

概要设计说明书

一、引言

1.编写目的

此份概要设计编写的目的是用于给此软件优化的工程师以及想要了解本软件的设计理念的人准备的。

2.背景

(1) 本软件的名称为"随手算", 手写输入计算器。

(2) 此项目任务的提出者以及开发者为第 12 组全体成员, 用户为需要手写公式而不是键盘输入从计算的计算器的人, 为的是使节省生活、工作以及学习中需要用计算器的时候更加省时省力。该软件的初步运行环境为个人计算机, 后期可以逐渐扩展到移动设备当中。

3.定义

MNIST 数据集: MNIST (Mixed National Institute of Standards and Technology database) 是一个计算机视觉数据集, 它包含 70000 张手写数字的灰度图片, 其中每一张图片包含 28×28 个像素点。每一张图片都有对应的标签, 也就是图片对应的数字。

分割笔画: 在按笔画在水平方向上最左端的坐标, 将笔画有小到大排序后, 我们从最左边开始扫描所有笔画。如果一个笔画还没有分组, 我们就为它指定唯一分组编号, 然后再看其右侧有哪些笔画和当前笔画在水平方向上的投影是有效重合的, 并将这些重合的笔画定为属于同一组, 直到所有笔画都被扫描。

4.参考资料

1.软件工程——西安电子科技大学出版社

2.<https://blog.csdn.net/SoftwareTeacher/article/details/82875980>

二、总体设计

1.需求规定

本软件的输入为用户的手写公式, 输出为识别出的手写计算公式以及计算结果。本软件处理的过程主要就是对用户手写公式的识别以及对识别出的公式进行计算。

2.运行环境

硬件环境: 具有数据的基本处理运算功能的硬件系统

支持环境: 初步为 windows 系统, 之后会有所扩展, 延伸至移动端环境

3.基本设计概念和处理流程结构

(1) 基本概念

数据集: 扩展后的 mnist 数据集

子程序: a.扩展 mnist 数据集

a1.收集并格式化数据

a2.训练模型

b.输入数据的处理

c.分割多个手写字符

c1.记录笔画产生过程

c2.分割笔画

c3.为每一个分组生成单一位图

d.批量推理

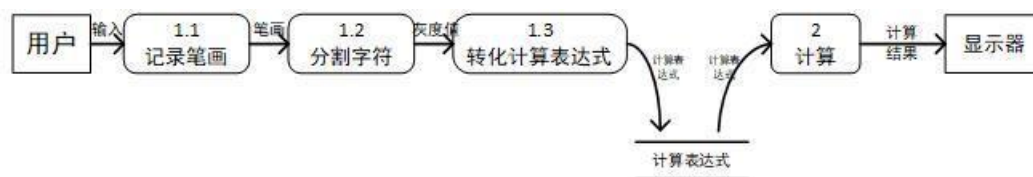
e.计算表达式

顺序性：在手写体识别前应将数据集扩展，并将扩展后的数据集作为数据库。

程序的运行顺序为：输入数据标准化→分割字符→批量推理→计算表达式

(2) 数据处理流程

数据处理流程如数据流程图，如下：



三、接口设计

1.用户接口

用户启动软件后，在输入框内输入要计算的公式，并以确认键确认以示输入完成。此时软件便会做出应答，将识别出的计算表达式进行显示并显示计算结果，之后等待用户的下一步操作。

2.内部接口

(A：手写输入模块 B：字符分割模块 C：表达式转换模块 D：计算模块
E：显示模块)

A->B:传递的是笔画信息，为多个 28*28 的灰度图像

B->C:记录的是字符信息，为多张 28*28 的灰度图像

C->D:传递的是一个可计算的表达式，字符串型数据

C->E:显示识别出来的计算表达式

D->E:传递的是一个实数，浮点型数据，为计算结果

四、运行设计

1.运行模块组合

当开启计算器的时候，输入模块会开始准备读入用户所输入的数据，用户输入的数据将由输入模块记录并存储，随后再将数据传入分割字符模块和识别模块，便可识别出输入的表达式；此时，即可将表达式数据传入显示模块进行回显，也会将表达式先传入计算模块计算出结果后，再到显示模块进行显示。

2.运行控制

开始与关闭手写计算器和正常软件一样，手写时利用鼠标或触屏在屏幕上拖动即可，之后只需按确认键即可显示结果，此外还有普通计算机该有的功能键可实现其相应的功能在此不再赘述。

3.运行时间

识别表达式的过程将占用相对较多的时间，计算及显示的时间和所占内存等资源较小可忽略不计。需要说明的是，识别所用的模型都是之前已经训练好的，故识别过程所占时间等资源也是在几十毫秒之内可以完成的，影响使用。

五、系统数据结构设计

1.逻辑结构设计

(1) 数据项名：输入

含义说明：用户通过手写界面输入的笔迹

数据类型：灰度图

长度：不限

取值范围：像素点大小 0~255

取值含义：用户输入的数字及字符的灰度范围

(2) 数据项名: 笔画

含义说明: 记录用户每次手写的笔画在纵向上的投影

数据类型: 一维灰度图

长度: 28 (单个字符投影大小)

取值范围: 256*28

取值含义: 256 为每一点在投影叠加后的灰阶级数, 28 为每一个字符投影后的图形大小

(3) 数据项名: 表达式灰度图组

含义说明: 每一个灰度图片表示一个字符

数据结构: 多个 28*28 大小的灰度图像

长度: 长度为数字, 符号的个数总和

取值范围: 灰度图像每一个像为 0-255

取值含义: 28*28*256 大小的数据代表一个灰度图, 每一个灰度图可表示一个符号或者数字, 便于识别

(4) 数据项名: 计算表达式

含义说明: 此为识别后存入系统的所要求结果的表达式

别名: 表达式

数据类型: 字符串型

长度: 根据用户输入所定

取值范围: 所有与计算有关的数字和非数字字符

取值含义: 数字字符单独或多个组合为具体数字, 符号字符为计算的计算符

(5) 数据项名: 计算结果

含义说明: 表达式计算的最终结果计算的结果

别名: 结果

数据结果: 双精度浮点型数字

长度: 8 字节

取值范围: 绝对值范围大约是: $2.23 \times 10^{-308} \sim 1.79 \times 10^{308}$

取值含义: 表示最终计算并要显示的结

数据之间的逻辑关系: 见上文数据流程图

2.设计软件系统结构

(1)模块划分:

手写输入模块, 字符分割模块, 表达式转换模块, 计算模块, 显示模块

(2)模块功能:

A: 手写输入模块: 记录手写输入的图形, 笔画等信息

B: 字符分割模块: 根据输入的笔画等信息将字符分割开来

C: 表达式转换模块: 将分割的字符转化为计算机可识别的计算表达式

D: 计算模块: 用于计算表达式的结果并记录的模块

E: 显示模块: 用于回显计算结果

(3)模块之间的调用关系:

总的来说: A ... E 前者可依次调用后者, 实现依次顺序调用

其次: D 可调用 E, 实现对识别出来的表达式进行回显

(4)模块之间的接口:

A->B:传递的是笔画信息，为多个 28*28 的灰度图像

B->C:记录的是字符信息，为多张 28*28 的灰度图像

C->D:传递的是一个可计算的表达式，字符串型数据

C->E:显示识别出来的计算表达式

D->E:传递的是一个实数，浮点型数据，为计算结果

(5)评价模块结构的质量:

每个模块可实现单独的功能，可分开调试，利于实现整体功能，并且便于改进添加新的功能与模块，且模块间传递的信息较为清晰，利于分辨与处理。但是，模块主要为顺序结构调用与运行，某个模块的失误就可能导致整体程序出错，所以设计模块时要格外仔细。

六、系统出错处理设计

1.出错信息

(1) 可能用户输入的表达式不符合规范

解决方案：回显表达时同时提示“输入有误，请重新输入”

(2) 手写板发送命令不返回数据问题

解决方案：a、打开串口时，要以 O_RDWR(可读写)，O_NOCTTY(防止将终端机当成进程控制终端机)，O_NDELAY(以不可阻断的方式打开文件，即无论有无数据读取或等待，都会立即返回进程之中)

b、设置正确的比特率，校验位，停止位

c、取消软件流控制，防止出现丢码

(3) 手写板接收数据不稳定

解决方案：在向手写板发送命令时，手写板需要时间返回数据，故在 write 后，用 usleep 一段时间，再 read，保证每次 write 后均能 read 到数据

(4)手写板返回数据错误

解决方案: 手写板开发时, 不能死板协议上的返回数据, 要根据实际的测试数据来比对, 以防协议和实际返回的数据有误

2.补救措施

a.后背技术说明:

每次的识别出的表达式都会记录在表达式的存储库当中，一旦出现系统数据问题时可到表达式的存储库当中复原之前的表达式。

3.系统恢复设计

恢复再启动技术说明:

当系统崩溃时软件可自动重启，恢复工作。