

阿里巴巴集团

SDK 参考手册

iOS 媒体推流器 SDK 参考手册

POWERED BY QUPAI

文档修订历史

版本	作者	工作描述	修订历史	修改日期
1.0	刘亚钊	iOS 推流 SDK 文档		2016-06-21
1.1	刘亚钊	iOS 推流 SDK 文档		2016-07-14
1.2	刘亚钊	iOS 推流 SDK 文档		2016-11-02

文档说明

本文档面向所有使用该 SDK 的开发人员、测试人员以及对此感兴趣的用戶，要求读者具有一定的 iOS 开发能力

1. 简介

iOS 推流 SDK 是在 iOS 平台上使用的软件开发工具包 (Soft Development Kit)，为 iOS 开发者提供简单易用的接口，帮助开发者实现 iOS 平台上的推流应用开发。

1.1 功能说明

1. 方便快捷、低门槛实现媒体推流功能。用户无须关心内部实现细节，只需要自定义界面既可以实现专业级的推流应用。

2. 推流支持格式：rtmp

3. 编码目前为硬编

1.2 安装包说明

推流器 SDK 的完整下载包中包含 test、lib、doc 等：

1. test：主要存放了调用 SDK 的示例工程，可以帮助用户了解如何使用该 SDK

2. lib：推流器 SDK 开发包, 包含推流器 framework 文件, 需要在您的工程中进行引用

3. doc：存放 SDK 相关接入文档。

1.3 推流器性能

1. 目前推流 SDK 推流采用的是硬编。
2. 推流采用 librtmp 推流
3. SDK 的大小：使用 SDK 库安装之后的程序大小在 1M 左右。即为你的程序增加 1M 左右的大小。

1.4 注意事项

1. 推流器 SDK 目前只支持单实例。不能够同时开 2 个推流实例，同时只能存在一个实例，需要另开实例的时候，需要关闭之前存在的实例。
2. 操作系统版本要求 iOS8.0 以上。

2. 使用说明

2.1 开发环境配置

1. 需要准备 iOS 的运行环境(Xcode8.0 以上版本, iOS SDK8.0 以上版本)，以及硬件 CPU 支持 ARMv7、ARMv7s 或 ARM64 的 iOS 设备。

2.2、开通直播

参考链接：

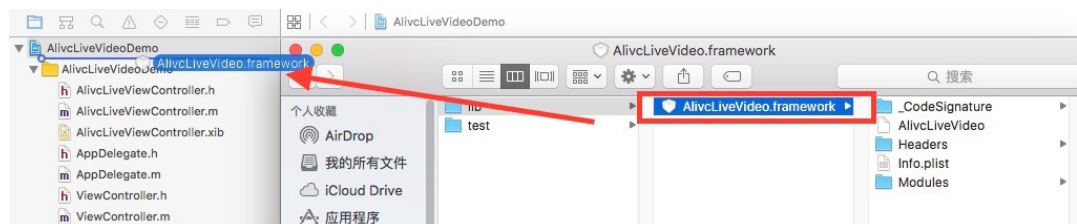
https://help.aliyun.com/document_detail/29957.html

2.3、SDK 下载

—阿里云提供下载地址。

2.4、SDK 集成

拖拽 sdk 文件夹到自己的 Xcode 项目中。



勾选 Copy items if needed, 点击 Finish。
打开项目的 app target, 在 Build Phases 中的 Link Binary With Libraries 添加以下依赖库。

```
libz.tbd  
VideoToolbox.framework  
AudioToolbox.framework  
libstdc++.tbd  
SystemConfiguration.framework  
CoreTelephony.framework
```

2.5、iOS10 适配说明

由于 iOS10.0 对于 iOS 设备权限获取的方式不同于之前的版本, 因此特别说明:

推流 SDK 需要获取 iOS 设备的摄像头和麦克风权限, 需要在开发项目的 plist 文件中添加如下权限配置:

1. Privacy - Camera Usage Description
2. Privacy - Microphone Usage Description

3. SDK 使用指南

3.1、开发环境配置:

1. 需要准备 Xcode 运行环境, 以及硬件配置 iOS8.0 以上的苹果手持设备。
2. 权限开通, 在阿里云上申请推流 SDK 开发权限

3.2、开发步骤

在需要直播功能的控制器头部导入#import
<AlivcLiveVideo/AlivcLiveVideo.h>, 并设置代理 AlivcLiveSessionDelegate
使用用媒体推流器 SDK 的调用顺序为:

1. 调用[[AlivcLiveSession alloc] initWithConfiguration:configuration]创建 session
类

2. 调用 `alivcLiveVideoStartPreview` 开启预览
3. 调用 `alivcLiveVideoConnectServer` 开始推流
4. 调用 `alivcLiveVideoDisconnectServer` 停止推流
5. 调用 `alivcLiveVideoStopPreview` 停止预览
6. `session = nil` 销毁直播 session

3.3、Demo 示例

在 SDK 中提供了 Demo, 此 Demo 是用推流 SDK 开发了一个完整的推流器, 用户可以参考 Demo 进行推流的开发。

下面给出了部分重要的 Demo 中调用 SDK 的代码。

一. 设置直播参数

1. 初始化 config 配置类

```
AlivcLConfiguration *configuration = [[AlivcLConfiguration alloc] init];
```

2. 设置推流地址

```
configuration.url = pushUrl;
```

3. 设置最大码率

```
configuration.videoMaxBitRate = 1500 * 1000;
```

4. 设置当前视频码率

```
configuration.videoBitRate = 600 * 1000; /
```

5. 设置最小码率

```
configuration.videoMinBitRate = 400 * 1000;
```

6. 设置音频码率

```
configuration.audioBitRate = 64 * 1000;
```

7. 设置直播分辨率

```
configuration.videoSize = CGSizeMake(360, 640);
```

8. 设置横屏 or 竖屏

```
configuration.screenOrientation = AlivcLiveScreenVertical;
```

9. 设置帧率

```
configuration.fps = 20;
```

10. 设置摄像头采集质量

```
configuration.preset = AVCaptureSessionPresetiFrame1280x720;
```

11. 设置前置摄像头或后置摄像头

```
configuration.position = AVCaptureDevicePositionFront;
```

12. 设置水印图片¹

```
configuration.waterMaskImage = [UIImage imageNamed:@"watermask"];
```

13. 设置水印位置

```
configuration.waterMaskLocation = 1;
```

14. 设置水印相对x边框距离

```
configuration.waterMaskMarginX = 10;
```

15. 设置水印相对y边框距离

```
configuration.waterMaskMarginY = 10;
```

16. 设置重连超时时长

```
configuration.reconnectTimeout = 5;
```

二.创建直播 session

1. 初始化 liveSession 类

```
_liveSession = [[AlivcLiveSession alloc] initWithConfiguration:configuration]
```

2.设置 session 代理

```
_liveSession.delegate = self;
```

3.开启直播预览

```
[_liveSession alivcLiveVideoStartPreview];
```

4.开启直播

```
[_liveSession alivcLiveVideoConnectServer];
```

¹ 注：文档红色字体标明部分为本版本更新部分

5.获取直播预览视图

```
[_liveSession previewView]
```

可将获取的直播视图添加到当前控制器上，实现直播预览功能。

6.停止预览

```
[_liveSession alivcLiveVideoStopPreview];
```

注意：停止预览后将 liveSession 置为 nil

7.关闭直播

```
[_liveSession alivcLiveVideoDisconnectServer];
```

8.设置闪光灯模式

```
_liveSession.torchMode = AVCaptureTorchModeOn//关闭
```

```
AVCaptureTorchModeOff
```

9.开启美颜

```
[_liveSession setEnableSkin:YES];
```

10.缩放

```
[_liveSession alivcLiveVideoZoomCamera:1.0f];
```

11.聚焦

```
[_liveSession alivcLiveVideoFocusAtAdjustedPoint:percentPoint
```

```
autoFocus:YES];
```


12.调试信息

```
AlivcLDebugInfo *i = [_liveSession dumpDebugInfo];
```

三、直播功能的相关代理方法

@required

1. 直播出错的代理方法

```
-(void)alivcLiveVideoLiveSession:(AlivcLiveSession *)session  
error:(NSError *)error;
```

2. 网速较慢时的代理方法

```
-(void)alivcLiveVideoLiveSessionNetworkSlow:  
(AlivcLiveSession *)session;
```

@optional

1. 音频初始化失败

```
-(void)alivcLiveVideoLiveSession:(AlivcLiveSession *)session  
OpenAudioError:(NSError *)error;
```

2. 视频初始化失败

```
-(void)alivcLiveVideoLiveSession:(AlivcLiveSession *)session  
OpenVideoError:(NSError *)error;
```

3. 音频编码器初始化失败

```
-(void)alivcLiveVideoLiveSession:(AlivcLiveSession *)session  
EncodeAudioError:(NSError *)error;
```

4. 视频编码器初始化失败

```
-(void)alivcLiveVideoLiveSession:(AlivcLiveSession *)session  
EncodeVideoError:(NSError *)error;
```

5. 音频设备获取成功

- (void)alivcLiveVideoOpenAudioSuccess:(AlivcLiveSession *)session;

6. 视频设备获取成功

- (void)alivcLiveVideoOpenVideoSuccess:(AlivcLiveSession *)session;

7. 推流连接成功

- (void)alivcLiveVideoLiveSessionConnectSuccess:(AlivcLiveSession *)session;

8. 码率变化

-(void)alivcLiveVideoLiveSession:(AlivcLiveSession *)session
bitrateStatusChange:(ALIVC_LIVE_BITRATE_STATUS)bitrateStatus;

9. 重连超时

- (void)alivcLiveVideoReconnectTimeout:(AlivcLiveSession *)session

4. 接口说明

4.1 AlivcLiveSession

类名: AlivcLiveSession

功能: 推流器接口类, 提供推流控制

成员

initWithConfiguration	初始化推流器
StartPreview	开始预览
StopPreview	停止预览
RotateCamera	转换摄像头
FocusAtAdjustedPoint: autoFocus	对焦摄像头
ZoomCamera	缩放摄像头
enableSkin	美颜
torchMode	闪光灯
enableMute	静音
ConnectServer	开始推流
DisconnectServer	断开推流
dumpDebugInfo	调试信息
VideoBitRate	码率

Version	版本号
---------	-----

各成员具体使用;

```
/*!
```

```
 * 当前版本号
```

```
 *
```

```
 * @return 版本号
```

```
 */
```

```
+ (NSString *)alivcLiveVideoVersion;
```

```
/*!
```

```
 * init
```

```
 *
```

```
 * @param configuration 配置
```

```
 *
```

```
 * @return AlivcLiveSession
```

```
 */
```

```
- (instancetype) initWithConfiguration: (AlivcLConfiguration  
*) configuration;
```

```
/*!
```

```
 * 预览View
```

```
 *
```

```
 * @return 预览View
```

```
 */
```

```
- (UIView *)previewView;
```

```
/*!
```

```
 * 开始
```

```
 */
```

```
- (void) alivcLiveVideoStartPreview;
```

```
/*!
```

```
 * 停止
```

```
 */
```

```
- (void) alivcLiveVideoStopPreview;
```

```

/*!
 * 转换摄像头
 */
- (void)alivcLiveVideoRotateCamera;

/*!
 * 对焦
 *
 * @param point 位置
 * @param autoFocus 是否自动对焦
 */
- (void)alivcLiveVideoFocusAtAdjustedPoint: (CGPoint)point
autoFocus: (BOOL)autoFocus;

/*!
 * 缩放
 *
 * @param zoom 缩放倍数
 */
- (void)alivcLiveVideoZoomCamera: (CGFloat)zoom;

/*!
 * 更新live配置，可以更新码率和帧率，但是只能在connectServer之前调用
 *
 * @param block 配置
 */
-
(void)alivcLiveVideoUpdateConfiguration: (void (^)(AlivcLConfiguration
*configuration))block;

/*!
 * 推流连接
 */
- (void)alivcLiveVideoConnectServer;

```

```

/*!
 * 推流断开连接
 */
- (void)alivcLiveVideoDisconnectServer;

/*!
 * 调试信息
 *
 * @return 调试信息
 */
- (AlivcLDebugInfo *)dumpDebugInfo;

/*!
 * 当前码率
 *
 * @return 码率
 */
- (NSInteger)alivcLiveVideoBitRate;

```

4.2 delegate监听回调

成员

error	推流错误
NetworkSlow	网速慢
ConnectSuccess	连接成功
OpenVideoSuccess	摄像头获取成功
OpenAudioSuccess	麦克风获取成功
openAudioError	音频设备打开失败
openVideoError	视频设备打开失败
encodeAudioError	音频编码失败
encodeVideoError	视频编码失败
bitrateStatusChange	码率变化
ReconnectTimeout	重连超时

4.3 AlivcLDebugInfo

类名: AlivcLDebugInfo

功能： 提供推流实时调试信息
成员

cameraPresent	摄像头方向
connectStatus	连接状态
fps	当前编码帧数
encodeSpeed	编码速度
speed	当前上传速度 单位byte
localBufferSize	本地buffer大小
localBufferAudioCount	本地buffer音频帧数
localBufferVideoCount	当前buffer视频帧数
localDelay	编码耗时 单位ms
pushSize	当前上传数据大小 单位 byte
keyFrameDTS	上一个关键帧dts
currentVideoPTS	当前输出流video pts
currentAudioPTS	当前输出流audio pts
encodeFrameCount	所有编码帧数
pushFrameCount	所有发送帧数
videoDiscardFrameCount	视频丢帧数
audioDiscardFrameCount	音频丢帧数
cycleDelay	周期性延迟 单位 ms

5. 注意事项

无

6. 版权声明

版权所有，切勿盗版