

[首页](#) [资讯](#) [精华](#) [论坛](#) [问答](#) [博客](#) [专栏](#) [群组](#) [更多](#) ▼

[您还未登录！](#) [登录](#) [注册](#)



## Just Code

- [博客](#)
- [微博](#)
- [相册](#)
- [收藏](#)
- [留言](#)
- [关于我](#)



## UITouch 触摸事件处理（实例）

博客分类：

- [Phone / IOS / Objective-C / Swift](#)

### 1. UITouch 的主要方法：

C代码  

1. - (void)touchesBegan:(NSSet \*)touches withEvent:(UIEvent \*)event;
2. - (void)touchesMoved:(NSSet \*)touches withEvent:(UIEvent \*)event;
3. - (void)touchesEnded:(NSSet \*)touches withEvent:(UIEvent \*)event;
4. - (void)touchesCancelled:(NSSet \*)touches withEvent:(UIEvent \*)event;
- 5.
6. - (void)motionBegan:(UIEventSubtype)motion withEvent:  
(UIEvent \*)event \_\_OSX\_AVAILABLE\_STARTING(\_\_MAC\_NA,\_\_IPHONE\_3\_0);
7. - (void)motionEnded:(UIEventSubtype)motion withEvent:  
(UIEvent \*)event \_\_OSX\_AVAILABLE\_STARTING(\_\_MAC\_NA,\_\_IPHONE\_3\_0);
8. - (void)motionCancelled:(UIEventSubtype)motion withEvent:  
(UIEvent \*)event \_\_OSX\_AVAILABLE\_STARTING(\_\_MAC\_NA,\_\_IPHONE\_3\_0);

## 2. 触摸时，图片移动（实例）

### C代码

```
1. - (void)viewDidLoad
2. {
3.     UIImageView *image = [[UIImageView alloc] initWithFrame:CGRectMake(20.0, 50.0, 45.0, 45.0)];
4.     image.image = [UIImage imageNamed:@"baby.png"];
5.     image.tag = 100;
6.     [self.view addSubview:image];
7.
8.     [super viewDidLoad];
9.     // Do any additional setup after loading the view, typically from a nib.
10. }
11.
12. -(void)touchesEnded:(NSSet *)touches withEvent:(UIEvent *)event
13. {
14.     UITouch *touch = [touches anyObject];
15.     UIImageView *view1 = (UIImageView*)[self.view viewWithTag:100];
16.     CGPoint point = [touch locationInView:self.view];
17.     CGRect frame = view1.frame;
18.     frame.origin = point;
19.     view1.frame = frame;
20. }
```

iphone/ipad无键盘的设计是为屏幕争取更多的显示空间，大屏幕在观看图片、文字、视频等方面为用户带来了更好的用户体验。而触摸屏幕是iOS设备接受用户输入的主要方式，包括单击、双击、拨动以及多点触摸等，这些操作都会产生触摸事件。

在Cocoa中，代表触摸对象的类是UITouch。当用户触摸屏幕后，就会产生相应的事件，所有相关的UITouch对象都被包装在事件中，被程序交由特定的对象来处理。UITouch对象直接包括触摸的详细信息。

UITouch类中包含5个属性：

**window**：触摸产生时所处的窗口。由于窗口可能发生变化，当前所在的窗口不一定是从一开始的窗口。

**view**：触摸产生时所处的视图。由于视图可能发生变化，当前视图也不一定是最初的视图。

**tapCount**：轻击（Tap）操作和鼠标的单击操作类似，tapCount表示短时间内轻击屏幕的次数。因此可以根据tapCount判断单击、双击或更多的轻击。

**timestamp**：时间戳记录了触摸事件产生或变化时的时间。单位是秒。

**phase**：触摸事件在屏幕上有一个周期，即触摸开始、触摸点移动、触摸结束，还有中途取消。而通过phase可以查看当前触摸事件在一个周期中所处的状态。phase是UITouchPhase类型的，这是一个枚举配型，包含了

- UITouchPhaseBegan（触摸开始）
- UITouchPhaseMoved（接触点移动）
- UITouchPhaseStationary（接触点无移动）
- UITouchPhaseEnded（触摸结束）
- UITouchPhaseCancelled（触摸取消）

UITouch类中包含如下成员函数：

- (CGPoint)locationInView:(UIView \*)view：函数返回一个CGPoint类型的值，表示触摸在view这个视图上的位置，这里返回的位置是针对view的坐标系的。调用时传入的view参数为空的话，返回的时触摸点在整个窗口的位置。

- (CGPoint)previousLocationInView:(UIView \*)view：该方法记录了前一个坐标值，函数返回也是一个CGPoint类型的值，表示触摸在view这个视图上的位置，这里返回的位置是针对view的坐标系的。调用时传入的view参数为空的话，返回的时触摸点在整个窗口的位置。

当手指接触到屏幕，不管是单点触摸还是多点触摸，事件都会开始，直到用户所有的手指都离开屏幕。期间所有的UITouch对象都被包含在UIEvent事件对象中，由程序分发给处理者。事件记录了这个周期中所有触摸对象状态的变化。

只要屏幕被触摸，系统就会报若干个触摸的信息封装到UIEvent对象中发送给程序，由管理程序UIApplication对象将事件分发。一般来说，事件将被发给主窗口，然后传给第一响应者对象(FirstResponder)处理。

关于响应者的概念，通过以下几点说明：

### 响应者对象（Response object）

响应者对象就是可以响应事件并对事件作出处理。在iOS中，存在UIResponder类，它定义了响应者对象的所有方法。UIApplication、UIView等类都继承了UIResponder类，UIWindow和UIKit中的控件因为继承了UIView，所以也间接继承了UIResponder类，这些类的实例都可以当作响应者。

## 第一响应者（First responder）

当前接受触摸的响应者对象被称为第一响应者，即表示当前该对象正在与用户交互，它是响应者链的开端。

## 响应者链（Responder chain）

响应者链表示一系列的响应者对象。事件被交由第一响应者对象处理，如果第一响应者不处理，事件被沿着响应者链向上传递，交给下一个响应者（next responder）。一般来说，第一响应者是个视图对象或者其子类对象，当其被触摸后事件被交由它处理，如果它不处理，事件就会被传递给它的视图控制器对象（如果存在），然后是它的父视图（superview）对象（如果存在），以此类推，直到顶层视图。接下来会沿着顶层视图（top view）到窗口（UIWindow对象）再到程序（UIApplication对象）。如果整个过程都没有响应这个事件，该事件就被丢弃。一般情况下，在响应者链中只要由对象处理事件，事件就停止传递。但有时候可以在视图的响应方法中根据一些条件判断来决定是否需要继续传递事件。

## 管理事件分发

视图对触摸事件是否需要作回应可以通过设置视图的userInteractionEnabled属性。默认状态为YES，如果设置为NO，可以阻止视图接收和分发触摸事件。除此之外，当视图被隐藏（setHidden: YES）或者透明（alpha值为0）也不会收事件。不过这个属性只对视图有效，如果想要整个程序都步响应事件，可以调用UIApplication的beginIgnoringInteractionEvents方法来完全停止事件接收和分发。通过endIgnoringInteractionEvents方法来恢复让程序接收和分发事件。

如果要让视图接收多点触摸，需要设置它的multipleTouchEnabled属性为YES，默认状态下这个属性值为NO，即视图默认不接收多点触摸。

首先触摸的对象是视图，而视图的类UIView继承了UIResponder类，但是要对事件作出处理，还需要重写UIResponder类中定义的事件处理函数。根据不同的触摸状态，程序会调用相应的处理函数，这些函数包括以下几个：

```
-(void)touchesBegan:(NSSet *)touches withEvent:(UIEvent *)event;  
-(void)touchesMoved:(NSSet *)touches withEvent:(UIEvent *)event;  
-(void)touchesEnded:(NSSet *)touches withEvent:(UIEvent *)event;  
-(void)touchesCancelled:(NSSet *)touches withEvent:(UIEvent *)event;
```

当手指接触屏幕时，就会调用touchesBegan:withEvent方法；

当手指在屏幕上移时，动就会调用touchesMoved:withEvent方法；

当手指离开屏幕时，就会调用touchesEnded:withEvent方法；

当触摸被取消（比如触摸过程中被来电打断），就会调用touchesCancelled:withEvent方法。而这几个方法被调用时，正好对应了UITouch类中phase属性的4个枚举值。

上面的四个事件方法，在开发过程中并不要求全部实现，可以根据需要重写特定的方法。对于这4个方法，都有两个相同的参数：NSSet类型的touches和UIEvent类型的事件。其中touches表示触摸产生的所有UITouch对象，而event表示特定的事件。因为UIEvent包含了整个触摸过程中所有的触摸对象，因此可以调用allTouches方法获取该事件内所有的触摸对象，也可以调用touchesForView: 或者 touchesForWindows: 取出特定视图或者窗口上的触摸对象。在这几个事件中，都可以拿到触摸对象，然后根据其位置，状态，时间属性做逻辑处理。

例如：

C代码  

```
1. -(void)touchesEnded:(NSSet *)touches withEvent:(UIEvent *)event
2. {
3.     UITouch *touch = [touches anyObject];
4.     if(touch.tapCount == 2)
5.     {
6.         self.view.backgroundColor = [UIColor redColor];
7.     }
8. }
```

上面的例子说明在触摸手指离开后，根据tapCount点击的次数来设置当前视图的背景色。不管是一个手指还是多个手指，轻击操作都会使每个触摸对象的tapCount加1，由于上面的例子不需要知道具体触摸对象的位置或时间等，因此可以直接调用touches的anyObject方法来获取任意一个触摸对象然后判断其tapCount的值即可。

检测tapCount可以放在touchesBegan也可以touchesEnded，不过一般后者跟准确，因为touchesEnded可以保证所有的手指都已经离开屏幕，这样就不会把轻击动作和按下拖动等动作混淆。

轻击操作很容易引起歧义，比如当用户点了一次之后，并不知道用户是想单击还是只是双击的一部分，或者点了两次之后并不知道用户是想双击还是继续点击。为了解决这个问题，一般可以使用“延迟调用”函数。

例如：

C代码  

```
1. -(void)touchesEnded:(NSSet *)touches withEvent:(UIEvent *)event
2. {
3.     UITouch *touch = [touches anyObject];
4.     if(touch.tapCount == 1)
5.     {
6.         [self performSelector:@selector(setBackground:) withObject:[UIColor blueColor] afterDelay:2];
7.         self.view.backgroundColor = [UIColor redColor];
8.     }
9. }
```

上面代码表示在第一次轻击之后，没有直接更改视图的背景属性，而是通过performSelector: withObject: afterDelay: 方法设置2秒中后更改。

C代码  

```
1. -(void)touchesEnded:(NSSet *)touches withEvent:(UIEvent *)event
2. {
3.     UITouch *touch = [touches anyObject];
4.     if(touch.tapCount == 2)
5.     {
6.         [NSObject cancelPreviousPerformRequestsWithTarget:self selector:@selector(setBackground:) object:
7.         [UIColor redColor]];
8.         self.view.backgroundColor = [UIColor redColor];
9.     }
10. }
```

双击就是两次单击的组合，因此在第一次点击的时候，设置背景色的方法已经启动，在检测到双击的时候先要把先前对应的方法取消掉，可以通过调用NSObject类的cancelPreviousPerformRequestWithTarget:selector:object方法取消指定对象的方法调用，然后调用双击对应的方法设置背景色为红色。

下面举个例子创建可以拖动的视图，这个主要通过触摸对象的位置坐标来实现。因此调用触摸对象的locationInView: 方法即可。

例如：

C代码  

```
1. CGPoint originalLocation;
2. -(void)touchesBegan:(NSSet *)touches withEvent:(UIEvent *)event
3. {
4.     UITouch *touch = [touches anyObject];
5.     originalLocation = [touch locationInView:self.view];
6. }
```



```

7.
8. -(void)touchesMoved:(NSSet *)touches withEvent:(UIEvent *)event
9. {
10.     UITouch *touch = [touches anyObject];
11.     CGPoint currentLocation = [touch locationInView:self.view];
12.     CGRect frame = self.view.frame;
13.     frame.origin.x += currentLocation.x-originalLocation.x;
14.     frame.origin.y += currentLocation.y-originalLocation.y;
15.     self.view.frame = frame;
16. }

```

这里先在touchesBegan中通过[touch locationInView:self.view]获取手指触摸在当前视图上的位置，用CGPoint变量记录，然后在手指移动事件touchesMoved方法中获取触摸对象当前位置，并通过于与原始位置的差值计算出移动偏移量，再设置当前视图的位置。

部分来源: <http://www.cnblogs.com/spiritstudio/archive/2011/05/24/2054907.html>

实例:



C代码  

```

1. //对画面进行单次点击时所触发的函数式
2. -(void)touchesBegan:(NSSet *)touches withEvent:(UIEvent *)event {
3.
4. //宣告一个UITouch的指标来存放事件触发时所撷取到的状态
5. UITouch *touch = [[event allTouches] anyObject];
6.
7. //将XY轴的座标资讯正规化后输出
8. touchX.text = [NSString stringWithFormat:@"%0.0f", [touch locationInView:touch.view].x];
9. touchY.text = [NSString stringWithFormat:@"%0.0f", [touch locationInView:touch.view].y];
10. }

```

C代码  

```

1. //对画面进行拖曳动做时所触发的函数式
2. -(void)touchesMoved:(NSSet *)touches withEvent:(UIEvent *)event {
3.
4. //宣告一个UITouch的指标来存放事件触发时所撷取到的状态

```



```
5. UITouch *touch = [[event allTouches] anyObject];
6.
7. //将XY轴的坐标资讯正规化后输出
8. moveX.text = [NSString stringWithFormat:@"%0.0f", [touch locationInView:touch.view].x];
9. moveY.text = [NSString stringWithFormat:@"%0.0f", [touch locationInView:touch.view].y];
10. }
```

## C代码

```
1. //手指离开画面（结束操作）时所触发的函数
2. -(void)touchesEnded:(NSSet *)touches withEvent:(UIEvent *)event {
3.
4. //宣告一个UITouch的指标来存放事件触发时所撷取到的状态
5. UITouch *touch = [[event allTouches] anyObject];
6.
7. //取得并输出连点资讯，tapCount可保留一定时间内的连点次数
8. tapCountLabel.text = [NSString stringWithFormat:@"%d", [touch tapCount]];
9. }
```

来源: <http://furnacedigital.blogspot.com/2010/12/touch-panel.html>



分享到:  

[UIActivityIndicatorView、UIProgressView ... | UIDeviceOrientation 和 UIInterfaceOrien ...](#)

- 2012-04-02 21:52
- 浏览 71180
- [评论\(1\)](#)
- 分类: [移动开发](#)
- [相关推荐](#)

参考知识库

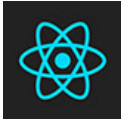


[区块链知识库](#) 168 关注 / 72 收录





[深度学习知识库](#) 1419 关注 / 212 收录



[React Native知识库](#) 1046 关注 / 492 收录

# iOS

[iOS知识库](#) 933 关注 / 1146 收录

评论

1 楼 [hbb19900618](#) 2016-05-13

我想问一下 touchesMoved 能否方便实现 视频播放时左右滑动屏幕快进、快退的功能，当然要准确的区别上下滑动的方向，这个方向能判断的准吗？一般是如何实现的？用手势去实现吗

发表评论



[您还没有登录,请您登录后再发表评论](#)



天梯梦

- 浏览: 7362405 次
- 性别: ♀
- 来自: 洛杉矶
- 我现在离线

最近访客

[更多访客>>](#)



[chrislee1982](#)



[littlesheep](#)

[tkintc](#)[cdphantom](#)

## 文章分类

- [全部博客 \(1999\)](#)
- [Php / Pear / Mysql / Node.js \(378\)](#)
- [Javascript / JQuery / Bootstrap / Web \(435\)](#)
- [Phone / IOS / Objective-C / Swift \(137\)](#)
- [Ubuntu / Mac / Github / Aptana / Nginx / Shell / Linux \(335\)](#)
- [Perl / Koha / Ruby / Markdown \(8\)](#)
- [Java / Jsp \(12\)](#)
- [Python 2 / Wxpython \(25\)](#)
- [Codeigniter / CakePHP \(32\)](#)
- [Div / Css / XML / HTML5 \(179\)](#)
- [WP / Joomla! / Magento / Shopify / Drupal / Moodle / Zimbra \(275\)](#)
- [Apache / VPN / Software \(31\)](#)
- [AS3.0/2.0 / Flex / Flash \(45\)](#)
- [Smarty \(6\)](#)
- [SEO \(24\)](#)
- [Google / Facebook / Pinterest / SNS \(80\)](#)
- [Tools \(22\)](#)

## 社区版块

- [我的资讯 \(0\)](#)
- [我的论坛 \(16\)](#)
- [我的问答 \(0\)](#)

## 存档分类

- [2016-09 \(5\)](#)
- [2016-08 \(50\)](#)
- [2016-07 \(70\)](#)
- [更多存档...](#)

## 评论排行榜

- [Proxy、SSH 和 VPN 的区别](#)

## 最新评论

- [masuweng](#): 帮助到我了。。尤其是第一个参数，我就是这直接载入 ...  
[轻量级的 JavaScript 弹出框脚本: TinyBox](#)

- [一只电兔子](#): 谢谢,辛苦啦  
[给开发者准备的 10 款最好的 jQuery 日历插件](#)
- [youxianyen](#): ...  
[微信红包随机生成算法 \(PHP版\)](#)
- [潇洒欧兄](#): 有什么方法,把下载按钮去掉  
[jquery在线预览PDF文件,打开PDF文件](#)
- [aiq缘诀录](#): [b][b][b][b][b][b][b][b][b][b] ...  
[OAuth2.0认证和授权原理](#)



佘山租房



上海招聘会



恒大佘山首府



移动硬盘

声明: ITeye文章版权属于作者,受法律保护。没有作者书面许可不得转载。若作者同意转载,必须以超链接形式标明文章原始出处和作者。

© 2003-2016 ITeye.com. All rights reserved. [ 京ICP证110151号 京公网安备110105010620 ]