数据处理中的矩阵方法

--读书笔记

基于动态语音帧检测的子空间语音增强算法研究

硕士研究生：欧粤

学号：201828013726017

学科专业：物理电子学

所在单位：电子所高功率微波源实验室

完成时间：2019 年 5 月 23 日

摘 要

子空间算法与其它经典语音增强算法相比有很好的去噪效果，但是在低信噪比环境下，仍存在大量残留噪声 .针对这一点，本文提出在动态对数谱语音帧检测方法下对信号子空间优化的方法。结果表明，相比传统的谱减法、维纳滤波法、传统子空间算法，本文提出的改进子空间优化算法效果更佳，不仅能有效的抑制背景噪声，改善语音质量，并且在极强干扰的情况下仍能保持优异的性能。

【关键词】 语音端点检测 子空间 特征值分解 Nelder-Mead 优化

目录

1. 介绍

语音信号在传输过程中容易受到背景噪声的干扰，语音增强是解决噪声污染的有效方法，它的目标是尽可能从带噪语音信号中恢复出纯净的语音信号，改善语音听觉质量以减轻听觉上的疲劳。经典的语音增强算法包括谱减法、维纳滤波法和子空间算法等，首先谱减法[1-3]是最简单常见的语音增强算法，但是它容易产生音乐噪声。维纳滤波法 [4-6] 是一种基于最小均方准则的方法，通过设置滤波器系数尽可能消除噪声。该方法可以有效地抑制音乐噪声，但是低信噪比下语音失真度较大。

子空间 [7-10] 方法是近年兴起的一种语音增强算法，它利用投影矩阵把含噪语音投影到两个相互正交的子空间上，两个矩阵分别为信号子空间(对应纯净语音信号) 和噪声子空间(对应噪声信号)，通过将噪声子空间置零，从语音子空间中获得增强的语音信号。但由噪声分布在整个空间的，语音子空间也不可避免地残留一些噪声成分，因此，传统子空间方法不能完全消除噪声，尤其在低信噪比下增强语音中往往残留有较多的噪声成分。

为了解决这个问题，本文在经典子空间方法的基础上，提出用动态语音帧检测的方法对其进行优化。当算法检测出当前语音帧为噪声帧时，利用谱减法的思路进行处理，加快算法运行时间。而当前语音帧为信号帧时，对当前的噪声谱进行实时更新，进而改善降噪效果。数据结果表明，本文算法能有效的抑制背景噪声，改善语音质量，同时在低信噪比的情况下仍能有效不失真的增强语音信号。

本文的结构安排如下，第二章介绍基于对数谱距离的语音帧检测，第三章介绍子空间语音增强算法，第四章介绍改进后的算法。第五章为算法结果比较。

2 语音帧检测

语音活动检测(Voice Activity Detection,VAD)又称语音帧检测，是指在噪声环境中检测语音的存在与否,通常用于语音编码、语音增强等语音处理系统中,起到降低语音编码率、节省通信带宽、减少移动设备能耗、提高识别率的作用。

本文采用的语音帧检测方法为基于对数谱距离的端点检测，令含噪语音信号为 **x(n)**,分帧处理后得到的第 **i** 帧语音信号为$\mathbf{x_i(m)}$,每帧长为 **N** 对$\mathbf{x_i(m)}$进行 DFT 可得离散频谱为:

$$\boldsymbol{X_i(k) = \sum_{m=0}^{N-1} x_i(m) exp\left(-j\frac{2\pi km}{N}\right)\ 0 < k \le N-1} \tag{2-1}$$

DFT 后的频谱$\boldsymbol{X_i(k)}$取模值再取对数有:

$$\boldsymbol{\hat{X}_i(k) = log|X_i(k)|} \tag{2-2}$$

设有两个不同信号$\boldsymbol{x_0(n)}$和$\boldsymbol{x_1(n)}$,其第 i 帧的对数频谱分别为$\boldsymbol{\hat{x}_i^0(k)}$和$\boldsymbol{\hat{x}_i^1(k)}$，下标 i 表示第 **i** 帧，上标 0 和 1 表示不同的信号$\boldsymbol{x_0(n)}$和$\boldsymbol{x_1(n)}$，这两个信号的对数频谱距离表示为:

$$d_{spec}(i)=\frac{1}{N2}\sum_{k=0}^{N2-1}(\hat{X}_i^0(k)-\hat{X}_i^1(k))^2 \quad (2\text{-}3)$$

（上式 **N2** 只取正频率部分，当帧长为 **N** 时，**N2=N/2+1**）。

根据上述原理，编制的对数谱语音帧检测算法见下图 2.1，NOISEMARGIN 是语音段和噪声段之间的最小距离，是需要动态调节的量。若输出的 SPEECHFLAG=1，则本帧为语音帧，进行子空间语音增强。SPEECHFLAG=0,则本帧为噪声帧，直接利用谱减法的思路直接进行滤波，加快程序运行速度，同时跟踪记录噪声。

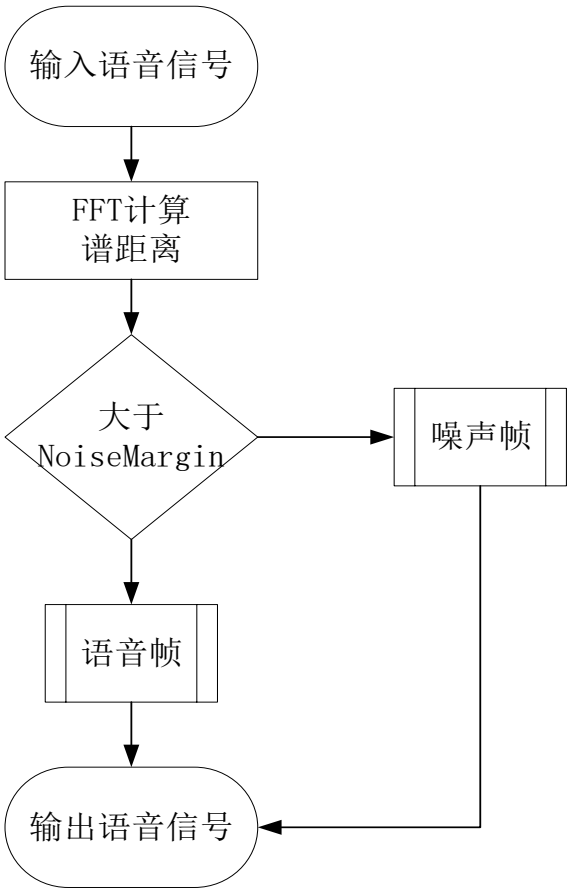

图 2.1 对数谱语音帧检测算法

输入任意语音信号，对其施加白色噪声，用对数谱距离法进行端点检测。其结果如下图 2.2 所示，可以看出在强噪声低信噪比的情况下，动态对数谱距离能够有效判断当前帧的状态。

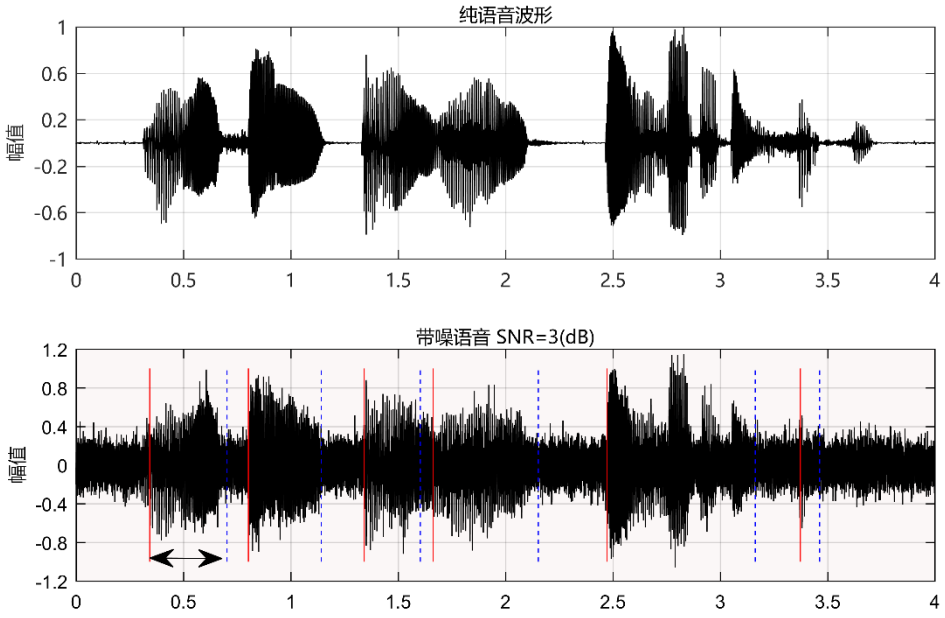

图 2.2 对数谱语音帧检测结果

从上文可以得出,对数谱语音帧检测结果与 **NOISEMARGIN**(谱距离门限值)有着重要的影响。将在纯净语音情况下语音帧检测结果记录为 **Ideal**,带噪声情况下语音帧检测结果记录为 **Real**，语音帧数定义为 $\boldsymbol{Q}$，精确度定义为

$$\boldsymbol{Precison} = \frac{\boldsymbol{Ideal==Real}}{\boldsymbol{Q}} \tag{2-4}$$

固定初始输入信号，施加不同强度的白噪声，通过改变 **NOISEMARGIN 阈值**，绘制出阈值与精确度的关系于下图 2.3

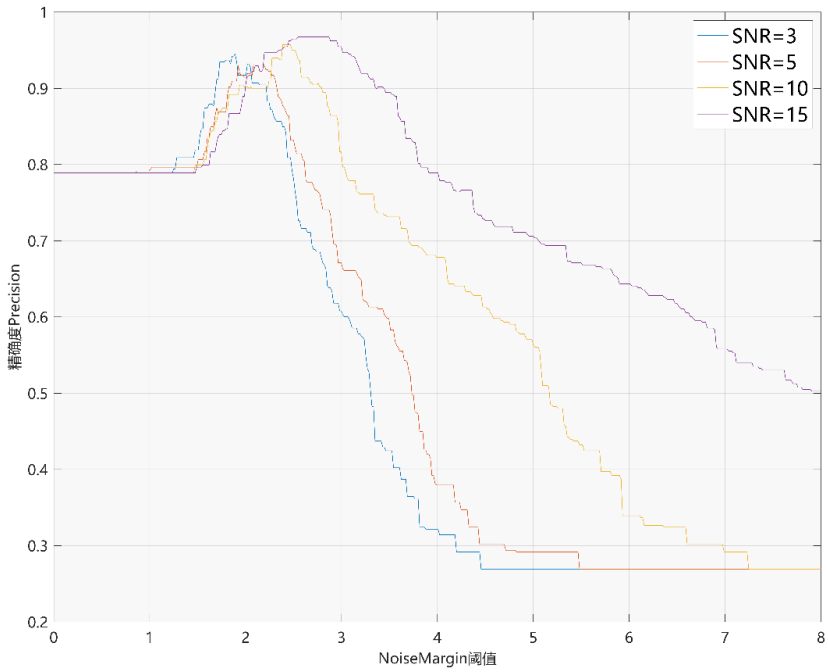

图 2.3 谱门限阈值与语音帧检测结果图

从上图可以看出，在同一初始信噪比的情况下，设置不同阈值，其语音帧检测结果不同。在同一阈值的情况下，不同信噪比的初试信号也会带来相异的预测结果。所以后文将会通过算法动态调节语音阈值参数，以此有效提升辨识能力，从而为为后续的子空间语音增强打下坚实的基础。

3. 子空间语音增强

子空间方法是通过空间分解，将整个空间划分为两个独立子空间，即噪声子空间和叠加噪声的信号子空间。然后通过去除噪声子空间并用最优估计器估计语音特征值来实现语音增强。该方法经过严格的数学推导，既满足了在保证残差信号的能量和频谱的同时，还能使估计信号的失真最小。

假设带噪语音信号 Y 和噪声信号 N 互不相关，且纯净语音信号为 S，令带噪语音为:

$$\boldsymbol{Y} = \boldsymbol{S} + \boldsymbol{N} \tag{3-1}$$

上式中，Y，S 和 N 分别为 K 维的带噪语音矢量，纯净语音矢量和噪声信号矢量。

令 $\boldsymbol{R_y}$ ，$\boldsymbol{R_s}$ 和$\boldsymbol{R_n}$ 分别表示 Y, S 和 N 的协方差矩阵，H 为 K X K 的线性预测器，则增强语音可以表示为:

$$\hat{\boldsymbol{S}} = \boldsymbol{HY} \tag{3-2}$$

预测值和真实值的误差为:

$$\boldsymbol{\varepsilon} = \hat{\boldsymbol{S}} - \boldsymbol{S} = (\boldsymbol{H} - \boldsymbol{I}) \cdot \boldsymbol{S} + \boldsymbol{H} \cdot \boldsymbol{N} = \boldsymbol{\varepsilon_S} + \boldsymbol{\varepsilon_N} \tag{3-3}$$

其中，$\boldsymbol{\varepsilon}_S$和$\boldsymbol{\varepsilon}_N$ 分别表示语音信号的失真和增强后残留的噪声，相应的能量为:

$$\overline{\boldsymbol{\varepsilon_S^2}} = \boldsymbol{E}[\boldsymbol{\varepsilon_S^T}\boldsymbol{\varepsilon_S}] = \mathbf{tr}(\boldsymbol{E}[\boldsymbol{\varepsilon_S^T}\boldsymbol{\varepsilon_S}]) = \mathbf{tr}\{(\boldsymbol{H}-\boldsymbol{I})\boldsymbol{R_S}(\boldsymbol{H}-\boldsymbol{I})^{\boldsymbol{T}}\} = \boldsymbol{tr}(\boldsymbol{HR_SH^T} - \boldsymbol{HR_S} - \boldsymbol{RH^T} + \boldsymbol{R_S}) \tag{3-4}$$

$$\overline{\boldsymbol{\varepsilon_N^2}} = \quad \boldsymbol{E}\left[\boldsymbol{\varepsilon_N^T}\boldsymbol{\varepsilon_N}\right] = \boldsymbol{tr}\left(\boldsymbol{E}\left[\boldsymbol{\varepsilon_N^T}\boldsymbol{\varepsilon_N}\right]\right) = \boldsymbol{tr}(\boldsymbol{HR_NH^T}) \tag{3-5}$$

最优约束估计器设计的思想是在约束条件下失真信号的能量最小，即:

$$\min_{\boldsymbol{H}} \overline{\boldsymbol{\varepsilon_S^2}} \ \text{（在（}\mathbf{1}/\boldsymbol{k}\text{）}\ \overline{\boldsymbol{\varepsilon_N^2}} \leq \boldsymbol{\sigma^2}\ \text{条件下}$$

根据该准则得到的估计器，对所有残留噪声范围为$\mathbf{k}\boldsymbol{\sigma}^{\mathbf{2}}$的线性滤波器的信号失真都做了最小化处理。当$\mathbf{k} \geq \mathbf{1}$时,满足约束条件且能得到最小信号失真的滤波器为 H=I。

对于上式的约束最优化可以用 Lagrange 乘子法来解决。它满足如下的 Lagrange 梯度方程

$$\boldsymbol{L}(\boldsymbol{H}, \boldsymbol{\mu}) = \overline{\boldsymbol{\varepsilon}}_{\boldsymbol{s}}^{\boldsymbol{2}} + \boldsymbol{\mu}(\overline{\boldsymbol{\varepsilon}}_{\boldsymbol{N}}^{\boldsymbol{2}} - \boldsymbol{\alpha K \sigma}_{\boldsymbol{N}}^{\boldsymbol{2}}) \tag{3-6}$$

由梯度$\nabla_H L(H, \mu) = 0$ 可以求得最优约束估计器:

$$\boldsymbol{H}_{\text{opt}} = \boldsymbol{R}_S(\boldsymbol{R}_S + \mu \boldsymbol{R}_N)^{-1} \tag{3-7}$$

对上式的协方差矩阵应用特征值分解,即$\boldsymbol{R_S} = \boldsymbol{U\Lambda_s U^T}$，可将最优约束估计器改写为:

$$\boldsymbol{H}_{\mathrm{opt}} = \boldsymbol{U}\boldsymbol{A}_S(\boldsymbol{A}_S + \mu\boldsymbol{A}_N)^{-1}\boldsymbol{U}^{-\mathrm{T}} \tag{3-8}$$

最终获得纯净语音信号的最优估计，表达式为：

$$\hat{\boldsymbol{S}} = \boldsymbol{H}_{opt} \cdot \boldsymbol{Y} \tag{3-9}$$

输入任意语音信号，并对其施加白色噪声，用经典子空间法对语音进行增强。其结果如下图 3.1 所示：

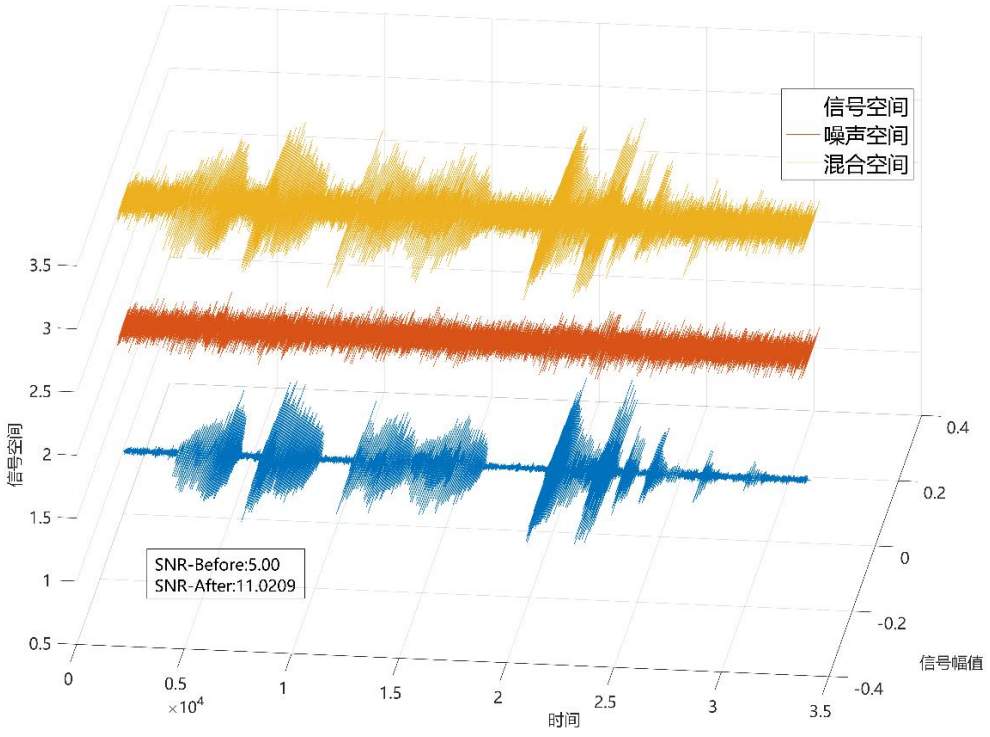

图 3,1 子空间语音增强结果图

当输入的信号信噪比为 5dB 时，经过子空间语音增强后，输出信号的信噪比强度为 11dB，可以看出在强噪声低信噪比的情况下，经典子空间法可以有效提升信号信噪比，因此达到增强语音信号的效果。

同时从上文公式说明$\boldsymbol{\mu}$对子空间语音增强算法有着重要的影响，图 3.2 绘制了不同初试信噪比情况下，不同 $\boldsymbol{u}$ 值对最后输出语音信号信噪比的影响。

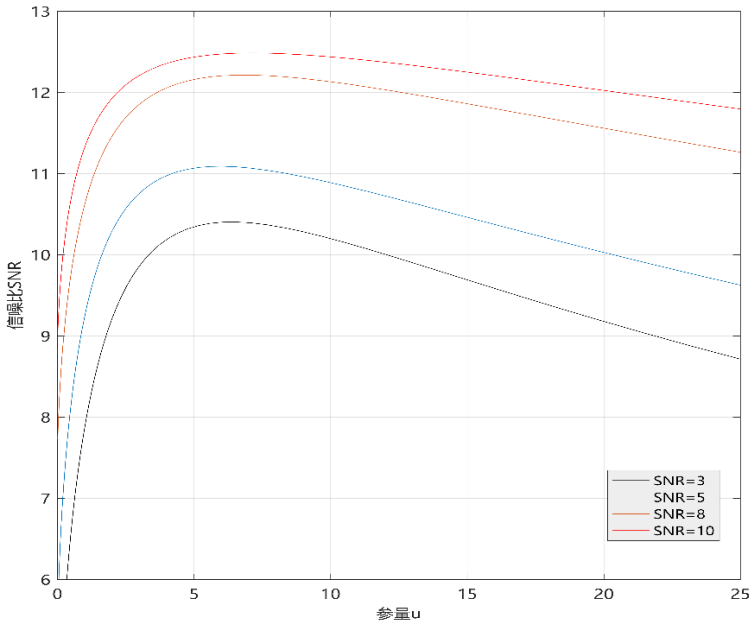

图 3.2 参量μ对子空间算法的影响

从上图可以看出$\boldsymbol{\mu}$是一个极其重要的参量，它可以决定子空间语音增强算法的性能，所以如何在算法中动态调整这个参量将会在下一部分进行讨论。同时值得注意的是在经典子空间算法中，其噪声值估计往往是取前 3000 个无话段语音点，这显然不具有普适性，所以在算法的运行过程中，需要时时更新噪声估计值。而语音帧检测可以有效解决这个问题。详细的说明见第四章。

4 动态语音帧算法介绍

经过第三章的讨论，可以发现传统的子空间方法滤波效果对参量 $\boldsymbol{u}$ 依赖程度较高，且算法本身无法判断当前语音帧的情况，导致对所有语音帧进行了特征值分解，这让算法运行速度下降。针对这种情况，本文提出了基于动态语音端点检测的方法对其进行优化，算法流程图见下图 4.1.

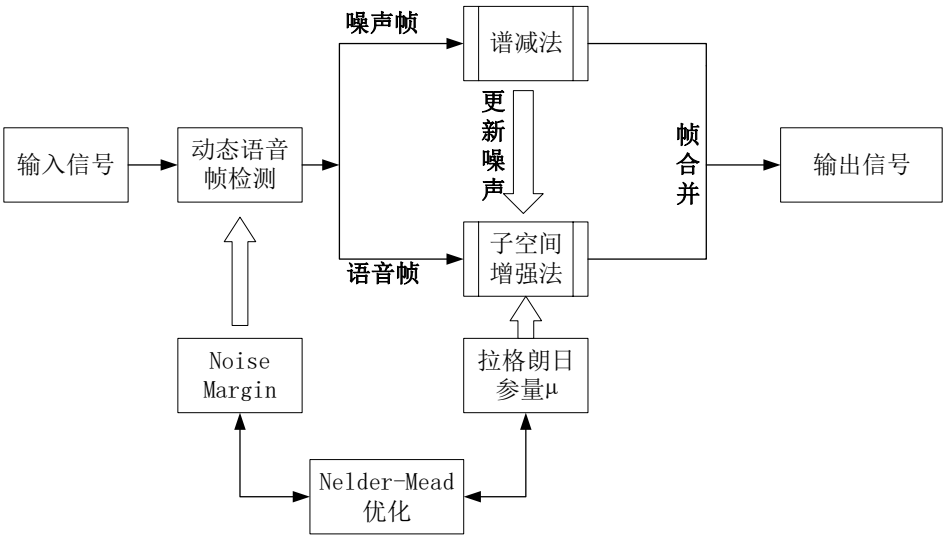

图 4.1 动态语音帧检测子空间增强法流程图

上方提出的算法含有两个重要的参量，一个是动态语音帧检测的谱距离门限值 ***NoiseMargin***，另一个是拉格朗日参量μ。固定初始语音信号信噪比，绘制出两个参量与输出信噪比关系图像 4.2。

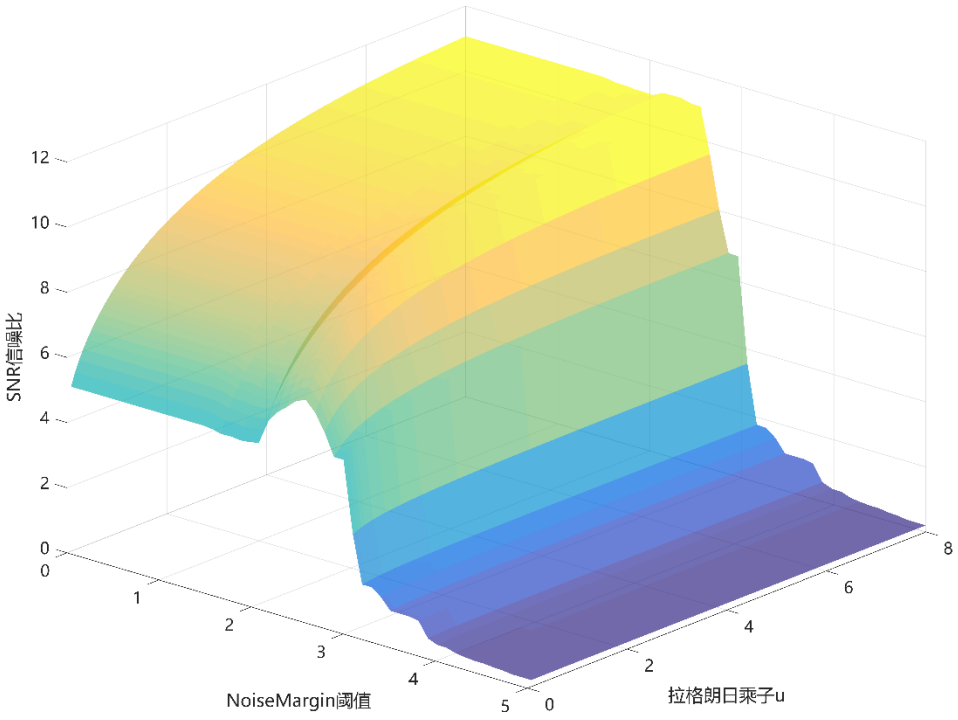

图 4.2 参量与输出信噪比的关系

可以看出这两个参量对输出语音信号的信噪比有着重大的影响，需要通过算法优化。但是因无法获知参量与语音信号的函数关系，传统的梯度下降法，牛顿法无法使用。所以

经过思考，选择了无约束非线性规划的 Nelder-Mead 单纯形法（搜索法）[9,10]作为优化算法。

Nelder-Mead 单纯形法作为一种特殊的算法，具有算法简单、容易实现、对目标函数连续或可导无要求等特点，它由 Nelder 和 Mead 首次提出，并得到广泛应用，该方法是求解无约束优化问题的局部搜索算法。Nelder-Mead 单纯形法的基本搜索思想可表述为：先初始化单纯形 $\boldsymbol{p_0}$、$\boldsymbol{p_1}$、$\boldsymbol{p_2}$、$\boldsymbol{p_3}$。然后在每一次迭代中计算单纯形每个点的目标函数 $\boldsymbol{\phi g(a)}$，将具有最大目标函数的点用其他点替代。重复迭代更新单纯形直至其收敛到函数最小值附近，单纯形法算法描述见表 4-1。

Logical Decisions for the Nelder-Mead Algorithm

IF $f(R) < f(G)$, THEN Perform Case (i) {either reflect or extend}
ELSE Perform Case (ii) {either contract or shrink}

```
BEGIN {Case (i).}
IF f(B) < f(R) THEN
    replace W with R
ELSE
    Compute E and f(E)
    IF f(E) < f(B) THEN
        replace W with E
    ELSE
        replace W with R
    ENDIF
ENDIF
END {Case (i).}
```

```
BEGIN {Case (ii).}
IF f(R) < f(W) THEN
    replace W with R
Compute C = (W + M)/2
or C = (M + R)/2 and f(C)
IF f(C) < f(W) THEN
    replace W with C
ELSE
    Compute S and f(S)
    replace W with S
    replace G with M
ENDIF
END {Case (ii).}
```

表 4-1 Nelder-Mead 单纯形法算法描述

Matlab 中包含了 Nelder-Mead 优化算法，但只能求解所给函数最小值，所以需要进行转换：

$$\underset{\mu,N}{\boldsymbol{max}}\boldsymbol{SNR(\mu,N)} = \underset{\mu,N}{\boldsymbol{min}}\ \boldsymbol{(-SNR(\mu,N))} \tag{4-1}$$

给定语音初始信噪比为 3db，任意给定初始值，经过 400 次优化，输出信噪比固定为 10.2017db，此时的$\boldsymbol{\mu = 2.1250}$，**NoiseMargin=**5.1250。下图 4.3 记录了优化过程。

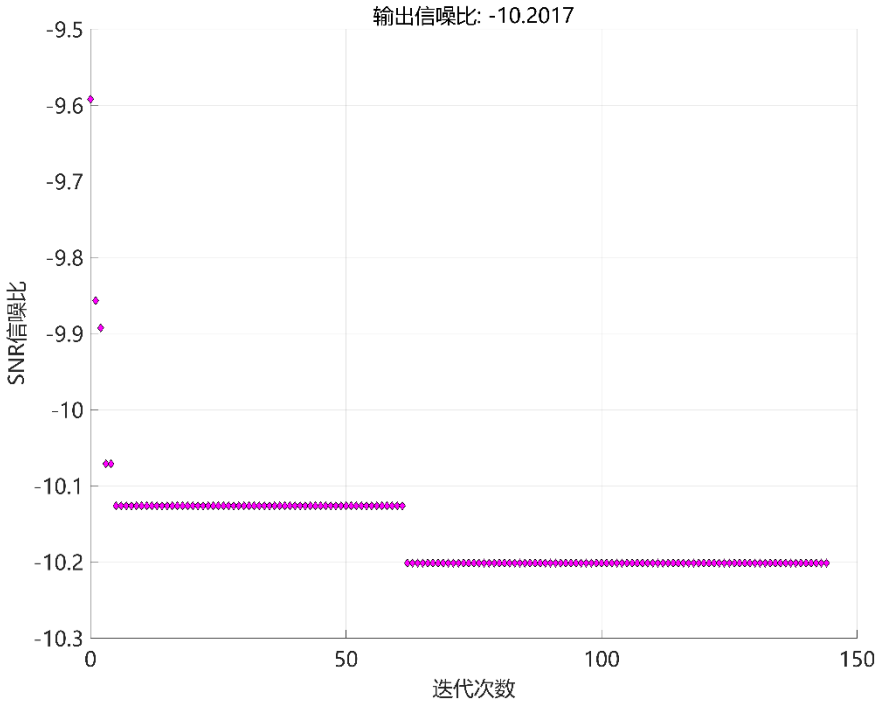

图 4.3 Nelder-Mead 优化过程

5 方法比较

传统的语音增强法有谱减法，维纳滤波法，子空间法。选取几组不同的初始语音信号，让其初始信噪比相同（统一设置为 8 dB），观察不同方法语音增强的结果将其记录于下表 5.1。

表 5-1 不同方法性能比较

方法选择 / 语音信号	谱减法	维纳滤波法	子空间法	增强子空间法
信号一（8dB）	20.2172dB	10.5413dB	9.6752dB	**13.4464dB↑**
信号二（8dB）	20.1499dB	10.5239dB	9.9098dB	**11.2568dB↑**
信号三（8dB）	21.3161dB	10.9099dB	9.7712dB	**14.1254dB↑**
信号四（8dB）	20.4275dB	9.8193dB	9.8066dB	**14.5532dB↑**

注：信号施加的噪声为高斯白噪声,精度保留小数点后四位

从表 5.1 可以看出，增强子空间法性能较维纳滤波法，子空间法性能有着显著的提升。**但谱减法性能表现的最好**，原因在于四组语音初始化时间过长，谱减法可以稳定判定非语音帧时高斯白噪声，并对所有帧进行同样的操作，而增强子空间法只在噪声帧采用了谱减法的思路，所以性能相对下降。

但在实际应用中，正是因为谱减法对先验噪声语音判定有着极高的要求，使得该方法泛化能力较差。选择另外一段语音，施加工厂噪音，并设定不同低信噪比条件，比较其于增强子空间算法的增强效果，记录于表 5.2。

表 5.2 工厂噪音影响下的性能比较

方法选择 / 语音信号	谱减法	增强子空间法
工厂语音一（3dB）	4.7389dB	**4.8485dB↑**
工厂语音二（5dB）	5.1161dB	**5.9726dB↑**
工厂语音三（8dB）	5.4276dB	**7.7011dB↑**
工厂语音四（10dB）	5.5432dB	**9.0852dB↑**

注：信号施加的噪声为工厂噪声，精度保留小数点后四位。

从上表可以看出，增强子空间法的性能表现比谱减法优异，更值得注意的是增强子空间法没有丢失掉原有信号的特征，而谱减法在一定情况下会使得输出的信号丢失掉原有信号的特征，这强有力的说明了**谱减法的不适应性**。具体细节可以见下图 5.1。

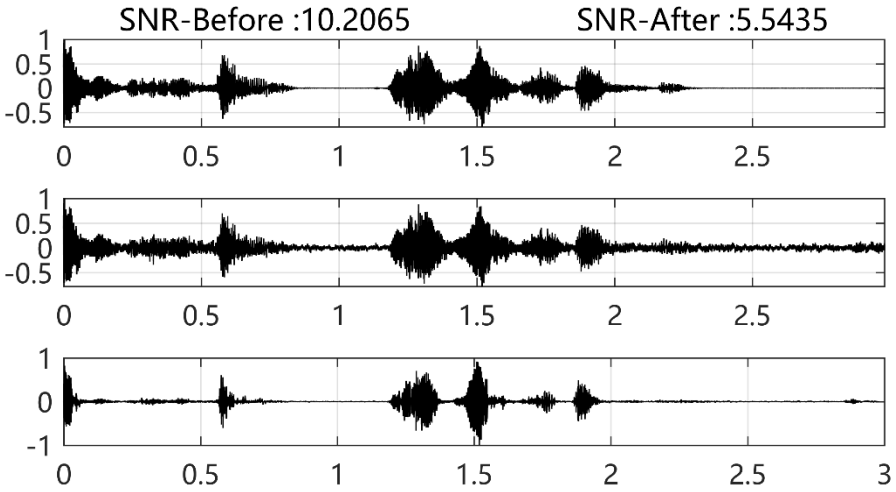

图 5.1（a）工厂信号谱减法输出结果

[注]:上方第三张子图为输出语音波形，可以看出语音信号已经完成失真。

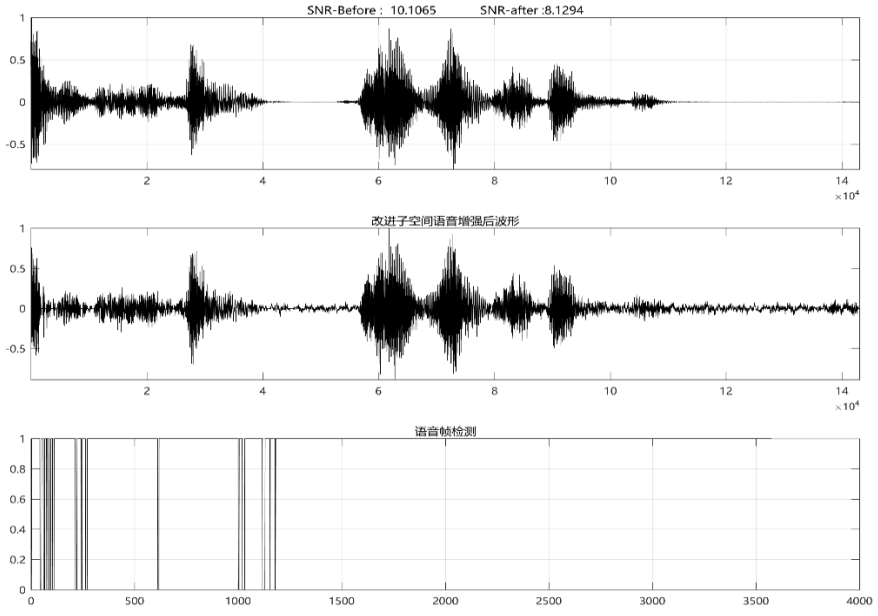

图 5.1(b) 工厂信号增强子空间法输出结果

算法比较还有一个重要指标是算法运行时间，对四种语音增强方法固定初始输入信号，记录算法运行时间，并绘制于下图 5.2

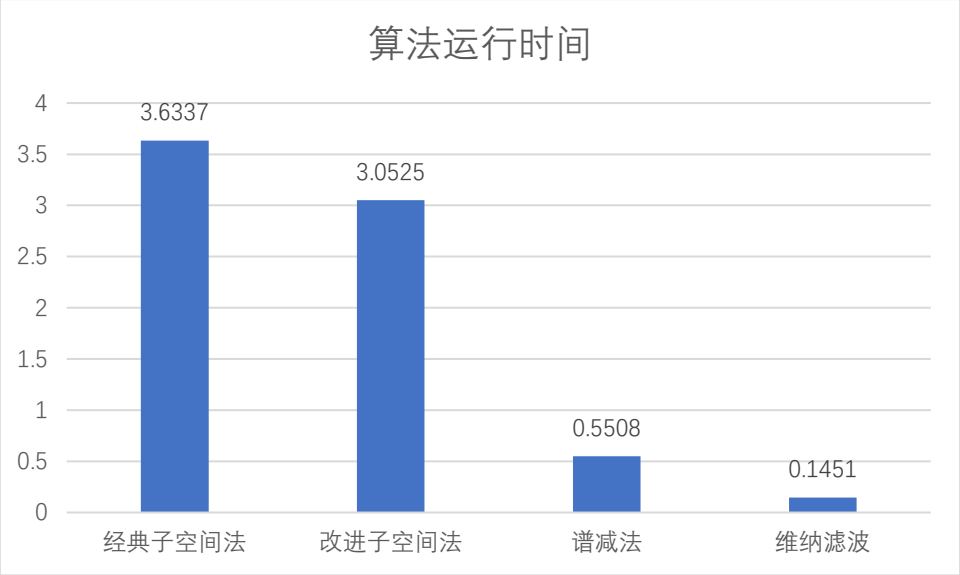

图 5.2 算法时间比较

子空间法因为需要对每一语音帧进行特征值分解，所以运行时间较维纳滤波，谱减法大大增加。改进的子空间法因为进行了语音帧判断，所以速度较经典子空间法提升了**约 20%**，但是因为**无法回避关键的语音帧特征值计算**，所以相比于简单的谱减法和维纳滤波，运算时间仍有非常大的差距。

6 结论

第五章给出了算法比较，从算法比较可以看出，改进后的子空间语音方法不仅能有效的抑制背景噪声，改善语音质量，并且在不同情况下仍能保持信号特征，具有普适性。

但仍存在着一些问题，例如运算时间并未得到较大程度的优化，相较于传统的算法运行时间，改进的子空间增强算法表现一般，所以如果有时间应当要对特征值分解内部细节进行再一次的深入，尤其是对 matlab 内置的特征值分解函数进行分析。同时，所用的方法可能还是没有达到老师口中“精妙“的程度，用的也是一些特征值分解这种基础的矩阵论知识，是否可以尝试引入瑞丽商的知识，这都是值得思考的内容。

7 致谢

感谢耿老师在课上的引导，为了增强所述内容的力度，耿老师尽可能用程序说话，这也是本读书笔记所坚持的一点，一切从实际出发，做到每一个步骤都有理可据，每一步处

理在程序中都有所体现。耿老师在课上教导我，学点最优化的知识没错，所以本文也没有仅仅拘泥于矩阵，而是以子空间为核心，进行外扩，通过优化算法让子空间算法变现更加优异。

同时也感谢自己，坚持到了最后，在黑暗的道路摸索，前期遇到了很大问题，没有他人的支持，只能照着图书馆的教程一步一步走。累了，玩玩游戏，写程序烦闷，出去走走。但发现问题，解决问题，这本来就是一个工程师应该做的。最后不妨引用两句诗作为本篇读书笔记的结尾：黑夜给了我黑色的眼睛，我却注定要用它来寻找光明!

8 参考文献

[1] Rezayee A , Gazor S . An adaptive KLT approach for speech enhancement[J]. IEEE Transactions on Speech and Audio Processing, 2001, 9(2):87-95.
[2]Boll S . Suppression of acoustic noise in speech using spectral subtraction[J]. IEEE Transactions on Acoustics, Speech, and Signal Processing, 2003, 27(2):113-120.
[3] Kamath S D , Loizou P C . A multi-band spectral subtraction method for enhancing speech corrupted by colored noise[J]. Acoustics, Speech, and Signal Processing, 1988. ICASSP-88. 1988 International Conference on, 2002, 4.
[4]Dendrinos M， Bakamidis S， Carayannis G . Speech enhancement from noise: A regenerative approach[J]. Speech Communication, 1991, 10(1):45-57.
[5]Cohen I， Berdugo B . Noise estimation by minima controlled recursive averaging for robust speech enhancement[J]. IEEE Signal Processing Letters, 2002, 9(1):12-15.
[6]EPHRAIM. Speech enhancement using a minimum mean square error short-time spectral amplitude estimator[J]. IEEE Trans, Acoust. Speech Signal Process. 1984, 32(6):1109-1121.
[7]Ephraim Y , Van Trees H L . A signal subspace approach for speech enhancement[J]. IEEE Transactions on Speech and Audio Processing, 1995, 3(4):251-266.
[8]Rezayee A， Gazor S . An adaptive KLT approach for speech enhancement[J]. IEEE Transactions on Speech and Audio Processing, 2001, 9(2):87-95.
[9]穆俊生. 基于特征值筛选的子空间语音增强算法研究[D].南昌航空大学,2015.
[10] Hansen P C , Jensen S H . Subspace-Based Noise Reduction for Speech Signals via Diagonal and Triangular Matrix Decompositions: Survey and Analysis[J]. EURASIP Journal on Advances in Signal Processing, 2007, 2007(1):092953.
[11]刘盛捷, 付翰初, 魏凯, et al. 基于 Nelder-Mead 单纯形法的逆合成孔径激光雷达联合补偿成像算法[J]. 光学学报, 2018, v.38； No.436(07):98-107.
[12] Lagarias J C , Reeds J A , Wright M H , et al. Convergence Properties of the Nelder--Mead Simplex Method in Low Dimensions[J]. SIAM Journal on Optimization, 1998, 9(1):112-147.