

# Street View House Number Recognition

## 深度学习技术与应用作业 #3

王宇涛

北京大学

2020 年 4 月 9 日

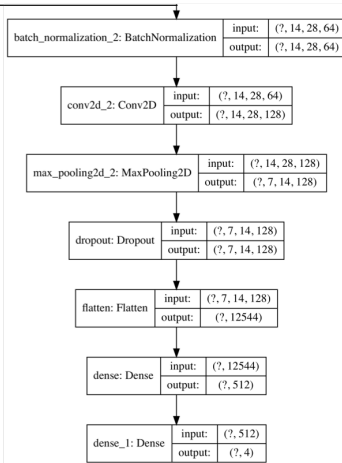
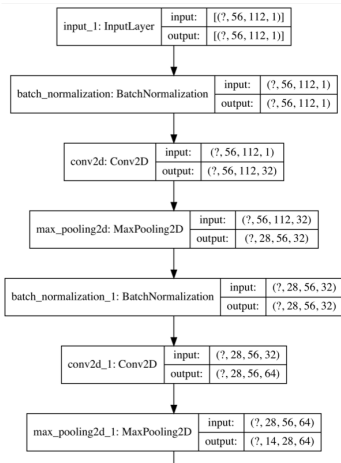
# 目录

数字定位网络

数字识别网络

结果与讨论

# 数字定位网络——结构



# 数字定位网络——超参数

```
inputs = Input(shape = (56,112,1))

model = layers.BatchNormalization()(inputs)
model = layers.Conv2D(32, (7,7), padding='same', activation='relu')(model)
model = layers.MaxPool2D(pool_size=2)(model)

model = layers.BatchNormalization()(model)
model = layers.Conv2D(64, (5,5), padding='same', activation='relu')(model)
model = layers.MaxPool2D(pool_size=2)(model)

model = layers.BatchNormalization()(model)
model = layers.Conv2D(128, (3,3), padding='same', activation='relu')(model)
model = layers.MaxPool2D(pool_size=2)(model)

model = layers.Dropout(0.5)(model)

model = layers.Flatten()(model)

model = layers.Dense(512, activation='relu')(model)

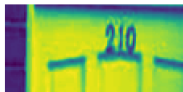
outputs = layers.Dense(4)(model)

model = Model(inputs=inputs, outputs=outputs)
```

# 数字定位网络——训练方法

## 输入图片

大小 56x112，经过去均值、归一化处理



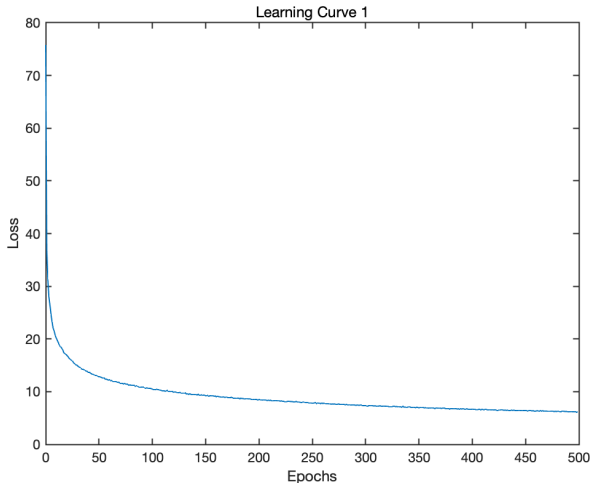
## 输出数组

整串数字的包围框信息，如  $[42.0, 15.6, 69.7, 41.1]$  分别表示包围框的左边界、上边界、右边界和下边界

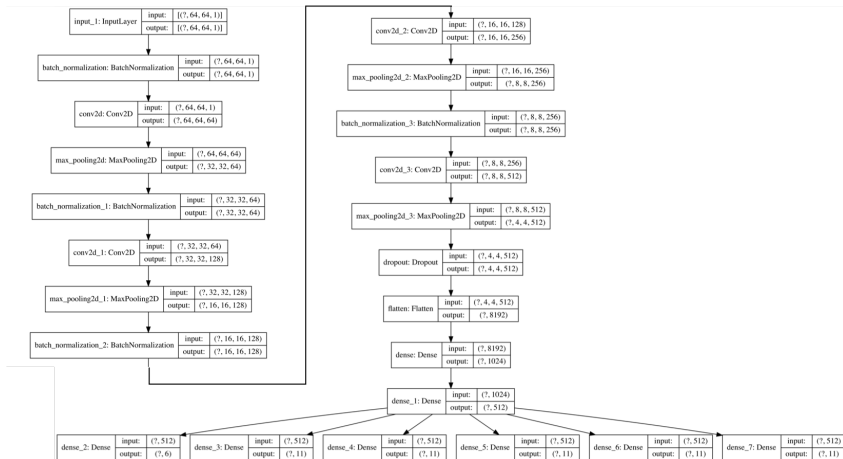
## 训练方法

使用 RMSprop 优化方法，训练 500 轮，损失函数为 MSE

# 数字定位网络——训练曲线



# 数字识别网络——结构



# 数字识别网络——超参数

```
inputs = Input(shape = (64,64,1))

model = layers.BatchNormalization()(inputs)
model = layers.Conv2D(64, (3,3), padding='same', activation='relu')(model)
model = layers.MaxPool2D(pool_size=2)(model)

model = layers.BatchNormalization()(model)
model = layers.Conv2D(128, (3,3), padding='same', activation='relu')(model)
model = layers.MaxPool2D(pool_size=2)(model)

model = layers.BatchNormalization()(model)
model = layers.Conv2D(256, (3,3), padding='same', activation='relu')(model)
model = layers.MaxPool2D(pool_size=2)(model)

model = layers.BatchNormalization()(model)
model = layers.Conv2D(512, (3,3), padding='same', activation='relu')(model)
model = layers.MaxPool2D(pool_size=2)(model)

model = layers.Dropout(0.5)(model)

model = layers.Flatten()(model)

model = layers.Dense(1024, activation='relu')(model)
model = layers.Dense(512, activation='relu')(model)

output_0 = layers.Dense(6, activation='softmax')(model)
output_1 = layers.Dense(11, activation='softmax')(model)
output_2 = layers.Dense(11, activation='softmax')(model)
output_3 = layers.Dense(11, activation='softmax')(model)
output_4 = layers.Dense(11, activation='softmax')(model)
output_5 = layers.Dense(11, activation='softmax')(model)

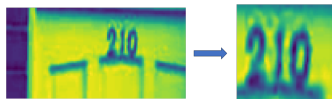
outputs = [output_0, output_1, output_2, output_3, output_4, output_5]
```



# 数字识别网络——训练方法

## 输入图片

大小 64x64，通过上述数字定位网络获得的包围框进行剪裁而得



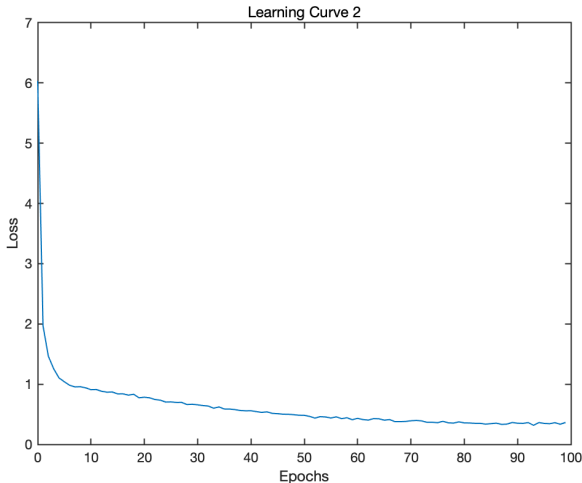
## 输出数组

六个 one-hot 输出，分别为数字个数和五位数字，并用 '10' 表示该位不存在数字，如 [3, 2, 1, 0, 10, 10] 表示共有 3 个数字，连起来为 210

## 训练方法

使用 RMSprop 优化方法，训练 100 轮，损失函数为交叉熵

# 数字识别网络——训练曲线



# 结果与讨论

## 识别准确率

- ▶ 忽略给定的位置信息 51.8%
- ▶ 给定整串数字包围框 83.0%

## 结果分析

- ▶ 受计算资源限制，无法构造更深的网络，无法更快地训练模型，故识别准确度较低
- ▶ 数字定位网络较为简单，可以考虑使用 Faster-RCNN 等网络更加精确地检测候选区域