



AVSSDK_For_Android 用户集成指南V2.4.2

深圳声联网科技有限公司

AVSSDK_For_Android_用户集成指南：修正记录

版本	发布日期	内容描述
V1.0	2016/06/20	正式版
V2.0	2017/03/20	优化了基础模型。更新了接口参数。增加了获取版本接口。
V2.2	2017/12/04	优化了基础模型、算法，更新了接口及参数。
V2.4.1	2017/12/20	优化了基础模型、算法，更新了接口及参数。
V2.4.2	2018/01/15	优化了基础模型、算法，修改了部分接口，更新了验证方式。

地址：广东省深圳市南山区松坪山路 1 号源兴科技大厦北座 406。



0755-33349168 33349598



0755-33349798



www.avsnest.com

重要声明

版权声明

版权归深圳声联网科技有限公司所有，保留所有权利。

商标声明

深圳声联网科技有限公司的产品是深圳声联网科技有限公司专有。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。本文档可能涉及深圳声联网科技有限公司的专利（或正在申请的专利）、商标、版权或其他知识产权，除非得到深圳声联网科技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

深圳声联网科技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。未得到深圳声联网科技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：深圳声联网科技有限公司



目录

1. 概述	1-4
2. 平台支持	2-4
3. SDK使用向导	3-4
3.1 开发准备	3-4
3.2 枚举定义	3-5
3.3 接口	3-5
4. 注意事项	4-7
5. 引擎重新激活	5-7
6. 错误码表	6-8

AVSNEST CONFIDENTIAL



1. 概述

AVSSDK_For_Android 是深圳声联网科技有限公司在 Android 平台上对声联网婴幼儿情感计算算法的封装。用户只需要简单的调用封装后的接口就可以在 Android 平台的产品上集成声联网婴幼儿情感计算算法的功能。

2. 平台支持

AVSSDK_For_Android 支持几乎所有 Android 系统的嵌入式平台。

3. SDK 使用向导

3.1 开发准备

在使用 AVSSDK_For_Android 进行开发之前，先准备开发需要的文件。

Jar 包:

文件名	说明
avssdk.jar	avssdk.jar 是 so 包的入口函数封装。 avssdk.jar 复制到工程的 app/libs 目录下并导入工程中即可。

参考源码文件:

文件名	说明
avssdkdemo	该工程为 avssdk 的综合使用工程，展示基本的哭声检测效果，具体应用场景不做详细说明。

动态库文件:

文件名	说明
Libs/平台文件夹/libAVSNestSDK.so	哭声检测算法核心库，对应 7 种不同的 CPU 架构 arm64-v8a、Armeabi、armeabi-v7a、mips、mips64、x86、x86_64 在 android 手机中只要调用 Armeabi、armeabi-v7a、x86 三种 CPU 即可，此时需要进行库的过滤设置，在 app.gradle 中加入（请参考 demo） ndk{ abiFilters "armeabi","armeabi-v7a","x86" } 将 libs 文件夹下的所有内容复制到工程目录 app/src/main/jniLibs 下



3.2 枚举定义

声音事件返回类型

```
public enum AVSEventE
{
    AVS_EVENT_ERROR(-1),           //错误状态
    AVS_EVENT_NOR (0),            //中间状态
    AVS_EVENT_MUTE(1),            //静音状态
    AVS_EVENT_PUBLIC(2),          //公共状态
    AVS_EVENT_CRY(3),             //哭声状态
    AVS_EVENT_LAUGH(4),           //笑声状态 (未实现)
    AVS_EVENT_SOUND_ABNORMAL(5), //异常声音 (AVSSDK 上未实现)
    AVS_EVENT_ACTIVE(6)
};
```

3.3 接口

■ public State avsInits(Context mContext,String eth,String deviceid);

声音事件检测引擎初始化。

参数:

mContext [in] 数据上下文

eth [in] 当前通信网卡名称字符串如 “wlan0”。

deviceid [in] 当前设备唯一序列号。(序列号应为长度小于 50 byte)

返回值: 成功返回 AVS_INIT_SUCC; 失败返回 AVS_INIT_FAIL

当返回失败时, 调用 **public int avsGetError();** 获取错误码。

设备初始化时会在内部进行网络通信, 网络通信超时为 5 秒。如果 5 秒没有连接上服务器, 会直接返回失败 (-1)。这时使用 **avsGetError ()** 可以获取到错误码。

■ public State avsActives (Context mContext,String eth,String deviceid);

声音事件检测引擎激活。

参数:

mContext [in] 数据上下文。

eth [in] 当前通信网卡名称字符串如 “wlan0”。



deviceid [in] 当前设备唯一序列号。（序列号应长度小于 50 byte）

返回值：成功返回 **AVS_INIT_SUCC**；失败返回 **AVS_INIT_FAIL**

当返回失败时，调用 **public int avsGetError()**；获取错误码。

设备初始化时会在内部进行网络通信，网络通信超时为 5 秒。如果 5 秒没有连接上服务器，会直接返回失败 (-1)。这时使用 **avsGetError ()**可以获取到错误码。

■ **public AVSEventE avsDetect(short[] data, short size);**

传入音频数据进行声音事件检测。

参数：

data [in] 需要传入的音频数据 buffer。

size [in] 需要传入的音频数据采样个数（范围：[160~8000]个采样点）。

返回值：

返回检测结果 **AVSEventE** 中的一种。

注明：当返回值为 **AVS_EVENT_ACTIVE** 时，表示检测引擎失活，需要重新联网激活。

激活方式详见[引擎重新激活](#)

■ **public void avsUinit();**

销毁声音事件检测引擎。

参数：

无

返回值：

无

■ **public String avsGetVersion();**

获取 AVSSDK 的版本号。

参数：

无

返回值：

String 类型版本号。

■ **public int avsGetError();**

当其 SDK 出错或者异常时，通过此接口获取出错描述编码。

参数：



无

返回值:

错误描述编码。

4. 注意事项

- 接口使用方法请参照 demo 中的例程。
- 若您拿到的是 16K 版本、录制的音频流必须为 16K 16bit PCM 格式。
- 若您拿到的是 8K 版本、录制的音频流必须为 8K 16bit PCM 格式。
- 请将下面权限配置代码复制到 AndroidManifest.xml 文件中

```
<uses-permission android:name="android.permission.READ_PHONE_STATE"/>
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.RECORD_AUDIO"/>
```

权限说明

名称	用途
android.permission.RECORD_AUDIO	获取录音权限，进行音频录制作为识别数据源
android.permission.ACCESS_WIFI_STATE	获取当前 WiFi 接入的状态以及 WLAN 热点的信息
android.permission.INTERNET	访问网络连接，可能产生 GPRS 流量
READ_PHONE_STATE	访问手机电话状态功能

5. 引擎重新激活

本“声音事件检测引擎”以下简称“引擎”，内置了联网验证和监控模块，引擎调用初始化接口 avsnits时会联网进行验证。验证成功才能进行事件检测。

引擎在初始化成功后，并非永远保持存活状态。判断引擎失活的重要条件即是检查 AVSEventE avsDetect(short[] data, short size) 返回值，当 avsDetect 返回值为 AVS_EVENT_ACTIVE 时表示引擎即将失活，需要在短期内重新激活一下，否则会引擎失效。

激活方式可以采用如下两种方式：

- 重新初始化方式



当 avsDetect 接口返回值出现 AVS_EVENT_ACTIVE 时，立即调用 avs_fini 接口销毁当前失活的引擎，然后重新调用 avsInits 建立新的检测引擎，继续调用 avsDetect 检测即可。

- 使用激活接口激活方式

当 avsDetect 接口返回值出现 AVS_EVENT_ACTIVE 时，立即调用 avsActives 接口联网激活当前引擎，激活成功后继续调用 avsDetect 进行检测。

avsInits 及 avsActives需要联网授权获取激活码。联网授权需要短暂耗时，耗时情况视网络状况而定。因此在产品使用中需要注意，当需要进行激活操作时，需要进行录音缓存。

6. 错误码表

avsGetError () 返回值请查表分析原因

错误码(int)	描述
100	基本错误码
101	未知错误
102	参数错误
103	内存操作错误
104	引擎内部错误
10000	获取MAC错误
10001	域名解析错误
10002	创建会话错误
10003	连接服务器错误
10004	发送错误
10005	接收错误
10006	关闭会话错误
10007	连接服务器超时
50005	非法授权
50006	用户被禁用
50007	未通过授权
50008	数量超限
50009	授权过期
60001	未知错误
其他值	如返回其他非上述错误码，请联系FAQ