iOS (http://lib.csdn.net/base/ios)

iOS (http://lib.csdn.net/base/ios) － 经典场景 (http://lib.csdn.net/ios/node/743) －
定位导航 (http://lib.csdn.net/ios/knowledge/1690)

👁 193　　💬 1

ios 视频播放器：AVPlayer（附：seektotime精准定位）

作者：Xoxo_x (http://my.csdn.net/Xoxo_x)

网上有很多AVPlayer的播放器，但对于seekToTime，时间跳转都不是很精确，尤其是在使用定时器的情况下，本人找了半天，都是误人子弟，所以写了这篇精准定位的方法：self.player seekToTime(time, toleranceBefore: kCMTimeZero, toleranceAfter: kCMTimeZero)

```
1   self.player seekToTime(time, toleranceBefore: kCMTimeZero, toleranceAfter: kCMTime
```

```
1   本文重点就是这句话，其他的是一些关于实现播放器的代码，大家可以不看
```

一直想做一个用H5写的播放器，实现跨平台，解决苹果不能播放flv的问题,如哪位大神有经验可以指导一下
Xoxo_x@126.com

ObjectIve - c:

```
1       添加库文件：AVFoundation.framework
2       包含头文件：#import <AVFoundation/AVFoundation.h>
```

简单实用:通过avplayerItem创建

```
1    // 加载网络视频
2        NSURL *movieUrl = [NSURL URLWithString:@"http://bos.nj.bpc.baidu.com/tieba-sma
3
4        AVPlayerItem *playerItem = [AVPlayerItem playerItemWithURL:movieUrl];
5
6        // 创建 AVPlayer 播放器
7        AVPlayer *player = [AVPlayer playerWithPlayerItem:playerItem];
8
9        // 将 AVPlayer 添加到 AVPlayerLayer 上
10       AVPlayerLayer *playerLayer = [AVPlayerLayer playerLayerWithPlayer:player];
11
12       // 设置播放页面大小
13       playerLayer.frame = CGRectMake(10, 30, self.view.bounds.size.width - 20, 200);
14
15       // 设置画面缩放模式
16       playerLayer.videoGravity = AVLayerVideoGravityResizeAspect;
17
18       // 在视图上添加播放器
19       [self.view.layer addSublayer:playerLayer];
20
21       // 开始播放
22       [player play];
```
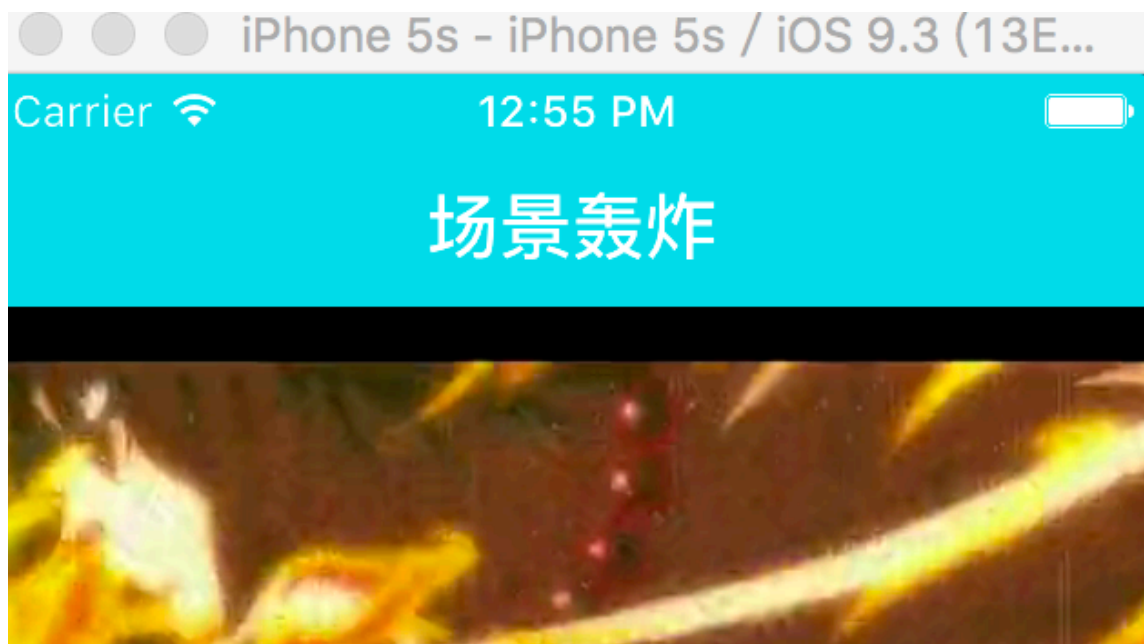
```
1   AVPlayer 本身并不能显示视频，而且它也不像 MPMoviePlayerController 有一个view 属性。如果 AV
2   有了 AVPlayerLayer 之添加到控制器视图的 layer 中即可。要使用 AVPlayer 首先了解一下几个常用的
3
4   AVAsset：主要用于获取多媒体信息，是一个抽象类，不能直接使用。
5   AVURLAsset：AVAsset 的子类，可以根据一个URL路径创建一个包含媒体信息的AVURLAsset对象。
6   AVPlayerItem：一个媒体资源管理对象，管理者视频的一些基本信息和状态，一个AVPlayerItem对应着一个
```
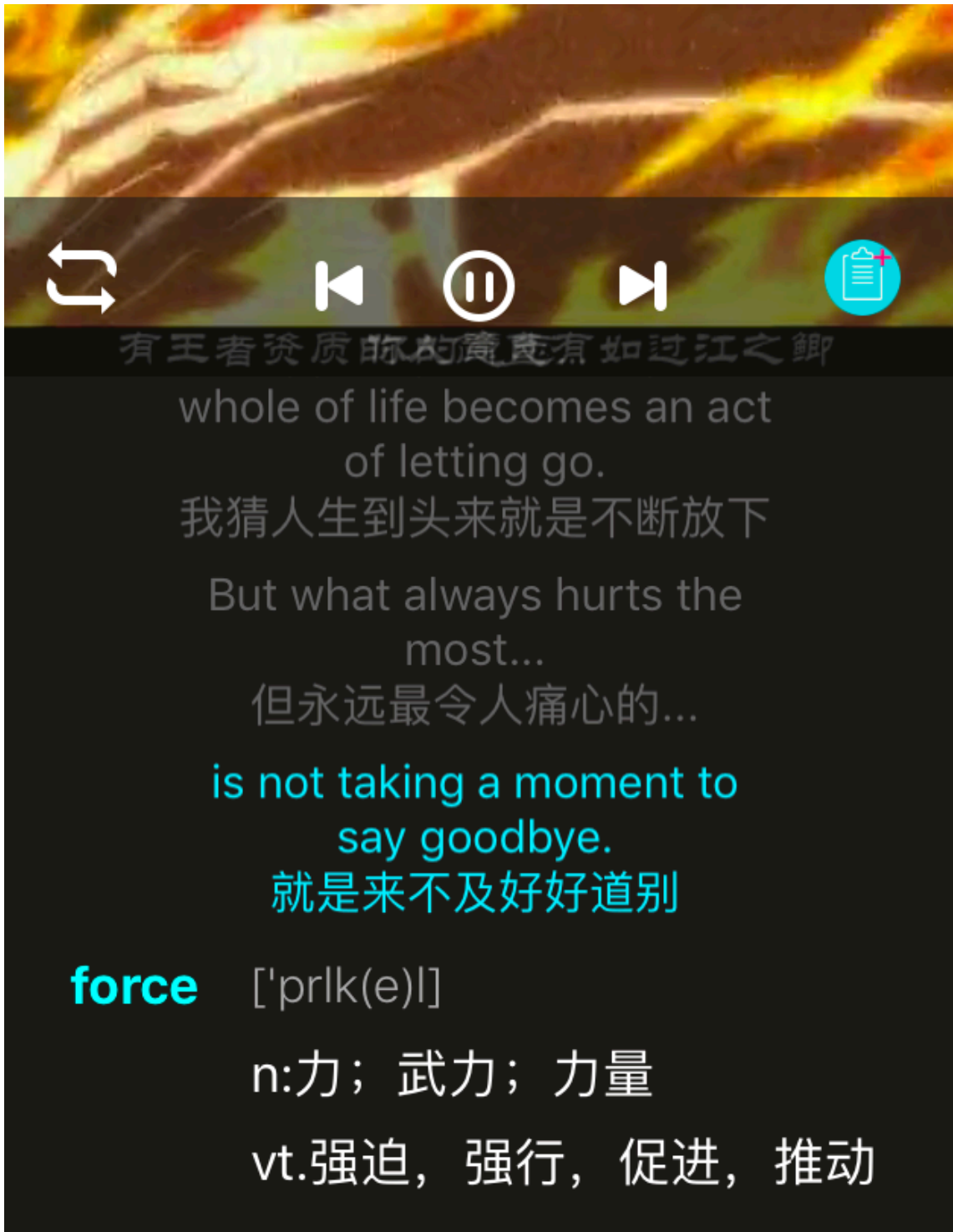
本文是swift 开发的. 使用

https://objectivec2swift.com/#/home/main (https://objectivec2swift.com/#/home/main)

可以转成Objective-C

效果图

```
1   MeidaPlayer 框架中的 MPMoviePlayerController 类和 MPMoviePlayerViewController 类是 iO
```

介绍：本博客介绍如何使用AVPlayer进行播放，暂停，视频切换，循环播放，跳转到指定时间，并精准定位，包含对AVPlayer播放器的监听等

添加播放器，播放视频

```
1    /*
2    根据playerItem创建播放器player
3    */
4            if player == nil {
5                playerItem = AVPlayerItem(URL: movieUrl)
6                player = AVPlayer(playerItem: playerItem)
7    /*
8    创建一个播放器层 AVPlayerLayer 用于展示
9    */
10               let playerLayer:AVPlayerLayer = AVPlayerLayer(player: player)
11               playerLayer.frame = self.view.frame
12               playerLayer.videoGravity = AVLayerVideoGravityResizeAspect
13               self.view.layer.addSublayer(playerLayer)
14   /*
15   设置playerLayer.videoGravity
16   */
17
18               playerLayer.videoGravity = AVLayerVideoGravityResizeAspectFill
19    /*
20   设置播放完成通知 Swift的方法有些特殊和oc不同#selector(SituationalEnglishVc.handleTapGest
21   */            NSNotificationCenter.defaultCenter().addObserver(self,
22                                                       selector: #selector(p
23                                                       name: AVPlayerItemDid
24                                                       object: playerItem)
25           addSomeObserver()
26           playState = true
27           player.status
28           let tapGesture = UITapGestureRecognizer(target: self, action: #selecto
29           tapGesture.numberOfTapsRequired = 1
30           let tapView = UIView.init(frame: playerLayer.frame)
31           avplayerView.addSubview(tapView)
32           tapView.addGestureRecognizer(tapGesture)
33           player.play()
34           /*
35           创建定时器，用于在特定时间跳转
36           */
37   getCurrentTimer = NSTimer.scheduledTimerWithTimeInterval(0.1, target: self, select
38           NSRunLoop.mainRunLoop().addTimer(getCurrentTimer, forMode: NSDefaultRu
39
40       }
```

暂停播放

```
1    player.pause()
```

切换电影

```
1    movieUrl = NSURL(string: "http://bos.nj.bpc.baidu.com/tieba-smallvideo/11772_3c435
2    playerItem = AVPlayerItem(URL: movieUrl)
3    //用这个方法就可以切换
4    player.replaceCurrentItemWithPlayerItem(playerItem)
```

备注：这是电影神灯的界面，神灯产品用户体验群：573431381 大家可以下载一下，看看有什么技术可以探讨一下

seekToTime精准定位

```
1  关于这个seektotime，是个坑，我跳了好多次，比如我要调到10秒，结果跳到了6秒，我要调15秒，跳到了18
```

```
1  我们不能用这个方法self.player.seekToTime(Time: CMTime)
2  使用这个方法，通过设置偏差tolerance，来精确设定的时间是多少，很管用
3    self.player.seekToTime(time, toleranceBefore: kCMTimeZero, toleranceAfter: kCMTi
```

还有一些冗余代码是对tableview的visibleCells进行操作的大家可以不看：如下

```
1  func scrollToCurrentRow(idx:Int){
2
3      currentRow = idx
4      if self.chineseArray.count > 0 {
5          let indexPath = NSIndexPath(forRow: idx, inSection: 0)
6          for item in sentenceTb.visibleCells{
7              let cell = item as! SentenceTableViewCell
8              cell.EnglishLb.textColor = colorFromHex("666666")
9              cell.ChineseLb.textColor = colorFromHex("666666")
10         }
11
12         let cell = sentenceTb.cellForRowAtIndexPath(indexPath) as? SentenceTab
13         if cell != nil {
14             cell!.EnglishLb.textColor = colorFromHex("5eecff")
15             cell!.ChineseLb.textColor = colorFromHex("5eecff")
16         }
17
18         switch currentRow {
19         case chineseArray.count - 1:
20             sentenceTb.scrollToRowAtIndexPath(indexPath, atScrollPosition: UIT
21             break
22         case chineseArray.count - 2:
23             sentenceTb.scrollToRowAtIndexPath(indexPath, atScrollPosition: UIT
24             break
25         default:
26             sentenceTb.scrollToRowAtIndexPath(indexPath, atScrollPosition: UIT
27             break
28         }
29     }
30 }
```

最后附上所有代码

```
1  //
2  //  SituationalEnglishVc.swift
3  //  eTimeMachine
4  //
5  //  Created by fsk-0-1-n on 16/9/18.
6  //  Copyright © 2016年 smartit. All rights reserved.
7  //
8
9  import UIKit
10
11 class SituationalEnglishVc: UIViewController,UITableViewDelegate,UITableViewDataSo
12     var movieUrl : NSURL!
```

```swift
13        var player : AVPlayer!
14        var playState : Bool!
15        var playerItem : AVPlayerItem! = nil
16        var getCurrentTimer : NSTimer! = nil
17        var replayTimer : NSTimer! = nil
18        var wordStart : CGFloat!
19
20        var replayNum : Int!
21        var isCyclePlay : Bool!
22        var isRePlayNow : Bool!
23        var currentRow:Int = 0
24
25        var xzlrc = XZLrc()
26        var timeArray = [CGFloat]()
27        var wordArray = [String]()
28        var chineseArray = [String]()
29
30        @IBOutlet weak var sentenceTb: UITableView!
31
32        @IBOutlet weak var wordView: UIView!
33
34        @IBOutlet weak var avplayerView: UIView!
35
36        @IBOutlet weak var controlView: UIView!
37        override func viewWillAppear(animated: Bool) {
38            self.navigationController?.navigationBarHidden = false
39                setStatusBarDefault(false)
40                setNavBarStyle()
41            var barBtn = UIBarButtonItem.init(title: "<", style: UIBarButtonItemStyle.
42            self.navigationItem.leftBarButtonItem = barBtn
43        }
44        override func viewDidLoad() {
45            super.viewDidLoad()
46            movieUrl = NSURL(string: "http://bos.nj.bpc.baidu.com/tieba-smallvideo/117
47
48            self.navigationItem.title = "场景轰炸"
49            self.hideTabBar()
50            replayNum = 0
51            wordStart = 10.100000
52            isCyclePlay = false
53            isRePlayNow = false
54            getTitleNet(2)
55            AVPlayerSetting()
56            LoadingGif()
57            AVPlayerControlView()
58            setSentenceTableView()
59            sentenceTb.separatorStyle = UITableViewCellSeparatorStyle.None
60        }
61        override func viewWillDisappear(animated: Bool) {
62            self.player.removeObserver(self, forKeyPath: "status")
63            if getCurrentTimer != nil {
64                getCurrentTimer.invalidate()
65                getCurrentTimer = nil
66            }
67            if replayTimer != nil {
68                replayTimer.invalidate()
```

```swift
 69              replayTimer = nil
 70          }
 71          self.player.pause()
 72          if self.player != nil {
 73              avplayerView.removeFromSuperview()
 74              avplayerView = nil
 75          }
 76      }
 77
 78      func getTitleNet(moiveId:Int) {
 79          //获取字幕信息
 80          xzlrc.getLrcWithUrl(moiveId) {
 81
 82              self.timeArray = self.xzlrc.timeArray
 83              self.wordArray = self.xzlrc.wordArray
 84              self.chineseArray = self.xzlrc.chineseArray
 85              self.sentenceTb.reloadData()
 86          }
 87      }
 88      func AVPlayerControlView()  {
 89          controlView.backgroundColor = colorFromHex( "1b1a18", alpha: 0.6)
 90          controlView.hidden = false
 91      }
 92      func AVPlayerSetting() {
 93
 94          if player == nil {
 95              playerItem = AVPlayerItem(URL: movieUrl)
 96              player = AVPlayer(playerItem: playerItem)
 97              let playerLayer:AVPlayerLayer = AVPlayerLayer(player: player)
 98              playerLayer.frame = avplayerView.frame
 99              playerLayer.videoGravity = AVLayerVideoGravityResizeAspect
100              avplayerView.layer.addSublayer(playerLayer)
101              playerLayer.videoGravity = AVLayerVideoGravityResizeAspectFill
102              NSNotificationCenter.defaultCenter().addObserver(self,
103                                                      selector: #selector(p
104                                                      name: AVPlayerItemDid
105                                                      object: playerItem)
106              addSomeObserver()
107              playState = true
108              player.status
109              let tapGesture = UITapGestureRecognizer(target: self, action: #selecto
110              tapGesture.numberOfTapsRequired = 1
111              let tapView = UIView.init(frame: playerLayer.frame)
112              avplayerView.addSubview(tapView)
113              tapView.addGestureRecognizer(tapGesture)
114              player.play()
115              getCurrentTimer = NSTimer.scheduledTimerWithTimeInterval(0.1, target:
116              NSRunLoop.mainRunLoop().addTimer(getCurrentTimer, forMode: NSDefaultRu
117
118          }
119
120      }
121      var webView:UIWebView!
122      func LoadingGif() {
123          let path = NSBundle.mainBundle().pathForResource("catLittle", ofType: "gif
124
```

```swift
125              let gifData = NSData(contentsOfFile: path)
126
127              webView = UIWebView(frame:avplayerView.frame)
128
129          self.view.addSubview(webView)
130          webView.layer.masksToBounds = true
131          webView.layer.cornerRadius = 5
132          webView.scalesPageToFit = true
133          webView.scrollView.scrollEnabled = false
134          webView.backgroundColor = UIColor.clearColor()
135          webView.opaque = false
136          webView.loadData(gifData!, MIMEType: "image/gif", textEncodingName: "", ba
137          self.view.addSubview(webView)
138      }
139      func addSomeObserver() {
140          //监听PlayerItem这个类
141          self.player.addObserver(self, forKeyPath: "status", options: NSKeyValueObs
142
143      }
144      override func observeValueForKeyPath(keyPath: String?, ofObject object: AnyObj
145          if (keyPath == "status") {
146              if player.status == AVPlayerStatus.ReadyToPlay
147              {
148                  print("准备播放")
149                  if self.webView != nil{
150                      self.webView.removeFromSuperview()
151                      webView = nil
152                  }
153              }
154              else if player.status == AVPlayerStatus.Failed || player.status == AVP
155                  player.pause()
156              }
157          }
158      }
159      func setSentenceTableView() {
160          sentenceTb.backgroundColor = colorFromHex("1b1a18")
161          sentenceTb.delegate = self
162          sentenceTb.dataSource = self
163      }
164      func tableView(tableView: UITableView, numberOfRowsInSection section: Int) ->
165          return timeArray.count
166      }
167      func tableView(tableView: UITableView, heightForRowAtIndexPath indexPath: NSIn
168          var ss = CGSize()
169          ss = strSize(chineseArray[indexPath.row], withMaxSize: CGSize(width: 200,h
170          var sss = CGSize()
171          sss = strSize(wordArray[indexPath.row], withMaxSize: CGSize(width: 200,hei
172
173          return ss.height+sss.height+8
174      }
175      func tableView(tableView: UITableView, cellForRowAtIndexPath indexPath: NSInde
176          var sentenceCell = tableView.dequeueReusableCellWithIdentifier("thirdID")a
177
178          if sentenceCell == nil
179          {
180              sentenceCell = SentenceTableViewCell(style: .Default, reuseIdentifier:
```

```
181
182            }
183            sentenceCell?.ChineseLb.text = self.chineseArray[indexPath.row]
184            sentenceCell?.EnglishLb.text = self.wordArray[indexPath.row]
185            var ss = CGSize()
186            ss = strSize(self.wordArray[indexPath.row], withMaxSize: CGSize(width: 200
187
188            sentenceCell?.EnglishLb.frame = CGRectMake(self.view.frame.size.width/2-10
189            var sss = CGSize()
190            sss = strSize(self.chineseArray[indexPath.row], withMaxSize: CGSize(width:
191            sentenceCell?.ChineseLb.frame = CGRectMake(self.view.frame.size.width/2-10
192            return sentenceCell!
193        }
194    func strSize(str: String, withMaxSize size: CGSize, withFont font: UIFont, wit
195        var s: CGSize
196
197        s = str.boundingRectWithSize(size, options: [.UsesLineFragmentOrigin, .Use
198
199        return s
200    }
201    func tableView(tableView: UITableView, didSelectRowAtIndexPath indexPath: NSIn
202        tableView.deselectRowAtIndexPath(indexPath, animated: true)
203    }
204    func handleTapGesture()
205    {
206        UIView.animateWithDuration(0.5) {
207            self.controlView.hidden = !self.controlView.hidden
208        }
209
210    }
211    func playerEnd() {
212        if isCyclePlay == true {
213            player.seekToTime(CMTimeMake(0, 1))
214            player.play()
215
216        }
217    }
218
219
220    @IBAction func playOrPauseClick(sender: AnyObject) {
221        if playState == true {
222            player.pause()
223            playState = false
224        }
225        else
226        {
227            player.play()
228            playState = true
229        }
230    }
231
232    @IBAction func leftMovieClick(sender: AnyObject) {
233        getTitleNet(1)
234        movieUrl = NSURL(string: "http://bos.nj.bpc.baidu.com/tieba-smallvideo/117
235        playerItem = AVPlayerItem(URL: movieUrl)
236        player.replaceCurrentItemWithPlayerItem(playerItem)
```

```
237        }
238
239        @IBAction func cyclePlayClick(sender: AnyObject) {
240            isCyclePlay = !isCyclePlay
241
242        }
243        @IBAction func rightMovieClick(sender: AnyObject) {
244            getTitleNet(4)
245            movieUrl = NSURL(string: "http://w2.dwstatic.com/1/5/1525/127352-100-14345
246            playerItem = AVPlayerItem(URL: movieUrl)
247            player.replaceCurrentItemWithPlayerItem(playerItem)
248
249        }
250
251        @IBAction func wordListBook(sender: AnyObject) {
252        }
253        func getCurrentMovieTime() {
254            scrollAccrodingTimeArray()
255        }
256
257        func scrollAccrodingTimeArray() {
258
259
260            let currentSecond:Double = CMTimeGetSeconds(self.playerItem.currentTime())
261            let s = String(format: "%.1f", currentSecond)
262            if isRePlayNow == false {
263                //判断当前时间是否要重复强调
264                if s == String(format: "%.1f", wordStart)
265                {
266                    isRePlayNow = true
267                    replaySetting()
268                }
269            }
270
271            for index in 0 ..< timeArray.count {
272                let ss = String(format: "%.1f", timeArray[index]/1000)
273                //判断当前时间是否要滚动句子
274                if ss == s {
275                    scrollToCurrentRow(index)
276                }
277
278            }
279        }
280        func replaySetting() {
281            replayTimer = NSTimer.scheduledTimerWithTimeInterval(1.5, target: self, se
282            NSRunLoop.mainRunLoop().addTimer(replayTimer, forMode: NSDefaultRunLoopMod
283
284
285        }
286        func replayThreeTimes() {
287            if replayNum  >= 3
288            {
289                replayTimer.invalidate()
290                replayTimer = nil
291                isRePlayNow = false
292                replayNum = 0
```

```swift
293                return
294            }
295            replayNum = replayNum+1
296            let timeScale = Int32(self.player.currentItem!.asset.duration.timescale)
297            let time = CMTimeMakeWithSeconds(Float64(wordStart), timeScale);
298            self.player.seekToTime(time, toleranceBefore: kCMTimeZero, toleranceAfter:
299            self.player.seekToTime(<#T##time: CMTime##CMTime#>)
300            let currentTime1:CMTime = self.playerItem.currentTime()
301            let currentSecond1:Double = CMTimeGetSeconds(currentTime1)
302            print("###### \(currentSecond1)")
303        }
304
305    func scrollToCurrentRow(idx:Int){
306
307        currentRow = idx
308        if self.chineseArray.count > 0 {
309            let indexPath = NSIndexPath(forRow: idx, inSection: 0)
310            for item in sentenceTb.visibleCells{
311                let cell = item as! SentenceTableViewCell
312                cell.EnglishLb.textColor = colorFromHex("666666")
313                cell.ChineseLb.textColor = colorFromHex("666666")
314            }
315
316            let cell = sentenceTb.cellForRowAtIndexPath(indexPath) as? SentenceTab
317            if cell != nil {
318                cell!.EnglishLb.textColor = colorFromHex("5eecff")
319                cell!.ChineseLb.textColor = colorFromHex("5eecff")
320            }
321
322            switch currentRow {
323            case chineseArray.count - 1:
324                sentenceTb.scrollToRowAtIndexPath(indexPath, atScrollPosition: UIT
325                break
326            case chineseArray.count - 2:
327                sentenceTb.scrollToRowAtIndexPath(indexPath, atScrollPosition: UIT
328                break
329            default:
330                sentenceTb.scrollToRowAtIndexPath(indexPath, atScrollPosition: UIT
331                break
332            }
333        }
334    }
335
336    override func didReceiveMemoryWarning() {
337        super.didReceiveMemoryWarning()
338        // Dispose of any resources that can be recreated.
339    }
340
341 }
342 class SentenceTableViewCell: UITableViewCell {
343
344    let EnglishLb = UILabel()
345    let ChineseLb = UILabel()
346    override init(style: UITableViewCellStyle, reuseIdentifier: String?) {
347
348        super.init(style: style, reuseIdentifier: reuseIdentifier)
```

```
349
350            self.contentView.addSubview(ChineseLb)
351            self.contentView.addSubview(EnglishLb)
352            self.contentView.backgroundColor = colorFromHex("1b1a18")
353            setUpviews()
354            self.backgroundColor = colorFromHex("1b1a18")
355        }
356
357        required init?(coder aDecoder: NSCoder) {
358            fatalError("init(coder:) has not been implemented")
359        }
360
361        func setUpviews() {
362            EnglishLb.frame = CGRectMake(self.contentView.frame.size.width/2-100 , 0,
363            EnglishLb.text = "['prlk(e)l]"
364            EnglishLb.font = UIFont.systemFontOfSize(15)
365            EnglishLb.textColor = UIColor.whiteColor()
366            EnglishLb.textAlignment = NSTextAlignment.Center
367            EnglishLb.numberOfLines = 0;
368
369            ChineseLb.frame = CGRectMake(self.contentView.frame.size.width/2-100, 20,
370            ChineseLb.text = "n:规划；计划"
371            ChineseLb.font = UIFont.systemFontOfSize(15)
372            ChineseLb.textColor = UIColor.whiteColor()
373            ChineseLb.textAlignment = NSTextAlignment.Center
374            ChineseLb.numberOfLines = 0
375        }
376
377
378
379
380    }
381
```

查看原文>> (http://blog.csdn.net/Xoxo_x/article/details/52606786)

看过本文的人也看了：

• iOS知识结构图
(http://lib.csdn.net/base/ios/structure)

• iOS逆向入门实践 — 逆向微信，伪装定...
(http://lib.csdn.net/article/ios/43842)

• iOS开发-定位服务和地理围栏
(http://lib.csdn.net/article/ios/42080)

• iOS开发-------地图定位显示周围微...
(http://lib.csdn.net/article/ios/42078)

• 【iOS】Mapkit的使用：地图显示、定位...
(http://lib.csdn.net/article/ios/42081)

• Android-导航栏特效-新闻类APP(仿iOS...
(http://lib.csdn.net/article/ios/42141)

发表评论

输入评论内容

发表

1个评论

(http://my.csdn.net/hwl348833288)

hwl348833288 (http://my.csdn.net/hwl348833288)

不对吧 是不是这样子

2016-10-10 23:35:16

回复

公司简介 (http://www.csdn.net/company/about.html) ｜ 招贤纳士 (http://www.csdn.net/company/recruit.html) ｜

广告服务 (http://www.csdn.net/company/marketing.html) ｜ 银行汇款帐号 (http://www.csdn.net/company/account.html)

｜ 联系方式 (http://www.csdn.net/company/contact.html) ｜ 版权声明 (http://www.csdn.net/company/statement.html) ｜

法律顾问 (http://www.csdn.net/company/layer.html) ｜ 问题报告 (mailto:webmaster@csdn.net) ｜

合作伙伴 (http://www.csdn.net/friendlink.html) ｜ 论坛反馈 (http://bbs.csdn.net/forums/Service)

网站客服　　杂志客服 (http://wpa.qq.com/msgrd?v=3&uin=2251809102&site=qq&menu=yes)

微博客服 (http://e.weibo.com/csdnsupport/profile)　　webmaster@csdn.net (mailto:webmaster@csdn.net)　　400-600-2320 ｜