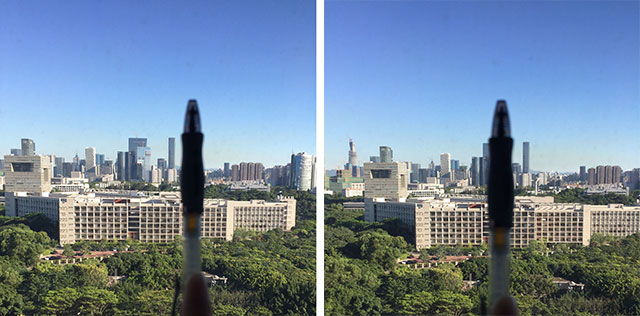
VR，全称virtual reality，中文叫虚拟现实，是模拟出三维空间的虚拟世界，提供用户关于视觉等感官的模拟，在视觉上产生立体感。当使用者头部移动时，也可以随之看到不同方向的画面，让人仿佛身临其境。

VR让人产生真实感的关键要素有三个：立体感，实时性，高交互性。

一，立体感。

人眼看现实世界之所以有立体感，是因为双眼之间有间距，造成双眼看到的东西视角有一点差别。

竖起中指，分别遮住左右眼，可以看到中指在远处景物的相对位置有偏差。如下两图所示：



（注：重点是笔相对背景的位置差异，左眼看到的笔相对位置偏右，而右眼看到的笔相对位置偏左。）

左眼与右眼图像的差异称为视差，大脑会自动将两眼看到的图像融合，产生出有空间感的立体视觉效果。

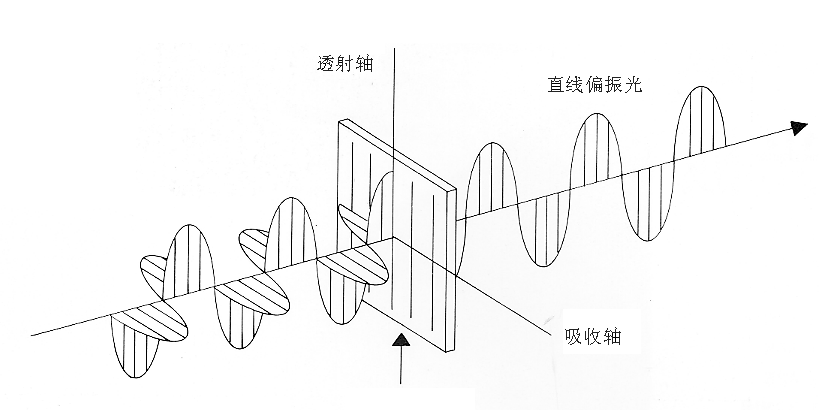
那么电子屏幕上如何让用户两眼看到不同的图像呢？配合3D眼镜，主要有几种方法：

1）红蓝3D眼镜：一只眼镜为红色，另一只为蓝色，屏幕上的物体同时具有红色和蓝色但位置有偏差，戴红色镜片的眼睛只能看到蓝色物体，戴蓝色镜片的眼睛只能看到红色物体。



红蓝方式模拟的立体画面缺少色彩效果不好，基本已经淘汰。

2）偏振光眼镜：利用光是横波的特性，屏幕上发射出来的光同时具有两个偏振方向（互相垂直），两边眼镜镜片根据偏振方向分别过滤掉另一个方向的光，从而使两眼看到的影像不同。



（如图，左边的光有两个偏振方向，其中一个能通过中间的镜片传到右边，另一个无法通过）

偏振光方案模拟的立体画面效果不错造价低廉，现在电影院3D电影基本是采用这个方案，它的缺点是两眼各自能看到的光只有一半，这就是在电影院戴上3D眼镜后画面偏暗的原因。

3）快门眼镜：屏幕上交替显示左眼和右眼图像，当显示左眼图像时，左眼镜片为透光状态，右眼镜片为不透光状态，反之同理。

这样屏幕频繁切换画面，两眼就能看到不同的画面，当频率足够快时，人眼就以为看到的是连续的画面。

这个方案效果也比较好，画面不会像偏振光方案一样太暗，但因为需要眼镜配合不停切换左右，眼镜需要供电体积偏大偏重，快门的不停闪动也导致眼睛看久了比较容易累。3D投影仪通常使用这个方案。

3）简单粗暴法：直接把屏幕分成两块，用隔板隔开，让两眼分别看到左右各半边屏幕。

这个方案简单粗暴，成本最低，文末的demo就用这个方法。

二，实时性。

实时性要求画面跟着头部转动，设备必须具有3D陀螺仪等，能判断头部移动方向，实时调整画面，让用户具有临场感。

三，高交互性。

可采取类似现实生活中习以为常的方式来操纵虚拟环境中的物体，比如伸手拾起画面里的东西。

除了这三要素，要营造VR的全面沉浸式体验，还需要环境、动作、音效等元素的有机结合。

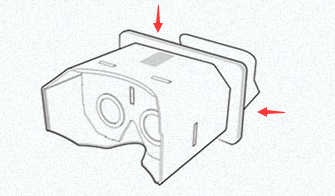
===================以上全是铺垫，以下才是正文===================

说了那么多，那跟前端有什么关系呢？

在VR的三要素里，至少“立体感”和“实时性”是可以在手机上用网页实现的，而高交互性受技术限制，暂时无法在手机上实现，可以期待下“眼部追踪”技术，通过红外线传感器跟踪瞳孔，可以知道你的眼睛盯在虚拟世界里的哪个地方，结合眨眼等动作就可能在手机上实现新形式的交互。

要体验VR网页，需搭配VR眼镜。3D方案采用上诉简单暴力法，把屏幕等分成两份，

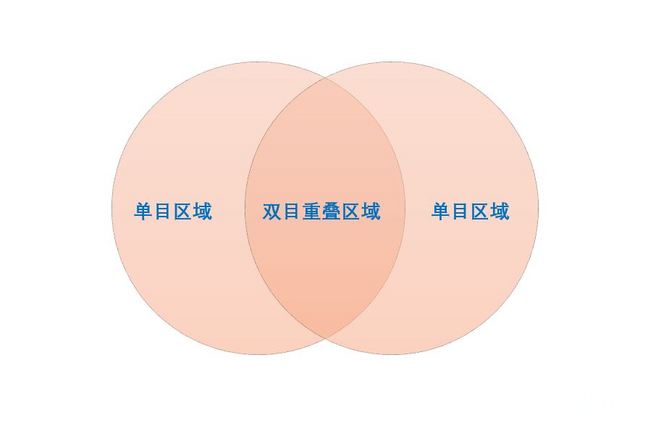
下图为暴风魔镜，很多手工VR眼镜包括google家的都大概是这个样子，原理一样。



红色箭头所示位置插手机，左边两个圆圈是凸透镜，目的是让人眼能看清手机屏幕，否则距离太近人眼无法对焦。

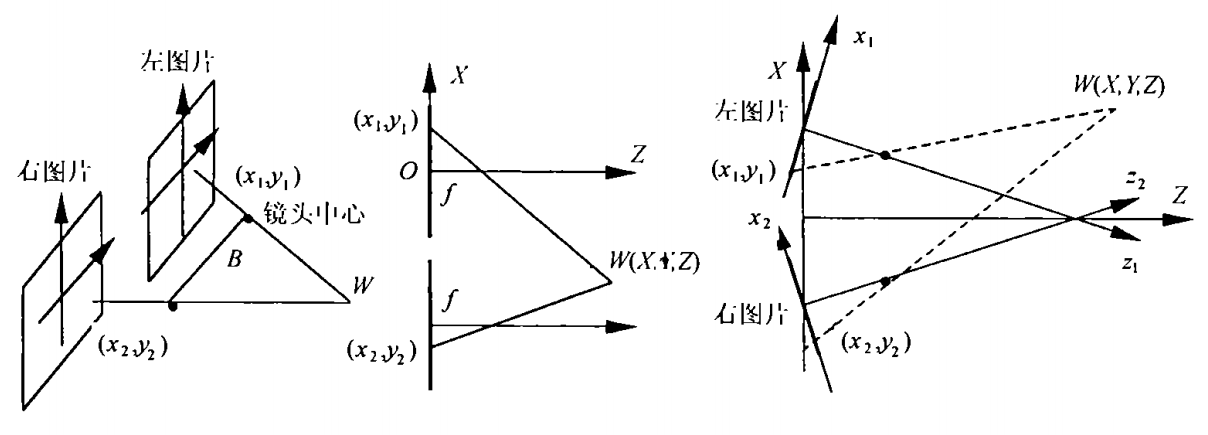
一，网页立体感的实现

如上所述，实现具有立体感的网页，要把屏幕分为两部分，分别为左右眼所见画面：



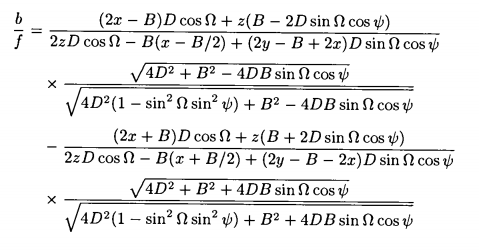
背景画面越远，重叠部分越多，如果是无限远的背景可以让它们完全重合。

物体越远，视差越大，下图是简单的双目立体视觉模型：



同一物体在左右屏幕上的投影坐标不一样，由于人眼是水平的，y值一样，只要调整x的差（也就是视差）就能改变物体的远近。

这个模型比较简单，但实际上眼球会转动等原因，视觉中心和双目视轴的连线不一定对称，实际上视差的计算很复杂，最符合现实的公式可能是这样的：



这里面引入了视觉中心目标点深度及光心偏角和光心极角等参数，但这公式太复杂了，我特意把它缩小了（反正放再大也看不懂），用模型一已经能满足大部分立体感的需求。

总之简单的做法是只要给元素设置不一样的x值，就能产生立体感，左眼看到的物体x值偏左，右眼看到的物体x值偏右，但偏差值不能太大，否则会变成斗鸡眼。我试了在iPhone 6的屏幕上视差值大于50px就有点小斗鸡了。

二，网页画面实时性的实现

画面实时性依靠手机陀螺仪和重力感应实现，单身重构妹纸tinnocui （面南看北斗）已将其封装成组件sensor供调用：



[戳我看tinnocui的重力感应demo](http://idemo.qq.com/view/lab?id=491)

[戳我看tinnocui的重力感应研究](http://km.oa.com/group/20932/articles/show/231251)

在VR网页中用此组件sensor可判断头部运动方向，从而随之改变画面，为了能够实现360°循环旋转，使用背景图和调整background-position值的方式来改变画面。若改进算法，应该可以用性能更优的方式比如调整translate3d值使画面旋转。

不过做的时候遇到一点坑，ios8和ios9/10的的y轴旋转角度范围不同，ios8的y轴旋转角度在0和正负180°之间，而ios9以后是0到正负90°度之间，同时y轴的正负也会影响z轴的正负，因此需要对y轴正负作判断：

if (y < 0){

positionX= (z - 180) \* (transR\_max / range);

positionY= - ( y + 90 ) \* (transH\_max / range);

}

else {

positionX= z \* (transR\_max / range);

positionY= - ( y - 90 ) \* (transH\_max / range);

}

 positionX和positionY分别为画面水平和垂直偏移量。

写到这里忽然想起忘了判断z轴旋转，有空再优化。

最后请看VR小demo：

[VR-太阳眼中的八大行星](http://idemo.qq.com/view/lab?id=660)

ps：该demo仅兼容ios 8 以上，暂未对其它机型作测试。