

**云知声opus音频编码开发指南**

**iOS 版**

北京云知声信息技术有限公司

Beijing USC Information Technology CO., LTD.

重要声明

版权声明

版权所有 © 2016，北京云知声信息技术有限公司，保留所有权利。

商标声明

北京云知声信息技术有限公司的产品是北京云知声信息技术有限公司专有。在提及其他公司及其产品时将使用各自公司所拥有的商标，这种使用的目的仅限于引用。本文档可能涉及北京云知声信息技术有限公司的专利（或正在申请的专利）、商标、版权或其他知识产权，除非得到北京云知声信息技术有限公司的明确书面许可协议，本文档不授予使用这些专利（或正在申请的专利）、商标、版权或其他知识产权的任何许可协议。

不作保证声明

北京云知声信息技术有限公司不对此文档中的任何内容作任何明示或暗示的陈述或保证，而且不对特定目的的适销性及适用性或者任何间接、特殊或连带的损失承担任何责任。本手册内容若有变动，恕不另行通知。本手册例子中所用的公司、人名和数据若非特别声明，均属虚构。未得到北京云知声信息技术有限公司明确的书面许可，不得为任何目的、以任何形式或手段（电子的或机械的）复制或传播手册的任何部分。

保密声明

本文档（包括任何附件）包含的信息是保密信息。接收人了解其获得的本文档是保密的，除用于规定的目的外不得用于任何目的，也不得将本文档泄露给任何第三方。

本软件产品受最终用户许可协议（EULA）中所述条款和条件的约束，该协议位于产品文档和/或软件产品的联机文档中，使用本产品，表明您已阅读并接受了EULA的条款。

版权所有©北京云知声信息技术有限公司

Copyrights © Beijing USC Information Technology CO., LTD.

目录

[1. 概述 1](#_Toc454801316)

[1.1. 目的 1](#_Toc454801317)

[1.2. 范围 1](#_Toc454801318)

[2. 使用说明 1](#_Toc454801319)

[2.1. 开发说明 1](#_Toc454801320)

[2.2. 开发前准备 1](#_Toc454801321)

[2.3. 支持的平台 1](#_Toc454801322)

[2.4. SDK运行简单时序图 2](#_Toc454801323)

[3. 口语评测 3](#_Toc454801324)

[3.1. 评测对象（USCRecognizer） 3](#_Toc454801325)

[3.1.1. 获取SDK版本号 3](#_Toc454801326)

[3.1.2. 创建对象 3](#_Toc454801327)

[3.1.3. 创建对象（混合识别引擎、离线引擎） 3](#_Toc454801328)

[3.1.4. 设置评测文本 3](#_Toc454801329)

[3.1.5. 设置代理 4](#_Toc454801330)

[3.1.6. 设置评测模式 4](#_Toc454801331)

[3.1.7. 设置音频格式 4](#_Toc454801332)

[3.1.8. 设置企业客户配置参数 4](#_Toc454801333)

[3.1.9. 设置延时评测 5](#_Toc454801334)

[3.1.10. 开始评测 5](#_Toc454801335)

[3.1.11. 停止评测 5](#_Toc454801336)

[3.1.12. 取消测评 6](#_Toc454801337)

[3.1.13. 读取pcm格式的音频文件进行识别 6](#_Toc454801338)

[3.1.14. 设置vad前置端点和后置端点的静音时间 6](#_Toc454801339)

[3.1.15. 设置设备唯一标识 7](#_Toc454801340)

[3.1.16. 设置打分系数 7](#_Toc454801341)

[3.1.17. 混合模式下模式控制 7](#_Toc454801342)

[3.1.18. 混合模式下设置离线结果等待时长 8](#_Toc454801343)

[3.2. 评测代理（USCRecognizerDelegate） 8](#_Toc454801344)

[3.2.1. 开始回调 8](#_Toc454801345)

[3.2.2. 结束回调 8](#_Toc454801346)

[3.2.3 结果回调 9](#_Toc454801347)

[3.2.4 结束回调 9](#_Toc454801348)

[3.2.5 音量大小回调 9](#_Toc454801349)

[3.2.6 VAD超时回调 10](#_Toc454801350)

[3.2.7 录音数据回调 10](#_Toc454801351)

[3.2.8 获取录音数据url（混合识别引擎、在线引擎） 11](#_Toc454801352)

[3.2.9 引擎初始化结束（混合识别引擎、离线引擎） 11](#_Toc454801353)

[3.2.10 延时评测结果回调 11](#_Toc454801354)

[4. 使用示例 12](#_Toc454801355)

[4.1 Project设置 12](#_Toc454801356)

[4.2 测评示例 13](#_Toc454801357)

[4.3 OralTask参数说明 13](#_Toc454801358)

[5. 结果格式说明 15](#_Toc454801359)

[5.1 返回结果格式（Json） 15](#_Toc454801360)

[5.2 数据格式说明 19](#_Toc454801361)

[附1： 错误代码说明 20](#_Toc454801362)

[附2：新旧错误码对应说明 21](#_Toc454801363)

[附3：版本变更说明 23](#_Toc454801364)

# 概述

opus音频编码SDK旨在使第三方应用便利的集成和使用云知声的opus音频编码服务。

## 目的

本文档对opus音频编码SDK接口定义进行说明。

文档读者为使用opus音频编码SDK进行开发的产品设计师、软件工程师。

## 范围

本文档定义opus音频编码的使用说明、体系结构、API接口。

不包含核心部件的性能定义，也不包含其它配套或附赠产品的使用说明。

# 使用说明

## 开发说明

在开发应用程序时，仅需关注文档中所提供的接口函数而不用了解具体实现。

## 开发前准备

对于个人开发者，使用opus音频编码服务，需要经过我们的授权，请到“<http://dev.hivoice.cn>”注册成为我们的开发者。

## 支持的平台

支持iOS 6.0 及以上系统。

# opus音频编码

## 编码对象（UniOpus）

## opus编码函数

函数原型

-(void)appendAudioData:(NSData \*)data;

参数

|  |  |
| --- | --- |
| 名称 | 描述 |
| data | 传入需要编码的音频数据（按照buffer进行编码） |

## 停止编码

函数

-(void)stopEncode;

## 退出编码

函数

-(void)cancelEncode;

## 设置代理

函数

@property (nonatomic,assign)id <UniOpusDelegate>delegate;

参数

|  |  |
| --- | --- |
| 名称 | 描述 |
| delegate | Opus编码回调接口代理 |

## 编码代理（UniOpusDelegate）

## 返回编码数据

函数

-(void)opusDataDidEncode:(NSData \*)encodeData;

说明

1. 编码完成后，返回编码数据。
2. 建议每次返回buffer时进行编码，录音完成后再编码，由于音频文件过大编码耗时较多可能会造成线程阻塞。

## 编码完成回调

函数

-(void)opusDataDidFinished;

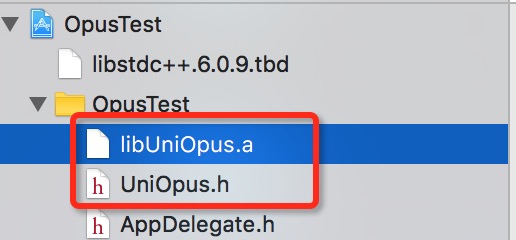
说明

1.将音频buffer传入后，会进行编码，当所有的buffer编码完成后会回调这个函数。

# 4. 使用示例

## 4.1 Project设置

1、将UniOpus\_Libs目录下所有文件拖到Xcode工程中的libs文件夹下，在弹出的窗口中勾选 Copy items into destination group’s folder(if needed)，确保TARGETS->Build Phases->Link Binary With Libraries引用libUniOpus.a



2、在 TARGETS->Build Phases->Link Binary With Libraries 标签下，点击按钮“+”，在弹出的窗口中选中AVFoundation.framework ，libstdc++.6.0.9.dylib等添加至工程

3、然后在你的程序中导入头文件#import "UniOpus.h"

4、将Info.plist文件中的Bundle identifiter 替换成你公司自己的Bundle identifiter

5、在TARGETS->Build Settings->Code Signing Identity中替换自己的证书

说明

1、目前sdk只提供真机版本。

2、静态库采用c++混编，请将相应的.m文件改为.mm文件，开启c++编译选项。

3、为了防止编译错误，请将C++ Standard Library设置为Compiler Default或libc++。

4、目前SDK不支持armv6的设备。

## 4.2 代码示例

//在调用startOpus之后，调用该函数进行编码

- (void)appendAudioData:(NSData \*)audioData

{

[uniOpus appendAudioData:audioData];

}

#pragma mark - UniOpus

-(void)startOpus

{

//启动opus编码模块

if (uniOpus)

{

uniOpus = nil;

}

uniOpus = [[UniOpus alloc]init];

uniOpus.delegate = self;

}

-(void)stopOpus

{

if (uniOpus)

{

[uniOpus stopEncode];

}

}

-(void)cancelOpus

{

if (uniOpus)

{

[uniOpus cancelEncode];

}

}

#pragma mark - UniOpusDelegate

-(void)opusDataDidEncode:(NSData \*)encodeData

{

//backup for recognition

if (socketRecognition)

{

[socketRecognition appendData:encodeData];

}

//backup for HttpFront

[httpData appendData:encodeData];

}

-(void)opusDataDidFinished

{

USCLog(@"recognition -> opusDataDidFinished");

opusFinish = YES;

[self finishRecognition];

}

# 附1：版本变更说明

|  |  |  |  |
| --- | --- | --- | --- |
| 版本号 | 日期 | 修订人 | 描述 |
| 1.0 | 2016/7/5 | 裴新华 | UniOpusSDK第一个版本实现 |