q肌电信号采集系统软件分析程序

姓名： 学号： 专业：

1. 概述

本次作业我选择用实际项目作为书写内容。

本学年作者在研究一项关于人体肌表电信号(sEMG)和肌肉乏力、疼痛相关的项目，项目需要使用前级信号放大及采集装置采集sEMG肌电信号，而后通过串口传给PC，PC通过科学计算分析sEMG的时域、频域特征，以查找原始信号和特定病情的联系。

本次作业我将以上述项目作为背景，描述项目实施过程中用到的Python科学计算中涉及到的所有思想、流程和现象。

1.1信号采集装置

项目中需要搭建信号放大、滤波和采集的前级装置，已制作完毕，如图1所示。它能够通过STM32单片机同时采集六路肌电信号，采样频率为1kHz。部署后如图2所示。

|  |  |
| --- | --- |
| C:\Users\Don Fantom Juan\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.Word\IMG_2560.jpg  图1肌电系统外观 | 图2部署后图 |

1.2数据格式

肌电采集系统搭载的模数转换器（ADC）采样精度为12bits，即对应0~4095十进制数，单片机采样频率为1kHz，即1秒钟将上传1000组肌电数据。肌电信号以”123412341234123412341324\n”的格式传输并存储。其中每个”1234”串为一个通道数据，故一组数据共25Byte。在后续工作中需要将数据读取、显示并处理。

1.3sEMG典型信号

典型sEMG信号为交流信号，如图3所示。其中信号幅度较大的为收缩部分，平稳的是静止部分。

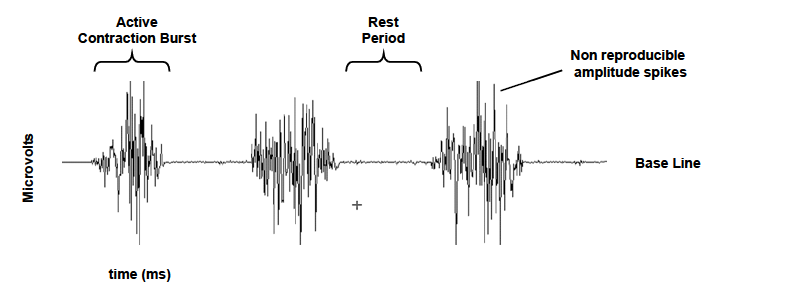


图3典型sEMG信号

1.4实验目的

为探究sEMG信号和身体肌肉疲惫、疼痛之间的关系，需要将原始信号在频域、时域上进行分析，提取其各种特征，而后用机器学习的方法对其进行学习。本阶段只需要能够对采集到的数据进行读取、显示和快速傅里叶变换（fft）即可。所以本次作业主要围绕文件读取和显示，快速傅里叶变换而写。

1. 具体实现

2.1读取文件

获取到的信号存储在.txt文件中，格式如1.2节所述。由于原始数据是包含了6路信号的综合数据，所以在读取文件时需要将数据分离。可根据每组信号末尾的’\n’为结束符分割不同的数据。

2.1.1 open()方法

open方法用于打开文件，调用方法为：

open(dir+filename, mode)

其中：读写模式为r只读，w新建（可覆盖原文件），a追加，b二进制读取等。路径和文件名：如果在当前目录下的.txt，则可直接写文件名，Python会自行查找到。

如果读写模式=NULL，则默认以只读方式打开文件。本实验中就是缺省读写模式：

fh=open(dir)

2.1.2 f.read

上文将指定一个文件进行打开，返回一个句柄，在拿到句柄后，我们可以通过read、readline函数对句柄所指向的文件进行读取操作：

f.read([size]) size未指定的话就会返回所有文件中的内容，如果文件过大，PC内存会吃不消。在实验中只要size=25即可读取一行数据。

f.readline() 无参数，返回一行数据，在实验中也可以使用f.readline()方法进行读取一行数据。

2.1.3遍历文件内容

由于我们需要将所有文件中的内容读取出来，即对文件进行遍历操作，python提供了更简单的方法，本实验最终也是采用这种方法：

for data infh:

ch1.append(int(data[0:4]))

其中，ch1~ch5是一维数组，ch.append是在数组后面添加数据的方法。由于遍历时每次读取出的数据都是字符型，需要int()转化为整型数据。

其中第一通道数据占用data[0]~data[3]；第二通道数据占用data[4]~data[7]；第三通道数据占用data[8]~data[11]；第三通道数据占用data[12]~data[15]；第三通道数据占用data[16]~data[19]；第三通道数据占用data[20]~data[23]。

而后我们将数据返回：return ch1,ch2,ch3,ch4,ch5,ch6，以tuple的形式返回即可。

2.2多路信号显示

当将数据从硬盘文件读取到内存中后，我们需要将数据进行显示。

显示需要用到matplotlib.pyplot中的plot方法。这个方法与Matlab中同名方法用法类似，引入语句为：import matplotlib.pyplot as pl; plot的使用方法如下：

plot(x,y) 使用默认颜色及线形绘制曲线(x,y);

plot(x,y,’bo’) 使用蓝色圆形线性绘制曲线(x,y);

plot(y) 默认用x=0:1:len(y)作为x绘制曲线(x,y);

本实验中规定了曲线颜色，线性应用默认的线形即可，因为数据太大，花哨的线形反而会引起视觉上的混乱。

由于通过readtxt()方法已经将硬盘内容读取出来到tuple类型变量中，则在show()方法中只需要对tuple数组进行显示即可。Tuple数组的组织方式是tuple[0]为第一路信号；tuple[1]为第二路信号；tuple[2]为第三路信号；tuple[3]为第四路信号；tuple[4]为第五路信号；tuple[5]为第六路信号；显示调用函数为：

pl.plot(range(len(result[0])-1),result[0][1:len(result[0])],'r')

在实验中，六路信号时域显示如图4所示。其中红颜色的为第一路肌电信号，其他颜色的为第2~6路肌电信号，为防止颜色太多引起的视觉混乱，在实验中将2~6路AD采集器接地，所以在图中我们会看到第一路通道信号在中间，其他信号在最底下，且含有环境噪声。

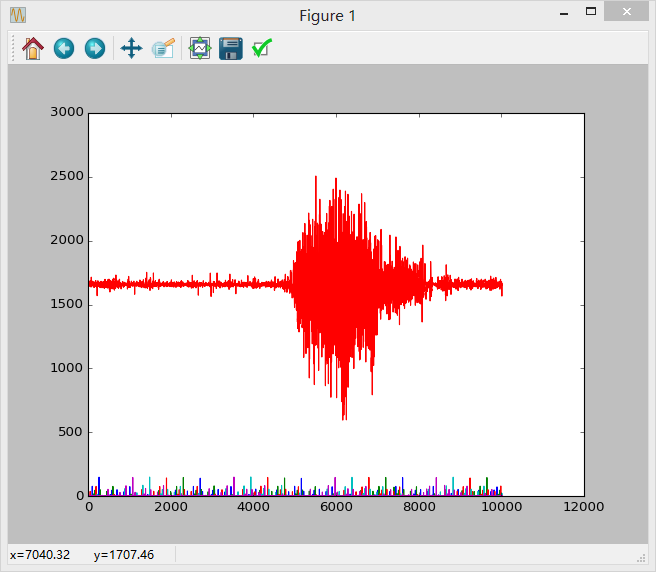


图4原始肌电时域信号

2.3快速傅立叶变换

快速傅里叶变换FFT能够将数据从时域变成频域，用很多不同频率的sine函数去拟合曲线，从而提取出曲线中频域的信号。

在Python的Numpy包中，有快速傅立叶变换方法np.fft.fft()，调用即可将时域信号变成频域信号：

numpy.fft.fft(a,n=None,axis=-1,norm=None)

其中a是待分析的时域信号，n是傅里叶变换出来后频域的范围，axis是坐标轴范围，norm是一般化方法，默认为None。而后它将返回一个复数形式的FFT变换后内容，我们需要对返回值进行取模运算，之后显示。取模运算为abs().

但由于傅里叶变换的特性，经过FFT方法转换出来的内容是对称的，我们只需要分析其功率谱一半的数据即可，结果如图5所示。由于本次处理的数据并没有经过修饰，之中包含了很多高频噪声，所以我们能够从FFT分析出的功率谱中发现很多>100Hz的高频噪声，这是因为线材伪迹和环境噪声引起的高频问题。

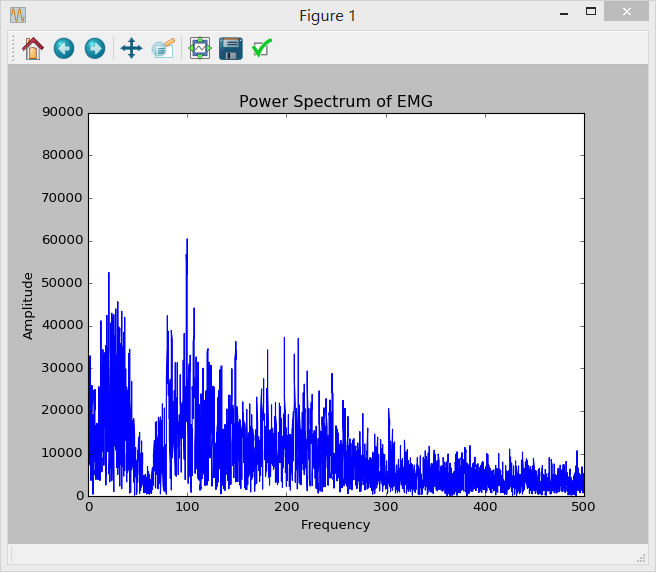


图5原始sEMG功率谱图

2.4时域分析

根据EMG方面权威的书，通常前人在处理时域sEMG信号的时候，需要将交流的信号转换成为直流的信号，即对信号进行整流操作。整流操作可以在电路中用整流电路进行，也可以在后续分析中用软件实现，本系统为了最大限度保证原始信号的可靠性，保留了交流信号，将在数据处理中对数据进行整流操作。

正常的sEMG时域信号积分=0，即>0数据的绝对值与<0数据的绝对值相等，根据这项特点，我们对原始信号进行积分操作，而后求取平均值，平均值将是信号的基线，根据基线，我们将原始信号整体减去基线高度并取绝对值，则信号整流完毕，如图6所示。核心代码为：

baseline=sum(result[0])/len(result[0])

foriin range(len(result[0])):

result[0][i]=abs(result[0][i]-baseline)

pl.plot(result[0])

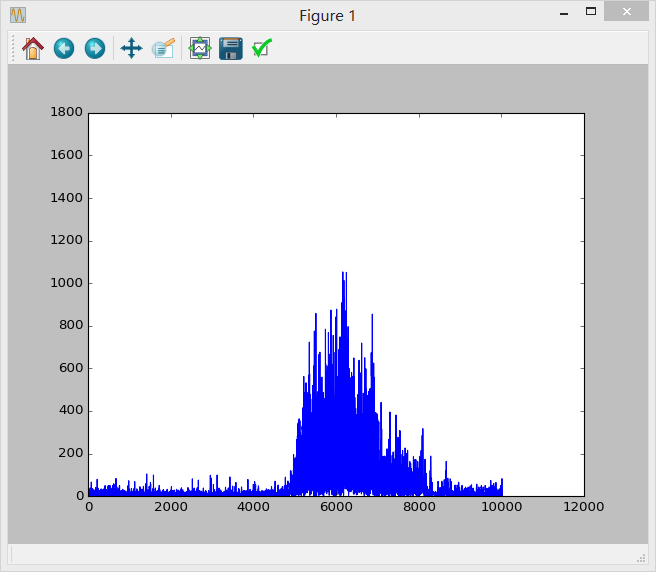


图6原始整流信号

整流后的信号呈现出很大的噪声，虽然能够显示出整体肌电动作的趋势，但是信号方差过大，不够平滑，我们需要对其进行平滑处理，利用滑动窗口方法，平滑后如图7所示。平滑的核心代码为：

fori,datain enumerate(result[0]):

meanN.append(sum(result[0][i:i+5])/N)

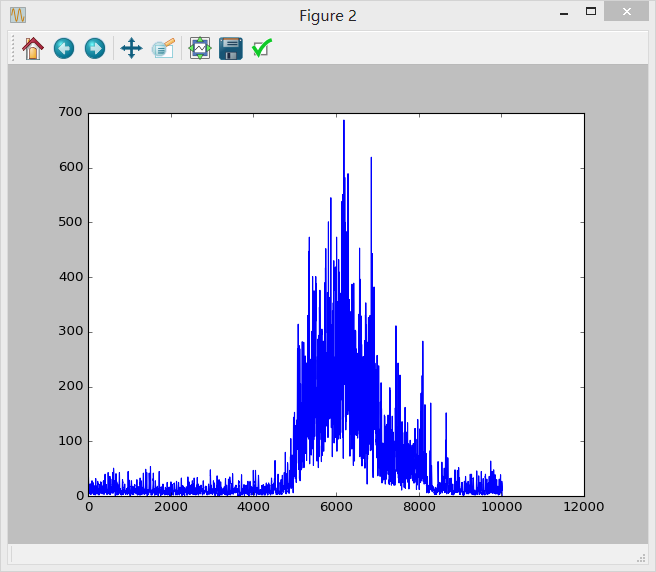


图7平滑后整流信号

1. 总结

经过上述的操作，我们已经能够对原始的肌电信号进行时域波形读取，时域波形整流与平滑；频域数据变换和展示功能，这些操作在sEMG信号分析领域是很基础的操作，一切的信号分析都是基于上述多种变换后的数据的特征提取而进行的。

在实验的过程中，我们涉及到了Python文件读取、Matplotlib绘图库、Numpy科学计算等等内容，是对所学科学计算及Python工程能力的很好实践。

在未来，我将实现Python串口操作，读取单片机通过串行口传来的数据，对其进行实时显示以及实时信号分析操作，而后提取出时域、频域中的直观sEMG信号，以完善项目。

注：当前工程代码已经在附件，请老师同学们参考！