



周围的人都比你厉害，你才会慢慢变强

公告

昵称： 山上有风景  
园龄： 1年11个月  
粉丝： 170  
关注： 19  
[+加关注](#)

<	2019年12月						>
日	一	二	三	四	五	六	
1	2	3	4	5	6	7	
8	9	10	11	12	13	14	
15	16	17	18	19	20	21	
22	23	24	25	26	27	28	
29	30	31	1	2	3	4	
5	6	7	8	9	10	11	

搜索

找找看

常用链接

[我的随笔](#)  
[我的评论](#)  
[我的参与](#)  
[最新评论](#)  
[我的标签](#)

我的标签

[STL\(18\)](#)  
[SDN\(15\)](#)  
[ThinkPHP3.2\(1\)](#)

积分与排名

积分 - 211835  
排名 - 1998

随笔分类

[C/C++\(74\)](#)  
[Html\(2\)](#)  
[Java\(33\)](#)  
[Javascript\(19\)](#)  
[OpenCV\(29\)](#)  
[PHP\(2\)](#)  
[Python\(155\)](#)  
[STL泛型编程\(18\)](#)  
[单片机笔记（复习用）\(3\)](#)  
[计算机网络\(32\)](#)  
[其他知识\(14\)](#)

# OpenCV---开闭操作

## 目录

- 一：开操作（先腐蚀后膨胀）  
特点：消除噪点，去除小的干扰块，而不影响原来的图像
- 二：闭操作（先膨胀后腐蚀）  
特点：可以填充闭合区域
- 三：利用开操作完成的任务  
(一) 提取水平垂直线  
原理：  
(二) 消除干扰线  
(三) 提取满足要求的形状

## 开操作（Open）

- ▶ 图像形态学的重要操作之一，基于膨胀与腐蚀操作组合形成的。
- ▶ 主要是应用在二值图像分析中，灰度图像亦可。
- ▶ 开操作=腐蚀+膨胀， 输入图像 + 结构元素

## 闭操作(Close)

- ▶ 图像形态学的重要操作之一，基于膨胀与腐蚀操作组合形成的。
- ▶ 主要是应用在二值图像分析中，灰度图像亦可。
- ▶ 闭操作=膨胀+腐蚀， 输入图像 + 结构元素

10

关注 | 顶部 | 评论

- 设计模式(27)
- 数据结构(57)
- 数据库(8)
- 算法习题(43)
- 算法训练营
- 随笔所想(4)
- 图形界面编程
- 正则表达式(2)
- 转载推文(4)
- 随笔档案

- 2019年11月(5)
- 2019年10月(32)
- 2019年9月(21)
- 2019年7月(10)
- 2019年5月(8)
- 2019年4月(25)
- 2019年3月(8)
- 2019年2月(1)
- 2019年1月(12)
- 2018年12月(19)
- 2018年9月(5)
- 2018年8月(95)
- 2018年7月(78)
- 2018年6月(26)
- 2018年5月(17)
- 2018年4月(22)
- 2018年3月(111)

- 最新评论
1. Re:python---websocket的使用  
网上的方法都不行，  
换成gbk会报如下的错：  
IndexError: index out of range  
--缘分天空0320
2. Re:python---websocket的使用  
@ 缘分天空0320太久没用，忘了。  
应该是字符编码问题吧。这类问题网上应该可以很容易找到方法解决。你看看Python的默认编码和代码是不是一致。一般就是gbk和utf8之间出错...  
--山上有风景
3. Re:python---websocket的使用  
您好，用了您的代码，报如下错误，麻烦问下如何解决呢？Traceback (most recent call last): File "server3.py", line 101, in <module...  
--缘分天空0320
4. Re:数据结构（六）查找---线性索引查找  
请问最后倒序排序的那个代码怎么实现的？可以发一下吗？  
--Viki-
5. Re:SDN实验---Ryu的源码分析  
@ 山上有风景谢谢！ ...  
--iRoy\_33

- 阅读排行榜
1. python---websocket的使用(17253)

# 开闭操作作用

- 去除小的干扰块-开操作
- 填充闭合区域-闭操作
- 水平或者垂直线提取

## 一：开操作（先腐蚀后膨胀）

特点：消除噪点，去除小的干扰块，而不影响原来的图像



```
import cv2 as cv
import numpy as np

def camp(val1,val2):
    pv = val1 + val2
    if pv > 255:
        return 255
    if pv < 0:
        return 0
    return pv

def open_demo(image):
    gray = cv.cvtColor(image,cv.COLOR_BGR2GRAY)
    for i in range(1000): #为灰度图像加一些噪点
        h = np.random.random_integers(0,gray.shape[0]-1)
        w = np.random.random_integers(0, gray.shape[1]-1)
        val = np.random.random_integers(0, 255)
        gray[h,w] = camp(gray[h,w],val)

    ret,binary = cv.threshold(gray,0,255,cv.THRESH_BINARY|cv.THRESH_OTSU) #生成二值化图像
    cv.imshow("binary",binary)
    kernel = cv.getStructuringElement(cv.MORPH_RECT,(3,3))
    binary = cv.morphologyEx(binary,cv.MORPH_OPEN,kernel) #开操作，先腐蚀后膨胀，会消除一些为1的白色噪点
    cv.imshow("open_demo",binary)

src = cv.imread("./o.png") #读取图片
cv.namedWindow("input image",cv.WINDOW_AUTOSIZE) #创建GUI窗口,形式为自适应
cv.imshow("input image",src) #通过名字将图像和窗口联系

open_demo(src)

cv.waitKey(0) #等待用户操作，里面等待参数是毫秒，我们填写0，代表是永远，等待用户操作
cv.destroyAllWindows() #销毁所有窗口
```



10

关注 | 顶部 | 评论

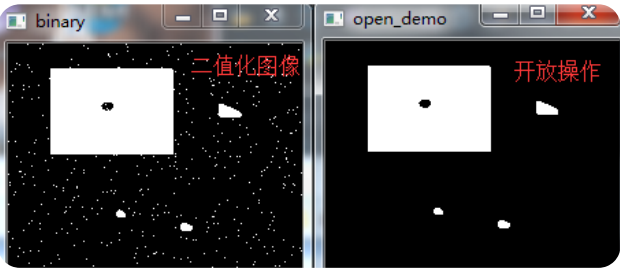
- 2. OpenCV---图像二值化(12513)
- 3. OpenCV---模板匹配matchTemplate(11534)
- 4. OpenCV---直线检测(8208)
- 5. python---基础知识回顾 (九) 图形用户界面-----WxPython(7986)

评论排行榜

- 1. python---基础知识回顾 (九) 图形用户界面-----Tkinter(4)
- 2. python---websocket的使用(3)
- 3. SDN实验---Ryu的源码分析(3)
- 4. 数据结构 (三) 串---KMP模式匹配算法之获取next数组(2)
- 5. 数据结构 (四) 树---树的存储结构(1)

推荐排行榜

- 1. 数据结构 (七) 排序---堆排序(11)
- 2. python---aiohttp的使用(6)
- 3. python---websocket的使用(4)
- 4. python---基础知识回顾 (九) 图形用户界面-----WxPython(3)
- 5. Python图像处理库PIL中图像格式转换(3)

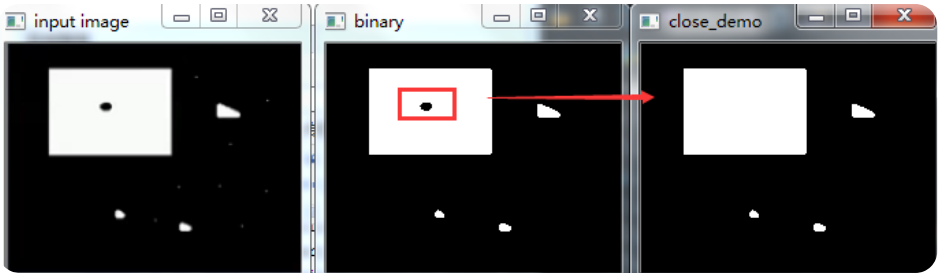


二：闭操作（先膨胀后腐蚀）

特点：可以填充闭合区域

```
def close_demo(image):
    gray = cv.cvtColor(image,cv.COLOR_BGR2GRAY)
    ret,binary = cv.threshold(gray,0,255,cv.THRESH_BINARY|cv.THRESH_OTSU)
    cv.imshow("binary",binary)
    kernel = cv.getStructuringElement(cv.MORPH_RECT,(15,15))
    binary = cv.morphologyEx(binary,cv.MORPH_CLOSE,kernel)
    cv.imshow("close_demo",binary)

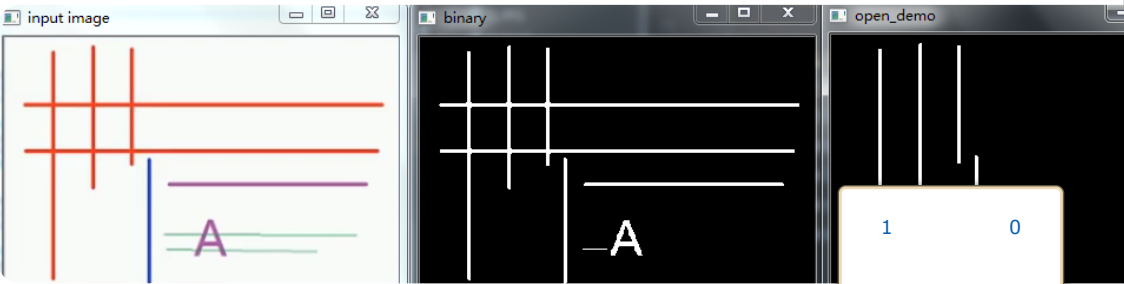
src = cv.imread("./o.png") #读取图片
cv.namedWindow("input image",cv.WINDOW_AUTOSIZE) #创建GUI窗口,形式为自适应
cv.imshow("input image",src) #通过名字将图像和窗口联系
```



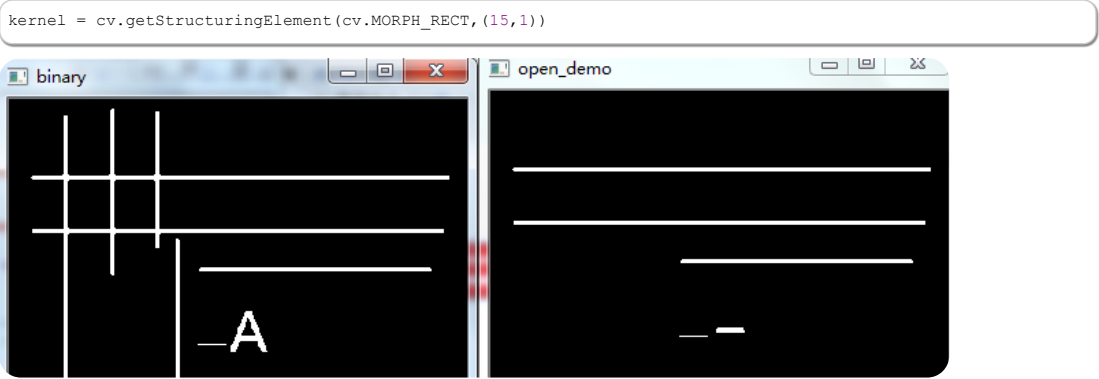
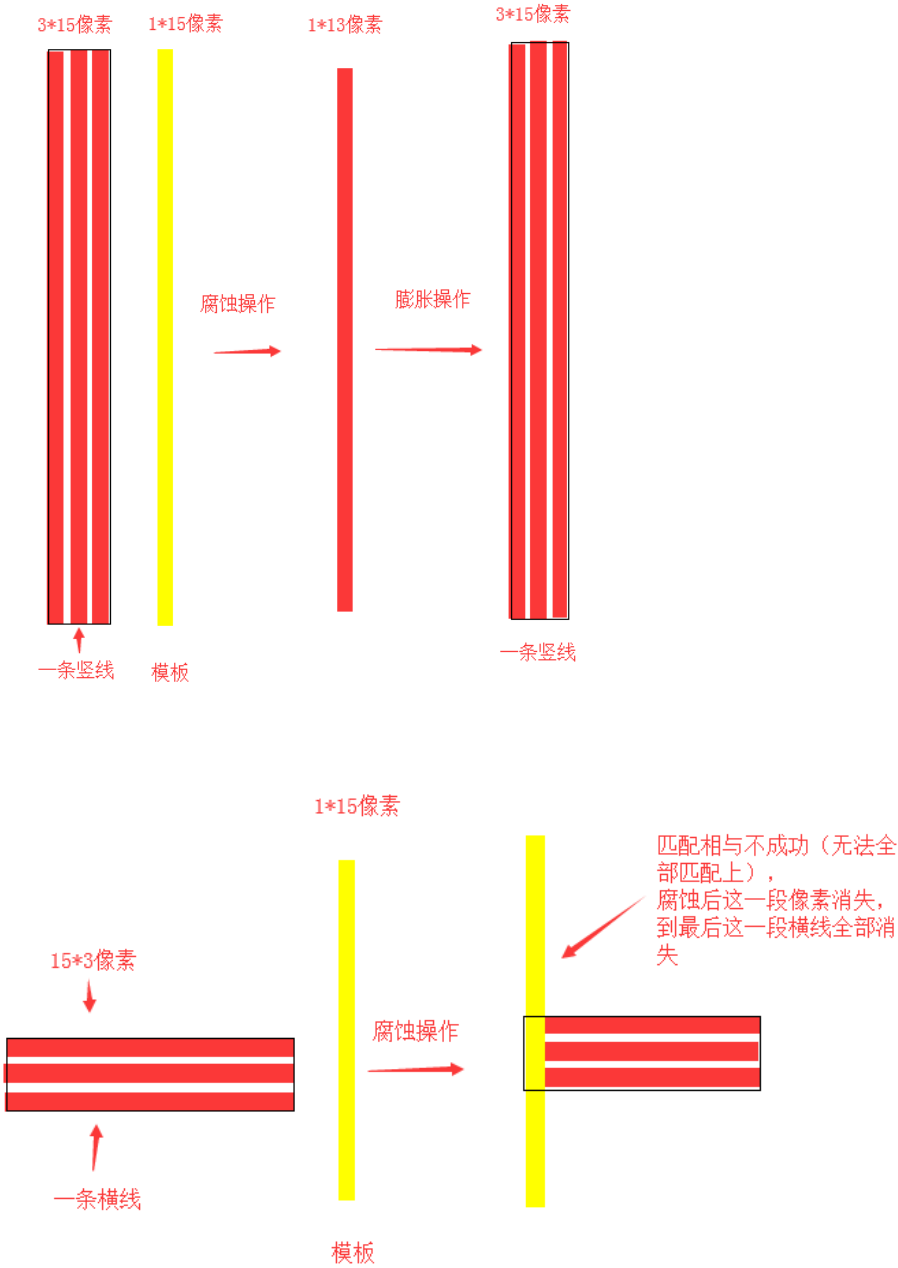
三：利用开操作完成的任务

(一) 提取水平垂直线

```
def open_demo(image):
    gray = cv.cvtColor(image,cv.COLOR_BGR2GRAY)
    ret,binary = cv.threshold(gray,0,255,cv.THRESH_BINARY_INV|cv.THRESH_OTSU)
    cv.imshow("binary",binary)
    kernel = cv.getStructuringElement(cv.MORPH_RECT,(1,15)) #修改内核为(15,1)可以提起横线
    binary = cv.morphologyEx(binary,cv.MORPH_OPEN,kernel)
    cv.imshow("open_demo",binary)
```



原理：



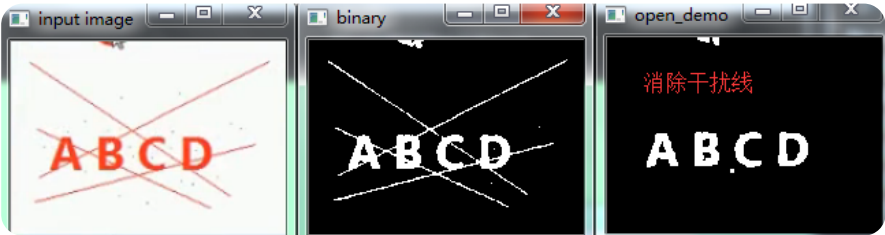
(二) 消除干扰线

```
def open_demo(image):  
    gray = cv.cvtColor(image,cv.COLOR_BGR2GRAY)  
    ret,binary = cv.threshold(gray,0,255,cv.THRESH_BINARY_INV|cv.THRESH_OTSU)
```

10

关注 | 顶部 | 评论

```
cv.imshow("binary",binary)
kernel = cv.getStructuringElement(cv.MORPH_RECT,(3,3)) #因为干扰线很细，小于我们想要的字母，先腐蚀后膨胀对字母无
影响，但是对于细线在腐蚀的时候就处理掉了
binary = cv.morphologyEx(binary,cv.MORPH_OPEN,kernel)
cv.imshow("open_demo",binary)
```



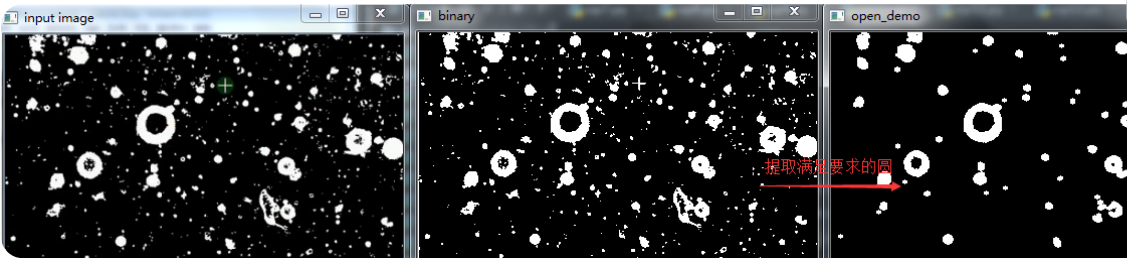
(三) 提取满足要求的形状

getStructuringElement我们设置的内核形状为矩形，是可以设置其他形状

矩形: MORPH\_RECT;  
交叉形: MORPH\_CROSS;  
椭圆形: MORPH\_ELLIPSE;

参考: OpenCV---膨胀与腐蚀

```
def open_demo(image):
    gray = cv.cvtColor(image,cv.COLOR_BGR2GRAY)
    ret,binary = cv.threshold(gray,0,255,cv.THRESH_BINARY|cv.THRESH_OTSU)
    cv.imshow("binary",binary)
    kernel = cv.getStructuringElement(cv.MORPH_ELLIPSE,(5,5))
    binary = cv.morphologyEx(binary,cv.MORPH_OPEN,kernel)
    cv.imshow("open_demo",binary)
```



作者: 山上有风景  
欢迎任何形式的转载，但请务必注明出处。  
限于本人水平，如果文章和代码有表述不当之处，还请不吝赐教。

分类: OpenCV

好文要顶

关注我

收藏该文

山上有风景  
关注 - 19  
粉丝 - 170  
+加关注

« 上一篇: OpenCV---膨胀与腐蚀  
» 下一篇: OpenCV---其他形态学操作

10

关注 | 顶部 | 评论

posted @ 2018-07-07 16:55 山上有风景 阅读(362) 评论(0) 编辑 收藏

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问](#) 网站首页。

Copyright © 2019 山上有风景

Powered by .NET Core 3.1.0 on Linux

1 0

[关注](#) | [顶部](#) | [评论](#)