由于HTML5 没有提供开箱即用的手势系统，但是提供了更底层一些的对 touch 事件的监听。基于此，我们可以做出自己的手势库。

常用的 HTML5 手势可以分为两类，单点手势和两点手势。单点手势有 tap(单击)，double tap(双击)，long tap(长按)，swipe(挥)，move(移动)。两点手势有 pinch(缩放)，rotate(旋转)。

**移动：**

关于移动手势检测就是在每次touchmove事件发生时，把两个位移点之间的坐标位置相减，就可以了。

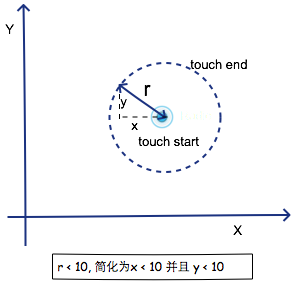
**单击：**

手势检测的关键是用 touchstart，touchmove，touchend 三个事件对手势进行分解。

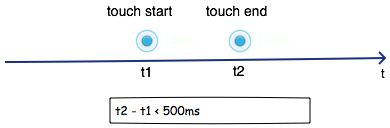
那么怎么分解单击事件呢?

1.在 touchstart 发生时进入单击检测，只有一个接触点。因为单击事件限制为一个手指的动作。

2.没有发生 touchmove 事件或者 touchmove 在一个很小的范围(如下图)。限制 touchmove 在一个很小范围，是为了给用户一定的冗余空间，因为不能保证用户手指在接触屏幕的时候不发生轻微的位移。



1. touchend 发生在 touchstart后的很短时间内(如下图)。这个时间段的阈值是毫秒级，用来限制手指和屏幕接触的时间。因为单击事件从开始到结束是很快的。



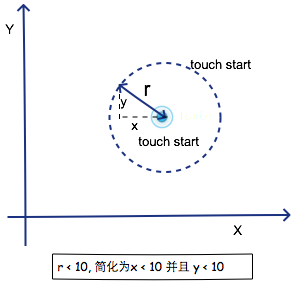
　有了上面的流程，就可以开始实现 tap 事件监测了。

**双击：**

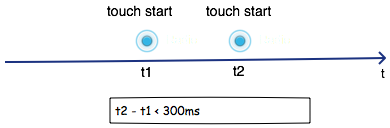
和单击一样，双击事件也需要我们对手势进行量化分解。

1.双击事件是一个手指的行为。所以在 touchstart 时，我们要判断此时屏幕有几个接触点。

2.双击事件中包含两次独立的单击行为。理想情况下，这两次点击应该落在屏幕上的同一个点上。为了给用户一定的冗余空间，将两次点击的坐标点距离限制在10个像素以内。



  3.　双击事件本质是两次快速的单击。也即是说，两次点击的间隔时间很短。通过一定的测试量化后，我们把两次单击的时间间隔设为300毫秒。



注意双击事件中我们检测了相邻两个 touchstart 事件的位移和时间间隔。

**长按：**

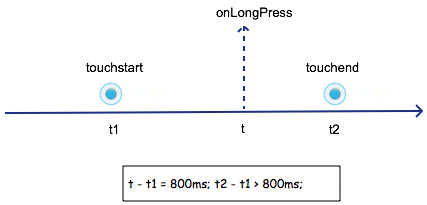
长按应该是最容易分解的手势。我们可以这样分解：在 touchstart 发生后的很长一段时间内，如果没有发生 touchmove 或者 touchend 事件，那么就触发长按手势。

1.长按是一个手指的行为，需要检测屏幕上是否只有一个接触点。

2.如果手指在空间上发生了移动，那么长按事件取消。

3.如果手指在屏幕上停留的时间超过800ms，那么触发长按手势。

4.如果手指在屏幕上停留的时间小于800ms，也即 touchend 在 touchstart 发生后的800ms内触发，那么长按事件取消。

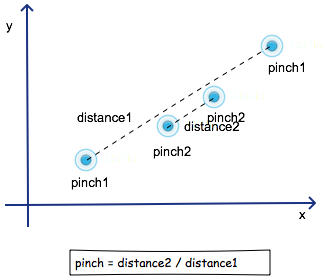


**缩放：**

缩放是一个非常有趣的手势，还记得第一代iPhone双指缩放图片给你带来的震撼吗?虽然如此，缩放手势的检测却相对简单。

1.缩放是两个手指的行为，需要检测屏幕上是否有两个接触点。

2.缩放比例的量化，是通过两次缩放行为之间的距离的比值得到，如下图。



所以缩放的核心是获取两个接触点之间的直线距离。

**旋转：**

旋转手势需要检测两个比较重要的值，一是旋转的角度，二是旋转的方向(顺时针或逆时针)。其中旋转角度和方向的计算需要通过向量的计算来获取，首先，需要获取向量的旋转方向和角度。

