

数字图像处理作业报告七

学号：71194506019 姓名：姜志刚 专业：计算机技术

题目

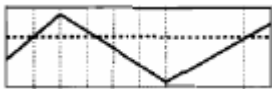
对一副二值图像进行膨胀、腐蚀、开、闭操作。

待处理图像：

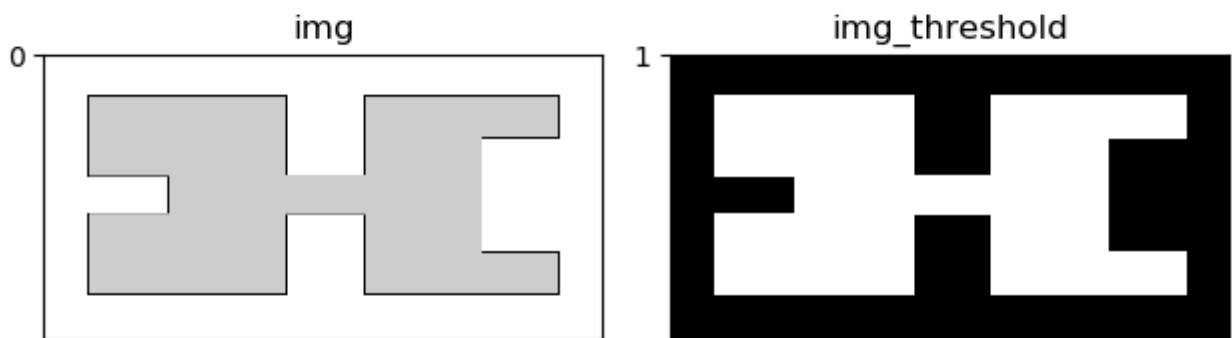
对图片进行二值化：

```
ret, img_threshold = cv2.threshold(img, 240, 255, cv2.THRESH_BINARY_INV)
```

在240~255之间取0，否则取255



破折线为将被阈值化的值；虚线为阈值



腐蚀

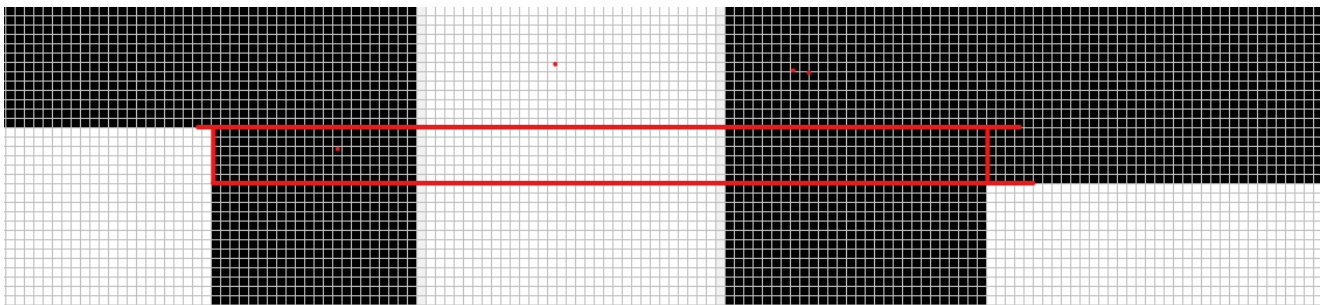
使用opencv, 结构元为50*50的方形, 进行腐蚀

```
kernel = np.ones((50, 50), np.uint8)
erosion = cv2.erode(img_threshold, kernel)
```

结果



可以看到腐蚀了50



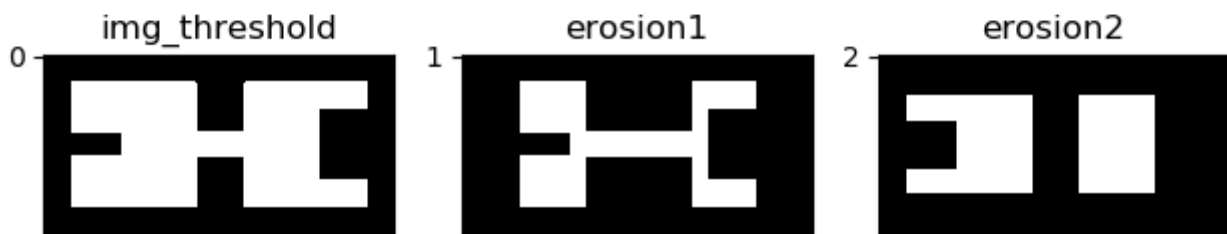
100*100的结构元时：



产生断裂

或者使用长方形的结构元进行腐蚀，kernel1 侧重水平方形，其结构元长大于宽
kernel12，侧重垂直反向，其结构元长小于宽

```
kernel1 = np.ones((1, 200), np.uint8)
kernel2 = np.ones((90, 1), np.uint8)
erosion = cv2.erode(img_threshold, kernel1)
erosion2 = cv2.erode(img_threshold, kernel2)
```



不同形状的结构元素，收缩或膨胀后的结果形状不同

膨胀

```
kernel1 = np.ones((1, 200), np.uint8)
kernel2 = np.ones((90, 1), np.uint8)
dilate1 = cv2.dilate(erosion, kernel1)
dilate2 = cv2.dilate(erosion, kernel2)
```

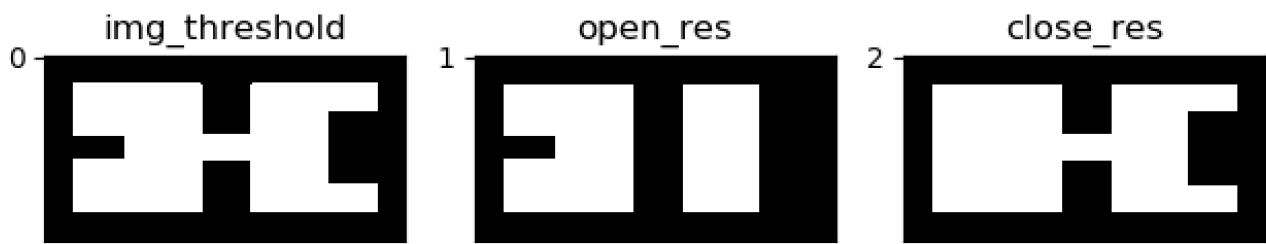


开运算与闭运算

用100*100的方形结构元，进行如下操作：

```
def open_(img, kernel):  
    erosion = cv2.erode(img, kernel)  
    dilate = cv2.dilate(erosion, kernel)  
    return dilate
```

```
def close_(img, kernel):  
    dilate = cv2.dilate(img, kernel)  
    erosion = cv2.erode(dilate, kernel)  
    return erosion
```

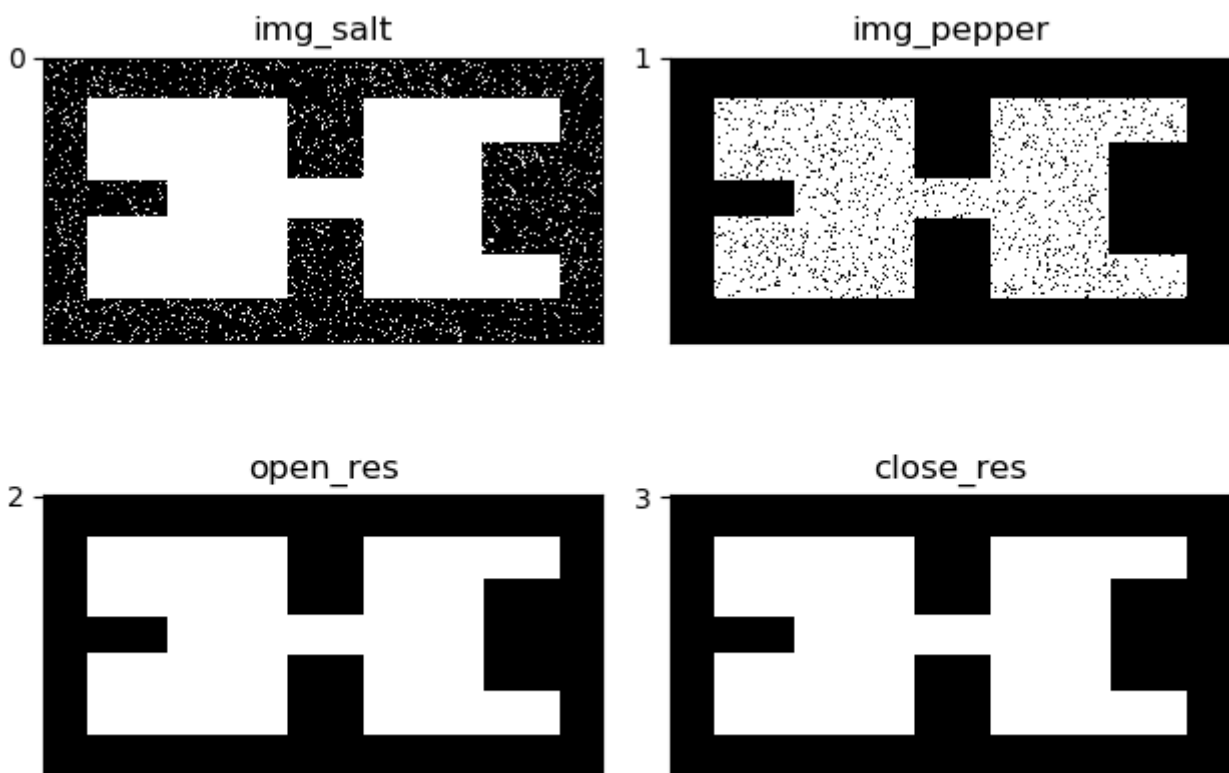


开运算先进行腐蚀，这回导致凸起（如右上角和左下角）和狭长（中心部分）被腐蚀掉，然后再膨胀回“原来大小”

闭运算先进行膨胀，这回导致凹陷（右边中间）被膨胀填充，然后再腐蚀回“原来大小”

形态学去噪

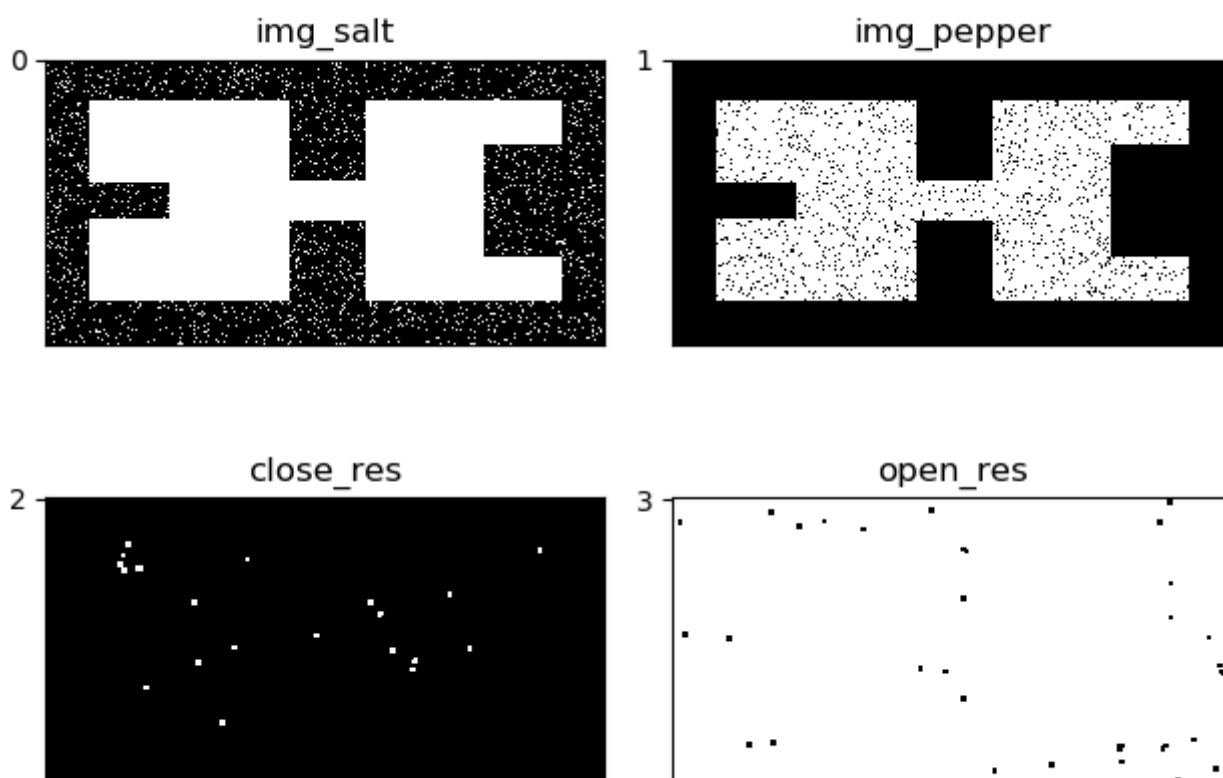
有两副分别是盐噪声和胡椒噪声的图，用 10×10 的核（其实可以更小）



对盐噪声图像进行开运算，先腐蚀会导致盐噪声被腐蚀掉，如果对其进行闭运算，中则整个图像会被膨胀，因为噪声之间靠的很近，膨胀后有可能复原不了

对胡椒噪声图像进行闭运算，会膨胀掉胡椒噪声，如归对其开运算，整个图像可能会被腐蚀，与盐噪声同理

错误的运用：



参考

[python-opencv 图像二值化，自适应阈值处理](#)