

## [转] [投稿] Speex回声消除代码分析

2015年12月03日 18:47:20 阅读数：25023

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

先说明下，这里的代码流程是修改过的Speex流程，但与Speex代码差异不大，应该不影响阅读。

- (1) 用RemoveDCOffset函数进行去直流
- (2) 远端信号预加重后放入x[i+frame\_size]，近端信号预加重后放入input缓冲区
- (3) 前M-1帧的远端频域信号移位，为当前帧频域信号腾出空间
- (4) 用spx\_fft函数进行FFT变换，变换后的系数存在X中
- (5) 计算当前远端信号当前帧的方差Sxx。（去直流操作后，意味着均值可以视为零）
- (6) 当前远端时域信号移位，x[i] = x[i+frame\_size]，50%交叠处理
- (7) 先用前景滤波器进行滤波
- (8) 将滤波后的输出进行反变换
- (9) 再用近端信号（麦克风语音）减去滤波输出得到前景滤波器的语差信号
- (10) 再用mdf\_inner\_prod计算前景滤波器的时域误差总功率
- (11) 使用mdf\_adjust\_prop计算背景滤波器各段（每个）相同频率的系数幅度，存放在prop（这里进行了归一化处理）。
- (12) 用weighted\_spectral\_mul\_conj函数计算背景滤波器频域系数需要调整多少量，（其思想为LMS：步长乘以W\*E）

power\_1 最优步长因子，具体可以参考Mader算法,与prop相乘得到最优步长  
prop 为归一化后的背景滤波器各频率的系数幅度。与power\_1相乘  
X 为转换到频域后的远端信号  
E 为转换到频域后的误差信号  
PHI 输出结果：背景滤波器频域系数需要调整的量

- (13) 背景滤波器频域系数调整： $W[j*N+i] = W[j*N+i] + PHI[i]$
- (14) 防止循环卷积处理（重叠保留法变循环卷积为线性卷积，把相关块的FFT系数中的后半部分置为零）
- (15) 根据调整后的背景滤波器对远端信号再进行频域滤波处理
- (16) 滤波后的输出进行反变换，存到y
- (17) 在时域计算两次滤波之间的误差及两次滤波之间误差的功率Dbf
- (18) 在时域计算背景滤波器滤波后的误差和背景滤波器滤波后误差的方差See。至此我们得到：  
Sff 前景滤波后的方差  
See 背景滤波后的方差  
Dbf 前景滤波与背景滤波之间方差
- (19) 利用Sff、See、Dbf来计算是否需要更新前景滤波器（总的来说，当Dbf值比较大时需要更新前景滤波器系数）
- (20) 如果需要更新前景滤波器系数则把背景滤波器系数给前景滤波器，并平滑背景滤波后的时域误差（前景滤波时域语差加上背景滤波器的时域输出），得到总误差，存在e[i+frame\_size]，这里的总误差，就是总的远端信号回声。
- (21) 根据总的远端信号回声，计算回声消除后的语音信号，并进行预加重处理，存在out输出缓冲区中（这个步骤是我自己调的）
- (22) 误差缓冲区移动，为下一帧处理做好准备
- (23) 计算背景滤波器输出与误差的互相关Sey、背景滤波器的自相关Syy
- (24) 再把语差信号与输出信号（前面都置为0）都是变换到频域
- (25) 再计算变换后的E、Y、X的功率谱，对应的分别是Rf、Xf、Xf
- (26) 通过Xf平滑得到远端信号功率谱power

$$power[j] = (ss\_1 * power[j]) + 1 + (ss * Xf[j]);$$

- (27) 当（前一帧的）泄露系数小于一个阈值（0.5）时，重新估算远端信号功率谱power
- (28) 计算ey协方差，得到e和y的相关系数Pey，标准差Pyy。（Pey为相关系数：ey协方差/y标准差）
- (29) 计算远端信号泄露系数

参考论文：On Adjusting the Learning Rate in Frequency Domain Echo Cancellation With Double-Talk

先计算递归平均系数：

alpha = beta0\*Syy/See; 这里beta0为泄露估计的学习速率  
alpha\_1 = 1-alpha;

再递归计算相关系数Pey和标准差Pyy

最后得到泄露系数： $\text{leak\_estimate} = \text{Pey}/\text{Pyy}$

(30) 计算残余回声与误差的比率，并做上下限处理。

$\text{RER} = (0.0001 * \text{Sxx} + 3.0 * \text{leak\_estimate} * \text{Syy}) / \text{See}$ ；

下限： $\text{Sey} * \text{Sey} / (1 + \text{See} * \text{Syy})$

上限：0.5

(31) 判断收敛条件： $\text{sum\_adapt} > \text{M}$  且  $\text{leak\_estimate} > 0.03$

如果滤波器收敛：

$r = 0.7 * \text{leak\_estimate} * \text{Yf} + 0.3 * \text{RER} * (\text{RF} + 1)$  // 计算残余回声的功率

// 计算最优步长因子，具体可以参考Mader算法，或者到本文作者那里寻求帮助

$\text{power\_1}[i] = r / (\text{e}^{*(\text{power}[i] + 10)})$ ;

如果没有收敛

当  $\text{Sxx} > \text{N} * 1000$  时，按如下计算adapt，否则adapt\_rate值为0

$\text{adapt\_rate} = 0.25 * \text{Sxx} / \text{See}$ ;

$\text{sum\_adapt} += \text{adapt\_rate}$

(32) 把残余回声保存在last\_y，这是为了语音增强时去掉回声的影响

icoolmedia声明：

本文为原创，欢迎转发到博客上，转发时第一页的版权信息必须保留！谢谢！

文章标签：

Speech

音频

回声消除

个人分类：[音频编码](#)

此PDF由spygg生成,请尊重原作者版权!!!

我的邮箱:liushidc@163.com