## 问题描述

之前我们实现边框都是使用border: 1px solid #ccc这样的方法，但是在retina屏上因为设备像素比的不同，在移动设备上的表现也不相同，1px会被渲染成1.5px, 2px, 2.5px, 3px....，在用户体验上略差，现在要解决就是在retina屏幕上也显示1px。

## 实现方案

通过google了解到解决方案倒是不少，有阴影、软图片、缩放、0.5px等，下面就分析一下各个方案的利弊。

PS：目前项目中使用的是软图片的方式

### 1、软图片

// retina上模拟1px, 2px...

// -----------------------------------------------------------------------------

.retina(@top: transparent, @right: transparent, @bottom: transparent, @left: transparent, @w: 1px) {

@media only screen and (-webkit-min-device-pixel-ratio: 2),

only screen and (min-device-pixel-ratio: 2) {

border: none;

background-image:

linear-gradient(180deg, @top, @top 50%, transparent 50%),

linear-gradient(270deg, @right, @right 50%, transparent 50%),

linear-gradient(0deg, @bottom, @bottom 50%, transparent 50%),

linear-gradient(90deg, @left, @left 50%, transparent 50%);

background-size: 100% @w, @w 100%, 100% @w, @w 100%;

background-repeat: no-repeat;

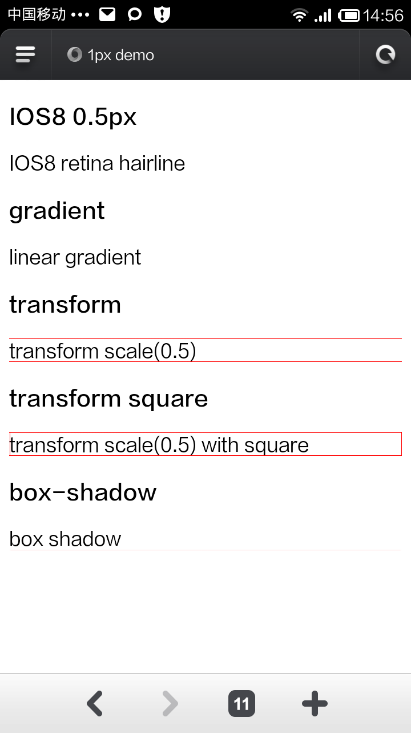
background-position: top, right top, bottom, left top;

}

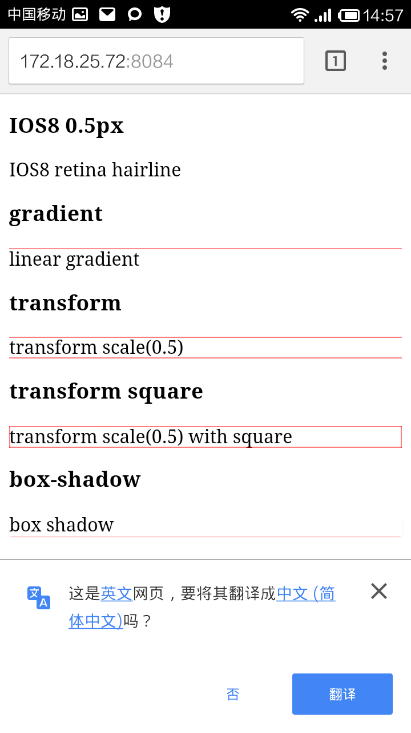
}

额，有些问题这样写，测试小米2上的自带浏览器、手机百度，百度浏览器都显示不出来上边框，只有chrome是好的

小米2手机百度，自带、百度浏览器



小米2chrome



加了浏览器前缀-webkit，修改代码如下：

background-image:

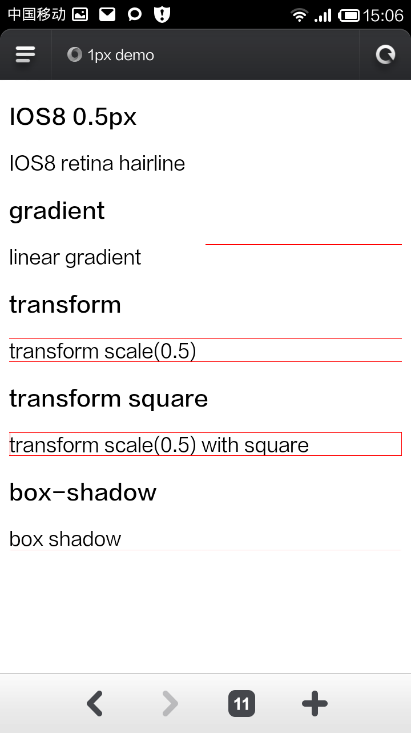
-webkit-linear-gradient(180deg, @top, @top 50%, transparent 50%),

-webkit-linear-gradient(270deg, @right, @right 50%, transparent 50%),

-webkit-linear-gradient(0, @bottom, @bottom 50%, transparent 50%),

-webkit-linear-gradient(90deg, @left, @left 50%, transparent 50%);

表现更怪异，如果只写带有-webkit前缀的代码的话（不包含规范的代码，这个不应该的。。。仅测试~），检测小米2上的自带、百度、chrome浏览器，手机百度表现都一致（iphone4表现也是这样），如下图：



最后研究了半天才发现，是因为加了-webkit的渐变方向和规范定义的不一致（规范定义的0deg是自下而上，在-webkit-前缀下浏览器实现的是自左向右），再次修改代码：

background-image:

-webkit-linear-gradient(270deg, @**top,** @top 50%, transparent 50%),

-webkit-linear-gradient(180deg, @right, @right 50%, transparent 50%),

-webkit-linear-gradient(90deg, @bottom, @bottom 50%, transparent 50%),

-webkit-linear-gradient(0, @left, @left 50%, transparent 50%);

background-image:

linear-gradient(180deg, @**top,** @top 50%, transparent 50%),

linear-gradient(270deg, @right, @right 50%, transparent 50%),

linear-gradient(0deg, @bottom, @bottom 50%, transparent 50%),

linear-gradient(90deg, @left, @left 50%, transparent 50%);

这样就OK了！

优点：相比使用缩放，不会依赖DOM，纯CSS  
缺点：

* 很明显代码特别长；
* 无法实现圆角
* 使用时可能需要配合padding，如设置子元素的背景可能会挡住父元素所设置的1px软图片；
* 如果有背景颜色，要写成background-color，不然会不小心覆盖掉；
* 对于非retina屏，需要写border: 1px solid #f00进行适配

### 2、缩放

.transform-scale {

position: relative;

&:after,

&:before {

content: '';

position: absolute;

left: 0;

top: 0;

height: 1px;

width: 100%;

-webkit-transform: scaleY(0.5);

transform: scaleY(0.5);

background: #f00;

}

&:after {

top: auto;

bottom: 0;

}

}

优点：实现一条线倒是很简单  
缺点：

* 无法实现圆角；
* 四条边框比较纠结

### 3、阴影

.shadow {

-webkit-box-shadow:0 1px 1px -1px rgba(255, 0, 0, 0.5);

box-shadow:0 1px 1px -1px rgba(255, 0, 0, 0.5);

}

没觉得这个方法好用，模拟的效果差强人意，颜色也不好配置。。。

### 4、0.5px

// IOS8 hairline

.hairline(@color, @style:solid) {

@media (-webkit-min-device-pixel-ratio: 2) {

border: 0.5px @style @color;

}

}

优点：“原生”，支持圆角~  
缺点：目前只有IOS8+才支持，在IOS7以下，安卓系统都是显示为0px

我们可以检测UA是不是IOS8来兼容：

**var** ua = navigator.userAgent;

**if** (/iPhone|iPod|iPad/i.test(navigator.userAgent)) {

**var** version = ua.match(/OS\s\*(\d+)\_(\d+)\_?(\d+)?/i);

version = verstion && version[1];

**if** (version > 7) {

}

}

### 5、viewport&&rem

[再谈mobile web retina 下 1px 边框解决方案](http://www.ghugo.com/css-retina-hairline/)这篇文章介绍了viewport结合rem，这个方案就是让我们像以前写1倍像素那样写页面，感觉有点别扭。。。  
如devicePixelRatio=2下设置

<meta name="viewport" content="initial-scale=0.5, maximum-scale=0.5, minimum-scale=0.5, user-scalable=no">

再设置rem，假设header的高度是30px（设备像素比为1）：

html {

font-size: 20px;

}

header {

height: 3rem;

}

### 6、border-image

又见[神文](https://github.com/maxzhang/maxzhang.github.com/issues/4)，使用border-image模拟。

使用的图片如下：

http://wiki.baidu.com/download/attachments/73308126/QQ%E6%88%AA%E5%9B%BE20150413125428.png?version=3&modificationDate=1428901159600&api=v2

.border-image-1px {

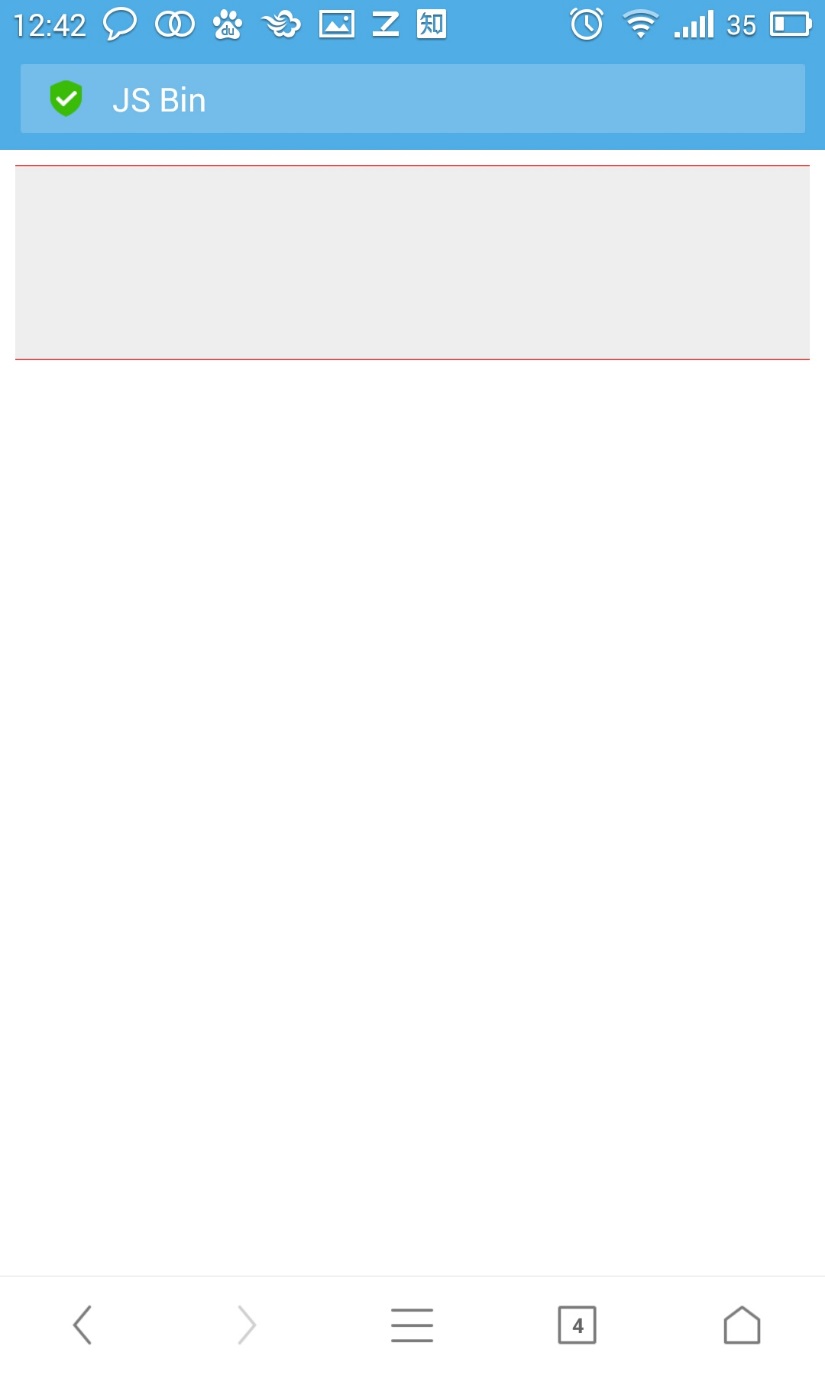
border-width: 1px 0px;

-webkit-border-image: url("border.png") 2 0 stretch;

border-image: url("border.png") 2 0 stretch;

}

实际效果如下，魅族3(M356，设备像素比 2.5)下的UC浏览器：



优点：额，，，  
缺点：

* 边框颜色更改不灵活；
* 放到PS里面看边框，是有点模糊的(因为带有颜色部分是1px，在retina屏幕上拉伸到2px肯定会有点模糊)

## 总结

1、0.5px，相信浏览器肯定是会慢慢支持的；目前而言，如果能用的话，可以hack一下；

2、阴影的方案不建议使用（用了你就知道。。。），border-image也是种方案，但是效果不是很好啊。。。

3、背景图片和缩放可以在项目中配合使用，如单个线条使用缩放，四条框用背景图片模拟，额，如果要圆角的话，无能无力了