实验三、语音增强算法实现

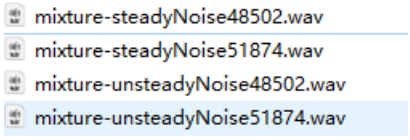
实践一、基于谱减法的语音增强（必做）

一、实验目的 ：

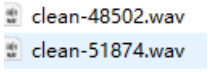
利用谱减法对语音进行降噪，了解谱减法的基本算法，分析不同种类的噪声对谱减法性能的影响，以及谱减法应用前后语音性能提升的效果。

二、数据准备

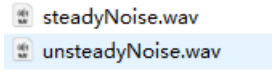
4 组带噪音频(mixture)：



2 组干净人声音频（clean）



2 组噪声（noise） 包括稳态噪声和非稳态噪声



注：广义来讲稳态噪声指信号随着时间变化不明显

非稳态噪声随时间变化比较明显

三、谱减法介绍

谱减法基于一个简单的假设：

语音信号中的噪声只有加性噪声，将带噪语音的频谱中噪声部分估计出，用原始音频的幅度谱减去噪声音频的幅度谱就可以获得干净的语音。

谱减法的基本流程：

x(k)—带噪语音 n(k)--噪声 s(k)—干净音频 （x(k) = n(k)+s(k)加性噪声）

1、对含噪语音进行傅里叶变换 X(w) = STFT(x(k))

2、同理，估计出噪声频谱 N(w)

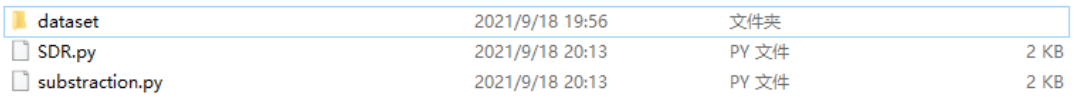
3、使用幅度谱进行谱减法操作

$$\hat{S}(w) = \begin{cases} |X(w)| - |N(w)| & \text{if} |X(w)| > |N(w)| \\ 0 & \text{else} \end{cases}$$

4、依据$\hat{S}(w)$和原始信号(mixture)的相位谱恢复成语音 $\hat{S}(k)$（estimate）

详细的可以参考文章：https://www.cnblogs.com/riddick/p/6848673.html

四、实践内容：



已经提前给出数据和谱减法代码(substraction.py)

一般情况下，我们不能拿到噪声的音频，因此只能利用混合音频(mixture)估计噪声频谱。一般认为前五帧是只包含噪声不包含人声的静音段，可以用此来估计噪声。

利用谱减法给每条混合音频降噪。

并对降噪的语音进行分析，可以从听感，语谱图以及 SDR 等方面进行分析。

注：SDR 全称 signal-to-distortion ratio，可以对语音进行估计，数值越大代表音频越纯净。

- 本实验提供的谱减法代码仅仅为基础版，谱减法也有很多变体。同学们也可参考文章自行修改编写其他种类的谱减法。

实践二、多通道语音增强（必做）

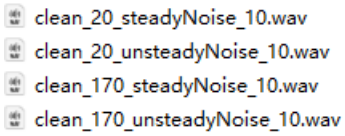
一．实验目的：

应用波束形成算法对多通道音频降噪，了解麦克风阵列数据降噪的基本流程。

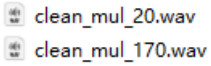
二．数据准备：

所有数据均为 5 通道音频数据。

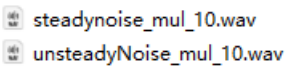
4 组带噪音频 mixture：例如 clean_20_steadyNoise_10 代表人声传来的角度是 20，稳态噪声传来的角度是 10。



2 组干净音频 clean：



2 组噪声 noise：

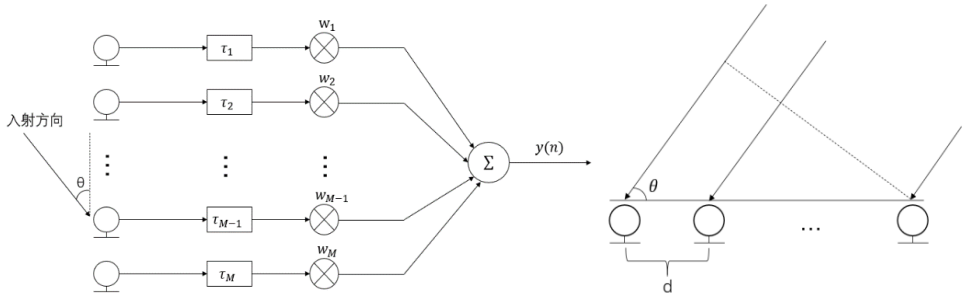


三．固定系数的波束形成介绍(delay and sum beamforming)：

对于麦克风阵列而言，由于各个麦克风的分布位置不同，阵元接收的语音信号会存在一定的时间差。利用这一信息可以确定声源的方向和位置。通过对齐各个通道的信号，相位差异可以将干扰部分抵消掉，增强目标语音

信号。

本次实验主要构建固定波束形成的方法(延迟求和波束形成 delay-and-sum beamforming)，即加权系数保持不变。



以右图线性麦可风为例，假设麦克风之间的距离是 d，则加权系数

$$\boldsymbol{w_i = e^{\frac{-j2\pi f(i-1)d\cos\theta}{c}}},$$

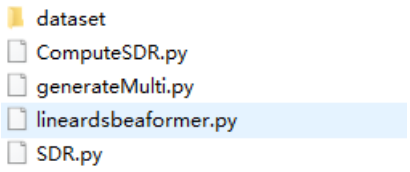
其中 c 代表声音传播速度 340m/s, f 代表频率，(i-1)d 代表第 i 个麦克风和第一个麦克风之间的距离，j 代表虚部。增强后的语音为

$$\mathbf{y(n) = \sum_{i=1}^{M} w_i x_i}$$

x_i代表第 i 个通道的语音。

注：delay-and-sum beamforming 是一种非常简单的多通道增强算法，因此增强后的效果不是很好。

四．实践内容：



本实验已经提供数据 dataset。

Lineardsbeamformer.py: delay-and-sum 算法的具体实现，通过更改 doa 的数值可以增强对应角度的声音而抑制其他角度的声音。 一般来说当人声角度和噪声角度之间的差异越大，音频增强后的效果越好。如果两个角度相差不大则不容易区分。

ComputeSDR.py:用来计算 clean 和 mixture 以及 estimate 和 clean 之间的 SDR 大小。注意：多通道音频无法直接计算 SDR 一般提取第 0 个通道的音频（单通道）来计算 SDR。

GenerateMulti.py: 用来生成多通道的音频数据，需要指定声源的角度和声源的距离以及单通道的声音信号。由于本实验仅仅提供了 4 组音频，同学

们可以自行生成其他角度的多通道音频信号。（选做）

五. 实验分析

可以通过对比音频增强前后的 SDR、听感差异、语谱图差异等方面来分析 delay-and-sum 的效果。

六. 参考资料

https://www.cnblogs.com/LXP-Never/p/12239399.html

https://blog.csdn.net/weixin_38260878/article/details/99709915

https://blog.csdn.net/miao0967020148/article/details/107075951

有关波束形成的详细介绍可以参考以上

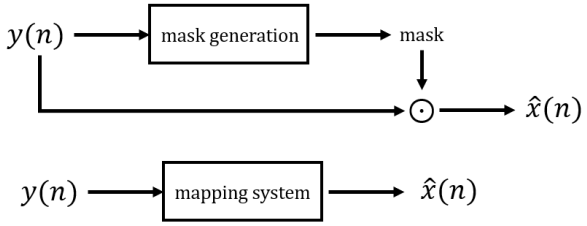
实践三、基于深度学习的语音增强（可选）

一、实验内容

利用深度学习来构建语音增强算法（单通道即可）。可以使用自己电脑 GPU 进行训练，CPU 也可以（训练速度会很慢）。

二、实验内容

基于深度学习的语音增强方法具有很强非线性拟合能力，在处理非稳态的噪声时，能得到很好的性能。 基于深度学习的语音增强，主要分为两类：基于 masking 的目标和基于 mapping 的目标。



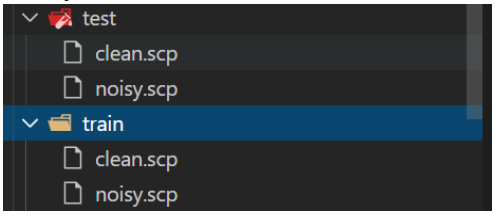
本实验中可以利用任意目标，任意算法结构构建基于深度学习的语音增强算法。

详细有关语音增强的 DNN 算法可以自行查找网络资料，此处给一些参考网址：https://zhuanlan.zhihu.com/p/139423710

https://www.cnblogs.com/LXP-Never/p/14142108.html

三、 数据准备：

Noisy： 带噪声的语音 clean： 干净的语音



文件包含数据路径 格式为：名称.wav 路径

四 、 代码可以自行查找 此处给出一些参考代码

使用 Conv-tasnet 网络进行语音增强。

参考论文：https://arxiv.org/abs/1809.07454
代码（非原始论文代码）：https://github.com/mrjunjieli/Conv_Tasnet