

# West's blog (<http://szuwest.github.io>)

## 声纹识别之Android手机录音采集探讨 (<http://szuwest.github.io/sheng-wen-shi-bie-zhi-androidshou-ji-lu-yin-cai-ji-tan-tao.html>)

2017-10-08

## 声纹识别之Android手机录音采集探讨

声纹识别是人工智能语音方向的一个重要分支。最近人工智能的兴起也带动了声纹识别这一领域的发展。声纹识别也越来越准确。不过在声音采集方面，特别是手机端的声音采集，是一个很讲究的问题。我加入人工智能声纹识别公司也有两个多月了，对手机端的语音采集也有了一定的了解，准备整理一下做成文章，跟大家探讨。今天重点说Android手机的录音采集方面的东西。

### 语音采集的一些基础指标

首先要搞清楚的一点是我们的人声是一些声波，是模拟信号，我们计算机的采集是将这些模拟信号转化为数字信号存储起来。我们播放语音文件的时候，是将数字信号转化为模拟信号。说实话，这一块的知识我大学也学得不好，模电我们软件工程的学生没有专门的课程，数电倒是有。不过数电模电的转化这个概念这倒不难理解。这里面就涉及到了采样率和采样位数这两个指标。

- 采样率，或者叫采样频率，指每秒钟取得声音样本的次数。采样率越高，数据越精确。我们也需要了解常用的采样率是多少：8k(8000)，16k, 44.1k, 48k。8k是电话所用的采样率，对于我们人说的声音的频率，基本在这个采样率之内。所以电话就才采用这个频率采样。48k采样率是CD，DVD所采用的。超过这个频率人耳是分辨不出来的了。
- 采样位数，说的是每个采样数据占的位数。每个采样数据记录的是振幅，采样精度取决于采样位数的大小。常用的位数是8位，也就是一个字节。还有16位或者32位。

除此之外，还有一个声道数也很重要。声道数也叫通道数，即声音的通道的数目。常见的单声道和立体声（双声道），现在发展到了四声环绕（四声道）和5.1声道。我现在接触到的只有单声道和双声道。

- 单声道.单声道的声音只能使用一个扬声器发声，有的也处理成两个扬声器输出同一个声道的声音，当通过两个扬声器回放单声道信息的时候，我们可以明显感觉到声音是从两个音箱中间传递到我们耳朵里的，无法判断声源的具体位置。
- 双声道 双声道就是有两个声音通道，其原理是人们听到声音时可以根据左耳和右耳对声音相位差来判断声源的具体位置。声音在录制过程中被分配到两个独立的声道，从而达到了很好的声音定位效果。这种技术在音乐欣赏中显得尤为有用，听众可以清晰地分辨出各种乐器来自的方向，从而使音乐更富想象力，更加接近于临场感受。双声目前最常用途与两个，在卡拉OK中，一个是奏乐，一个是歌手的声音；在VCD中，一个是普通话配音，一个是粤语配音。

## 音频格式

说完录音采集的指标，需要说说声音的存储。那些指标对于我们大部分人来说，根本就没有多少了解，反而音频格式听得比较多。音频存储下来是按照一定的格式的，例如我们用的最多的MP3。实际上它是一种经过编码压缩过的音频。经过录音设备采集的声音，未经编码，我们叫PCM数据。怎么算它的存储空间呢，这里有个概念叫比特率，也叫码率，就是一秒中数据有多大。计算方式也易理解：采样率x采样位数x声道数，单位就是bit/秒，当然我们要用字节数来表述的话，需要再除以8。

在手机平台中，一般都是 $(16000 \times 16 \times 1) / 8 = 32000 \text{ byte/s} \sim 32 \text{ KB/s}$ 。一分钟的原始音频大约1.92MB, 4分钟的大约7.68MB，还是挺大的。所以我们的音乐格式一般都要经过编码压缩，例如MP3，4分钟的歌曲大概也就4MB左右。但是对于人工智能的声纹识别来讲，是不能压缩的，因为压缩就会失真，一些重要的细节就会丢失，这些细节对声纹识别却很重要。一般我们就直接使用pcm数据或者WAV数据。WAV格式的音频，就是在pcm数据前面再加个文件头。文件头里说明了采样率，采样位数，声道数和文件大小等信息，然后很多音频播放器就能读取这个头然后播放音频。没有这个头的话，音频播放器是播放不了的。

## Android录音类

说了这么多录音相关的指标概念，接下来说说Android中是怎么录音的，将这些相关的指标如何对应的。

Android系统中提供了两个录音相关的类，一个是MediaRecorder，一个是AudioRecord。MediaRecorder集成了录音，编码等功能，使用前配置好参数，编码格式，音频存储路径，发起开始录音和结束录音命令，最终你就会得到一个音频文件。AudioRecord需要使用者自己去配置参数，设定音频存储buffer大小，然后需要自己读取数据，处理录音数据。

MediaRecorder较简单易用，音频存储空间小，而AudioRecord需要对音频实时处理，它输出的是pcm语音数据，空间占用大。对于一些只需要记录语音的应用来说，MediaRecorder更好用，而对于像声纹识别这些应用场景来说，AudioRecord是必然选择。

## 录音源

Android录音采集还有一个重要的参数，叫录音源audioResource。我们看一下AudioRecord的构造函数。

```
public AudioRecord(int audioSource, int sampleRateInHz, int channelConfig, int audioFormat, int bufferSizeInBytes)
```

第一个参数audioSource就是录音源，sampleRateInHz是采样率，channelConfig是声道数，audioFormat是采样位数，bufferSizeInBytes是采样时缓冲区大小，硬件设备采样的数据会先存到这个缓存区，我们程序从这个缓存区读取数据。

Android系统中还专门为这个audioSource定义不少的常量。一般来说我们的录音源不就是麦克风吗，为啥还会有这么多的取值？

```
/** 默认声音 */
public static final int DEFAULT = 0;

/** 麦克风声音 */
public static final int MIC = 1;

/** 通话上行声音 */
public static final int VOICE_UPLINK = 2;

/** 通话下行声音 */
public static final int VOICE_DOWNLINK = 3;

/** 通话上下行声音 */
public static final int VOICE_CALL = 4;

/** 根据摄像头转向选择麦克风 */
public static final int CAMCORDER = 5;

/** 对麦克风声音进行声音识别，然后进行录制 */
public static final int VOICE_RECOGNITION = 6;

/** 对麦克风中类似ip通话的交流声音进行识别，默认会开启回声消除和自动增益 */
public static final int VOICE_COMMUNICATION = 7;

/** 录制系统内置声音 */
public static final int REMOTE_SUBMIX = 8;

/**对麦克风录音不处理，Android7.0才加入的 */
public static final int UNPROCESSED = 9;
```

我也不太明白为什么要搞这么多值，但是从它们的定义来看都是针对不同的场景。这里有三个是针对电话通话的，而且一般只给系统用。最常用的是MIC，这个就是主麦克风，这适用于大部分的场景。对于声纹识别场景来说，希望能做得更好，希望得到更原始更真实的数据。VOICE\_RECOGNITION和VOICE\_COMMUNICATION貌似也挺适合声纹识别场景，但是官方没有更多的资料，网上也很少。还有

数，我发现采用这个录音源获取到的声音数据，比较接近麦克风采集的原始数据，它的语谱图跟专门的麦克风采集的音频的语谱图很相似。这里说一下语谱图和声波图，这两种图都是鉴别声音质量和信息的方式。我看做算法的同事经常参考这两个图。我对这两种图也还不够了解。

经过我自己的一些试验，还有查阅的一些资料，我得出一个结论：Android手机的录音其实都在底层（可能是驱动层）做了处理，例如噪声抑制和自动增益，上层拿到的数据其实已经不是原始的数据了。为什么这样做呢，因为手机的使用环境很复杂，可能会很嘈杂，并且我们说话的时候，手机跟嘴之间还是有一定的距离的。如果不做自动增益和噪声抑制的话，那声音可能很小，还有噪声，这样再次播放出来就听不清了。所以底层肯定有做处理的。这个我不知道官方有没有文档说明，但是我觉得事实应该就是这样的。由于这是做在底层，而且上层API无法修改，后来Android4.2之后加入了一些类：NoiseSuppressor，AutomaticGainControl，AcousticEchoCanceller。可以控制是否开启自动增益，噪声抑制等功能。实际上测试结果是并不是所有的硬件都支持，而且开启了之后效果有一点，但是不是很明显。我得出的结论就是原本Android底层（驱动层）就开启了噪声抑制和自动增益的，而且这是可能是关不了的。在4.2之后的系统里，虽然有了相关API是取开启或者关闭这些功能，但是有些硬件就是不支持（摊手）。就算有支持了，也只是在底层的基础上，在进一步的加强而已。到了Android7.0系统，终于加了一个UNPROCESSED定义，支持这个定义的硬件，就真的去除了底层的自动增益和噪声抑制等效果的。这个定义对声纹识别还是很有用的。奈何这个是7.0之后才支持，市面上的这个系统的手机还是比较少。

以上的一些结论都是我个人得出的，不一定正确。如果有人指出我的不对或者补充，那将是更好。欢迎讨论（szhanfeng203@gmail.com）

## Android手机麦克风和双MIC降噪

说了这么多软件的东西，说一下硬件的东西。

Android手机至少有一个麦克风，位于手机底部。但是很多手机同时会有两个或以上的麦克风，一般位于手机顶部或者背部摄像头旁边。现在很多手机厂商会采用双麦克风进行降噪。双MIC降噪一般用在打电话的时候。因为打电话的时候，主麦克风（手机底部）靠近声源，而另外一个麦克风在听筒附近，离声源比较远。语音到达这两个MIC的幅度相差6个dB,被保留。而噪音到达两个MIC的幅度相当，被过滤。网上有不少资料说明，感兴趣的可以查查看。

不过在软件层面，audioResource指定为MIC是采用主麦克风，而CAMCORDER是摄像头侧的麦克风，貌似没办法指定两个麦克风都用。估计双MIC降噪只有在打电话时，系统自己采用的。不知道有没有做系统的同学知道这个。

对于Android手机的录音，其实有很多要讨论的。例如手机是如何对VOICE\_RECOGNITION和VOICE\_COMMUNICATION支持，NoiseSuppressor和AutomaticGainControl各厂商是如何处理的，都值得探讨，可惜这方面的资料太少，讨论的人也不多，我也没做过Android系统，所以很多是未知数，需要更多是试验。欢迎留言或者发邮件讨论（szhanfeng203@gmail.com）。

## Comments

0条评论

West's Blog

1 登录 ▾

♥ 推荐

🔗 分享

评分最高 ▾



开始讨论...

通过以下方式登录

或注册一个 DISQUS 帐号 (?)



姓名

来做第一个留言的人吧！

在 WEST'S BLOG 上还有

### 再谈iOS开发和Android开发之间的切换

1条评论 • 2年前



Ho YuanHou — 您好，請問您接不接Android源碼轉ios源碼的外包工程http://midisheetmusic.sourc...我的qq號 ...

### Coturn在iOS端应用和libevent的编译

3条评论 • 5个月前



MaoTao Tian — 你好，请问你的问题解决了么？

### 升级Xcode8后Jenkins打包问题

1条评论 • 1年前



琥珀川 — 你好，请问你写的那个脚本写在哪里呢？Jenkins 的 execture shell 里面吗？

### iOS系统相册上传不得不说的有些事儿

1条评论 • 2年前



constantine\_jerry — 你好，我想问一下，如何后台批量上传相册中的照片或视频呢？假如是这样的需求：1. 从相册选择 N 个照片或视频并上传2. ...

📧 订阅    在您的网站上使用 Disqus添加 Disqus添加    隐私

## SITE

[Archives \(http://szuwest.github.io/archives.html\)](http://szuwest.github.io/archives.html)

[Tags \(http://szuwest.github.io/tags.html\)](http://szuwest.github.io/tags.html)

## CATEGORIES

[Android \(http://szuwest.github.io/category/android.html\)](http://szuwest.github.io/category/android.html)

第5页 共6页

[编程架构 \(http://szuwest.github.io/category/bian-cheng-jia-gou.html\)](http://szuwest.github.io/category/bian-cheng-jia-gou.html)

2017/12/5 下午4:05

声明:本站为手机录音采集探讨

http://szuwest.github.io/sheng-wen-shi-bie-zhi-andr...

- [IOS \(http://szuwest.github.io/category/ios.html\)](http://szuwest.github.io/category/ios.html)
- [技术 \(http://szuwest.github.io/category/ji-zhu.html\)](http://szuwest.github.io/category/ji-zhu.html)
- [生活 \(http://szuwest.github.io/category/sheng-huo.html\)](http://szuwest.github.io/category/sheng-huo.html)
- [算法 \(http://szuwest.github.io/category/suan-fa.html\)](http://szuwest.github.io/category/suan-fa.html)

## LINKS

---

- [About \(http://szuwest.github.io/pages/about.html\)](http://szuwest.github.io/pages/about.html)
- [我的生活博客 \(http://szhanfeng203.blog.163.com/\)](http://szhanfeng203.blog.163.com/)
- [RSS \(http://szuwest.github.io/feeds/all.atom.xml\)](http://szuwest.github.io/feeds/all.atom.xml)

## SOCIAL

---

- [Github \(https://github.com/szuwest\)](https://github.com/szuwest)
  - [Twitter \(https://twitter.com/szuwest\)](https://twitter.com/szuwest)
  - [Facebook \(https://www.facebook.com/profile.php?id=1084651913\)](https://www.facebook.com/profile.php?id=1084651913)
  - [微博 \(http://weibo.com/szuwest\)](http://weibo.com/szuwest)
- 

West's blog by West