Handwritten Number Recognition Project Report

唐林峥，虞果，杨润康，

Our division of labor:

虞果:　AI

唐林峥：Usage of Linux and Raspberry-Pi, LED-control, pi-camera-control(code), Button-control(code)

杨润康: Button-control, LED-control(code), circuit design and assembly, pi-camera-control

Problems:

The AI part：

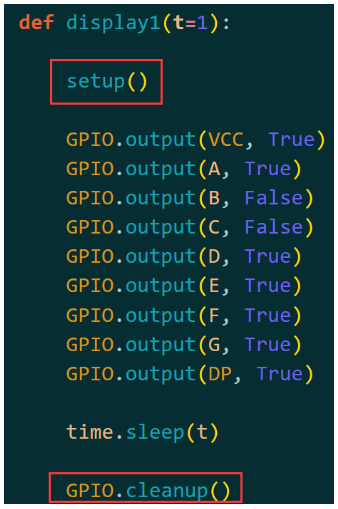
In this part, the first major problem we've encountered lies in how to automatically recognize the edge of each number and than cut it. After counting the value of each column to determine the current state, we were perplexed and didn't know what to do with the 'flag' parameter. Eventually, we decided to compare the value with the maximum value of the white and black, thus judging which state it is in. To complete it, we added a threshold to make sure that what it has identified is truly the numbers, through this, we are able to output rather satisfactory results.

The second problem is that when processing the double-line pictures, the upper and lower line is not exactly asymmetric, causing a massive decline in cutting quality. As the principle used in row split is to transpose twice the main matrix used in column split, we chose to rewrite the given order of cutting, making it a row first, then a column, which turned out to be an effective solution.

The third problem is the improvement in the one-line recognition rate. In the beginning, the best result we get is three correct answers out of ten, which is a disappointing one. Knowing the keys to it are the value of K, the portion of the training set and testing set, the threshold in the binarization, and how it was being cut, we used the variable-control approach to test each combination by group and successfully raised the recognition rate to 80 percent.

The Raspberry Pi and the circuit part:

In this part， we did not meet many troubles that were very serious, the only one really matter problem was when we test the LED light, we found out that the displaying performance is very unstable, we tested the displaying of numbers 0 – 9, one by one and for 1s per number. But in these testing cases, only a few of them could display normally and what is weirder is the randomness of the numbers that could be displayed normally. At first, we suspected our code went wrong somehow, so we tried many ways to fix this the ultimate and best result we achieve is that when displaying one number we always set up the GPIOs and then clean up them. But even if this was the best there are still 2 -3 numbers that don’t display or just blink. then we decide to directly try the code online, to see whether it is the problem of the codes or not. And in fact, the codes online did not work perfectly too. So, we began to check the LED and Raspberry Pi, to see where the issue truly was. Because all the displaying was “correct”, they only had the problem that they just blink instead of lighted up for 1s, so we thought the problem probably came here. We used our backup LED to try displaying first and still failed, which means it is not about the LED, only about the Raspberry Pi. Thus, we borrowed a multimeter to measure the current between LED’s VCC and the GPIO, and indeed, we found the reason why sometimes LED only blink was the output voltage of the GPIO (board number 7) is unstable, or in other words, this GPIO is actually faulty. Then we change to another GPIO, and the issue has finally gone.

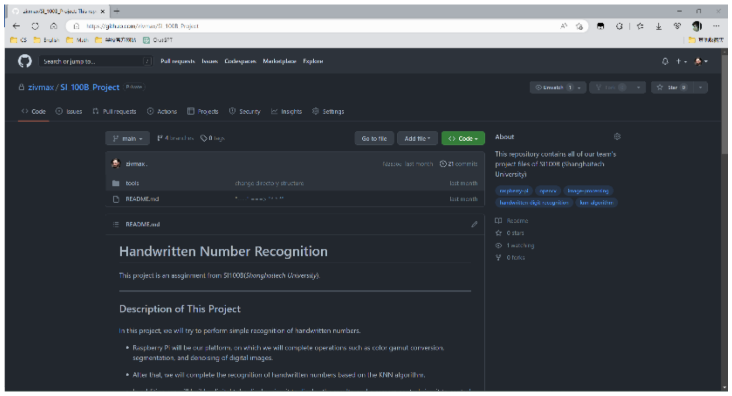


```
def display1(t=1):

    setup()

    GPIO.output(VCC, True)
    GPIO.output(A, True)
    GPIO.output(B, False)
    GPIO.output(C, False)
    GPIO.output(D, True)
    GPIO.output(E, True)
    GPIO.output(F, True)
    GPIO.output(G, True)
    GPIO.output(DP, True)

    time.sleep(t)

    GPIO.cleanup()
```

Innovations:

Our innovations are mainly reflected in improving the convenience and efficiency of development.

First, it is worth mentioning that we used Git and GitHub during the developing, to help us cooperate and control the source code better, and also, in the early time when we didn’t use the SSH extension of VScode, this really help sync the code in our computer and the Raspberry Pi.



https://github.com/zivmax/SI_100B_Project
Pull requests Issues Codespaces Marketplace Explore
zivmax / SI_100B_Project Private
Code Issues Pull requests Actions Projects Security Insights Settings
main
Go to file
Add file
Code
zivmax
21 commits
tools
change directory structure
last month
README.md
last month
About
raspberry-pi opencv image-processing handwritten-digit-recognition knn-algorithm
Readme
0 stars
1 watching
0 forks

This repository contains all of our team's project files of SI100B (Shanghaitech University)

Handwritten Number Recognition

This project is an assignment from SI100B(*Shanghaitech University*).

Description of This Project

In this project, we will try to perform simple recognition of handwritten numbers.

- Raspberry Pi will be our platform, on which we will complete operations such as color gamut conversion, segmentation, and denoising of digital images.
- After that, we will complete the recognition of handwritten numbers based on the KNN algorithm.

Second, we found a way to connect the Raspberry Pi wirelessly, this not only removes the limit of cable, and what’s more important it allows Raspberry Pi to connect multiple devices at the same time, which allows us to develop on Raspberry Pi at the same time. The process of building connect is quite twisted, we mainly referred to these materials below to achieve it:

1. 使用 wpa_supplicant.conf 配置树莓派的网络及配置文件常用字段解析_玩转智能机器人的博客-CSDN 博客_wpa_supplicant 配置文件 https://blog.csdn.net/u011198687/article/details/123312666

2. 树莓派 3B+/4B 连接“手机热点“或“WiFi“ 后无法上网（必解）_wjd956574941 的博客-CSDN 博客_树莓派连接 wifi 无法上网 https://blog.csdn.net/wjd956574941/article/details/123257114

3. 树莓派设置程序开机自启动（通过桌面启动） _吉大小姐 l 的博客-CSDN 博客_树莓派设置开机自启动 https://blog.csdn.net/weixin_51156135/article/details/122252423

During the process, we also learned some basic knowledge of computer networks such as dynamic IP address allocation, gateway, port, and the setup of the static IP address.

Finally, we learned how to use the SSH extension of VScode, and we then apply it to the developing immediately. We feel the advantages of remote development, especially before this, we have to push to GitHub or develop through VNC and terminal(putty), which is not a good experience.

Final result showcase: