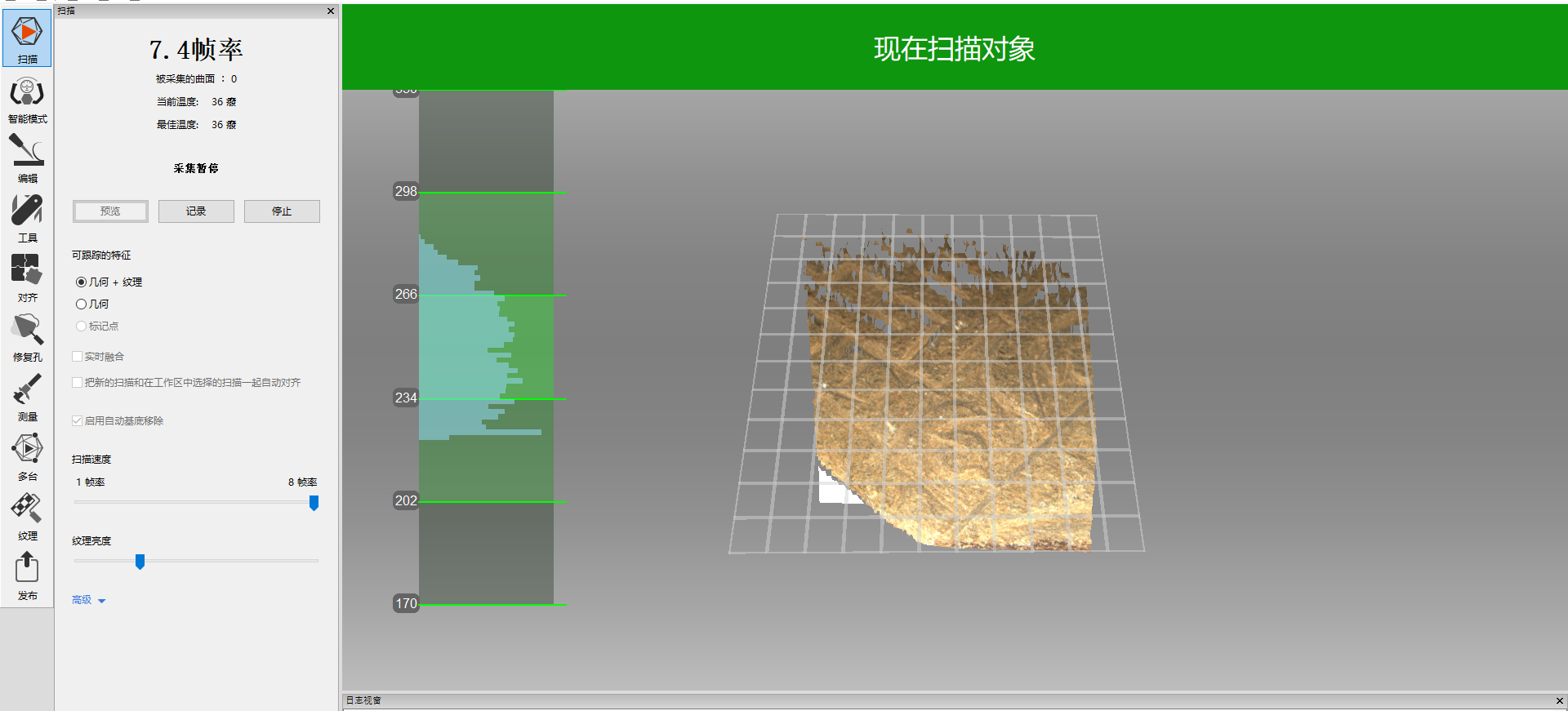


图3-2-2 Artec Studio扫描界面

（4）将取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，接着点击“对齐”，然后选中扫描数据，点击“自动对齐”。如果“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。如图所示：

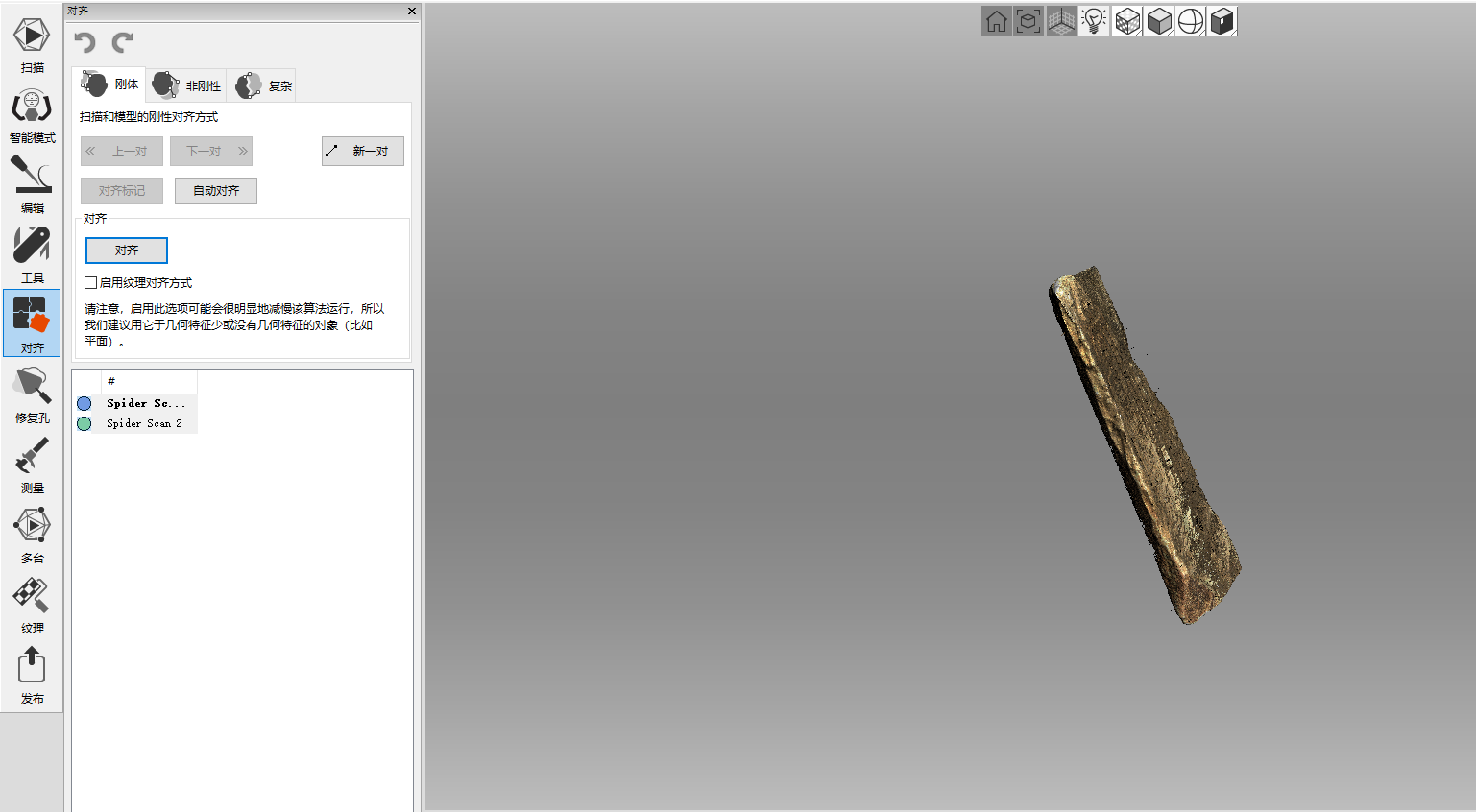


图3-2-3 Artec Studio对齐界面

（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。如图所示：

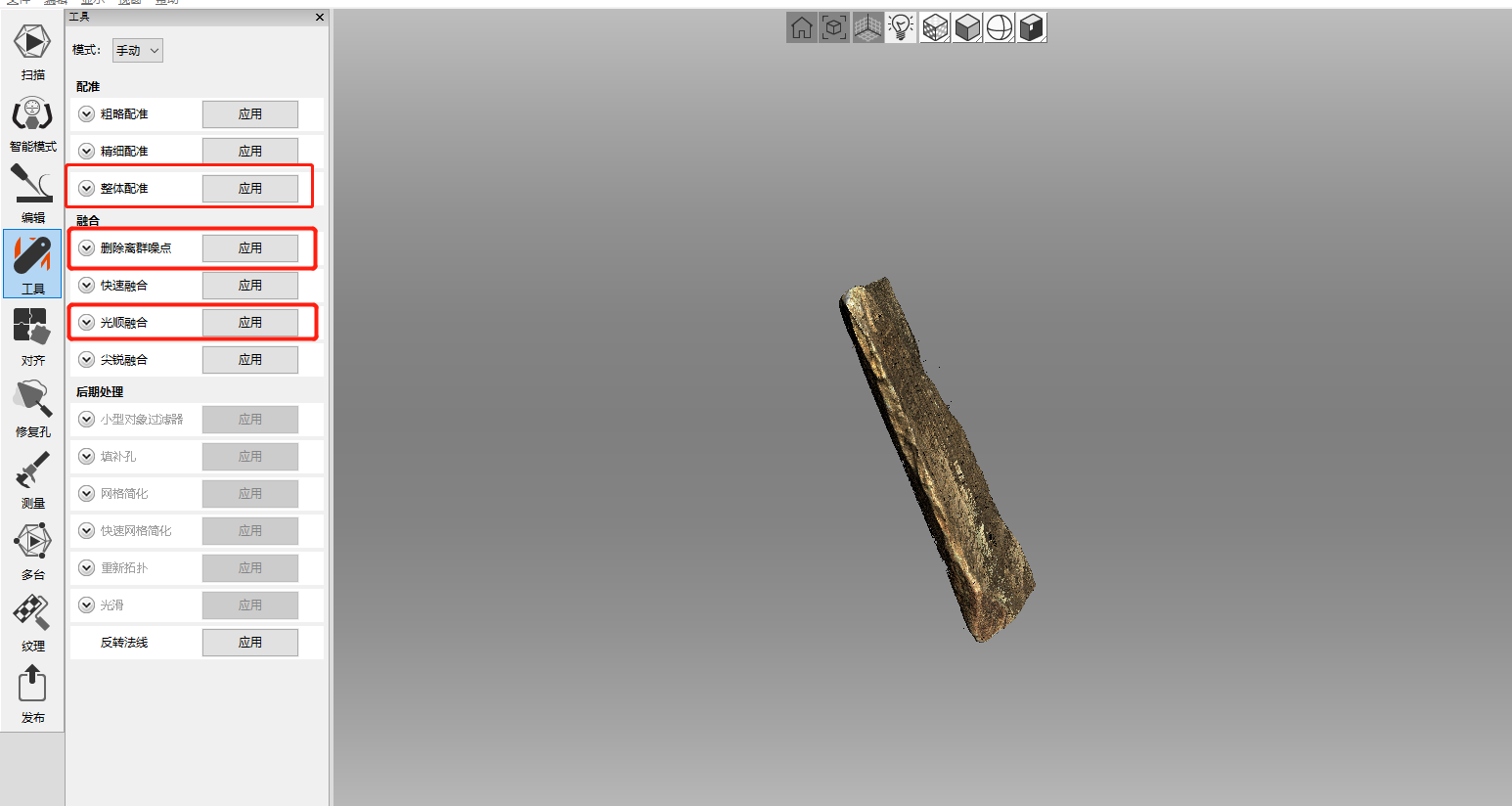


图3-2-4 Artec Studio工具界面

（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。如图所示：

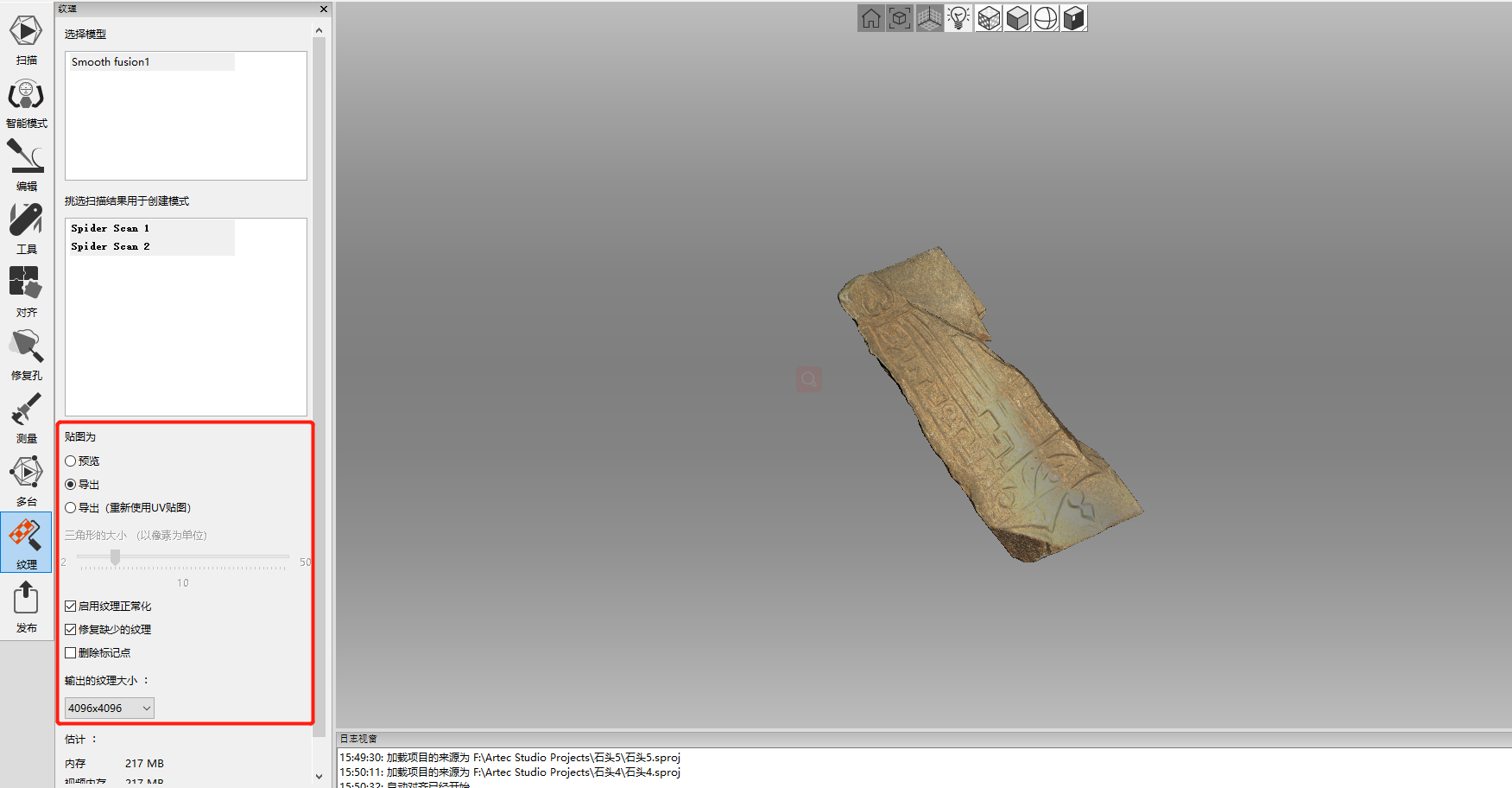


图3-2-5 Artec Studio纹理导出界面

（7）导出模型。点击菜单栏的“文件”，然后再点击“导出”，保存为所需要的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，这样就成功了。如图所示：

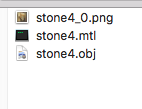


图3-2-6 Artec Studio模型导出文件

3.3 Artec扫描建模注意事项

（1）对于复杂的玛尼石模型扫描，当旋转台旋转的时候，需要及时的对扫描的位置和角度进行调整，尽量保持artec扫描仪与玛尼石的角度和距离保持一致，因此在扫描的时候要根据玛尼石的旋转来调整位置，同时还要保持artec扫描仪的平稳性，尽量不要抖动，如果artec扫描仪对物体的距离和角度发生了改变，就会有可能无法扫描，并且在Artec Studio中会发生错位的现象，构建的模型就不准确，由此反复的话会耗费时间。

（2）扫描时，尽量保持在旋转台旋转一到两周之内完成，因为artec扫描是会不断的叠加的，扫描的时间越长，那么就会越占内存空间，而占的内存空间越大，设备的流畅性就会受到影响，会发生卡帧的可能，因此为了保持设备的流畅性，同时也缩短扫描所消耗的时间，尽量在旋转台旋转到一到两周内完成，避免重复扫描。

（3）在融合选项中，提供了三种不同的融合方式：快速融合、尖锐融合和光顺融合。快速融合就是能够加快模型的加载效率，但是融合的效果相对较低，存储空间占据最小。尖锐融合是模型融合效果最好，但是模型的加载效率比较低，存储空间占据最大。光顺融合则是一种比较平衡的方式，模型融合效果和模型加载效率都相对适中，则本课题选择模型融合效果和模型加载效率都适中的光顺融合。

（4）在导出的模型格式中首先必须有个obj格式，然后是管理材质纹理贴图的mtl格式，这个格式可以查找jpg、jpeg、png等图片格式，将这几个格式文件放在一个文件夹中，并且不能参杂其他多余的格式的贴图，否则在载入模型的时候就不能识别并且呈现。

第四章 详细设计

4.1 设计工具

4.1.1 WebStorm

WebStorm 是一个适用于JavaScript 的开发工具。WebStorm为开发者提供适用于现代JavaScript的全部有效功能，并且能够使用智能代码，即时bug检测，庞大的JavaScript导航系统和重新构造的能力，同时拥有TypeScript功能，样式表语言和目前最流行的框架。在IDE中轻松调试客户端和Node.js应用程序，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。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。 与此同时，开发者可以在IDE编辑器中上传文件，然后可以通过查看更改，来处理各种矛盾。WebStorm平台提供了有效的工具，可适用于debug，测试，跟踪客户端和Node.js应用程序，WebStorm可以快速并且有效率的完成以上目标。WebStorm与用于Web开发的常用命令行工具结合，为开发者提供高效，简单的开发体验，并且能够实时的保存，避免电脑突然的关机丢失文件。

特性：

（1）智能编码协助

WebStorm为开发者带来了基于JavaScript、Node.js、HTML和CSS的智能编码辅助功能。实现代码完成，强大的导航功效，即时bug检测以及所有这些语言的重组。

（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应用程序供应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内部实现运行和debug测试，以良好的可视化形式检测结果，并且实时导航到测试代码。

（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技术的目标是在电脑屏幕或者手机屏幕上讲虚拟模型、场景镶嵌在现实场景中并且实现交互。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手机若是要实现AR功能，需要通过摄像头或者创建新的传感器，然后定位开发者要拍摄的物体，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

firefox是一个相对其他浏览器来说比较简洁的浏览器。不会包含大多数用户所不需要的功能和插件，因此，firefox不太容易崩溃，同时firefox的安全性也是本课题选择它的原因，迄今为止，firefox还没有成为一些黑客们的重要攻击对象。同时Firefox有一个优势，用户可以从一个浏览器窗口的单个标签页面中打开各个网站，这样就避免了在浏览器窗口之间的频繁切换，只需要点击工具栏下显示的标签，就是可以在两个或者多个不同的网页之间切换。同时firefox的用户量也相对较多，这也为webVR/AR的发展奠定了一定的用户基础。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 设计思路

本课题主要设计一个博物馆玛尼石展示的一个webAR案例，由于传统的博物馆展示馆中的藏品比较昂贵，有时候距离较远，不能近距离的观察，在这样的基础上加上了webAR技术，可以通过扫描藏品旁边的标记marker图像，在手机上立刻呈现一个3D立体效果，并且附上文字介绍，可以让参观者更加直观更有沉浸感的了解和感受玛尼石藏品，同时也增加了趣味性和参观者的参与性，如果想要更仔细的观察，也可以将藏品放大，摆脱了真实藏品的麻烦性，增加了观察者的参与感，让观察者更加能够参与其中。

4.2.2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取1 种或 2 种都采取，而本课题所需要的手持设备（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，主要还是靠视频透视式技术。要想实现webAR技术这里需要三步：设备识别、追踪感应器和3D渲染，最重要的步骤就是设备识别和追踪感应器。

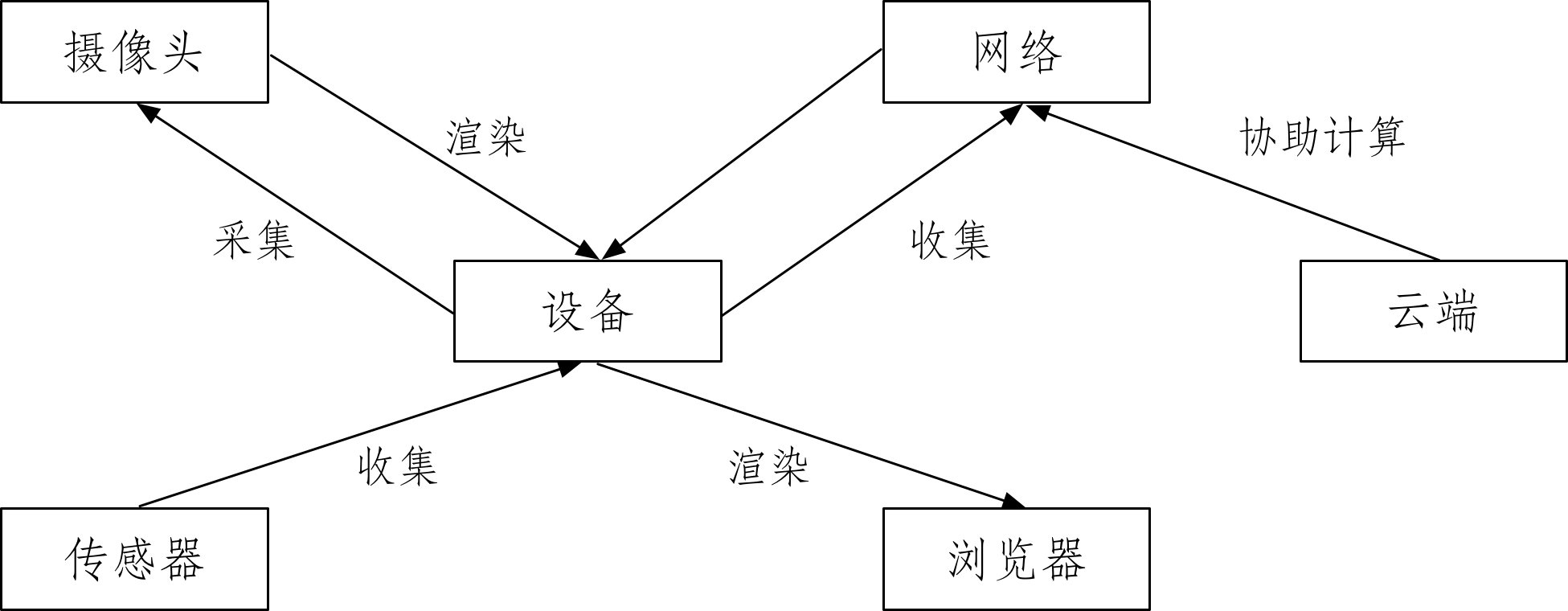


图4-2-1 webAR流程图

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。

4.2.3 webRTC获取视频流

首先AR要识别，那就会用到 WebRTC 技术，WebRTC（全称Web Real-Time Communication），就是一个支持web浏览器进行实时语音或视频对话的技术。它其中有个很重要的接口函数getUserMedia()，可以实时获取所添加的摄像头的视频流，这样就可以调取摄像头，完成了webAR中的重要一步。

4.2.4 aframe渲染与交互

A-Frame是 Mozilla 团队在早年开发的一个做 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。在之前的章节中也介绍了webVR/AR技术是有一些内容是相似的，因此利用A-Frame框架中的标签可以让开发者用相对较少的代码来制作webAR。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直接提供了一个Raycaster类，能够实现ray casting算法。这个算法的实现理论很简单，就是把设置的摄像头（这里的摄像头不是指手机的摄像头，而是开发者在渲染时的Camera）看作视点，与开发者在屏幕上触碰的点坐标连起来构成一条射线，观察这条射线和开发者视图中的哪些物体有相交。事实上在实现AR技术的时候，设备识别、感应器追踪和3D渲染交互是同时进行的。

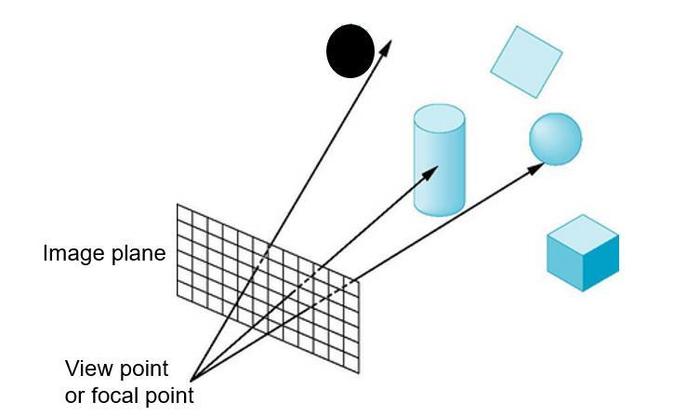


图4-2-2 Ray casting 算法

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

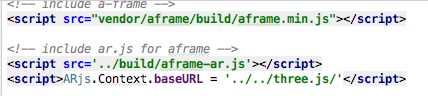


图4-3-2 加载js库

4.3.2 创建场景

添加场景是由<a-scene>元素来表示。场景是全局根对象，所有的实体组件都包含在场景中。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。在开始渲染之前加载。<a-scene>标签为我们处理了所有的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webAR，如果要包含AR.js，就需要包含aframe-ar.js。 然后在<a-scene>中初始化ar.js。如图所示：



图4-3-3 添加场景

Embedded是嵌入的意思，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的.OBJ文件和.MTL文件来加载3D模型和它的材质纹理贴图。我们可以通过指向指定.OBJ格式和.MTL格式文件路径的资源来加载.OBJ模型。

Obj:选择器指向.OBJ文件的<a-asset-item>资源库或者指向.OBJ文件的内联路径。

Mtl:选择器指向一个.MTL文件或者一个.MTL文件的内联路径的<a-asset-item>资源库。

如图所示：



图4-3-4 将obj和mtl文件放入资源



图4-3-5 加载模型

A-Frame框架通过<a-entity>标签元素来表示一个实体。可以在实体组件中加入我们想要添加的模型，并且设置位置、方向、大小等属性。在A-Frame框架中，实体本身与它的位置，旋转和缩放等组件是息息相关的。

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”。如图所示：



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，下载出标记。 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”marker标记图像。如图所示：



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，使其添加的颜色或纹理映射到内部。为了让虚拟场景呈现的效果是无缝的、连接的，我们选择的图片应该是等距的。如图所示：



图4-3-9 将图片放入资源库



图4-3-10 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，并将它放置在<a-assets>资源库中。<a-mixin>标签应该设置一个id或者class，当一个实体将该id设置为它的mixin属性时，那么这个实体就会查看并且吸纳所有的mixin属性。除此之外，还可以创建多个mixin属性复合使用，与使用<a-entity>标签来为场景添加一个对象不同，我们使用<a-mixin>来创建可多次使用的元素，使用它们就像使用一般的实体一样。为了让多个属性的组件能够整合在一起，需要用<a-mixin>标签或者多个实体去define。在这里用来添加文字到资源库，设置文字居中。如图所示：

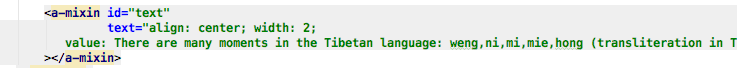


图4-3-11 文字内容添加到资源库

这里用<a-text>标签，这是包装文本的一个组件。可以用这个组件来添加文字，调整文字的位置、方向、字宽以及每行的字数。如图所示：



图4-3-12 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。通过在A-Frame框架中添加动画效果，并将<a-animation>元素作为实体的子元素进行动画处理，这样处理方便并且有效。

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。如图所示：

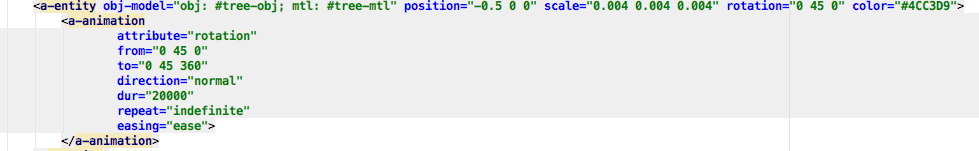


图-4-3-13 添加动画

attribute：这个属性是设置动画效果的选项。为了实现动画效果，就要指定组件的属性，这里设置为旋转。

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

在<script>中加载aframe.min.js和aframe-ar.js这两个库，并插入artoolkit插件，首先在<body>中添加3D场景<a-scene embedded arjs='trackingMethod: best;'>，在场景中嵌入ar.js，接着在资源库<a-assets>中加载想要展示的模型、文字和虚拟场景的背景图片。这里放入了4号、5号和6号玛尼石。接下来添加一个锚点anchor，加强增强现实（AR）,然后就是添加模型，在属性中调整模型的位置，方向以及大小，这里是将模型固定在marker标记图像中，然后添加模型的动画效果，这里添加的是旋转，能够360度的无限旋转，转速为20000毫秒，接下来是将文字介绍放置在模型之上，设置位置、方向以及每行的字数。设置好实体组件了以后就是添加一个天空盒<a-sky>，由于此次课题需要做一个博物馆的展示，因此添加了一个博物馆的背景图。以上这几个组件都是写在锚点中，为了能够加强增强现实，最后是添加照相机，这里添加了一个静态照相机。到这里一个webAR案例就完成了。代码如下：

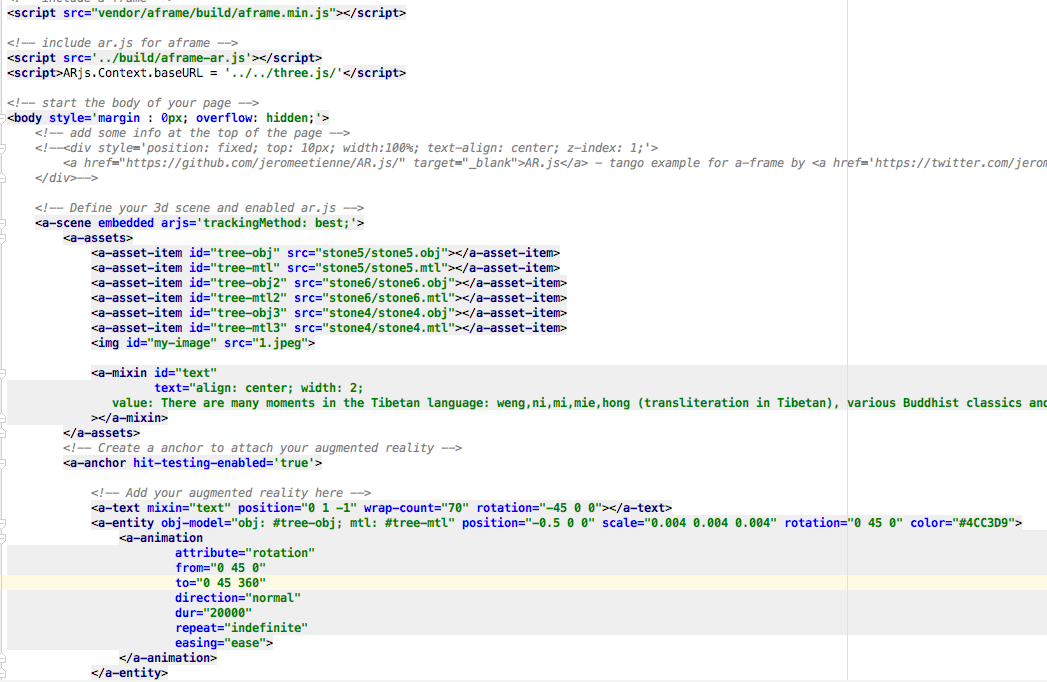


图4-4-1 代码展示1

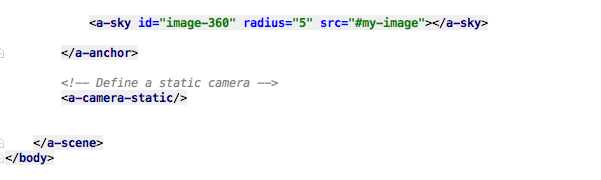


图4-4-2 代码展示2

（2）案例展示（PC端）

本课题将相关代码push到了自己的服务器，由于兼容性的问题，选择在firefox中打开网页，<http://www.tjuwebvr.com/AR.js-master/aframe/examples/basic.html>，首先会弹出一个是否允许打开摄像头，点击“同意”之后，则出现了一个摄像头打开的界面，之后将识别标记marker图像对准摄像头，则出现了以下界面，如图所示：

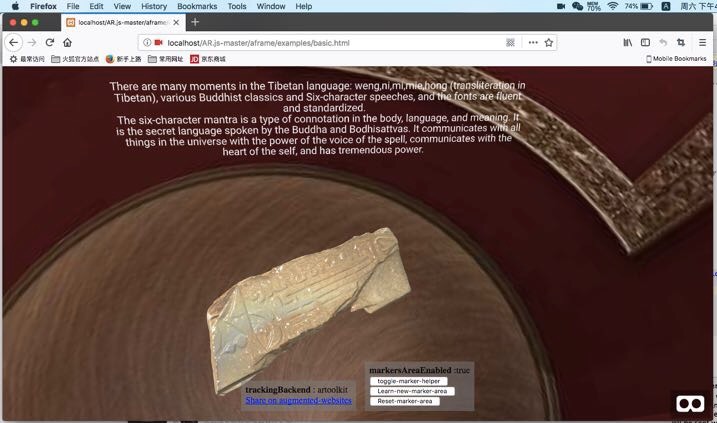


图4-4-3 4号石头PC端1

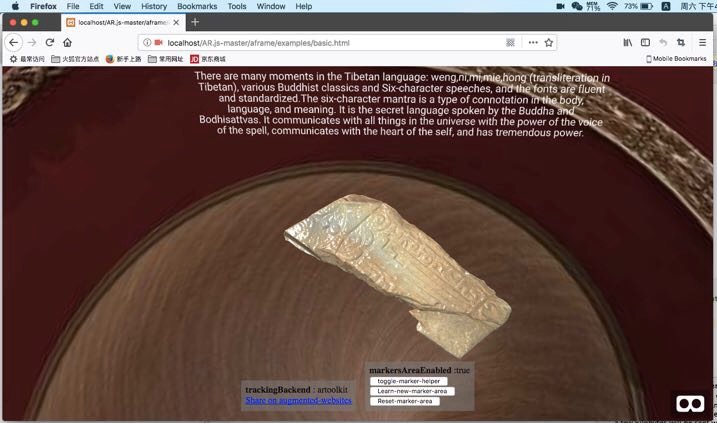


图4-4-4 4号石头PC端2

（3）案例展示（手机端）

在手机端的firefox中打开网页（<http://www.tjuwebvr.com/AR.js-master/aframe/examples/basic.html>），这里需要等待一些时间，因为模型的精度比较高，需要加载的时间，手机端也会弹出摄像头允许通知，点击“同意”，则打开了摄像头界面，这里打开的是手机后置摄像头，此时将手机摄像头对准打印好的标记marker图像，便能呈现如下场景，如图所示：



图4-4-5 手机端摄像头打开界面

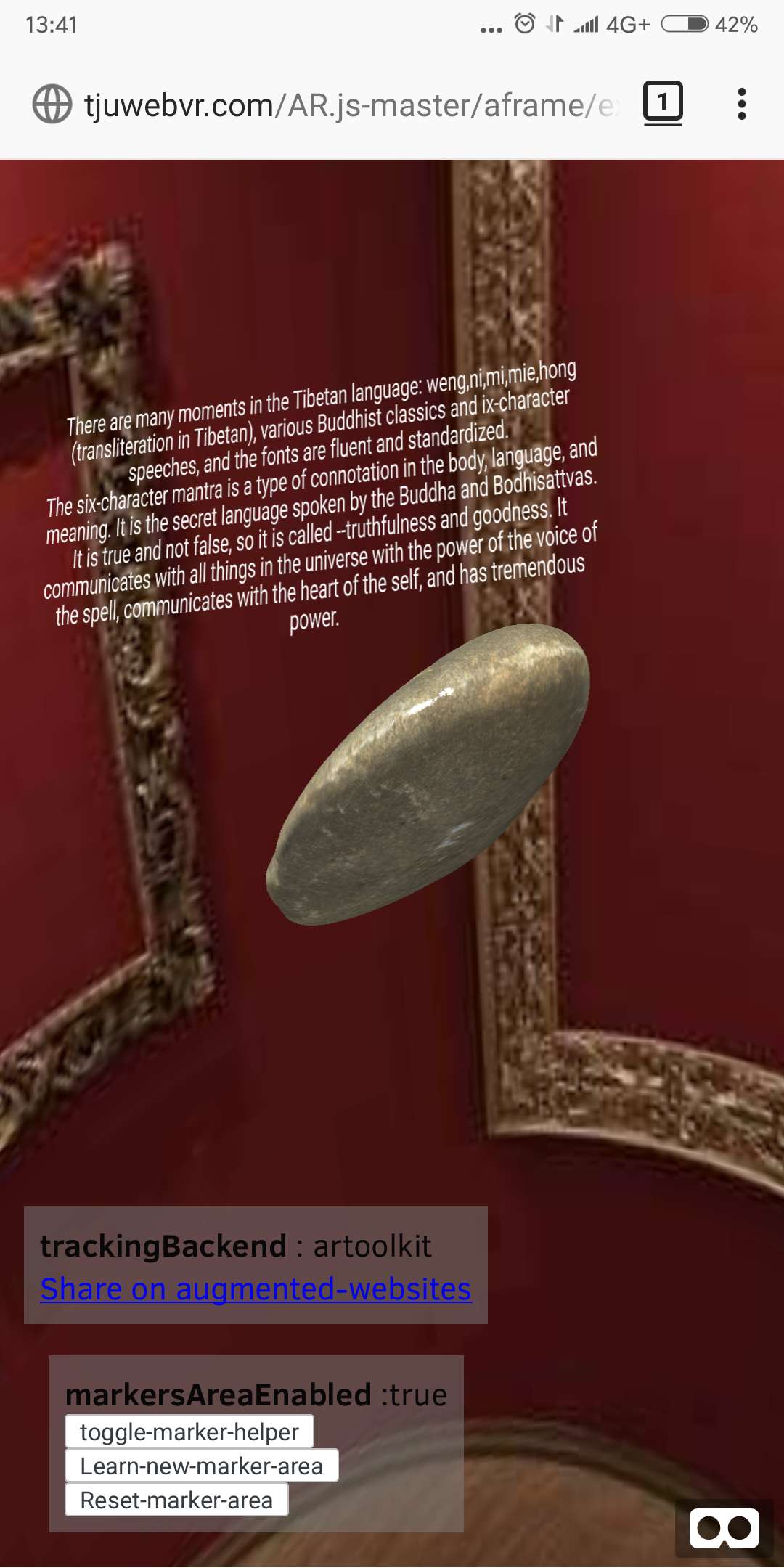


图4-4-6 5号石头手机端

第五章 案例应用

5.1 5G时代与移动端WebAR应用

在我国信息技术的大力推广下，我们即将迎来了5G时代在国内的商业试点，况且它的技术正在不断的成熟化。不同于4G时期，此次将会融合了大数据、移动互联网、人工智能、VR、AR等新兴技术，由此可见5G时代不再只是局促的通信范畴规模，更是经过与大数据、移动互联网、人工智能、VR、AR等新兴技术的结合，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3D模型的一个一个交互环境，而AR是技术是在VR技术的基础之上，VR技术的到来使得用户再也不户再也不只是通感观过视觉的感观效果所感知，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，这使得用户接收的信息量更大了，更真实了，这都是符合人类的应用习惯的。而物联网技术使得人们生活的信息来源更加朴实朴实、真实和有效率。这使得我们的生活正逐步向着智能化和信息化发展，这也是我们所追求的，随着时代的发展，这两个技术能够实现结合，也为这种新的实现交互的方式奠定了一定的技术基础。技术的普及应用，也将会改变人们目前固前固对于外部信息的接收方式式以及互习惯，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。

而对于webAR技术，特别是对于智能移动终端的webAR技术，也在慢慢的得到普及与接纳，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webAR技术在各个平台的实现奠定了坚实基础，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，开发者只需要在社交应用中用超链接的形式通过微信朋友圈、新浪微博、微信公众号等方式进行传播，就能加快webAR技术对用户的影响力。用户只需要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新媒体的应用

尽人皆知，传统的H5都是只能局限在屏幕上对它进行点击操作，而当webAR加入到了H5中，就可以把用户带到了真实世界里，在真实世界里操控这些本来只能在屏幕上操控的功能，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。能够让用户参与到其中，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 文化遗产保护应用

在文化遗产中引用webAR技术还在初步阶段，很多博物馆以及历史文化遗产的保护也都开始涉足于webAR技术，可以通过遗产展示、文字介绍等一系列手段，给文化遗产的信息化保护增加了很多的可能性，并且让体验用户有很强烈的参与感，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。

5.4 汽车领域中的应用

随着VR/AR技术的流行，汽车行业领域也越来越多的使用VR/AR技术。首先是汽修教育，通过一个webVR系统，不再只是枯燥的课本或者ppt的学习，可以更直观的通过3D立体效果来对一辆汽车进行维修教程，这样的方式更加的直观，学员只需点开一个网页，就可以进入VR视野进行学习，随时随地的学习，学员能够体验3D视野的汽车维修，方便而且高效。其次是汽车评审，目前的一下企业评审人员一般都是异地的，而普通的汽车评审需要经过汽车进行油泥模型喷漆等等工序，耗时大概几个月，而如果加入了webVR/AR技术，则可以建立汽车模型导入，对汽车的零部件以及外观进行评审，既缩短了评审时间，又能高效率的完成，也能促进汽车行业的发展。

5.5 医学领域应用

随着医学技术不断发展，医学与信息科技正在不断的融合。可以将webVR/AR技术与中医结合，建立一套标准的中医养生推播商业模式，可以让医学院的团队可以简易、方便的方式来设计想要推播的中医养生信息。摆脱了传统的文字图片的枯燥方式，而是用一种3D立体的效果，让其患者可以很直观的接受相关中医知识，可以让患者达到沉浸式学习与体验。

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，而非AR玩家就不能参与其中，AR玩家和非AR玩家之间就缺少了沟通，从而他们之间就会产生一定的隔阂，不利于AR技术的发展和应用，而且用户体验比较少，但是webAR技术的出现与传播，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技术，使得AR技术能够得到广泛的传播和更好的普及，同时也为webAR技术的到来和发展奠定了一定的用户基础。

（2）成本低

webAR技术不需要头显设备，比如，HTC vive、Oculus Rift、hololens、AR镜片等头显设备，这些设备的价格都相对高，例如HTC vive价格为5488元，Oculus Rift价格为4578元，hololens价格为23488元，对于普通的学生或者一般的开发者这样的价格太贵，开发的成本太高了。同时webAR技术不需要下载专门的硬件，不需要下载特定的app，只需要用一个简单的aframe框架以及ar.js库，便可以进行webAR的开发，对于任何的开发者甚至小白都是个不错的选择。

（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相似的地方：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

因为体验设备的硬件需求太低，因此webAR对于其他的AR技术，仍然存在着许多的问题，这也是开发者可以继续研究探讨的一部分，有缺陷才能更加的完善也会更加促进webAR技术的优化。

（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，并且目前只能在firefox浏览器打开，对用户的使用受到一定的局限性，不能满足广大用户的要求，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。当摄像头离开标记marker图像或者有点偏移的时候，就没办法识别，用户体验有点差。

（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。因此对模型的制作要求就会非常的高，尤其是模型的纹理贴图材质，如果模型没有按照指定要求的来制作，就无法将模型加载进webAR中。在一定程度上模型的加载也有一定的局限性。

（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。

（5）加载慢

本课题所需要的玛尼石模型因为纹理材质贴图的精度比较高，在当初制作模型的时候，融合方式选择了光顺融合，光顺融合相对于快速融合的加载效率没有那么高，因此，在网页每一次刷新的时候都会耗费一定的时间，并且模型的加载花费的时间也很大，有时候天空盒加载出来了，但是模型还未加载，就会影响用户的体验效果。

第七章 总结与展望

7.1 总结

本课题实现了西藏玛尼石的展示的webAR案例，可以让用户360度的观察玛尼石，拜托了传统的观赏方式，提供了一些新的思路。的通过本次课题的研究，我基本掌握了webVR/AR技术的开发方法，对于webVR/AR的技术背景、设计原理、操作方法以及最后的呈现方式和交互都有了一定的了解，最后呈现效果达到了最初的设计目的。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，webVR/AR技术相对于其他呈现VR/AR的方式还是有一定的不同的，相对于别的方式，webVR/AR适合各个开发者，上手比较简单，能够很快的适应和掌握，加上aframe框架的简化，只需要一些标签组件便可以制作一个简单的webVR/AR,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，亚马逊的一个适用于webVR/AR技术的平台Sumerian出现在大众的视野中，这个平台的出现标志着开发者不再是用复杂的代码才能开发webVR/AR,只需要用很少的代码就可以实现webVR/AR的开发。Sumerian是一个能够在web端实现webAR的新型平台，降低了对开发者的要求，开发者不需要具备很专业的编程能力，只需要将需要的3D模型拖放到构建好的场景中，或者导入来自第三方资源库的3D模型，Sumerian这个编辑器可以在任意一个web浏览器中运行，能够支持ARkit和ARCore，兼容性特别好。如果开发者具有一定的coding能力，那么Sumerian还会提供一个能够写代码的接口和Sumerian命令行进行更加深入的制作场景，并且编写复杂的应用程序。这个平台的出现对于webVR/AR内容有着比较大的意义。我们相信随着webAR技术的深入研究，能够探索到更多实现交互的可能。不久的将来，webVR/AR技术会应用到更多的领域，还会有更多的交互，这都是有可能的，未来，webVR/AR将会是大势所趋，成为主流。