

# 《计算机视觉》

## 实验指导书

武汉科技大学信息科学与工程学院

# 目 录

实验一	基于灰度阈值的图像分割 .....	1
实验二	Haar 人脸检测.....	3
实验三	卷积神经网络图像分类 .....	7
实验四	多位手写数字识别 .....	10

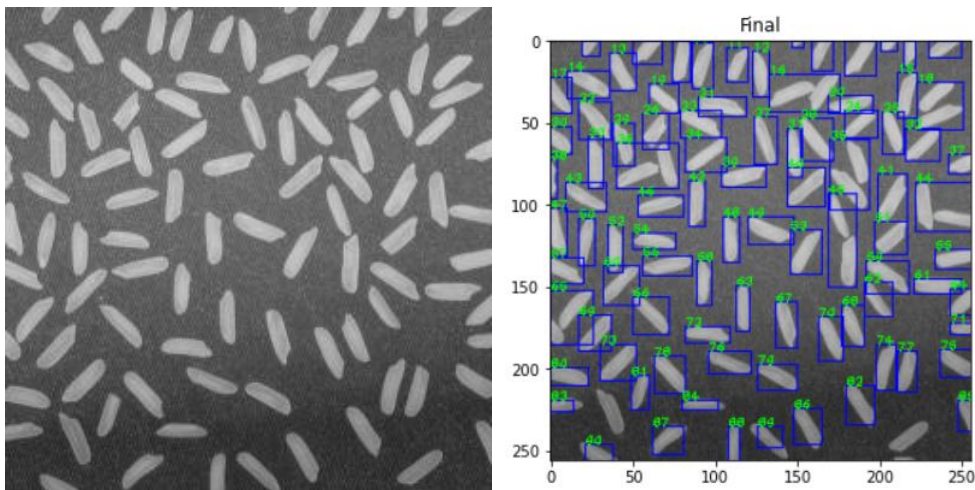
# 实验一 基于灰度阈值的图像分割

## 一、实验目的

- 1) 掌握 Python 下 OpenCV 基本用法
- 2) 掌握大津算法的使用
- 3) 熟悉图像的显示、变换、处理等操作
- 4) 熟悉灰度阈值分割应用

## 二、实验内容

给一张包括很多米粒的图片，采用大津（Otsu）灰度阈值分割方法，将米粒和背景分离，然后用边界框方法框出所有米粒，最后统计米粒的数量。



## 三、实验仪器、设备及材料

每个学生上机使用一台 PC 机进行程序开发，PC 机的硬件、软件环境如下：

- 1) 硬件：CPU 酷睿 4 核 3.0GHZ 4GB 内存，80G 硬盘空间。
- 2) 软件：

操作系统：Windows 7 或 10

开发工具：Anaconda, Visual Studio Code

## 四、实验原理

- 1、采用 OpenCV 中 `imread` 函数读取图片, `cvtColor(image, cv.COLOR_BGR2GRAY)` 函数将图像转为灰度图
- 2、调用 OpenCV 中的大津算法 `thr, bw = cv.threshold(gray, 0, 0xff, cv.THRESH_OTSU)` 获得阈值和二值图 (米粒白, 背景黑)
- 3、调用 OpenCV 中的轮廓提取函数 `cnts, _ = cv.findContours(seg, cv.RETR_EXTERNAL, cv.CHAIN_APPROX_SIMPLE)`, 得到所有米粒的轮廓存入 `cnts` 列表
- 4、循环 `cnts` 列表, 调用 `cv.contourArea(c)` 计算轮廓面积, 过滤面积过小的结果, 用如下代码在原图像上绘制边界框和数字:

```
# 区域画框并标记
x, y, w, h = cv.boundingRect(c)
cv.rectangle(image, (x, y), (x+w, y+h), (0, 0, 0xff), 1)
cv.putText(image, str(count), (x, y), cv.FONT_HERSHEY_PLAIN, 0.5, (0, 0xff, 0))
```

## 五、实验报告要求

每位同学必须独立书写实验报告, 注意: 不得抄袭他人的报告 (或给他人抄袭), 一旦发现, **成绩为零分**。实验报告的内容应包括:

按照实验内容和实验步骤要求完成实验任务后, 报告中给出包含详细注释的源程序清单, 并对调试分析过程进行描述: 包括测试数据、测试输出结果, 以及对程序调试过程中存在问题的思考 (列出主要问题的出错现象、出错原因、解决方法及效果等)。

## 六、实验注意事项

- 1) 每个同学必须按时参加实验, 尽可能在实验上机前完成全部或部分代码。
- 2) 实验中严禁随意拷贝其他同学的程序或给其他同学拷贝。
- 3) 在实验过程中, 爱惜实验器材, 避免对实验器材造成不必要的损害。如发生损坏照价赔偿。
- 4) 在实验用机上使用移动存储设备之前, 必须用杀毒软件进行检测, 否则不能使用。

## 实验二 Haar 人脸检测

### 一、实验目的

- 1) 掌握 OpenCV 中的人脸检测方法。
- 2) 掌握边界框的数据结构及绘制方法。
- 3) 熟悉 Haar 级联分类器的使用方法。
- 4) 熟悉人脸检测应用。

### 二、实验内容

给定一副图像，将其中的人脸部分准备框出来。首先定义分类器，这里采用了 Haar 级联分类器；然后采用分类器的多尺度检测函数返回所有预测的人脸边界框；最后将所有边界框绘制到图像上。如下图为原始图像：



下图为检测结果：



需要不断调整检测参数才能得到理想的结果。

### 三、实验仪器、设备及材料

每个学生上机使用一台 PC 机进行程序开发，PC 机的硬件、软件环境如下：

1) 硬件：CPU 酷睿 4 核 3.0GHZ 4GB 内存，80G 硬盘空间。

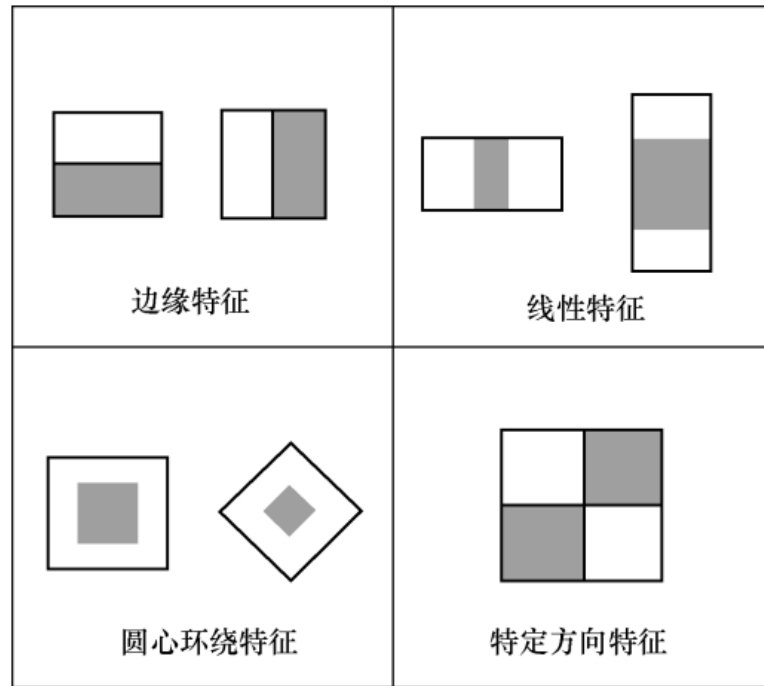
2) 软件：

操作系统：Windows 7 或 10

开发工具：Anaconda, Visual Studio Code

### 四、实验原理

1、人脸识别使用 Haar 级联分类器，通过对比分析相邻图像区域来判断给定图像或子图像与 已知对象是否匹配。Haar 特征分为 4 种类型：边缘特征、线性特征、中心特征和对角线特征。将这些特征组合成特征模板，特征模板内有白色和黑色两种矩形，并定义该模板的特征值为白色矩形像素之和减去黑色矩形像素之和。Lienhart R. 等人对 Haar-like 矩形特征库做了进一步扩展，扩展后的特征大致分为 4 种类型——边缘特征、线性特征、圆心环绕特征和特定方向特征，如图所示。



Haar 特征的提取简单来说就是通过不断改变模板的大小、位置和类型，将白色矩形区域像素之和减去黑色矩形区域像素之和，从而得到每种类型模板的大量子特征。

Haar 级联是一个基于 Haar 特征的级联分类器，级联分类器能够把弱分类器串联成强分类器。弱分类器可以理解为性能受限的分类器，它们没有办法正确地分辨所有事物。当问题很简单时，弱分类器产生的结果是可以接受的，但是问题一旦复杂起来，结果就会出现很大的偏差。强分类器可以正确地对数据进行分类，建立一个实时系统来保证分类器运行良好并且足够简单。

2、所提供图像数据需要转换颜色通道才能正常显示，比如源图像是 BGR 格式的，需要转换成 RGB 格式才能正常显示颜色，代码如下：

```
image = cv.cvtColor(image, cv.COLOR_BGR2RGB)
```

3、初始化 haar 级联分类器对象：

```
haar = cv.CascadeClassifier('haarcascade_frontalface_default.xml')
```

4、调用 detectMultiScale 函数检测人脸并返回边界框：

```
rects = haar.detectMultiScale(image, scaleFactor=1.2, minNeighbors=5)
```

image 是用 cv.imread 函数读取的图像数据，scaleFactor 的含义是检测尺度“每次缩小几倍”，minNeighbors 直译过来即最小邻居数量，也就是一个矩形框旁边有多少个矩形框，至少有多少个矩形框，该位置才算是人脸。minNeighbors 最

开始设置为 0，scaleFactor 从 1.05 开始，逐步上升到 1.5；然后提高 minNeighbors，观察检测结果。直到获得比较理想的结果。

## 五、实验报告要求

每位同学必须独立书写实验报告，注意：不得抄袭他人的报告（或给他人抄袭），一旦发现，**成绩为零分**。实验报告的内容应包括：

按照实验内容和实验步骤要求完成实验任务后，报告中给出包含详细注释的源程序清单，并对调试分析过程进行描述：包括测试数据、测试输出结果，以及对程序调试过程中存在问题的思考（列出主要问题的出错现象、出错原因、解决方法及效果等）。

## 六、实验注意事项

- 1) 每个同学必须按时参加实验，尽可能在实验上机前完成全部或部分代码。
- 2) 实验中严禁随意拷贝其他同学的程序或给其他同学拷贝。
- 3) 在实验过程中，爱惜实验器材，避免对实验器材造成不必要的损害。如发生损坏照价赔偿。
- 4) 在实验用机上使用移动存储设备之前，必须用杀毒软件进行检测，否则不能使用。



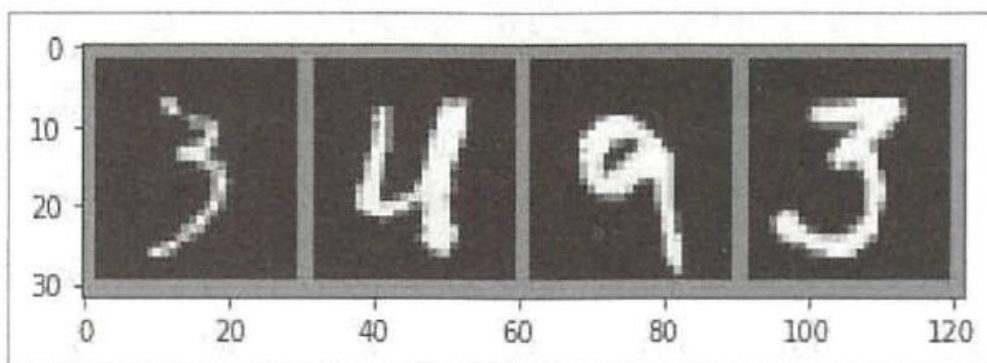
## 实验三 卷积神经网络图像分类

### 一、实验目的

- 1) 掌握卷积神经网络图像分类。
- 2) 掌握数据集的载入、预览及训练。
- 3) 掌握卷积神经网络的设计、损失函数和优化器设置。

### 二、实验内容

设计卷积神经网络进行手写数字识别，采用 MNIST 数据集进行训练和验证。



具体设计参考教材第六章的“6.4 实战手写数字识别”。要求能输出每一个 Epoch 的损失、准确度及最终测试集上的准确度。并且能对随机挑选的数字图像进行正确预测，比如上图要求输出为 3，4，9，3。

要求能保存模型的训练权重，并能直接读取进行前向预测。

### 三、实验仪器、设备及材料

每个学生上机使用一台 PC 机进行程序开发，PC 机的硬件、软件环境如下：

- 1) 硬件：CPU 酷睿 4 核 3.0GHZ 4GB 内存，80G 硬盘空间。
- 2) 软件：

操作系统：Windows 7 或 10

开发工具：Anaconda, Visual Studio Code

## 四、实验原理

具体原理参考教材第六章 “PyTorch 基础”。主要是参照 LeNet 进行设计卷积神经网络，为了节省训练时间，要对网络进行一些简化。

卷积层，下图中的红线部分改为 1,6 和 6, 16

```
self.conv1=torch.nn.Sequential(  
    torch.nn.Conv2d(1,64,kernel_size=3,stride=1,padding=1),  
    torch.nn.ReLU(),  
    torch.nn.Conv2d(64,128,kernel_size=3,stride=1,padding=1),  
    torch.nn.ReLU(),  
    torch.nn.MaxPool2d(stride=2,kernel_size=2))
```

全连接层，下图中红线部分改为 14\*14\*16

```
self.dense=torch.nn.Sequential(  
    torch.nn.Linear(14*14*128,1024),  
    torch.nn.ReLU(),  
    torch.nn.Dropout(p=0.5),  
    torch.nn.Linear(1024, 10))
```

模型的训练权重采用如下方法保存：

```
torch.save({'mycnn':model.state_dict()}, 'mycnn.pt')
```

已保存的模型权重可以通过如下方法读取：

```
state_dict = torch.load('mycnn.pt')  
model.load_state_dict(state_dict['mycnn'])
```

## 五、实验报告要求

每位同学必须独立书写实验报告，注意：不得抄袭他人的报告（或给他人抄袭），一旦发现，成绩为零分。实验报告的内容应包括：

按照实验内容和实验步骤要求完成实验任务后，报告中给出包含详细注释的源程序清单，并对调试分析过程进行描述：包括测试数据、测试输出结果，以及对程序调试过程中存在问题的思考（列出主要问题的出错现象、出错原因、解决方法及效果等）。

## 六、实验注意事项

- 1) 每个同学必须按时参加实验，尽可能在实验上机前完成全部或部分代码。
- 2) 实验中严禁随意拷贝其他同学的程序或给其他同学拷贝。
- 3) 在实验过程中，爱惜实验器材，避免对实验器材造成不必要的损害。如发生损坏照价赔偿。
- 4) 在实验用机上使用移动存储设备之前，必须用杀毒软件进行检测，否则不能使用。

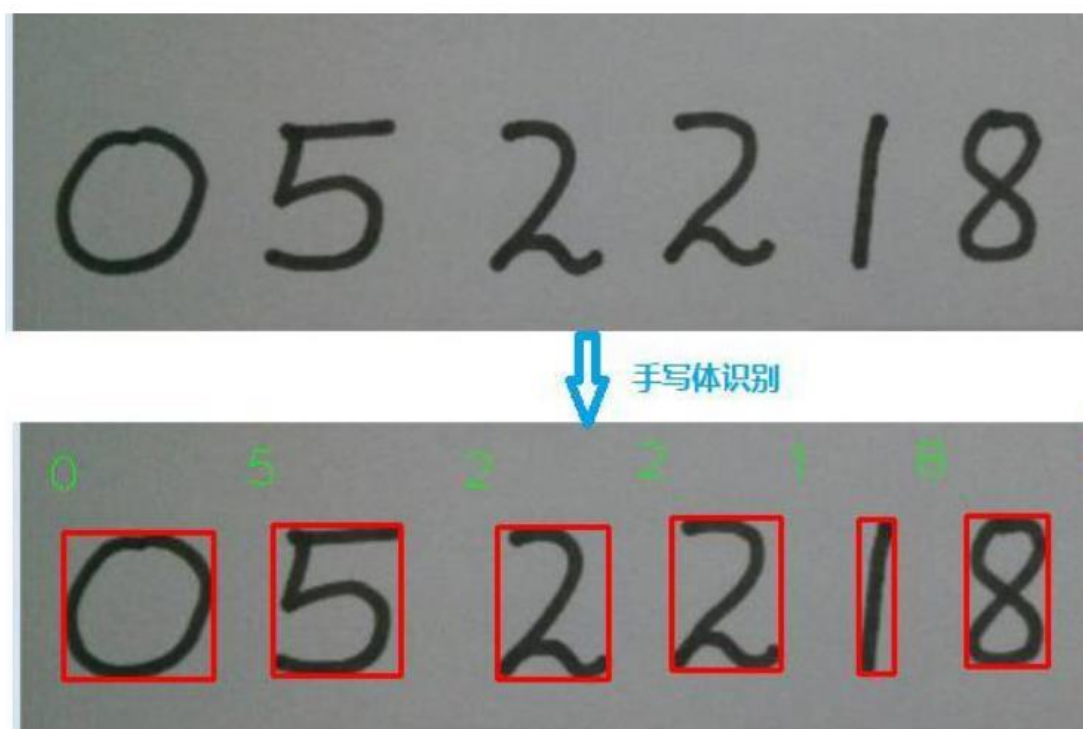
## 实验四 多位手写数字识别

### 一、实验目的

- 1) 掌握卷积神经网络的预测流程;
- 2) 掌握 OpenCV 中的边界框检测;
- 3) 熟悉 CNN 识别多位数字应用;
- 4) 熟悉 OpenCV 与 Pytorch 的组合应用。

### 二、实验内容

CNN 只能识别单个数字，但实际应用中往往需要识别多位数字，并且组合成一个结果，典型的应用比如煤气表数值的自动读取。首先读入图像，然后应用灰度阈值分割方法提取前景和背景；然后调用边界框检测函数获取每个数字的边界框，将数字从图像中提取出来；最后将提取出来的图像交给 CNN 进行预测。具体结果如下图所示：



### 三、实验仪器、设备及材料

每个学生上机使用一台 PC 机进行程序开发，PC 机的硬件、软件环境如下：

1) 硬件：CPU 酷睿 4 核 3.0GHZ 4GB 内存，80G 硬盘空间。

2) 软件：

操作系统：Windows 7 或 10

开发工具：Anaconda, Visual Studio Code

### 四、实验原理

1、灰度阈值分割及边界框获取原理参见实验一。

2、将获得图像缩放成实验三中CNN可以接收的输入尺寸，代码如下：

```
image = cv.resize(image, [28, 28])
```

由于灰度阈值分割的背景是白，前景是黑，需要进行取反，以符合CNN训练用的数据特征：

```
image = 255 - bw
```

还需要进行变换，与MNIST的数据预处理保持一致：

```
img = transform(img)
```

获得的Tensor还缺一个批次维度，用unsqueeze\_函数补上：

```
img.unsqueeze_(0)
```

3、基于实验三的 CNN 训练结果保存的权重，这里只需要读入之前保存的权重文件，代码如下：

```
state_dict = torch.load('mycnn.pt')
```

```
model.load_state_dict(state_dict['mycnn'])
```

模型载入后就可以进行模型预测了

```
result = model(img)
```

所得到的结果还需要取最大列号，即为预测的数字值

```
print(torch.argmax(result, dim=-1))
```

4、结果的展示，将预测的结果显示在边界框上。

## 五、实验报告要求

每位同学必须独立书写实验报告，注意：不得抄袭他人的报告（或给他人抄袭），一旦发现，**成绩为零分**。实验报告的内容应包括：

按照实验内容和实验步骤要求完成实验任务后，在规定的实验报告封面之后，附上：实验源程序清单，实验测试数据、测试输出结果截图，以及对程序调试过程中存在问题的思考（列出主要问题的出错现象、出错原因、解决方法及效果等）。

## 六、实验注意事项

- 1) 每个同学必须按时参加实验，尽可能在实验上机前完成全部或部分代码。
- 2) 实验中严禁随意拷贝其他同学的程序或给其他同学拷贝。
- 3) 在实验过程中，爱惜实验器材，避免对实验器材造成不必要的损害。如发生损坏照价赔偿。
- 4) 在实验用机上使用移动存储设备之前，必须用杀毒软件进行检测，否则不能使用。