

EE231 实验报告

第五讲 – 手写数字识别

姓 名: 洪峰
学 号: 517021910418
学 院: 电子信息与电气工程学院
专 业: 信息工程
邮 箱: hf.fhong@foxmail.com

EE231 人工智能系统设计与实践

上海交通大学

2020 年 5 月 5 日

目录

I.	实验一.....	3
A.	完成 <code>recognize_hdr_lib.py</code> 脚本中 <code>TODO</code> 部份代码并截图.....	3
B.	执行训练 (<code>train.py</code>) 及识别 (<code>recognize.py</code>)脚本， 并将结果截图贴上	3
II.	实验 2. 制作数张手写数字图片， 然后用实验 1 中训练出来的分类器去辨识.....	4
A.	将制作的手写数字图片截图贴上 (至少三张).....	4
B.	修改 <code>recognize.py</code> 脚本， 并将修改部份代码截图	4
C.	执行修改后的 <code>recognize.py</code> 脚本， 并将结果截图贴上	5
III.	实验 3.使用 MNIST 完整数据集训练， 再回去辨识实验 2 中的手写数字	6
A.	修改 <code>recognize.py</code> 脚本， 并将修改部份代码截图	6
B.	执行修改后的 <code>recognize.py</code> 脚本， 并将结果截图贴上	7
C.	比较实验 2 及实验 3 的结果， 是否有什么差异？请写下差异的分析与造成差异的原因 ..	7
IV.	实验 4. 实验 2、3 中若有无法辨识出手写数字，调校 <code>width_th</code> 、 <code>height_th</code> ，让手 写数字都可辨识.....	8

I. 实验一.

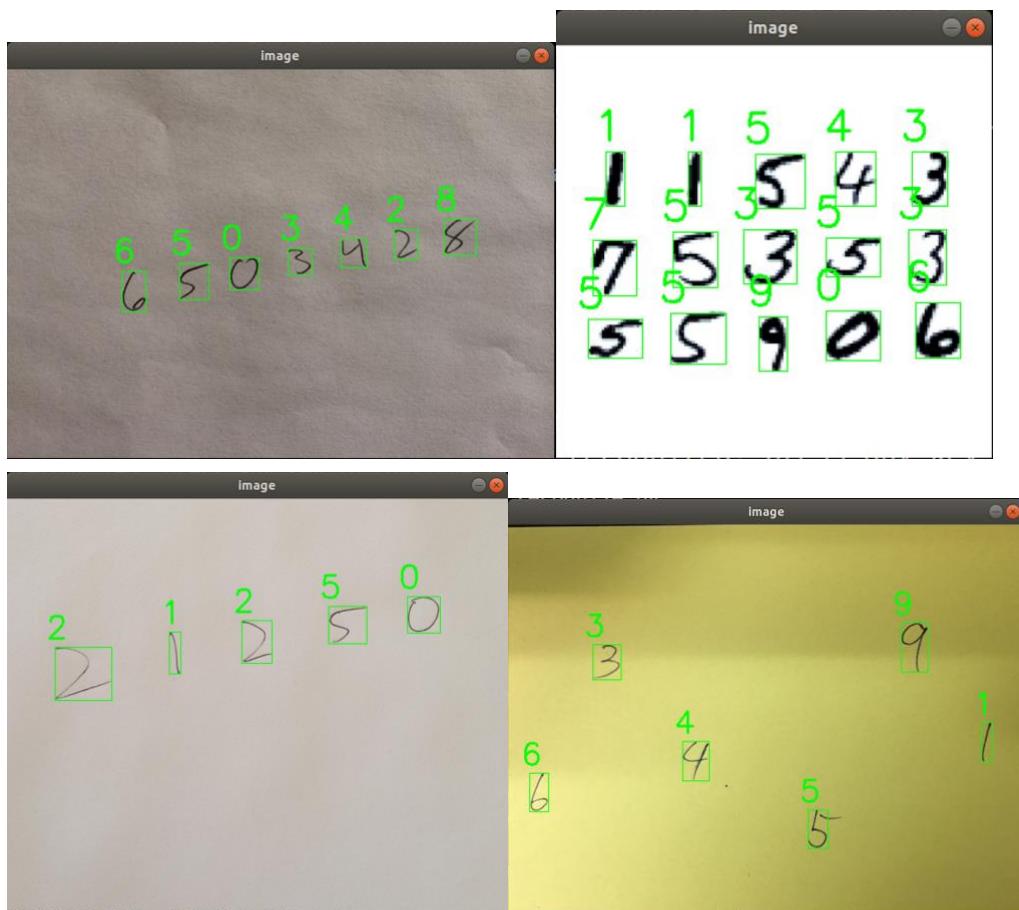
- A. 完成 recognize_hdr_lib.py 脚本中 TODO 部份代码
并截图

```
# TODO: convert it to grayscale
img_proc.convert_gray()

# copy the grayscale image
gray = img_proc.img[0]

# TODO: bluring the image, use gaussian bluring and structure element
#      (5, 5)
img_proc.blur('gaussian', 5, 5, 0)
```

- B. 执行训练 (train.py) 及识别 (recognize.py)脚本，并将结果截图贴上



II. 实验 2. 制作数张手写数字图片，然后用实验 1 中训练出来的分类器去辨识

A. 将制作的手写数字图片截图贴上 (至少三张)



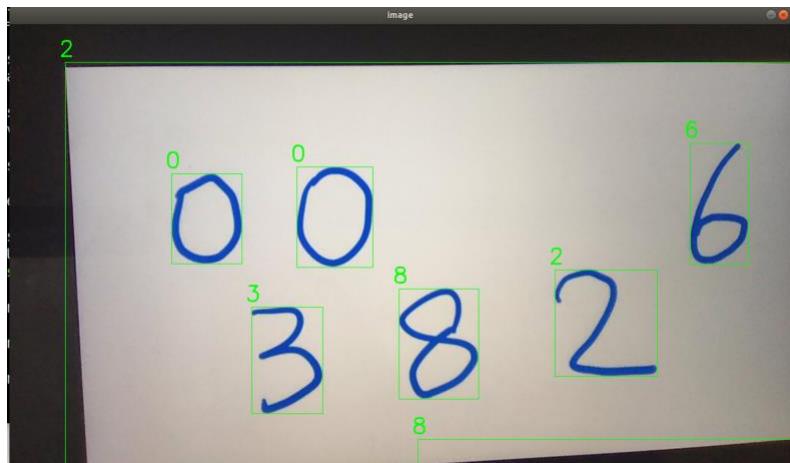
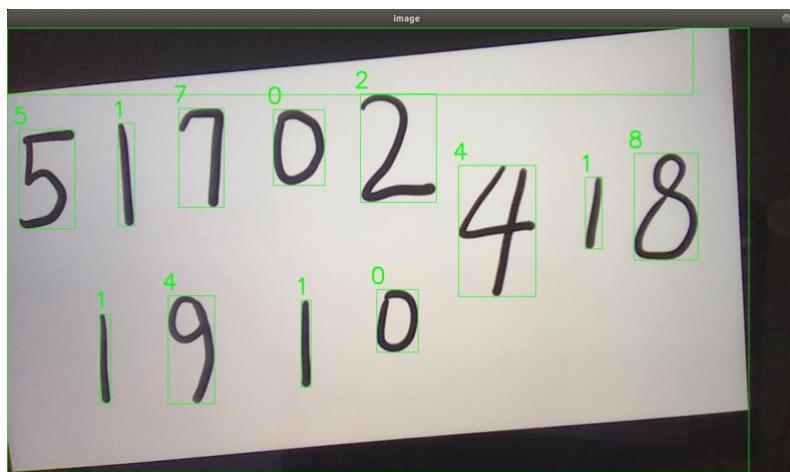
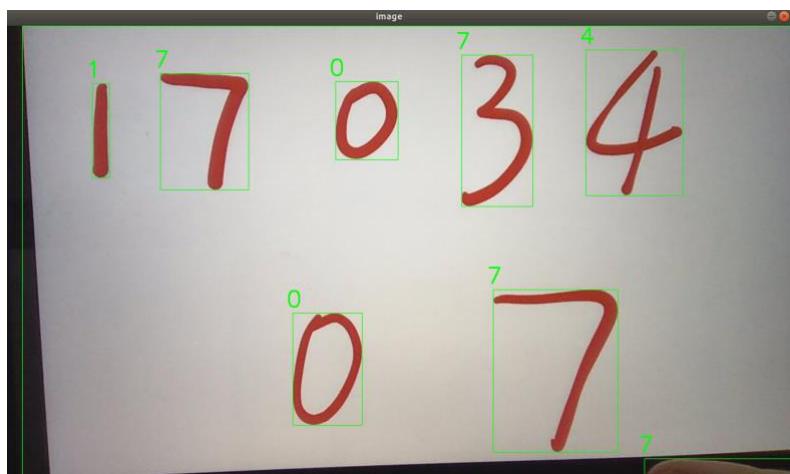
B. 修改 recognize.py 脚本，并将修改部份代码截图

```
import ailib.recognize_hdr_lib as recognize_hdr

# variables for parameters
model_path = 'models/mnist_5000_svm.bin'
image_path = 'myimages'
width_th = 7
height_th = 20

# train classifier
recognize_hdr.recognize(model_path, image_path, width_th, height_th)
```

C. 执行修改后的 recognize.py 脚本，并将结果截图贴上



容易将其他背景识别为数字
有个别错了

III. 实验 3. 使用 MNIST 完整数据集训练，再回去辨识实验 2 中的手写数字

A. 修改 recognize.py 脚本，并将修改部份代码截图

train.py

```
import ailib.train_hdr_lib as train_hdr

# variables for parameters
dataset_path = 'datasets/mnist_full.csv'
model_path = 'models/mnist_full.bin'

# train classifier
train_hdr.train(dataset_path, model_path)
```

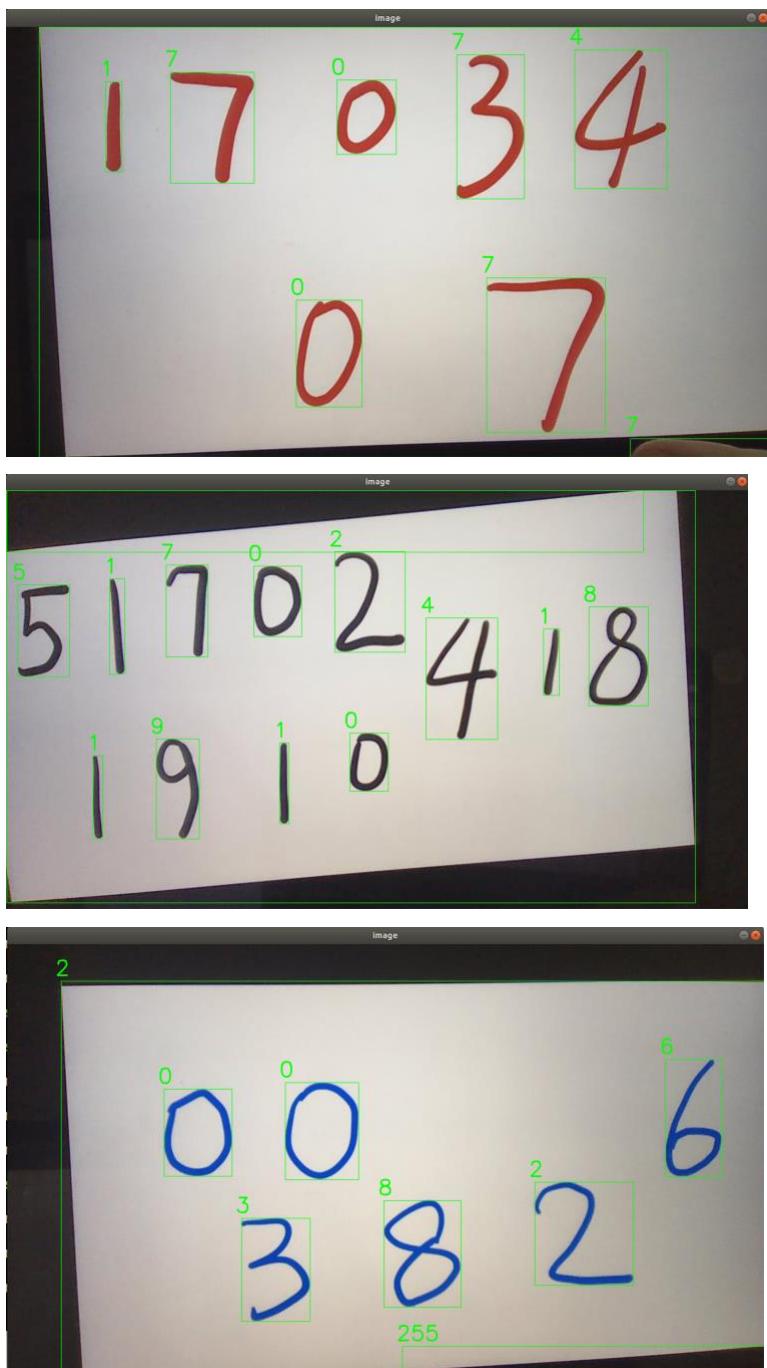
recognize.py

```
import ailib.recognize_hdr_lib as recognize_hdr

# variables for parameters
model_path = 'models/mnist_full.bin'
image_path = 'myimages'
width_th = 7
height_th = 20

# train classifier
recognize_hdr.recognize(model_path, image_path, width_th, height_th)
```

B. 执行修改后的 recognize.py 脚本，并将结果截图贴上



C. 比较实验 2 及实验 3 的结果，是否有什么差异？请写下差异的分析与造成差异的原因

1. 依然存在将背景识别为数字的问题
2. 原本识别错了 2 个错误，改为完整数据集后，修正了一个错误
3. 差异的原因在于数据集数量更大，更能表征整体特征

IV. 实验 4. 实验 2、3 中若有无法辨识出的手写数字，调校 width_th、height_th，让手写数字都可辨识

已经可以识别，不需要调整