**全局特征与局部特征提取神经网络设计**

**-----------以及LeNet对比实验**

**实验者：徐征**

目录

[实验一 探究不同因素对特征提取的影响 1](#_Toc30110656)

[一、实验目的 1](#_Toc30110657)

[二、实验原理 1](#_Toc30110658)

[三、实验步骤及结果 1](#_Toc30110659)

[四、实验结论 7](#_Toc30110660)

[实验二 探究全局特征提取之轮廓特征提取卷积核设计 8](#_Toc30110661)

[一、实验目的 8](#_Toc30110662)

[二、实验原理 8](#_Toc30110663)

[三、实验步骤及结果 8](#_Toc30110664)

[四、实验结论 14](#_Toc30110665)

[实验三 与传统LeNet神经网络的碰撞 15](#_Toc30110666)

[一、实验目的 15](#_Toc30110667)

[二、实验原理 15](#_Toc30110668)

[三、实验步骤及结果 15](#_Toc30110669)

[四、实验结论 19](#_Toc30110670)

# 实验一 探究不同因素对特征提取的影响

一、实验目的

探究卷积核的大小、卷积步长以及普通卷积、深度卷积对特征提取的影响。

二、实验原理

利用控制变量法，对卷积核大小、卷积步长以及卷积方式分别进行测试，测试图片为九张，利用LeNet进行试验，实验过程中可视化每一层神经网络卷积、激活、池化之后得到的图像，再进行视觉观察比较。

三、实验步骤及结果

（1）、探究卷积核大小对特征提取的影响

本次实验可视化第一层神经网络得到的卷积图像，控制卷积层输入数据通道数为3（RGB，in\_channels=3），输出数据通道30（卷积核数量，out\_channels=30），卷积方式为普通卷积（group=1），卷积步长为1（stride=1），padding=0，我们将卷积核大小从1\*1以2为步长递增到11\*11，以九张图片测试，控制变量以及结果如下：

表1-1 变量及值

|  |  |
| --- | --- |
| key | value |
| In\_channels | 3 |
| Out\_channels | 30 |
| Group | 1 |
| Stride | 1 |
| Padding | 0 |

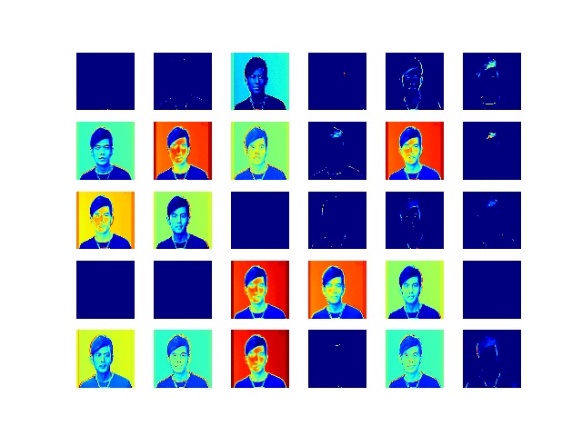
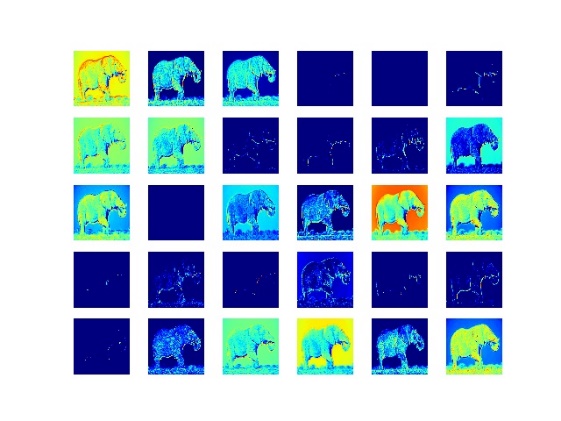


图1-1 卷积核大小 1\*1

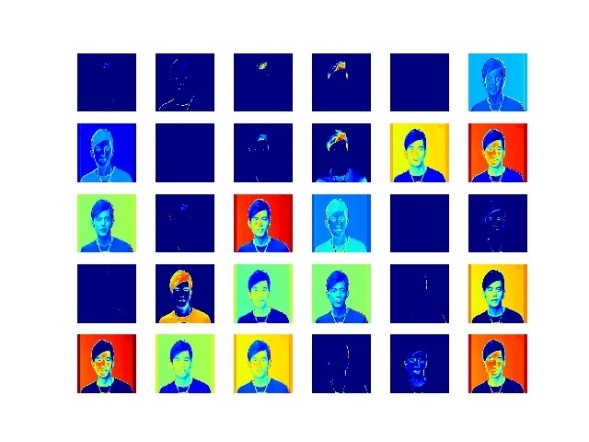
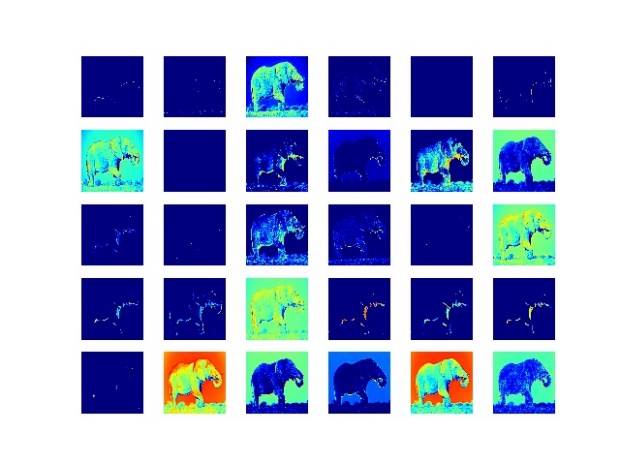


图1-2 卷积核大小 3\*3

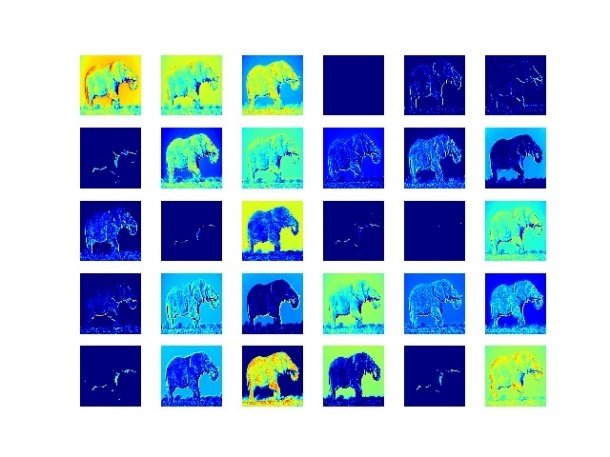


图1-3 卷积核大小 5\*5



图1-4 卷积核大小 7\*7

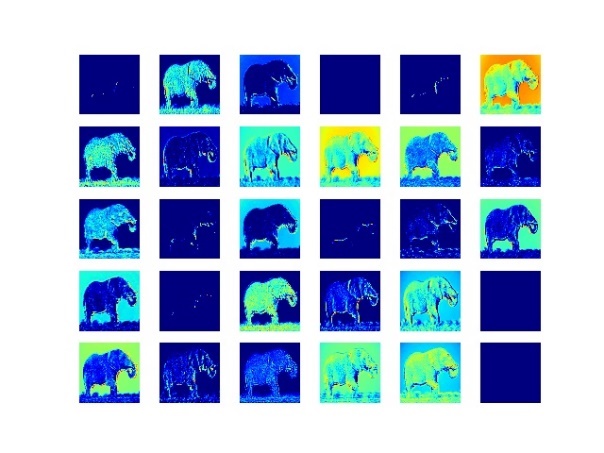


图1-5 卷积核大小 9\*9

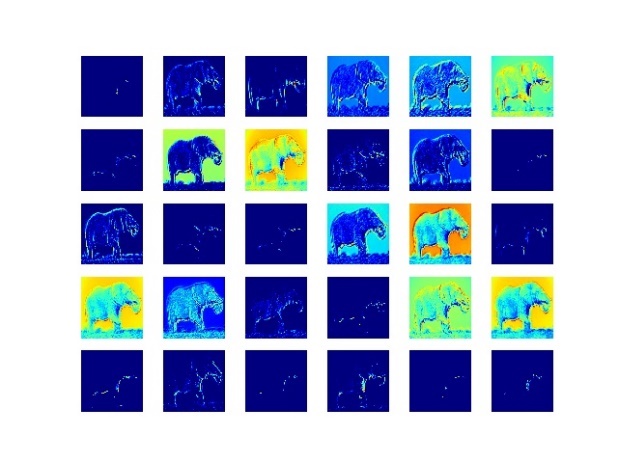


图1-6 卷积核大小 11\*11

（2）、探究卷积步长对特征提取的影响

本次实验可视化第一层神经网络得到的卷积图像，控制卷积层输入数据通道数为3（RGB，in\_channels=3），输出数据通道30（卷积核数量，out\_channels=30），卷积方式为普通卷积（group=1），卷积核大小为5\*5（kernel\_size），padding=0，我们将卷积步长stride从1递增到7，以九张图片测试，控制变量以及结果如下：

表2-1 变量及值

|  |  |
| --- | --- |
| key | value |
| In\_channels | 3 |
| Out\_channels | 30 |
| Group | 1 |
| Kernel\_size | 5\*5 |
| Padding | 0 |

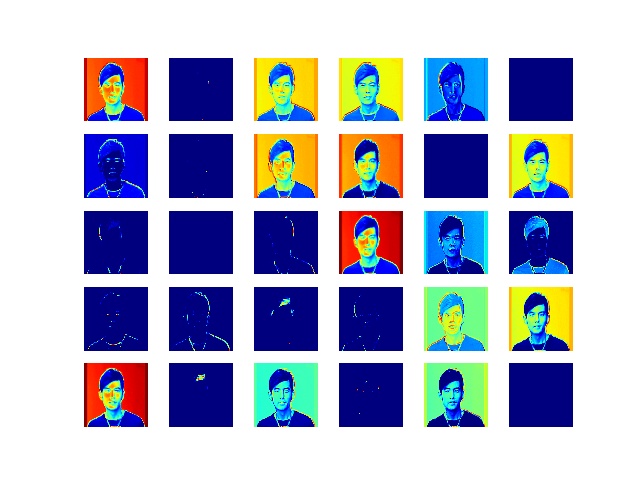


图2-1 Stride=1



图2-2 Stride=3



图2-3 Stride=5



图2-4 Stride=7

（3）、探究卷积方式对特征提取的影响

本次实验可视化第一层神经网络得到的卷积图像，控制卷积层输入数据通道数为3（RGB，in\_channels=3），输出数据通道30（卷积核数量，out\_channels=30），卷积方式为普通卷积（group=1），卷积核大小为5\*5（kernel\_size），padding=0，卷积步长（stride=1），分别利用普通卷积（group=1）以及深度卷积（group=in\_channels)，以九张图片测试，控制变量以及结果如下：

表3-1 变量及值

|  |  |
| --- | --- |
| key | value |
| In\_channels | 3 |
| Out\_channels | 30 |
| Group | 1 |
| Kernel\_size | 5\*5 |
| Stride | 1 |
| Padding | 0 |



图3-1 普通卷积

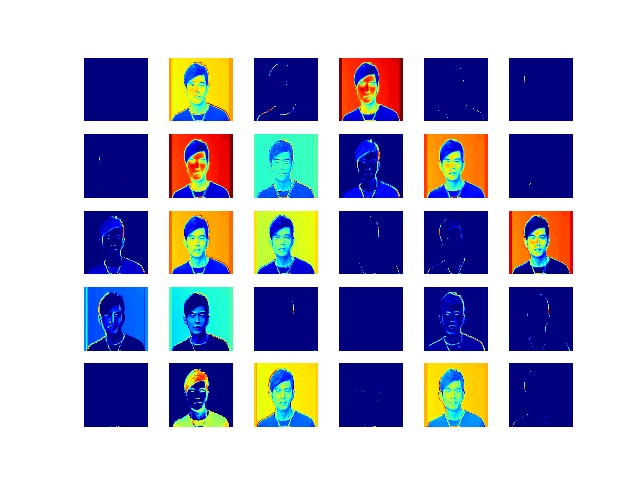


图3-2 深度卷积

四、实验结论

通过本次实验以及观察可以发现卷积核的大小对特征提取的影响不大，并不是卷积核越大越好；同时，卷积步长越大，特征提取轮廓斜街信息越粗糙，可见卷积步长对特征的提取具有一定影响，同时经过实验可以发现普通卷积与深度卷积对特征提取的影响也不大，由此可见，对特征提取的影响主要在于卷积核内容的设计以及神经网络的设计。

# 实验二 探究全局特征提取之轮廓特征提取卷积核设计

一、实验目的

探究设计卷积核及神经网络以达到提取全局特征之轮廓特征。

二、实验原理

参考Laplace算子自定义卷积核，参考Canny边缘检测算法对特征提取进行优化，并具体设计两层神经网络以提取轮廓特征。

三、实验步骤及结果

（1）、轮廓提取卷积核设计

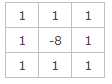
参考Laplace算子设计卷积核，Laplace算子的差分形式为：



写成filter mask的形式如下：



该mask的特点，mask在上下左右四个90度的方向上结果相同，也就是说在90度方向上无方向性。为了让该mask在45度的方向上也具有该性质，对该filter mask进行扩展定义为



将Laplace算子写成filter mask后，其操作大同小异于其他的空间滤波操作。将filter mask在原图上逐行移动，然后mask中数值与其重合的像素相乘后求和，赋给与mask中心重合的像素，对图像的第一，和最后的行和列无法做上述操作的像素赋值零，就得到了拉普拉斯操作结果。

故自定义设计卷积核的weight为：

对彩色图像RGB三个通道进行卷积，对得到的三个张量按列拼接。原图、可视化卷积后结果及简单二值化后结果分别为：

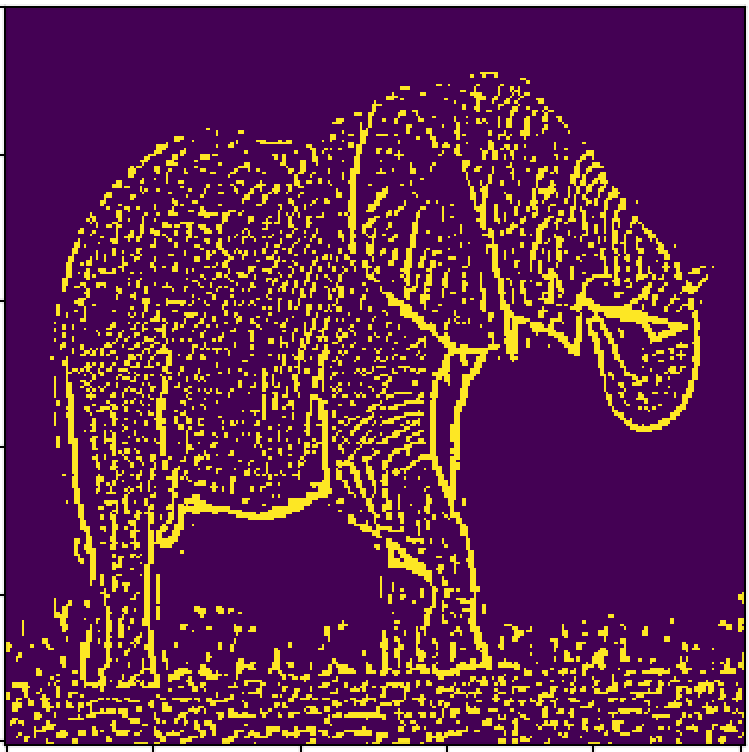




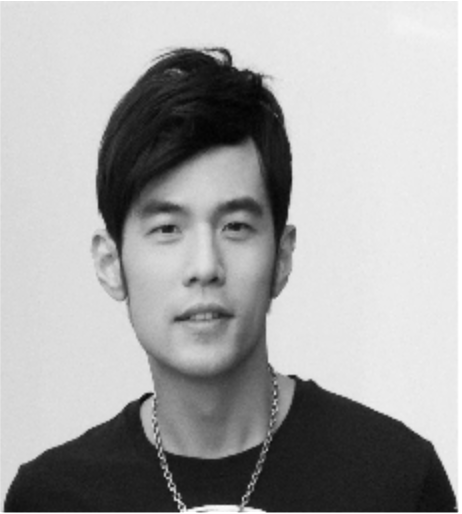


图1-1 轮廓提取

（2）、减少输入通道并提取轮廓特征

考虑到输入通道RGB三个通道的张量拼接会丢失特征信息，我们首先将原图进行灰度化处理，并予以二值化进行附加处理，进行卷积得到结果如下：

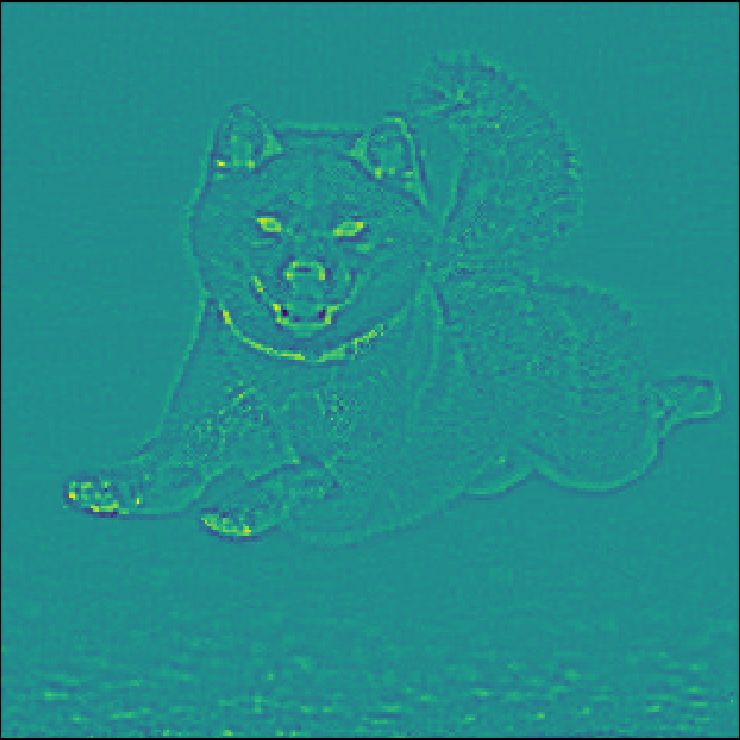
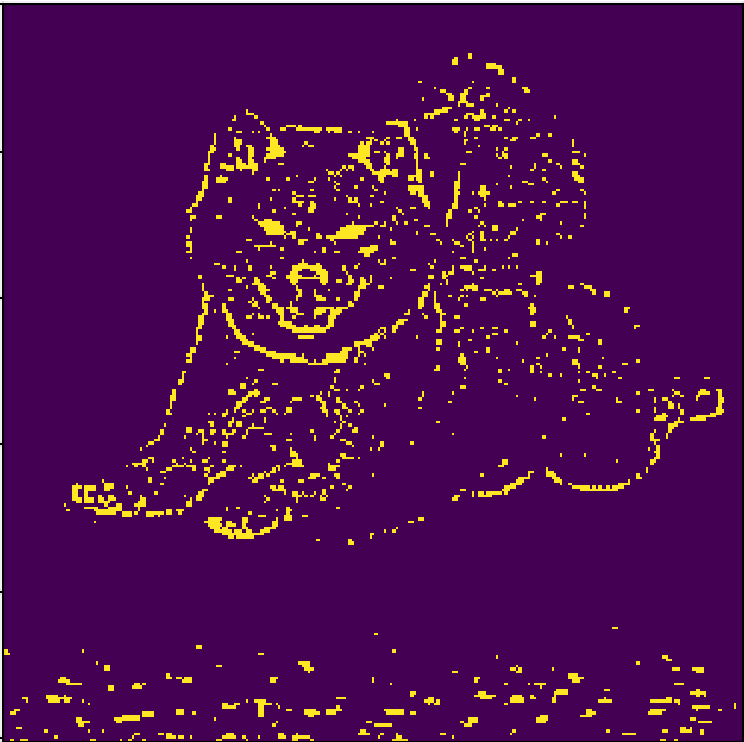
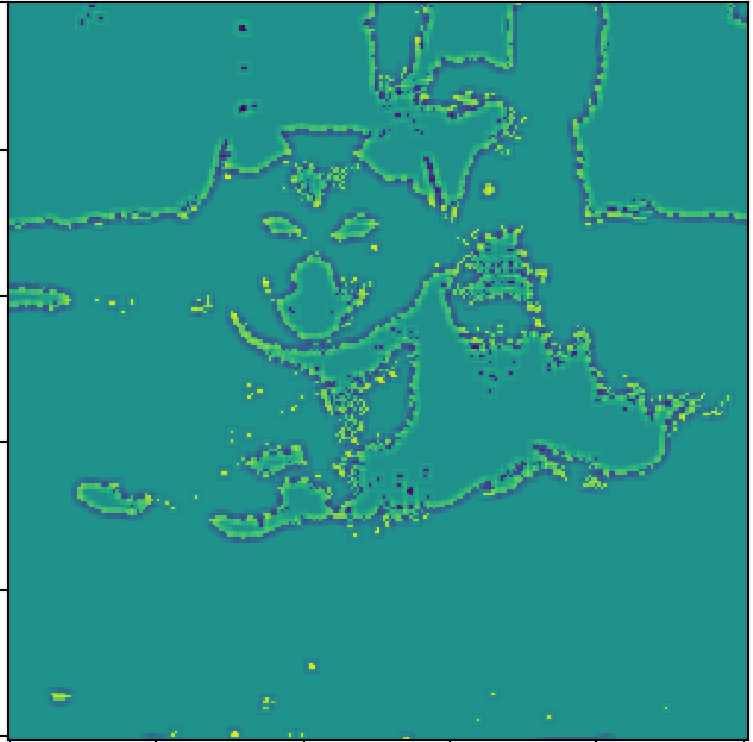
  

图2-1 轮廓提取（灰度化）





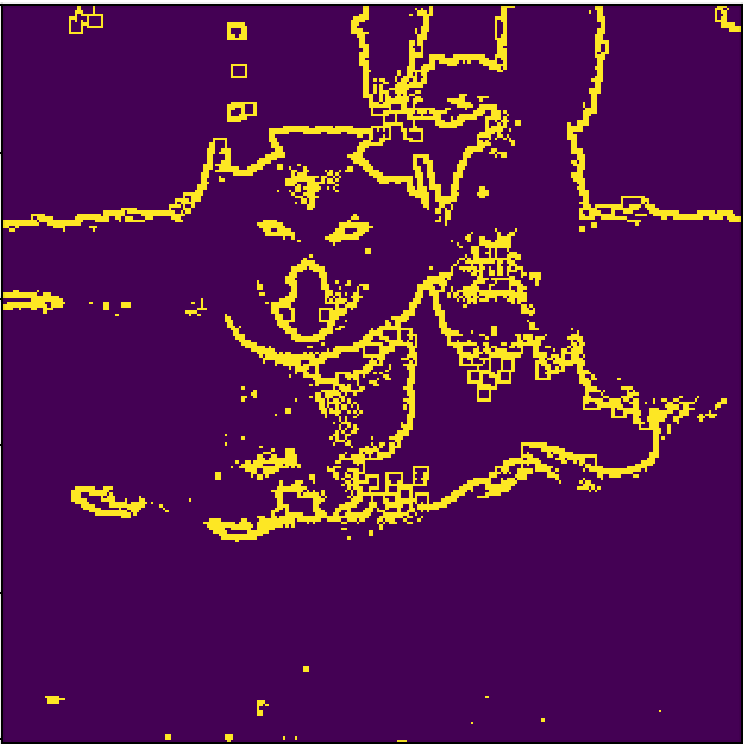


图2-2 轮廓提取（二值化）

（3）、轮廓特征提取优化

参考canny提取轮廓算法，首先利用高斯核对灰度图进行卷积滤波，高斯核weight的设计为：

再对卷积后的结果进行自定义卷积核的第二次卷积，在对卷积结果结果进行简单的二值化，原图、高斯卷积结果、自定义卷积核卷积结果及简单二值化后结果分别为：

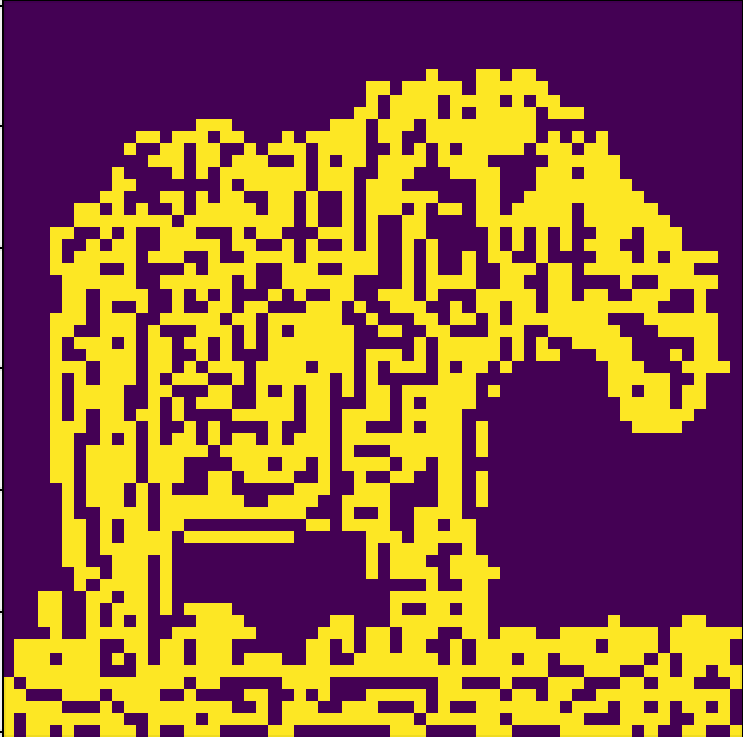


图3-1 轮廓提取（高斯滤波）

（4）、设计神经网络及可视化

设计两层神经网络，第一层神经网络采用自定义高斯核进行卷积，第二层神经网络采用自定义轮廓特征提取卷积核进行卷积，其神经网络流程如下：

可视化两次卷积之后的结果为：



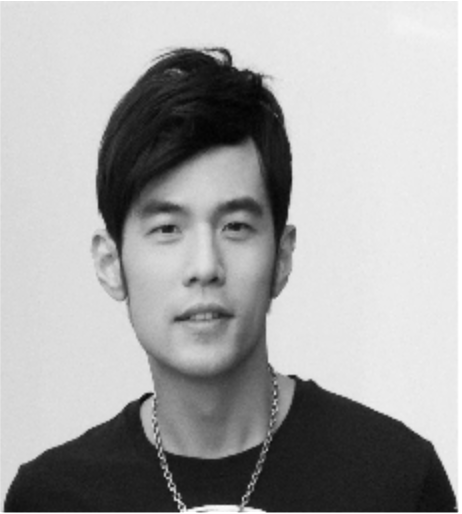


图4-1 神经网络卷积可视化

四、实验结论

通过本次实验以及观察可以发现此次卷积核的设计可以有效的提取轮廓信息，对全局特征提取的效果较好，初步优化之后可以实现自己设计的神经网络。

# 实验三 与传统LeNet神经网络的碰撞

一、实验目的

通过自定义神经网络与传统LeNet的比较，探究自定义神经网络是否能有效将全局变量与局部变量融合，以达到准确识别手写字体的目的。

二、实验原理

利用传统神经网络LeNet以及自定义神经网络分别进行手写字体识别的测试，通过对比探究此方法的可行性。

三、实验步骤及结果

（1）、LeNet与自定义神经网络的对比

控制全连接层之前得到的结果为60\*7\*7的tensor，即全连接层中m1:[60\*36]，m2:[25\*1500]，激活函数均采用ReLu激活函数，进行一次实验之后的结果为：

1. LeNet实验结果

epoch: 1, count:  6000, loss: 0.902

epoch: 1, count: 12000, loss: 0.435

epoch: 1, count: 18000, loss: 0.362

epoch: 1, count: 24000, loss: 0.310

epoch: 1, count: 30000, loss: 0.253

epoch: 1, count: 36000, loss: 0.274

epoch: 1, count: 42000, loss: 0.241

epoch: 1, count: 48000, loss: 0.241

epoch: 1, count: 54000, loss: 0.242

epoch: 1, count: 60000, loss: 0.232

epoch: 2, count:  6000, loss: 0.210

epoch: 2, count: 12000, loss: 0.191

epoch: 2, count: 18000, loss: 0.192

epoch: 2, count: 24000, loss: 0.177

epoch: 2, count: 30000, loss: 0.192

epoch: 2, count: 36000, loss: 0.176

epoch: 2, count: 42000, loss: 0.178

epoch: 2, count: 48000, loss: 0.169

epoch: 2, count: 54000, loss: 0.172

epoch: 2, count: 60000, loss: 0.153

epoch: 3, count:  6000, loss: 0.149

epoch: 3, count: 12000, loss: 0.130

epoch: 3, count: 18000, loss: 0.140

epoch: 3, count: 24000, loss: 0.153

epoch: 3, count: 30000, loss: 0.149

epoch: 3, count: 36000, loss: 0.146

epoch: 3, count: 42000, loss: 0.134

epoch: 3, count: 48000, loss: 0.139

epoch: 3, count: 54000, loss: 0.133

epoch: 3, count: 60000, loss: 0.130

Accuracy of the network on the 10000 test images: 96 %

1. 自定义神经网络实验结果

epoch: 1, count:  6000, loss: 6.840

epoch: 1, count: 12000, loss: 0.392

epoch: 1, count: 18000, loss: 0.327

epoch: 1, count: 24000, loss: 0.311

epoch: 1, count: 30000, loss: 0.278

epoch: 1, count: 36000, loss: 0.268

epoch: 1, count: 42000, loss: 0.231

epoch: 1, count: 48000, loss: 0.239

epoch: 1, count: 54000, loss: 0.218

epoch: 1, count: 60000, loss: 0.216

epoch: 2, count:  6000, loss: 0.195

epoch: 2, count: 12000, loss: 0.171

epoch: 2, count: 18000, loss: 0.148

epoch: 2, count: 24000, loss: 0.162

epoch: 2, count: 30000, loss: 0.172

epoch: 2, count: 36000, loss: 0.160

epoch: 2, count: 42000, loss: 0.163

epoch: 2, count: 48000, loss: 0.153

epoch: 2, count: 54000, loss: 0.154

epoch: 2, count: 60000, loss: 0.150

epoch: 3, count:  6000, loss: 0.118

epoch: 3, count: 12000, loss: 0.117

epoch: 3, count: 18000, loss: 0.102

epoch: 3, count: 24000, loss: 0.114

epoch: 3, count: 30000, loss: 0.126

epoch: 3, count: 36000, loss: 0.120

epoch: 3, count: 42000, loss: 0.106

epoch: 3, count: 48000, loss: 0.114

epoch: 3, count: 54000, loss: 0.113

epoch: 3, count: 60000, loss: 0.105

Accuracy of the network on the 10000 test images: 97 %

多次实验测试结果显示自定义神经网络识别准确率保持在97%左右，略优于LeNet神经网络。

（2）、LeNet与自定义神经网络的融合

将自定义神经网络与LeNet进行结合，首先利用自定义神经网络进行卷积，再利用传统LeNet神经网络进行卷积，其流程为：

其结果为：

epoch: 1, count:  6000, loss: 6.271

epoch: 1, count: 12000, loss: 0.489

epoch: 1, count: 18000, loss: 0.384

epoch: 1, count: 24000, loss: 0.327

epoch: 1, count: 30000, loss: 0.323

epoch: 1, count: 36000, loss: 0.293

epoch: 1, count: 42000, loss: 0.285

epoch: 1, count: 48000, loss: 0.245

epoch: 1, count: 54000, loss: 0.248

epoch: 1, count: 60000, loss: 0.239

epoch: 2, count:  6000, loss: 0.209

epoch: 2, count: 12000, loss: 0.205

epoch: 2, count: 18000, loss: 0.207

epoch: 2, count: 24000, loss: 0.182

epoch: 2, count: 30000, loss: 0.181

epoch: 2, count: 36000, loss: 0.163

epoch: 2, count: 42000, loss: 0.182

epoch: 2, count: 48000, loss: 0.171

epoch: 2, count: 54000, loss: 0.154

epoch: 2, count: 60000, loss: 0.171

epoch: 3, count:  6000, loss: 0.139

epoch: 3, count: 12000, loss: 0.138

epoch: 3, count: 18000, loss: 0.144

epoch: 3, count: 24000, loss: 0.145

epoch: 3, count: 30000, loss: 0.130

epoch: 3, count: 36000, loss: 0.135

epoch: 3, count: 42000, loss: 0.146

epoch: 3, count: 48000, loss: 0.139

epoch: 3, count: 54000, loss: 0.141

epoch: 3, count: 60000, loss: 0.135

Accuracy of the network on the 10000 test images: 96 %

经多次测试可知融合之后的准确率也保持在96%及以上，效果良好。

四、实验结论

通过本次实验多次测试以及观察可以发现自定义神经网络对手写字体识别的准确率保持在97%左右，其准确率要略优于LeNet的识别准确率，可以证明通过自定义的神经网络提取的全局特征与局部特征是有效且可利用的特征，对识别准确率有着较为突出的影响。同时通过自定义神经网络与LeNet神经网络的结合可以看到识别的准确率也保持在一个相对良好的范围内，此次实验可以证明自己设计的高斯卷积核与轮廓提取卷积核对特征提取有着极为突出的作用。