

基于 Matlab 的音频文件数字滤波处理

张 康, 刘 雅

(江西中医学院计算机系, 江西 南昌 330004)

摘要: 声音信号是一维连续信号, 而计算机只能处理离散信号。为了从离散信号还原连续信号, 根据采样定理, 可以确定采样频率的最小值。wav 文件是一种数字声音文件格式, 本文基于 Matlab 分析了 wav 声音文件频谱与声音的关系。通过一个数字滤波实例, 处理一个带有盲音的 wav 文件, 去除其中的噪声, 并给出了不同采样频率对输出声音信号的影响。

关键词: 数字信号处理; 数字滤波; Matlab; wav

中图分类号: TP391.42

文献标识码: A

Dealing with Audio Frequency File by Figure Filter Based on Matlab

ZHANG Kang, LIU Ya

(Department of Computer Science, Jiangxi University of Traditional Chinese Medicine, Nanchang 330004, China)

Abstract: Sound semaphore is one dimension concatenation signal, however computer only can process data by discrete signal. By way of revert concatenation signal from discrete signal according to the sampling theorem, it can confirm sampling frequency's least value. Wav file is a kind of figure sound file format. This paper analyses the relationship of wav sound file frequency chart between sounds based on Matlab. By giving an example of figure filter, dealing with a wav file with blinding sound, wiping off the noise and giving different sampling frequency reply export sound signal's effect.

Key words: digital signal processing; digital filtering; Matlab; wav

0 引言

自然声是以声波为载体传递信息的。是多媒体信息的一个重要组成部分, 也是表达思想和情感的一种必不可少的媒体。

随着计算机技术的发展, 声音信号实现了数字化。本文使用数字信号处理技术, 基于 Matlab 对 wav 文件进行分析处理。

1 音频信号基础

声音是由空气中分子的振动而产生的。自然界的声是一个随时间而变化的连续信号。声波的振幅表示声音信号的强弱程度。声波的频率反映出声音的音调, 高频信号声音尖锐, 低频信号声音低沉。

在日常生活中, 我们听到的声音一般都属于复音, 其声音信号由不同的振幅与频率的波合成而得到。

频率小于 20 Hz 的信号称为亚音, 频率范围为 20 Hz~20 kHz 的信号称为音频, 频率高于 20 kHz 的信号称为超音频。人们只能听到频率范围为 20 Hz~20 kHz 的音频信号。

2 数字音频信号频域滤波原理

要在数字信号处理中应用傅立叶变换, 需要解决两个问题: 一是在数学中进行傅立叶变换的 $f(t)$ 为连续信号, 而计算机处理的是数字信号; 二是数学上采用无穷大概念, 而计算机只能进行有限次计算。通常, 将受这种限制的傅立叶变换称为离散傅立叶变换。

假定以间隔 Δt 对一个连续声音信号 $f(t)$ 均匀采样, 离散化为 $\{f(t_0), f(t_0 + \Delta t), \dots, f(t_0 + (N-1)\Delta t)\}$, 将序列表示成 $\{f(n) = f(t_0 + n\Delta t)\}$ 。式中 n 为离散值 $0, 1, 2, \dots, N-1$ 。被抽样的离散傅立叶变换为:

$$F(k) = \frac{1}{N} \sum_{n=0}^{N-1} f(n) e^{-j2\pi kn/N}$$

收稿日期: 2006-11-14

作者简介: 张康 (1980-) 男, 江西南昌人, 江西中医学院计算机系助教, 研究方向: 信号处理; 刘雅 (1981-) 女, 湖南长沙人, 助教, 研究方向: 计算机技术。

$$f(n) = \sum_{k=0}^{N-1} F(k) e^{j2\pi kn/N} \quad (1)$$

表达式(1)为离散傅立叶变换对, 表达式(1)说明计算机在处理有限序列时, 把该信号作为周期信号来处理, 且 $f(n)$ 和 $F(k)$ 周期相同, 周期 $T=N$ 。所以, 对一个具有 N 个采样点的音频文件, 其傅立叶变换结果大小仍然为 N 。

声音文件持续的时间 $t = N * \Delta$, 其中 Δ 为采样周期, 采样频率 $f = 1/\Delta$ 。表达式(1)给出的傅立叶变换中, $k=0, 1, 2, \dots, N-1$ 分别对应于 $0, \Delta f, 2\Delta f, \dots, (N-1)\Delta f$ 处傅立叶变换的抽样值。可以证明 Δf 和 Δ 的关系为 $\Delta f = 1/(N * \Delta)$, 其中 Δ 的单位为 H (赫兹)。

3 基于 Matlab 的数字滤波实例

现有一带盲音的 wav 音频文件, 其杂音主要集中在低频, 而有用信号集中在高频部分。所以, 对该音频信号做傅立叶变换, 把变换结果通过一个高通滤波器, 再经傅立叶反变换得到处理后的音频信号。

原始 wav 音频文件在时域上的图形如图 1 所示。

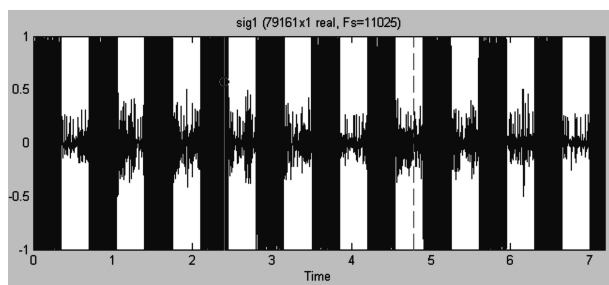


图 1 时域上的原始信号

现要去除其中的杂音, Matlab 代码如下:

```
[f,FS,bits] = wavread('origin.wav');
%读取源文件,其中 f 为数据,FS为
%采样频率,bits为量化位数
[m,n] = size(f); %m为采样点个数
F = fft(f,2*m-1); %对 零填充后进行傅立叶变换,结果
为 F
```

```
F1 = abs(F); % F1为 F 的频率振幅谱
plot(F1) %绘制 F 的频谱图
axis([0 2*m-2 0 500]);
H = lpfilter('butter',2*m-1,7000,20);
%设计 butterworth 低通滤波器
H = (1-H)';
H = 0.001+H; % H 为高通滤波器
figure; plot(H) %绘制滤波器图形
F = F * H; %频域滤波, F 为滤波结果
F2 = abs(F); % F2 为 F 的频率振幅谱
figure; plot(F2) %绘制 F 的频谱图
h = real(fft(F));
%对 F 进行傅立叶变换,结果为 h
h = h(1:size(f,1)+1:size(f,2));
```

```
wavwrite(h,FS,bits,'result.wav');
```

%将处理后的数据写入 result.wav 音频文件

其中, lpfilter() 为自定义函数, Matlab 如下:

```
function H = lpfilter(type,M,D0,n)
[U] = dfu(M);
D=U;
switch type
case 'ideal'
H = double(D<=D0); % D<=D0 返回一个逻辑值,
true 为 1, false 为 0
case 'butter'
if nargin==3
%只有三个输入参数时
n=1;
end
H = 1/(1+(D/D0)^(2*n));
case 'gaussian'
H = exp(-(D/2)^(2*(D0/2)));
otherwise
error('Unknown filter type');
end
```

其中, dfu() 为自定义函数, Matlab 代码如下:

```
function [U] = dfu(M)
u = 0:(M-1);
idx = find(u>M/2);
u(idx) = M-u(idx)-1;
U = u;
```

该文件采样频率 $FS=11025$ 共有 79161 个采样点。所以, 该声音文件持续时间为: $t = N * \Delta = 79161 * (1/11025) = 7.18 (s)$, 频率抽样间隔为: $\Delta f = 1/(N * \Delta) = 1/7.18 (Hz)$ 。

对原始信号进行离散傅里叶变换, 绘制出其频率振幅图如图 2 所示。

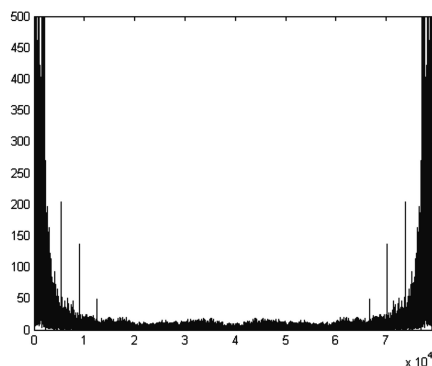


图 2 原始信号频率振幅谱

图 2 表明 0 至 835 赫兹的频率分量具有较高的能量, 而该低频成分正是噪音部分。

高通滤波函数 H 的图形如图 3 所示。

图 2 信号乘以图 3 的滤波器, 得到滤波结果, 如图 4 所示。

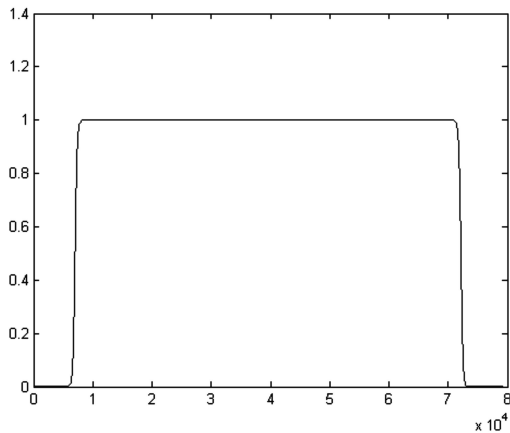


图 3 高通滤波器

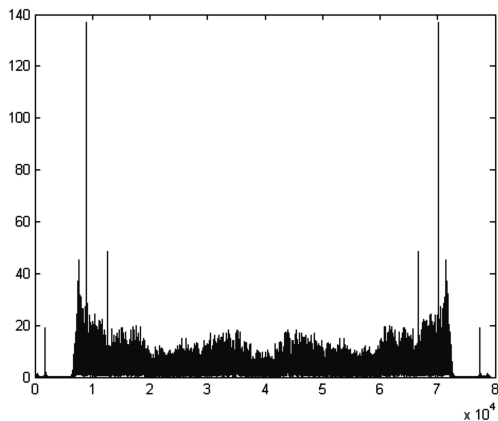


图 4 滤波结果频率振幅谱

从图中 4 我们可以看到, 滤波器把小于 835 赫兹的频率分量的能量降低, 而对高频成分没有影响。

(上接第 97 页)

$$S = H^T = (1001001) \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}^T = (111)$$

由 $S = (111)$ 可知 $e = (0000010)$, 这时可将接收的矢量 $r = (1001001)$ 译为: $c = r + e = (1001001) + (0000010) = (1001011)$, 即正确译码。

理论分析与实验结论一致, 电路设计正确。

设计译码电路时要注意的问题有:

(1) 数字信号传输的码字为串行输入串行输出, 输入时需要通过移位寄存器进行串并转换, 输出时需要将并行数据转换为串行输出;

(2) 为了同步和电路简单, 脉冲信号和运算过程中的控制脉冲可以利用数字信号发生器产生;

(3) 采用移位寄存器存在时序问题, 输入与输出有 8 个延时, 所以在 7 个移位寄存器后面要再加上一个移位寄存器进行延时, 保持同步。

4 结束语

处理后的 wav 文件时域图如图 5 所示。

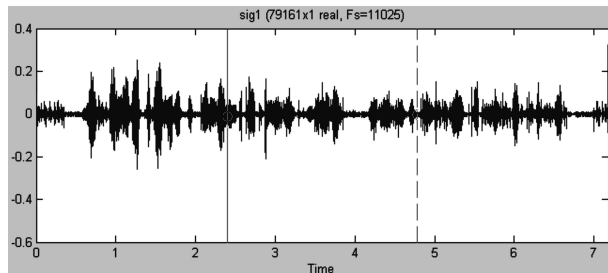


图 5 时域滤波结果

4 结束语

`wavwrite(f, FS, bits, result wav)` 语句中的 FS 为采样频率, 如果改变 FS 的结果, 则可以改变音频文件的播放速度。例如改为 $2 * FS$ 则新的 wav 音频文件的持续时间为: $t = N * \Delta t = 79161 * (1/2 * 11025) = 4.09$ (s) 比原音频文件的语速快一倍。

参考文献:

- [1] 张小虹. 信号与系统 [M]. 西安: 西安电子科技大学出版社, 2004
- [2] Edward W Kanep, Bonnie S Heck 信号与系统基础——应用 Web 和 Matlab [M]. 北京: 科学出版社, 2002
- [3] 阮沈勇, 王永利, 桑群芳. Matlab 程序设计 [M]. 北京: 电子工业出版社, 2004
- [4] Rafael C Gonzalez, Richard E Woods, Steven L Eddins 数字图像处理 (Matlab 版) [M]. 阮秋琦, 等译. 北京: 电子工业出版社, 2005

从仿真过程可以看出来, 采用 EWB 软件能较好地 对编码电路进行系统仿真。基于 EWB 创设的学习环境, 丰富了实验课堂教学方法, 拓展了教学内容的广度和深度, 为学生创造了更多的动手设计的机会, 使学生能够将理论与实践相结合, 提高了学习的主动性。

参考文献:

- [1] 姜丹, 钱玉美. 信息论与编码 [M]. 合肥: 中国科学技术大学出版社, 1992
- [2] 钟文耀, 段玉生, 何丽静. EWB 电路设计入门与应用 [M]. 北京: 清华大学出版社, 1999
- [3] 顾藏知, 陈战平. 数字电路与逻辑设计 [M]. 成都: 电子科技大学出版社, 1999
- [4] 陈运, 周亮. 信息论与编码 [M]. 北京: 电子工业出版社, 2002
- [5] 阎石. 数字电子技术基础 [M]. 北京: 高等教育出版社, 1998