# 脑机接口分析系统界面设计

## 语谱图文件交互接口结构v1.2

哈尔滨千澜科技有限公司

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 修正时间 | 撰写人 | 单位 | 备注 |
| V1.0 | 2019-07-02 | 丛俊锋 | 千澜科技 | 创建文本 |
| V1.1 | 2019-07-03 | 丛俊锋 | 千澜科技 | 修正文件头结构 |
| V1.2 | 2019-07-03 | 丛俊锋 | 千澜科技 | 修正刻度的坐标描述 |

1. 文件命名规则

文件名：eeg文件名+“.specgram”

说明：项目文件目录下的eeg文件名+“.specgram”。

如:项目文件夹下d:\workspace\01.cnt文件生成一个语谱图，生成后的语谱图文件名为：d:\workspace\01.cnt.specgram

1. 数据结构

|  |  |  |  |
| --- | --- | --- | --- |
| 头部结构 | | | |
| 名称 |  | 类型 | 说明 |
| 创建时间 | Createtime | Int | 创建文件时的时间戳 |
| Ax总数量 | Axcount | Int | Y轴刻度总数量 |
| Ax起始地址 | Axstart | Int | Ax数据的在文件中的起始地址 |
| 数据起始地址 | Datastart | int | 颜色数据开始地址 |
| 总宽度 | Width | Int | 图像的总宽度 |
| 总高度 | Height | Int | 图像的总高度 |
| 像素时间比例 | Rate | Int | X轴1像素代表的ms数，建议使用4毫秒或4的整数倍，用来计算宽度与时间同步 |

|  |  |  |  |
| --- | --- | --- | --- |
| 刻度结构 | | | |
| 名称 |  | 类型 | 说明 |
| 刻度序号 | Index | Int | 索引号0开始 |
| 刻度名称 | Name | Char[20] | 刻度名称 |
| Y轴坐标 | Y | Int | 刻度在Y轴的位置 |

|  |  |  |  |
| --- | --- | --- | --- |
| 颜色数据结构 | | | |
| 名称 |  | 类型 | 说明 |
| 颜色 | Color | Int | 颜色16进制值转10进制，如白色“#FFFFFF”该值为16777215（0xFFFFFF） |

1. 文件结构排列方式

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 文件头 | 刻度1 | 刻度2 | 刻度3 | ...... | 刻度n | 颜色1 |
| 颜色2 | 颜色3 | 颜色4 | 颜色5 | ...... | 颜色n |  |

其中颜色总数量 = width\*height

坐标体系采用屏幕坐标体系，既左上角为(0,0),向右向下延伸为正值

刻度1=[0,“0”,320],刻度2=[1,“500”,280] 刻度2=[2,“1000”,240]

颜色1=图像区域(0,0)位置颜色

颜色2=图像区域(1,0)位置颜色

颜色3=图像区域(2,0)位置颜色

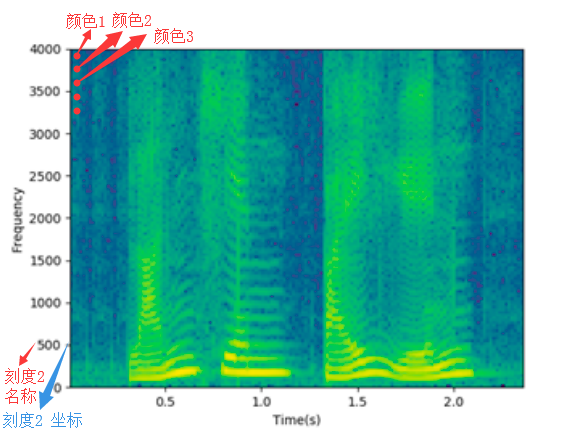


图1

1. C++结构体示例：

//头部结构

Struct HEADER

{

Int createtime;//创建时间

Int axcount; //y轴刻度总数量

Int axstart; //刻度数据起始地址

Int datastart; //数据起始位置

int width; //总宽度

Int height； //总高度

Int rate; //1像素代表的ms数

};

//刻度结构

Struct AX{

Int index; //刻度序号

Char name[20]; //刻度名称

Int y; //刻度y坐标

}

//颜色数据结构

Struct BODY

{

Int color;//颜色16进制值转10进制 ”如白色#FFFFFF”，该值为0xFFFFFF（16777215）

};