

本文是音频处理的朋友icoolmedia (QQ: 314138065) 的投稿。对音频处理有兴趣的朋友可以通过下面的方式与他交流：

作者：icoolmedia

QQ：314138065

音视频算法讨论QQ群：374737122

原文公式较多，因此直接贴上图片。

发现很多朋友想进入语音降噪处理的大门，却很容易被铺天盖地的理论弄的很迷惑，不知道从哪里开始比较好。网上给出的参考文章大多干说理论，没有代码实现。很不利于学习。于是打算写这篇语音降噪的文章，并给出相应的实现代码方便交流和进一步的学习。

### 一. 算法核心思想与流程概要

本文给出的降噪算法的核心流程很简单，主要是两个步骤：

1. 环境噪声的假设与估计
2. 频域维纳滤波降噪

这里简单说一下语音相位问题，通常在语音的降噪算法中，是不考虑纯净语音的相位问题的，这是因为理论上可以证明：**带噪语音相位就是纯净语音相位的最优估计！**

### 二. 环境噪声估计

首先我们假设环境噪声为随机平稳加性噪声、且与语音信号不相关。要注意这个假设对我们很重要，要记住这个算法的前提条件。这里介绍的噪声估计算法叫做**连续谱最小值跟踪**，资料出处为“语音增强-理论与实践”一书中第九章中的一部分内容。此算法利用了带噪语音信号在单个频带的功率通常会衰减到噪声的功率水平。即使在语音活动期间也是如此。

算法主要分为两个步骤：

1. 对各频点带噪语音功率谱进行平滑处理。短时平滑方式为：

$$P(\lambda, k) = \alpha P(\lambda, k) + (1 - \alpha) |Y(\lambda, k)|^2$$

这里， $P(\lambda, k)$ 为平滑后的第 $k$ 帧、 $\lambda$ 频点的语音信号功率谱， $\alpha$ 为平滑因子（通常取值为： $0.7 \leq \alpha \leq 0.9$ ）。

2. 对各频点带噪语音功率最小值进行非线性跟踪。

if  $P_{\min}(\lambda - 1, k) < P(\lambda, k)$

$$P_{\min}(\lambda, k) = \gamma P_{\min}(\lambda - 1, k) + \frac{1 - \gamma}{1 - \beta} (P(\lambda, k) - \beta P(\lambda - 1, k))$$

else

$$P_{\min}(\lambda, k) = P(\lambda, k)$$

end

这里，算法中的非线性跟踪会连续对噪声功率进行估计，这主要是由于上式中的第二项实现了一个一阶差分运算，是在离散情况下对求导的一种近似。当带噪语音功率 $P(\lambda, k)$ 增加时，导数值也会增加，因此差分 $P(\lambda, k) - P(\lambda - 1, k)$ 为正，当带噪语音功率下降的时候，导数为负，噪声估计减小。

这里算法唯一要注意的是：当语音谱上出现很窄的峰值时，可能会导致在语音活动期音对噪声的过估计，进而可能抑制语音。

### 三. 频域维纳滤波

使用维纳滤波进行语音降噪的过程，其实是把降噪过程视为一个线性时不变系统，当带噪语音通过这个系统时，在均方误差最小化准则下，使得系统的输出与期望的纯净语音信号最接近的过程。系统我们用下面的卷积过程来表示：

$$\hat{d}(n) = \sum_{k=0}^{M-1} h_k y(n-k)$$

其中 $y(n)$ 为输入信号， $\hat{d}(n)$ 为输出信号，如果我们转换到频域，表示为：

$\hat{D}(\omega) = H(\omega)Y(\omega)$ ，则频率 $\omega_k$ 处的估计误差为：

$$E(\omega_k) = D(\omega_k) - \hat{D}(\omega_k) = D(\omega_k) - H(\omega_k)Y(\omega_k)$$

在频域使其均方误差  $E[|E(\omega_k)|^2]$  最小化:

$$\begin{aligned} J_2 &= E[|E(\omega_k)|^2] \\ &= E\{[D(\omega_k) - H(\omega_k)Y(\omega_k)]^* [D(\omega_k) - H(\omega_k)Y(\omega_k)]\} \\ &= E[D(\omega_k)^2] - H(\omega_k)E[D^*(\omega_k)Y(\omega_k)] - H^*(\omega_k)E[D(\omega_k)Y^*(\omega_k)] \\ &\quad + |H(\omega_k)|^2 E[|Y(\omega_k)|^2] \\ &= E[D(\omega_k)^2] - H(\omega_k)P_{yd}(\omega_k) - H^*(\omega_k)P_{dy}(\omega_k) + |H(\omega_k)|^2 P_{yy}(\omega_k) \end{aligned}$$

这里  $P_{yy}(\omega_k) = E[|Y(\omega_k)|^2]$  为输入信号的功率谱,  $P_{yd}(\omega_k) = E[D^*(\omega_k)Y(\omega_k)]$  是输入信号  $y(n)$  与期望信号  $d(n)$  的互功率谱, 为了得到最优的滤波器  $H(\omega_k)$ , 我们对均方误差  $J_2$  求关于  $H(\omega_k)$  的复导数, 并令其等于 0, 可得:

$$\begin{aligned} \frac{\partial J_2}{\partial H(\omega_k)} &= H^*(\omega_k)P_{yy}(\omega_k) - P_{yd}(\omega_k) \\ &= [H(\omega_k)P_{yy}(\omega_k) - P_{dy}(\omega_k)]^* = 0 \end{aligned}$$

由于  $P_{yd}(\omega_k) = P_{dy}^*(\omega_k)$ , 求解  $H(\omega_k)$ , 得到维纳滤波的通用形式:

$$H(\omega_k) = \frac{P_{dy}(\omega_k)}{P_{yy}(\omega_k)}$$

再由于我们假设噪声为加性噪声, 且与语音信号不相关, 可得:

$$\begin{aligned} P_{dy}(\omega_k) &= P_{yx}(\omega_k) \\ P_{yy}(\omega_k) &= P_{xx}(\omega_k) + P_{nn}(\omega_k) \end{aligned}$$

将上面两个式子代入维纳滤波器的通用形式, 得到频域维纳滤波器:

$$H(\omega_k) = \frac{P_{yx}(\omega_k)}{P_{xx}(\omega_k) + P_{nn}(\omega_k)} = \frac{\zeta_k}{\zeta_k + 1}, \text{ 这里 } \zeta_k = \frac{P_{yx}(\omega_k)}{P_{nn}(\omega_k)} \text{ 为频率 } \omega_k \text{ 处的先验 SNR。}$$

下面的算法实现中将会用到这个结论。当然, 如果想使用平方根维纳滤波器, 也可以对  $H(\omega_k)$  再进行一次开方, 这个就不再多说了。

#### 四. 可能的改进方法

总的说来, 这是一个降噪算法的小实现, 可以改进的地方真的是太多了:

- 结合对先验信噪比与后验信噪比的调整算法 (如: 判决引导法)
- 结合语音存在的概率进行调整
- 使用超几何增益, 而不是维纳增益
- 结合盲信号处理与阵列方向信息

这里不再一一列举, 实现代码请到音视频算法讨论 QQ 群 (374737122) 中自行下载 (TestNs), 欢迎就其它改进方法一起讨论!

文章标签: 音频 语音 降噪

个人分类: 音频编码

此PDF由spyyg生成, 请尊重原作者版权!!!

我的邮箱: liushidc@163.com