

iOS_人脸检测_联网授权FAQ

- [iOS_人脸检测_联网授权FAQ](#)

- [<1> SDK支持哪些平台](#)
- [<2> SDK大小](#)
- [<3> SDK性能/影响检测时间的因素](#)
- [<4> 获取人脸的角度信息（3D Pose）](#)
- [<5> 怎么判断是否是同一张脸](#)
- [<6> 如何取出一张人脸照片](#)
 - [<6.1> 将视频帧转为图片](#)
- [<6.2> 如何获取一张质量相对较好的人脸照片](#)
- [<7> SDK可以检测多少个关键点](#)
- [<8> 检测到的关键点与人脸位置的对应关系](#)
- [<9> MGFaceppConfig各参数的意义](#)
- [<10> 如何正确设置config的orientation参数](#)
 - [<10.1> 检测视频流时orientation参数设置：](#)
 - [<10.2> 检测图片时orientation的参数设置：](#)
- [<11> 如何正确设置pixelFormatType参数](#)
- [<12> 为什么检测不到人脸](#)

<1> SDK支持哪些平台

SDK目前支持iOS和Andriod平台，其中iOS支持iPhone 4S 以上机型 和 iOS8 以上版本的操作系统，不支持模拟器

<2> SDK大小

版本号 0.5.2

SDK大小

包大小		Android	iOS
代码部分	32位	2.0M	19.7M
	64位	3.2M	19.8M
模型部分	人脸检测	4.8M	
	人脸关键点fast模式	track_fast 1.8M	
	人脸关键点robust模式	track_robust 2.5M	
	人脸属性	2.3M	
	3D pose	145K	

<3> SDK性能/影响检测时间的因素

测试环境：

环境参数	取值	对性能的影响
机型	iPhoen 6	cpu越好， 速度越快
系统版本	iOS 10.3.2	-
帧速(fps)	50	帧速越低， cpu占用率越低
关键点	81	-
摄像头	前置摄像头	-
最小人脸	100	值越小， 越消耗性能
检测间隔	40	值越小， 越消耗性能
分辨率	640*480	值越大， 越消耗性能
单脸跟踪	NO	-
fast模式	每帧时间-6.84ms 每帧内存-7M CPU-46.32%	
robust模式	每帧时间-9.03ms 每帧内存-7.6M CPU-60.12	

<4> 获取人脸的角度信息（3D Pose）

MGFaceInfo 类提供了人脸的角度信息，单位是 弧度制

```
14  MGFemale = 0,  
15  MGMale = 1,  
16  };  
17  
18  @interface MGFaceInfo : NSObject  
19  
20  /** tracking ID */  
21  @property (nonatomic, assign) NSInteger trackID;  
22  
23  /** 在该张图片中人脸序号 */  
24  @property (nonatomic, assign) int index;  
25  
26  /** 人脸的rect */  
27  @property (nonatomic, assign) CGRect rect;  
28  
29  /** 人脸点坐标 (NSValue -> CGPoint) */  
30  @property (nonatomic, strong) NSArray <NSValue *>*points;  
31  
32  /** 该张人脸质量 */  
33  @property (nonatomic, assign) CGFloat confidence;  
34  
35  
36  #pragma mark 以下属性需要SDK版本支持 请使用 [MGAlgorithmInfo SDKAbility] 属性，获取SDK支持功能  
37  
38  #pragma mark 需要主动调用 MGFacepp 相关方法获取  
39  //3D info  
40  @property (nonatomic, assign) float pitch;  
41  @property (nonatomic, assign) float yaw;  
42  @property (nonatomic, assign) float roll;
```

3D信息的参数解释如下

```
//抬头（上下）  
@property (nonatomic, assign) float pitch;  
//摇头（左右）  
@property (nonatomic, assign) float yaw;  
//旋转（平面旋转）  
@property (nonatomic, assign) float roll;
```

<5> 怎么判断是否是同一张脸

答：这个可以根据 trackID 来判断，如果同一张人脸一直在屏幕中并被检测到，那么这张人脸的 trackID 是不变的，反过来说，如果trackID相同，说明是同一张人脸 PS: 同一张人脸离开屏幕后重新进入，trackID 会发生改变

<6> 如何取出一张人脸照片

可以用如下代码将视频帧转为图片，`orientation` 的值和视频流的输出方向有关，建议转为图片之后，将图片保存到相册，来查看图片的方向是否是正确的

<6.1> 将视频帧转为图片

```
+ (UIImage*)imageFromSampleBuffer:(CMSampleBufferRef)sampleBuffer
    orientation:(UIImageOrientation) orientation{
    CVImageBufferRef imageBuffer = CMSampleBufferGetImageBuffer(sampleBuffer);
    CVPixelBufferLockBaseAddress(imageBuffer, 0);

    void *baseAddress = CVPixelBufferGetBaseAddress(imageBuffer);

    size_t bytesPerRow = CVPixelBufferGetBytesPerRow(imageBuffer);
    size_t width = CVPixelBufferGetWidth(imageBuffer);
    size_t height = CVPixelBufferGetHeight(imageBuffer);

    CGColorSpaceRef colorSpace = CGColorSpaceCreateDeviceRGB();
    CGContextRef context = CGContextCreate(baseAddress,
                                           width,
                                           height,
                                           8,
                                           bytesPerRow,
                                           colorSpace,
                                           kCGBitmapByteOrder32Little | kCG
    CGImageRef quartzImage = CGContextCreateImage(context);
    CVPixelBufferUnlockBaseAddress(imageBuffer, 0);

    UIImage *image = [UIImage imageWithCGImage:quartzImage scale:1.0 orientation:

    CGImageRelease(quartzImage);
    CGContextRelease(context);
    CGColorSpaceRelease(colorSpace);

    return image;
}
```

<6.2> 如何获取一张质量相对较好的人脸照片

可以根据人脸的一些属性来"筛选"质量较好的照片

- 判断人脸的角度信息,比如人脸的角度信息（详细看第<6>条）不超过45度(角度)
- 人脸框的位置 `rect`参数，比如人脸框在屏幕中央区域（根据鼻尖的关键点坐标判断），并

且大小在某个范围之内（比如 `100*100 ~ 200*200` ），具体请根据自己的业务场景设定。

<7> SDK可以检测多少个关键点

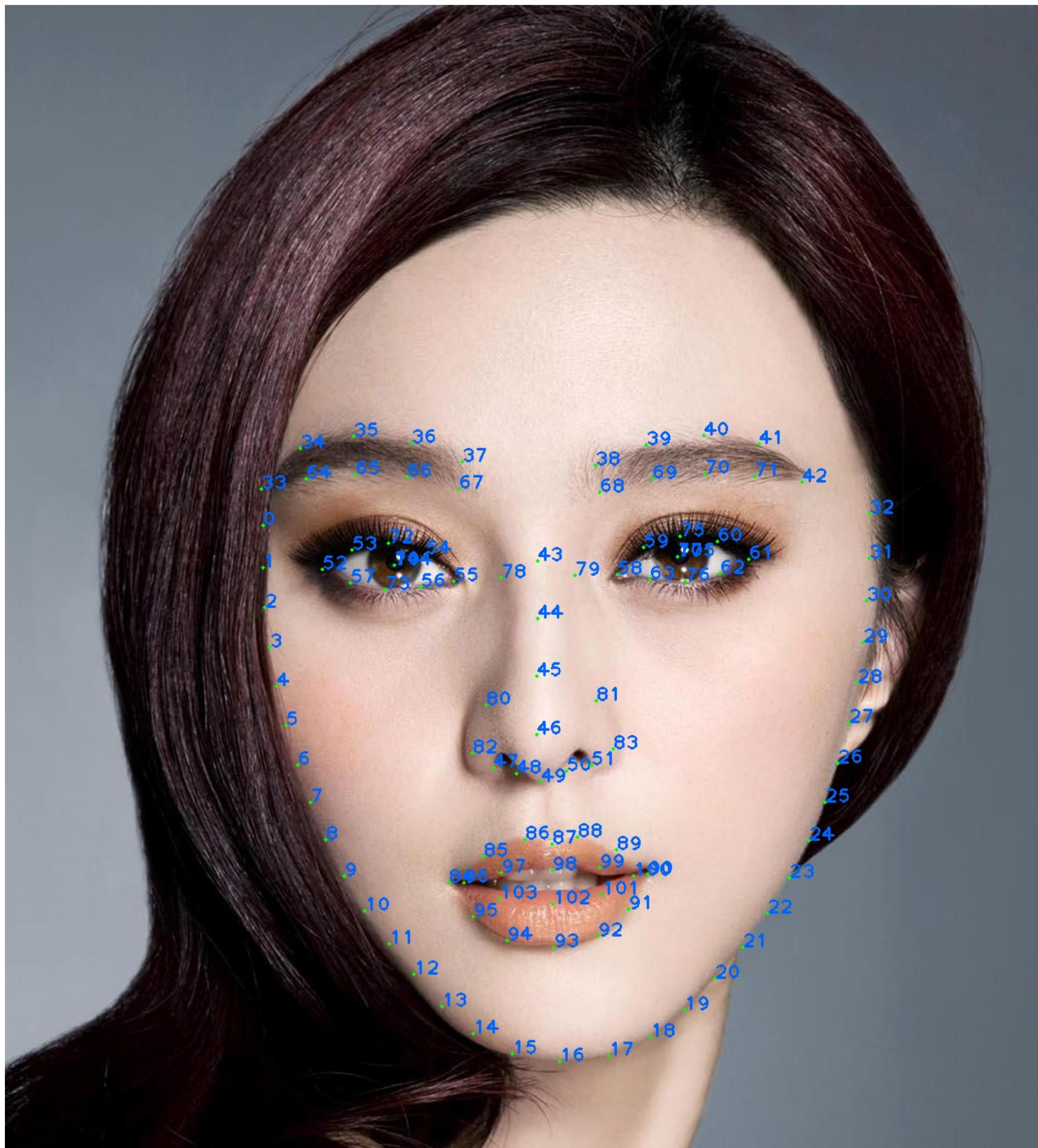
获取SDK检测的关键点数（81点 或 106点）， 代码设置如下：

```
int pointsNum = 81;//或者106  
[self.markManager GetGetLandmark:faceInfo isSmooth:YES pointsNumber:pointsNum];
```

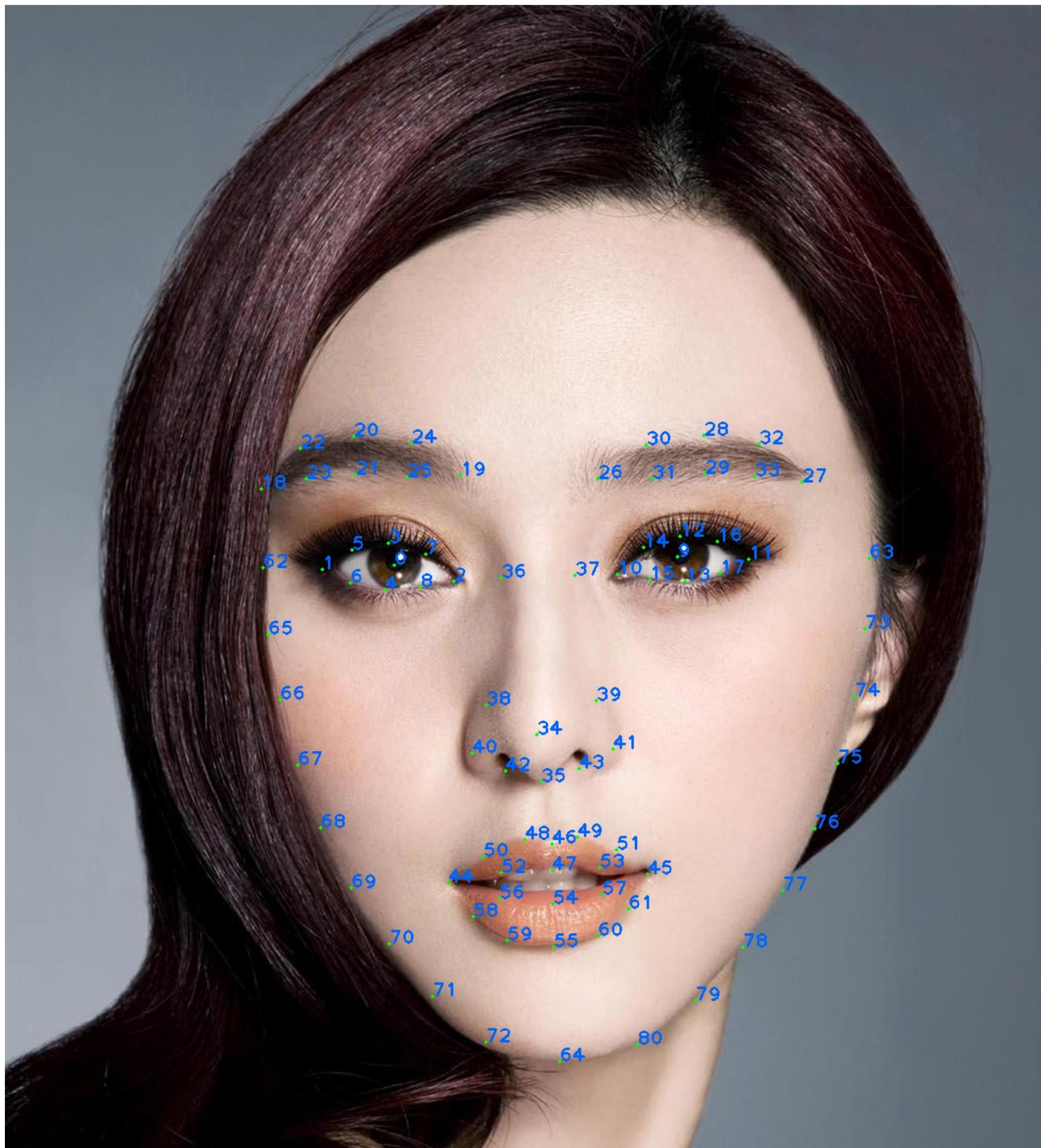
<8> 检测到的关键点与人脸位置的对应关系

在下载SDK包 `iOS/doc/xxx_point_position` 的路径下的图片(yyy表示81或106)， 上面有标示数组下标对应在人脸上的位置关系

106点坐标如下：



81点坐标如下:



<9> MGFaceppConfig各参数的意义

config的相关参数介绍如下:

```
/** 人脸大小 默认 100,低于 100*100像素的人脸将不会被检测到*/  
@property (nonatomic, assign) int minFaceSize;  
  
/** 重新全局检测间隔 40 */  
@property (nonatomic, assign) int interval;
```



```
/** 旋转角度 default 0, [0,90,180,270,360], 和视频帧的方向保持一致*/
@property (nonatomic, assign) int orientation;

/**
 * MGFppDetectionModeDetect = 0, //检测图片
 * MGFppDetectionModeTrackingFast = 3, //检测视频流,速度较快
 * MGFppDetectionModeTrackingRobust = 4, //检测视频流,精度较高,推荐
 */
@property (nonatomic, assign) MGFppDetectionMode detectionMode;

/** 废弃参数 */
@property (nonatomic, assign) MGDetectROI detectROI;

/** 设置视频流格式, 默认PixelFormatTypeRGBA ,注意要和你的视频流格式保持一致
 * PixelFormatTypeGRAY
 * PixelFormatTypeBGR
 * PixelFormatTypeNV21
 * PixelFormatTypeRGBA
 * PixelFormatTypeRGB
 */
@property (nonatomic, assign) MGPixelFormatType pixelFormatType;

/** 置信度低于该值的人脸将会被忽略, 防止误检*/
@property (nonatomic, assign) float faceConfidenceFilter;
```

<10> 如何正确设置config的orientation参数

<10.1> 检测视频流时orientation参数设置:

videoConnection.videoOrientation	config.orientation
不设置	90
AVCaptureVideoOrientationPortrait	0
AVCaptureVideoOrientationPortraitUpsideDown	180
AVCaptureVideoOrientationLandscapeRight	90
AVCaptureVideoOrientationLandscapeLeft	270

例如，视频流输出方向为：

```
videoConnection.videoOrientation = AVCaptureVideoOrientationLandscapeRight
```


那么，应该初始化检测器时config设置为: `config.orientation = 90;`

```
NSString *modelPath = [[NSBundle mainBundle] pathForResource:KMFACEMODELNAME ofType:@"model"];
NSData *modelData = [NSData dataWithContentsOfFile:modelPath];
self.markManager = [[MGFacepp alloc] initWithModel:modelData
                                                    faceppSetting:^(MGFaceppConfig *config) {
//config
config.orientation = 90;
}];
```

如果已经创建过检测器，那么可以这样更新方向参数：

```
[self.markManager updateFaceppSetting:^(MGFaceppConfig *config) {
config.orientation = 90;
}];
```

<10.2> 检测图片时orientation的参数设置：

- 方案一： 如果图片是直接来自相机或者相册获取的，那么方向很可能不对的，一般设置 `config.orientation = 90`
- 方案二： 对图片进行方向矫正,然后再送给检测器检测

```
- (UIImage *)normalizedImage {
    if (self.imageOrientation == UIImageOrientationUp) return self;

    UIGraphicsBeginImageContextWithOptions(self.size, NO, self.scale);
    [self drawInRect:(CGRect){0, 0, self.size}];
    UIImage *normalizedImage = UIGraphicsGetImageFromCurrentImageContext();
    UIGraphicsEndImageContext();
    return normalizedImage;
}
```

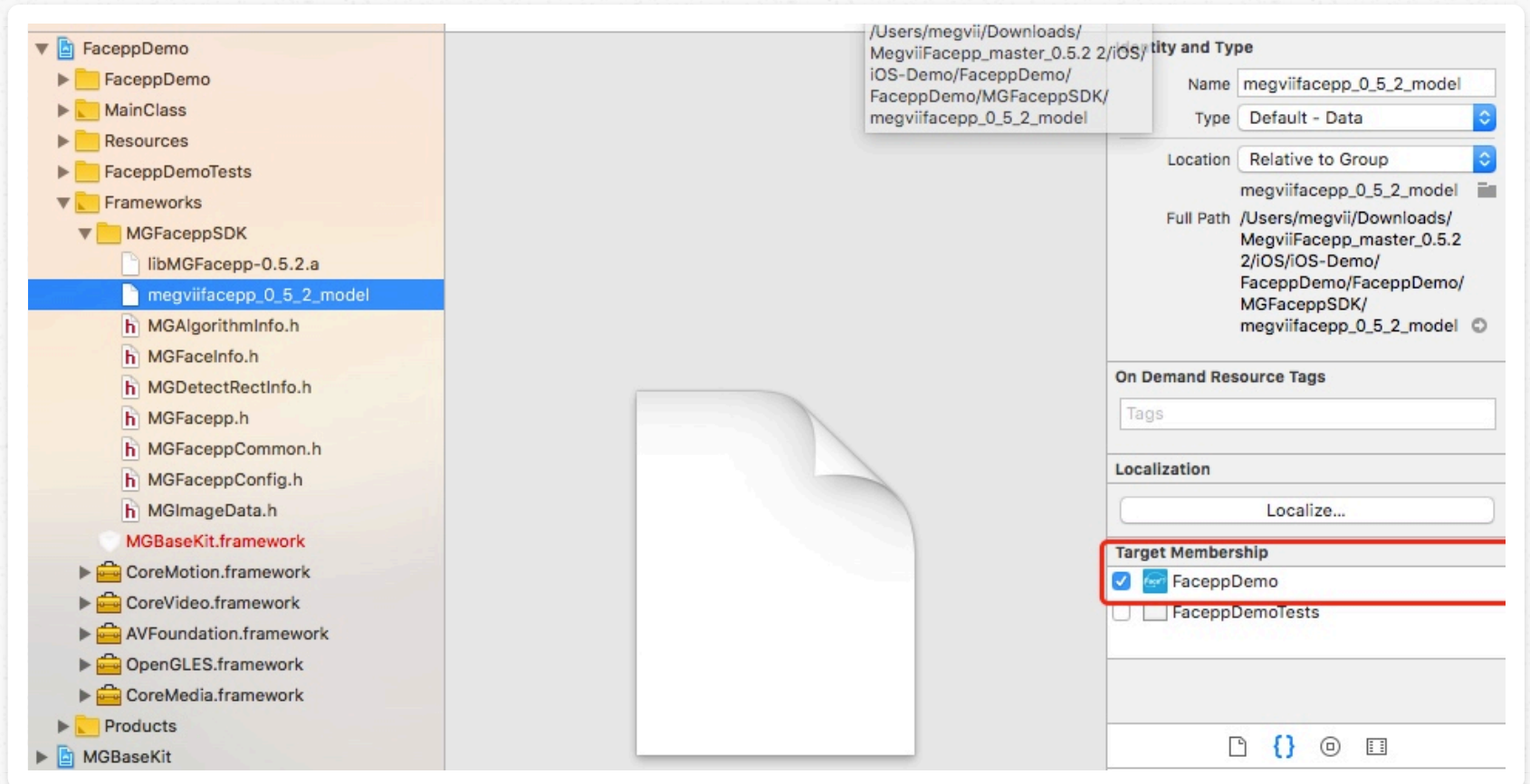
<11> 如何正确设置pixelFormatType参数

iOS 上常见的两种视频流格式

kCVPixelBufferPixelFormatTypeKey	config.pixelFormatType值	备注
kCVPixelFormatType_32BGRA	PixelFormatTypeRGBA	RGBA(黑

<12> 为什么检测不到人脸

1. 模型是否引用成功



2. 检查授权是否成功

- 联网授权请根据log日志来判断

```
[MGFaceLicenseHandle licenseForNetwokrFinish:^(bool License, NSDate *sdkDate)
{
    if (!License) {
        NSLog(@"联网授权失败 !!!");
    } else {
        NSLog(@"联网授权成功");
    }
}];
```

3. 视频流、图片方向是否OK

方向参数设置请参考 第<14条> 的设置

4. 视频流、图片格式是否OK

- 对于检测视频流，请参考 第<15条> 的设置
- 对于检测图片，默认设置即可

5. 人脸大小参数是否OK

可以参考 第<13>条 的 `minFaceSize` 参数，人脸大小小于 `minFaceSize` 时会被忽略掉

6. 有人脸，但被遮挡过多，或者脸的旋转角度过大，或者光线很暗

7. 视频里真的没有人脸