
语音信号特征的提取

摘要

随着计算机技术的发展,语音交互已经成为人机交互的必要手段,语音特征参数的精确度直接影响着语音合成的音质和语音识别的准确率。因此语音信号参数提取是语音信号处理研究中一项非常有意义的工作。

本文采用 Matlab 软件提取语音信号特征参数,首先对语音信号进行数字化处理,其次,进行预处理,包括预加重、加窗和分帧,本文讨论了预处理中各种参数的选择,以使信号特征提取更加准确。第三,讨论了各种时域特征参数的算法,包括短时能量、短时过零率等。

关键词: 语音信号, 特征参数, 提取, Matlab

目录

第一章 绪论

1. 语音信号特征提取概况

1.1. 研究意义

语音处理技术广泛应用于语音通信系统、声控电话交换、数据查询、计算机控制、工业控制等领域,带有语音功能的计算机也将很快成为大众化产品,语音将可能取代键盘和鼠标成为计算机的主要输入手段,为用户界面带来一次飞跃。

语音信号特征的提取是语音信号处理的前提和基础,只有分析出可表示语音信号本质特征的特征参数,才有可能利用这些参数进行高效的语音通信和准确的语音识别,才能建立语音合成的语音库。因此语音信号参数提取是语音信号处理研究中一项非常有意义的工作。

1.1.2 发展现状

语音信号处理是一门综合性的学科,包括计算机科学、语音学、语言学、声学和数学等诸多领域的内容。它的发展过程中,有过两次飞跃。第一次飞跃是1907年电子管的发明和1920年无线电广播的出现,电子管放大器使很微弱的声

音也可以放大和定量测量,从而使电声学和语言声学的一些研究成果扩展到通信和广播部门;第二次飞跃是在 20 世纪 70 年代初,电子计算机和数字信号处理的发展使声音信号特别是语音信号,可以通过模数转换器(A/D)采样和量化转换为数字信号,这样就可以用数字计算方法对语音信号进行处理和加工,提高了语音信号处理的准确性和高效性。

语音信号处理在现代信息科学中的地位举足轻重,但它仍有些基础的理论问题和技术问题有待解决,这些难题如听觉器官的物理模型和数学表示及语音增强的技术理论等,目前还有待发展。

1.2 本课题研究内容

本文主要介绍语音信号处理的理论及 Matlab 的相关内容,然后从 Matlab 仿真角度验证了录音、预处理、提取语音信号时域特征参数,主要讨论了预处理中各种参数的选择,以使信号特征提取更加准确。再次讨论了各种时域特征参数的算法,包括短时能量、短时过零率等,介绍了各环节的不同软件实现方法。最后对基于 Matlab 的语音信号特征参数提取进行总结。

第二章 Matlab 简介

MATLAB 是国际上仿真领域最权威、最实用的计算机工具。它是 MathWork 公司于 1984 年推出,它以强大的科学计算与可视化功能、简单易用、开放式可扩展环境,特别是所附带的 30 多种面向不同领域的工具箱支持,使得它在许多科学领域中成为计算机辅助设计和分析、算法研究和应用开发的基本工具和首选平台。

2.1 发展概况

Matlab 是 Matrix Laboratory(矩阵实验室的缩写),最初由美国 Cleve Moler 博士在 70 年代末讲授矩阵理论和数据分析等课程时编写的软件包 Linpack 与 Eispack 组成,旨在使应用人员免去大量经常重复的矩阵运算和基本数学运算等繁琐的编程工作。1984 年成立的 Math Works 公司正式把 Matlab 推向市场,并从事 Matlab 的研究和开发。1990 年,该公司推出了以框图为基础的控制系统仿真工具 Simulink,它方便了系统的研究与开发,使控制工程师可以直接构造系统框图进行仿真,并提供了控制系统中常用的各种环节的模块库。1993 年,Math Works 公司推出的 Matlab4.0 版在原来的基础上又作了较大改进,并推出了 Windows 版,

使命令执行和图形绘制可以在不同窗口进行。1994年推出了 Matlab4.2版本,并得到广泛的重视和应用。1999年,推出了 Matlab5.3版本,真正实现了32位运算,其速度更快、功能更完善、界面更友好,并提供了 Internet搜索引擎,可以协助用户寻求在线帮助。版本6.0、6.1又作了更精细的改进,增加了许多新的功能。版本7.0、7.1包括拥有数百个内部函数的主包和三十几种工具包。工具包又可以分为功能性工具包和学科工具包。功能工具包用来扩充 Matlab的符号计算,可视化建模仿真,文字处理及实时控制等功能。学科工具包是专业性比较强的工具包,控制工具包,信号处理工具包,通信工具包等都属于此类。

2.2 主要功能

MATLAB 之所以成为世界流行的科学计算与数学应用软件,是因为它有着强大的功能。

MATLAB 是一种应用于计算技术的高性能语言。它将计算、可视化和编程结合在一个易于使用的环境中,此而将问题解决方案表示成我们所熟悉的数学符号,其典型的使用包括:

- ◇数学计算
- ◇运算法则的推导
- ◇模型仿真和还原
- ◇数据分析,采集及可视化

MATLAB 是一个交互式系统,它的基本数据元素是矩阵,且不需要指定大小。通过它可以解决很多技术计算问题,尤其是带有矩阵和矢量公式推导的问题。

2.3 信号处理工具箱

Matlab 的信号处理工具箱是信号算法文件的集合,它的推出扩展了 Matlab 在信号处理领域的应用,为其研究和工程应用提供了有力的基础,应用此工具箱可直观、方便地进行信号分析、数值计算及系统设计等工作,从波形的产生到滤波器的实现和设计,参数建模,谱分析,简化了编程,节省了时间。它处理的基本对象是信号与系统,包含了信号处理中经常使用的近200个函数,根据其功能,可以分为20类,经常使用的函数如下所示。

滤波器分析: abs 求绝对值(幅值), conv 求卷积, filter 直接滤波器实现等; IIR滤波器设计: FIR 滤波器设计: fir1基于窗函数的FIR滤波器设计——标准响应, fir2基于窗函数的FIR滤波器设计等; 变换: czt 线性调频Z变换, idct 逆离散余弦变换, ifft 一维逆快速傅里叶变换等; 窗函数: boxcar 矩形窗,

hamming 海明窗, hanning 汉宁窗, chebwin 切比雪夫窗等; 参数化建模: lpc 线性预测系数等。

2.4 录音工具

在 Matlab 环境中, 可以通过多种编程方法驱动声卡, 实现对语音信号的采集和回放。同时由于 Matlab 是一个数据分析和处理功能十分强大的工程使用软件, 它的信号处理与分析工具箱提供了十分丰富的功能函数, 利用这些函数可以快捷地完成语音信号处理和分析以及信号的可视化, 使人机交互更加便捷。

在 Matlab 环境中, 主要可以通过以下 3 种方法驱动声卡, 采集语音信号:

①将声卡作为对象处理采集语音信号。Matlab 将声卡作为对象处理, 其后一切操作都不与硬件直接相关, 而是通过对该对象的操作来作用于硬件设备 (声卡)。

②调用 wavrecord 函数采集语音信号。wavrecord 函数使用 Windows 声音输入设备录制声音。函数调用方式是 “wavrecord(N, fs, ch, nbits)” 其中 N 是采集样本数据量; fs 是样本采集频率 (8000Hz、11025Hz、22050Hz、44100Hz), 其默认值是 11025Hz; ch 是样本采集通道, 1 为单声道, 2 为双声道, 默认值为 1 (单声道); nbit 是每个样本的位数 (或称解析度), ‘double’、‘single’、‘int16’ 为 16 位, ‘int8’ 为 8 位。

③运用 audiorecorder 对象采集语音信号。函数调用方式为 audiorecorder (fs, nbits, ch) 可以创建一个 audiorecorder 对象。其中 fs, nbits, ch 的意义同 wavrecorder 函数。对象创设后, 可以进行录音、暂停、停止、播放以及数据读取等操作。

本文使用第二种方法进行录音。因为本文是对已存信号进行处理, 不需要实时处理, 对语音要求不高, 所以只需选择处理简单的第二种方法。

第三章 语音信号分析

3.1 概述

语音信号分析是语音信号处理的前提和基础,只有分析出可表示语音信号本质特征的特征参数才有可能利用这些参数进行高效的语音通信、语音合成和语音识别等处理。而且,语音合成的好坏,语音识别率的高低也取决于对语音信号分析的准确性和精确性。因此语音信号分析在语音信号处理应用中具有举足轻重的地位。

贯穿于语音分析全过程的是“短时分析技术”。因为,语音信号从整体来看其特性及表征其本质特征的特征参数均是随时间而变化的,所以它是一个非平稳态过程,不能用处理平稳信号的数字信号处理技术对其进行分析处理。但是语音信号在一个短时间范围内其特性基本保持不变即相对稳定,即语音信号具有短时平稳性。所以任何语音信号的分析和处理必须建立在“短时”的基础上,将语音信号分为一段一段来分析其特征参数,其中每一段称为一“帧”,帧长一般取为10~30ms。这样,对于整体的语音信号来讲,分析出的是由每一帧特征参数组成

的特征参数时间序列。

3.2 语音数字化和预处理

在按帧进行语音分析，提取语音参数之前必须进行一些共同的短时分析技术，如语音信号的数字化、语音信号的端点检测、预加重、加窗和分帧等，这些也是不可忽视的语音信号分析的关键技术。

3.2.1 数字化

语音信号的数字化一般包括放大及增益控制、反混叠滤波、采样、A/D 变换及编码；预处理一般包括预加重、加窗和分帧等。

语音信号首先进行反混叠滤波，反混叠滤波的目的有两个：第一，抑制输入信号各个频域分量中频率超出 $f_s/2$ 的所有分量（ f_s 为采样频率），以防止混叠干扰；第二，抑制电源工频干扰。这样，反混叠滤波必须是一个带通滤波器，设其上、下截止频率分别是 f_H 和 f_L ，则对于绝大多数语音编译器， $f_H=3400\text{Hz}$ 、 $f_L=60\sim 100\text{Hz}$ ，采样频率为 $f_s=8\text{kHz}$ 。语音信号经过反混叠滤波和采样后由 A/D 变换器变换为二进制数字码。A/D 变换中要对信号进行量化，量化不可避免地会产生误差。量化后的信号值与原信号值之间的差值称为量化误差，又称为量化噪声。A/D 变换器分为线性和非线性两类。目前采用的线性 A/D 变换器绝大部分是 12 位的（即每一个采样脉冲转换为 12 位二进制数字）；非线性 A/D 变换器则是 8 位的，它与 12 位线性变换器是等效的。有时为了后续处理，要将非线性的 8 位码转换为线性的 12 位码。

3.2.2 预处理

由于语音信号的平均功率谱受声门激励和口鼻辐射影响，高频端大约在 800Hz 以上按 6dB/倍频程跌落，所以求语音信号频谱时，频率越高相应的成分越小，高频部分的频谱比低频部分的难求，为此要在预处理中进行预加重处理。预加重的目的是提升高频部分，使信号的频谱变得平坦，保持在低频到高频的整个频带中，能用同样的信噪比求频谱，以便于频谱分析或声道参数分析。预加重可在语音信号数字化时在反混叠器之前进行，这样不仅可以进行预加重，而且可以压缩信号的动态范围，有效地提高信噪比。但预加重一般是在语音信号数字化之后，在参数分析之前在计算机里用具有 6dB/倍频程的提升高频特性的预加重数字滤波器来实现，它一般是一阶的数字滤波器：

$$H(Z) = 1 - \mu z^{-1} \quad (3-1)$$

式(3-1)中， μ 值接近于1。

进行预加重数字滤波处理后，接下来就要进行加窗分帧处理。一般每秒的帧数约为33到100帧，视实际情况而定。分帧可以采用连续分段的方法，但一般要采用交叠分段的方法，这是为了使帧与帧之间平滑过渡，保持其连续性。前一帧和后一帧的交叠部分称为帧移。帧移与帧长的比值一般取0~1/2。分帧是用可移动的有限长度窗口进行加权的方法来实现的，即用一定的窗函数 $w(n)$ 乘 $s(n)$ ，从而形成加窗语音信号：

$$s_w(n) = s(n) w(n) \quad (3-2)$$

3.2.3 窗函数的选择

在语音信号处理中常用的窗函数是矩形窗和汉明窗等，它们的表达式如下（其中 N 为帧长）：

矩形窗：

$$w(n) = \begin{cases} 1, & 0 \leq n < (N-1) \\ 0, & \text{else} \end{cases} \quad (3-3)$$

汉明窗：

$$w(n) = \begin{cases} 0.54 - 0.46 \cos[2\pi n / (N-1)], & 0 \leq n < (N-1) \\ 0, & \text{else} \end{cases} \quad (3-4)$$

窗函数的选择（形状和长度）对于短时分析参数的特性影响很大。为此应选择合适的窗口，下面从窗口的形状和窗口的长度两方面来讨论这个问题。

一般一个好的窗函数的标准是：在时域因为是语音波形乘以窗函数，所以要减少时间窗两端的坡度，使窗口边缘两端不引起急剧变化而平滑过渡到零，这样可以使截取出的语音波形缓慢降为零，减少语音帧的截断效应；在频域要有较宽的3dB带宽以及较小的边带最大值。这里对矩形窗和汉明窗进行比较。矩形窗的单位函数响应的数字滤波器：

$$h(n) = \begin{cases} 1, & 0 \leq n < (N-1) \\ 0, & \text{else} \end{cases} \quad (3-5)$$

其频率响应：

$$H(e^{j\omega T}) = \sum_{n=0}^{N-1} e^{-jn\omega T} = \frac{\sin(N\omega T/2)}{\sin(\omega T/2)} e^{-j\omega T(N-1)/2} \quad (3-6)$$

它具有线性相位,其频率响应为第一个零值时所对应的频率为 f_s/N , 其中 f_s 为采样频率。而汉明窗的频率响应的第一个零值频率(即带宽)以及通带外的衰减都比矩形窗要大许多。矩形窗与汉明窗的一些参照数据示于下表。

表 4-1 矩形窗与汉明窗的比较

从表 3-1 可以看出,汉明窗的主瓣宽度比矩形窗大一倍,即带宽约增加一倍,同时其带外衰减也比矩形窗大一倍多。矩形窗的谱平滑性能较好,但损失了高频成分,使波形细节丢失;而汉明窗则相反,从这一方面来看,汉明窗比矩形窗更为合适。

3.3 语音信号的时域分析

语音信号的时域分析就是分析和提取语音信号的时域参数。进行语音分析时最先接触到并且也是最直观的是它的时域波形。语音信号本身就是时域信号,因而时域分析是最早使用,也是应用最广泛的一种分析方法,这种方法直接利用语音信号的时域波形。语音信号的时域参数有短时能量、短时过零率、平均幅度分析等,这是语音信号的一组最基本的各种语音信号数字处理技术中都要应用。

窗类型	旁瓣峰值	主瓣宽度	最小阻带衰减
矩形窗	-13	$4\pi/N$	-21
汉明窗	-41	$8\pi/N$	-53

3.3.1 短时能量及短时平均幅度分析

设语音波形时域信号为 $x(n)$ 加窗分帧处理后得到的第 n 帧语音信号为 $x_n(m)$, 则 $x_n(m)$ 满足:

$$x_n(m) = x(n-m) \quad 0 \leq m \leq N-1 \quad (3-8)$$

其中, $n=0, 1T, 2T, \dots$ 并且 N 是帧长, T 是帧移长度, $w(m)$ 是窗函数。

则第 n 帧语音信号的短时能量:

$$E_n = \sum_{m=0}^{N-1} x_n^2(m) \quad (3-9)$$

E_n 是一个度量语音信号幅度值变化的函数，但它有一个缺陷，即它对高电平非常敏感（因为它计算时用的是信号的平方）。为此，采用另一个度量语音信号幅度值变化的函数，即短时平均幅度函数 M_n ，它定义为：

$$M_n = \sum_{m=0}^{N-1} |x_n(m)| \quad (3-10)$$

M_n 也是一帧语音信号能量大小的表征，它与短时能量的区别在于计算时小取样值和大取样值不会因取平方而造成较大差异，在某些应用领域中会带来一些好处。

3.3.2 短时过零率分析

短时过零率表示一帧语音中语音信号波形穿过横轴（零电平）的次数。过零分析是语音时域分析中最简单的一种。对于连续语音信号，过零即意味着时域波形通过时间轴；而对于离散信号，过零率就是样本改变符号的次数。

定义语音信号 $x_n(m)$ 的短时过零率：

$$Z_n = \frac{1}{2} \sum_{m=0}^{N-1} |\text{sgn}[x_n(m)] - \text{sgn}[x_n(m-1)]| \quad (3-11)$$

式中， $\text{sgn}[\]$ 是符号函数，即：

$$\text{sgn}[x] = \begin{cases} 1, & (x > 0) \\ -1, & (x < 0) \end{cases} \quad (3-12)$$

在实际中求过零率时，需要十分注意的一个问题是如果输入信号中包含有 50Hz 的工频干扰或者 A/D 变换器的工作点有偏移（这等效于输入信号有直流偏移），往往会使计算的过零率很不准确。为了解决前一个问题，A/D 变换器前的防混叠带通滤波器的低端截频应高于 50Hz，以有效地抑制电源干扰。

第四章 MATLAB 仿真

本文采用 Matlab 对语音信号特征参数的提取进行仿真验证。如图 4-1，对语音信号进行数字化处理，从而利用软件提取特征参数并进行分析。



图 4-1 实验流程图

4.1 语音数字化和预处理

4.1.1 数字化

该实验以研究者本人的声音为分析样本。在 Matlab 中使用 wavrecord (n, fs, ch, dtype) 函数录取英文“Ada”的读音。因人类语音的频谱主要集中在 4kHz 以内，而根据采样定理，采样频率应大于信号中最高频率的两倍，所以采样频率 (fs) 取 8kHz；本实验将录音时间规定为 2s，因此采样的总点数 (n) 为 2*8000；通道数 (ch) 取 1，即为单通道；采样数据的存储格式 (dtype) 取为“double”，即 16 位采样精度。利用 wavplay 函数可以较清晰地听到读音，同时发现语音开始时存在短暂杂音。用 wavwrite 函数将语音信号保存为“a.wav”文件，应用于下面的处理。

通过 wavrecord 函数，语音信号便由计算机完成了采样、A/D 变换及编码过程。接下来进行滤波，该实验选用了 4 阶的比特沃斯带通滤波器，其上下截止频率分别为 3400Hz 以及 100Hz，这是因为语音信号一般在 4kHz 以内，虽然理论上可以将最高频率取为 4kHz，但实际上由于比特沃斯滤波器为使通带的幅度响应最大限度地平坦，而损失了截止频率处的下降斜度，所以上截止频率应小于 4kHz，从而抑制输入信号中各个频域分量中频率超出 4kHz 的分量；电源的工频干扰一般在 50Hz 左右，为了防止其干扰，所以下截止频率应大于 50Hz，本实验取为 100Hz。

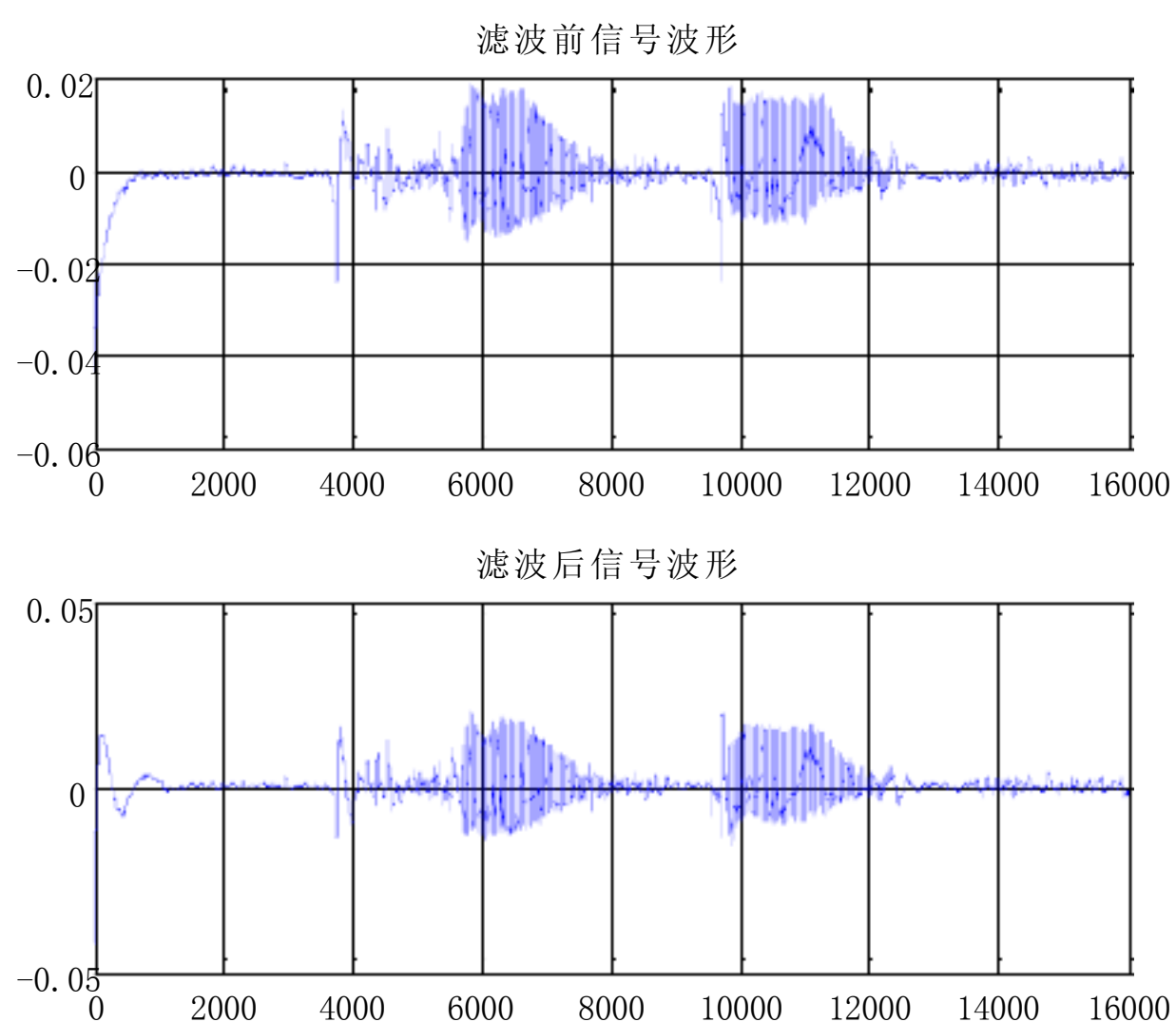


图 4-2 “Ada” 语音信号滤波前后的时域波形

如图 4-2 所示，滤波前后的信号波形差异不大，只有实验开始时的一段噪音以及整段语音中部分高频分量被抑制了，这是因为该实验是在低噪声的环境下完成的，所以通带外的噪声分量很小，滤波前后没有明显变化。

但仅仅根据信号幅度无法准确确认语音端点的具体位置，所以有必要进一步分析。如图 4-3 所示，因 $0 \sim \pi$ 间频谱与 $\pi \sim 2\pi$ 间频谱呈镜像对称，所以本文中频谱的数字角频率范围全部取为 $0 \sim \pi$ ，在整体频谱中，看不出低频段的变化，所以将低频段频谱放大进行观察，由右边两幅图可见，低频段被抑制了，电源干扰也相应地被排除了。

再次用 wavplay 函数收听滤波后的语音，语音更加清晰，语音开始段的杂音

被消除。可见，滤波可以提高得到的语音信号的质量。

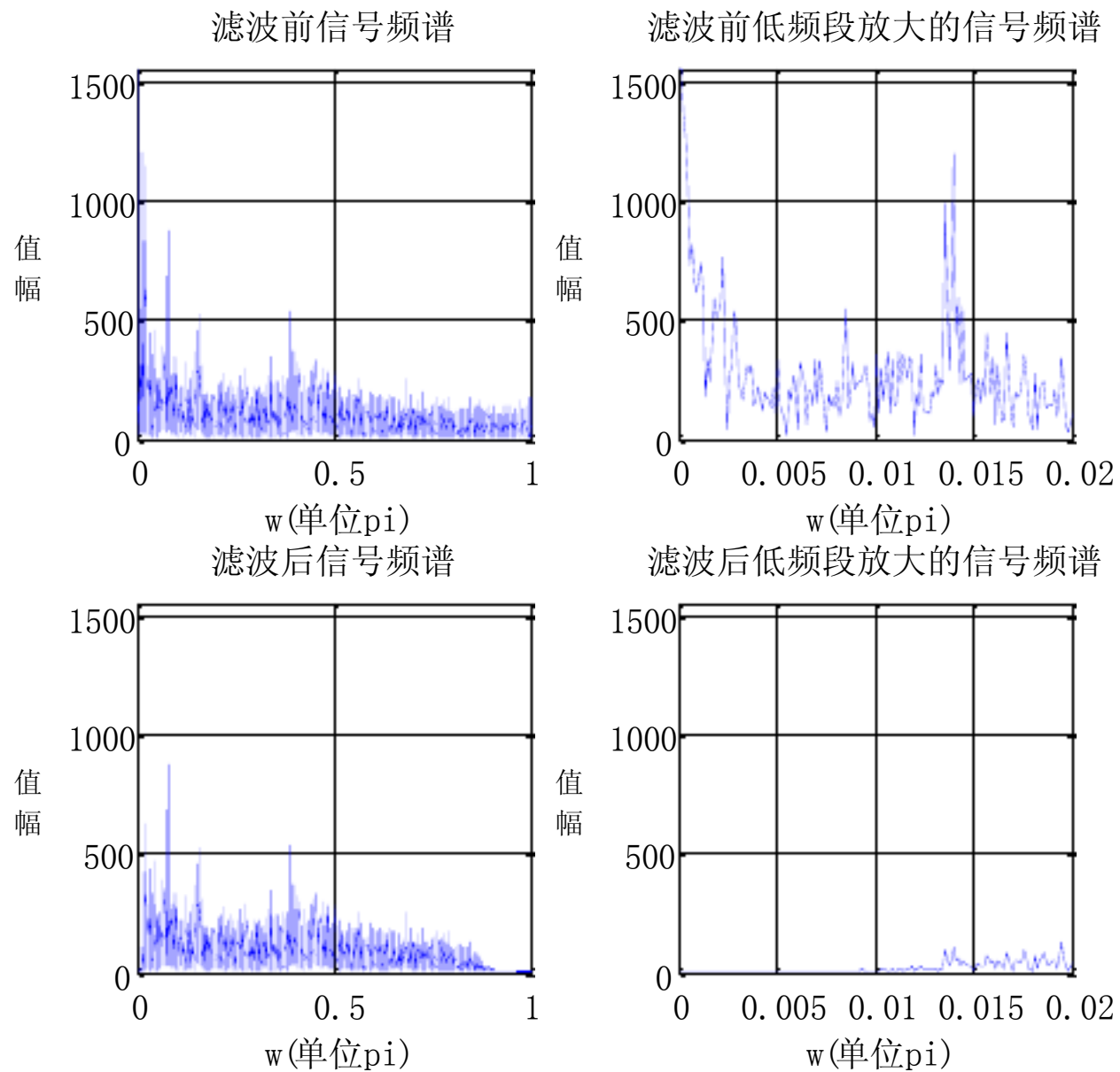


图 4-3 “ ” 语音信号滤波前后的频谱

4.1.2 预处理

用 wavwrite 函数将经过滤波的语音信号存储为 “b.wav” 文件，用于语音的预处理环节。本实验中，预处理包括预加重、加窗和分帧。

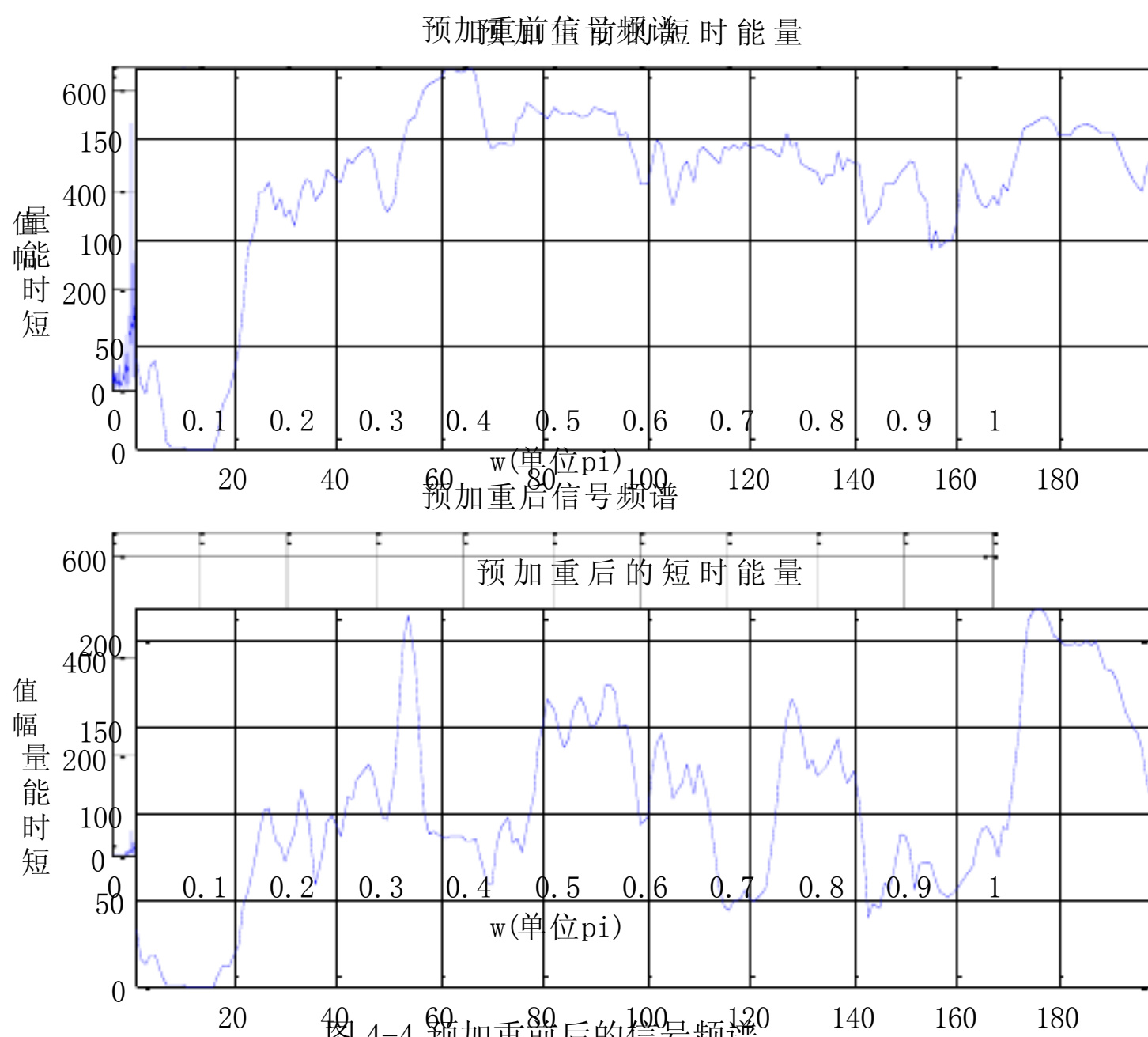
求语音信号频谱时，频率越高相应的成分越小，高频段大约在 800Hz 以上按 6dB/倍频程跌落，所以高频部分的频谱比低频部分的难求，为此要在预处理中进行预加重。本实验采用具有 6dB/倍频程的一阶数字滤波器提高高频特性，来实现预加重：

$$H(Z) = 1 - \mu Z^{-1} \quad (4-1)$$

其中 μ 值取 0.9357

如图 4-4 所示，就整体而言，下图比上图幅值低，下图最大幅值仅约为上图的 1/3，而高频段相对变化不大，所以预加重后高、低频段的幅值差异得到压缩。因此我们可以得到结论：第一，有用的高频部分的频谱得到提升，低频部分频谱进一步被抑制，起到了滤除低频干扰（包括工频干扰和随机噪声）的作用；第二，使信号的频谱变得平坦，压缩了信号的动态范围，保持整个频带中能用同样的信

噪比求频谱，便于频谱分析。如图 4-5 所示，下图比上图的动态范围小，整体幅值水平低，曲线更平滑，可见，预加重还起到了压缩短时能量的动态范围，消除直流漂移，抑制随机噪声的作用。将预加重后的语音用 wavwrite 函数存为“c.wav”文件，用于下面的处理。



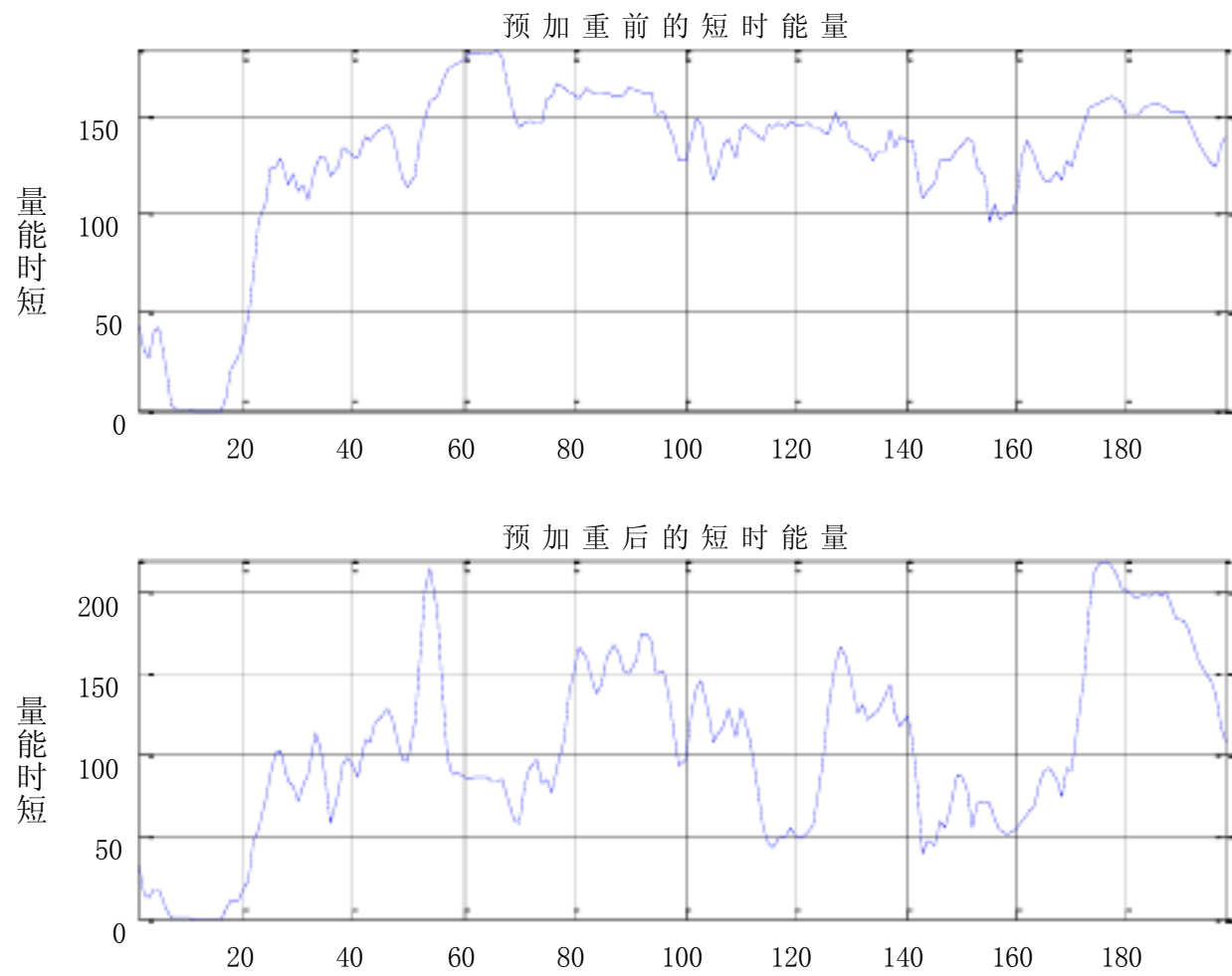


图 4-5 预加重前后的短时能量

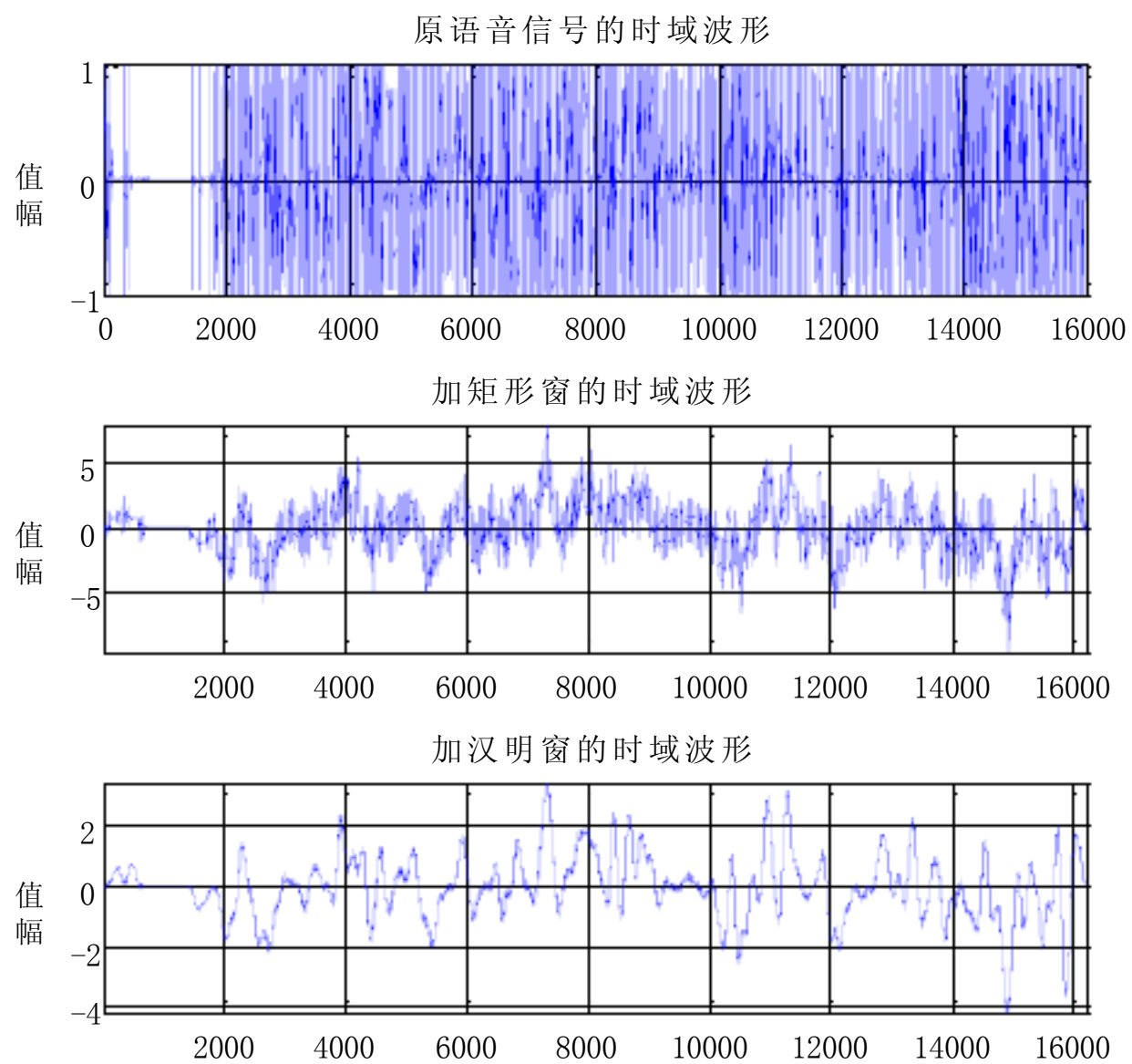


图 4-6 语音信号加窗前后时域波形比较

将预加重后的信号进行加窗分帧。窗函数的选择（形状和长度），对于短时

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/515033242112012002>