

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

找我看

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---模板匹配matchTemplate

### 目录

代码实现：  
补充：

**作用有局限性，必须在指定的环境下，才能匹配成功，是受到很多因素的影响，所以有一定的适应性**

模板匹配是一种最原始、最基本的模式识别方法，研究某一特定对象物的图案位于图像的什么地方，进而识别对象物，这就是一个匹配问题。它是图像处理中最基本、最常用的匹配方法。

模板匹配具有自身的局限性，主要表现在它只能进行平行移动，若原图像中的匹配目标发生旋转或大小变化，该算法无效。

模板匹配就是在整个图像区域发现与给定子图像匹配的小块区域

模板匹配

工作原理：在待检测图像上，**从左到右，从上向下**计算模板图像与重叠子图像的匹配度，**匹配程度越大，两者相同的可能性越大。**

### 代码实现：

```
import cv2 as cv
import numpy as np

def template_demo():
    tpl = cv.imread("./temp.png")
    target = cv.imread("./1.png")
    cv.imshow("template image",tpl)
    cv.imshow("target image",target)
    methods = [cv.TM_SQDIFF_NORMED,cv.TM_CCORR_NORMED,cv.TM_CCOEFF_NORMED] #各种匹配方法
    th,tw = tpl.shape[:2] #获取模板图像的高宽
```

20

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，麻烦问一下如何解决呢？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)


```
for md in methods:
    result = cv.matchTemplate(target,tpl,md)
    # result是我们各种算法下匹配后的图像
    # cv.imshow("%s"%md,result)
    #获取的是每种公式中计算出来的值，每个像素点都对应一个值
    min_val,max_val,min_loc,max_loc = cv.minMaxLoc(result)
    if md == cv.TM_SQDIFF_NORMED:
        tl = min_loc      #tl是左上角点
    else:
        tl = max_loc
    br = (tl[0]+tw,tl[1]+th)      #右下点
    cv.rectangle(target,tl,br, (0,0,255),2)      #画矩形
    cv.imshow("match-%s"%md,target)

src = cv.imread("./1.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
template_demo()
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
```

补充：

1.几种常见的模板匹配算法

TM_SQDIFF	$R(x,y) = \sum_{x',y'} (T(x',y') - I(x+x',y+y'))^2$
TM_SQDIFF_NORMED	$R(x,y) = \frac{\sum_{x',y'} (T(x',y') - I(x+x',y+y'))^2}{\sqrt{\sum_{x',y'} T(x',y')^2 \cdot \sum_{x',y'} I(x+x',y+y')^2}}$
TM_CCORR	$R(x,y) = \sum_{x',y'} (T(x',y') \cdot I(x+x',y+y'))$
TM_CCORR_NORMED	$R(x,y) = \frac{\sum_{x',y'} (T(x',y') \cdot I(x+x',y+y'))}{\sqrt{\sum_{x',y'} T(x',y')^2 \cdot \sum_{x',y'} I(x+x',y+y')^2}}$
TM_CCOEFF	$R(x,y) = \sum_{x',y'} (T'(x',y') \cdot I'(x+x',y+y'))$ <p>where</p> $T'(x',y') = T(x',y') - 1/(w \cdot h) \cdot \sum_{x'',y''} T(x'',y'')$ $I'(x+x',y+y') = I(x+x',y+y') - 1/(w \cdot h) \cdot \sum_{x'',y''} I(x+x'',y+y'')$
TM_CCOEFF_NORMED	$R(x,y) = \frac{\sum_{x',y'} (T'(x',y') \cdot I'(x+x',y+y'))}{\sqrt{\sum_{x',y'} T'(x',y')^2 \cdot \sum_{x',y'} I'(x+x',y+y')^2}}$



①**TM\_SQDIFF**是平方差匹配；**TM\_SQDIFF\_NORMED**是标准平方差匹配。利用平方差来进行匹配，最好匹配为0。匹配越差，匹配值越大。

②**TM\_CCORR**是相关性匹配；**TM\_CCORR\_NORMED**是标准相关性匹配。采用模板和图像间的乘法操作，数越大表示匹配程度较高，0表示最坏的匹配效果。

③**TM\_CCOEFF**是相关系数匹配；**TM\_CCOEFF\_NORMED**是标准相关系数匹配。将模板对其均值的相对值与图像对其均值的相关值进行匹配，1表示完美匹配，-1表示糟糕的匹配，0表示没有任何相关性（随机序列）。

总结：随着从简单的测量（平方差）到更复杂的测量（相关系数），我们可获得越来越准确的匹配（同时也意味着越来越

2

0

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)

相关性是越接近1越大越好

平方差是越小越好

所以TM\_SQDIFF在使用时和其他的是有所区别的

2.result = cv.matchTemplate(target,tpl,md)

opencv的目标匹配函数为matchTemplate，函数原型为: matchTemplate(image, templ, method[, result[, mask]]) -> result

image参数表示待搜索源图像，必须是8位整数或32位浮点。

templ参数表示模板图像，必须不大于源图像并具有相同的数据类型。

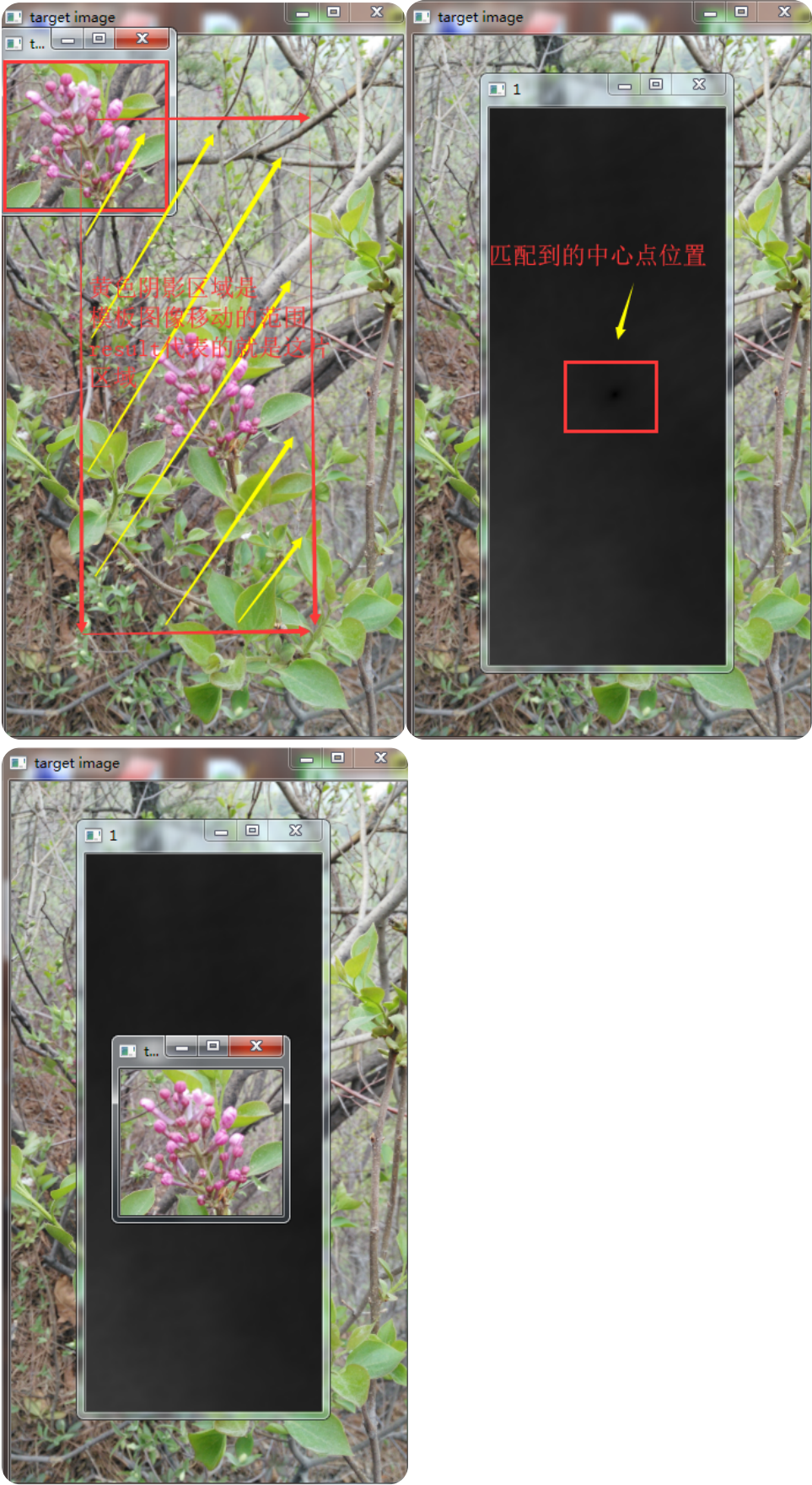
method参数表示计算匹配程度的方法。

result参数表示匹配结果图像，必须是单通道32位浮点。如果image的尺寸为W x H, templ的尺寸为w x h, 则result的尺寸为 (W-w+1) x (H-h+1)。

其中result是模板图像去匹配的区域位置图像

20

关注 | 顶部 | 评论



3.min\_val,max\_val,min\_loc,max\_loc = cv.minMaxLoc(result)



opencv的函数minMaxLoc: 在给定的矩阵中寻找最大和最小值,并给出它们的位置。 该功能不适用于多通道阵列。 最小或最大元素,要先将阵列重新解释为单通道。

20

关注 | 顶部 | 评论

函数minMaxLoc原型为: minMaxLoc(src[, mask]) -> minVal, maxVal, minLoc, maxLoc  
 src参数表示输入单通道图像。  
 mask参数表示用于选择子数组的可选掩码。  
 minVal参数表示返回的最小值, 如果不需要, 则使用NULL。  
 maxVal参数表示返回的最大值, 如果不需要, 则使用NULL。  
 minLoc参数表示返回的最小位置的指针 (在2D情况下); 如果不需要, 则使用NULL。  
 maxLoc参数表示返回的最大位置的指针 (在2D情况下); 如果不需要, 则使用NULL。



### 结合每种匹配算法, 我们看看获取的数值



```
result = cv.matchTemplate(target,tpl,md)
#获取的是每种公式中计算出来的值, 每个像素点都对应一个值
min_val,max_val,min_loc,max_loc = cv.minMaxLoc(result)
print("-----%s-----"%md)
print("min_val",min_val)
print("max_val",max_val)
print("min_loc",min_loc)
print("max_loc",max_loc)
print("-----%s-----" % md)
```



```
-----1----- #TM_SQDIFF_NORMED标准平方差匹配
min_val 0.0 #标准差是越小为0代表匹配上了
max_val 0.22279763221740723
min_loc (108, 248)
max_loc (3, 480)
-----3----- #TM_CCORR_NORMED标准相关性匹配
min_val 0.9228140115737915
max_val 1.0 #相关性是越接近1代表匹配上了
min_loc (9, 378)
max_loc (108, 248)
-----5----- #TM_CCOEFF_NORMED标准相关性系数匹配
min_val -0.10706906020641327
max_val 1.0 #相关性越接近1越好
min_loc (186, 248)
max_loc (108, 248)
-----5-----
```



### 查看min\_loc和max\_loc关系

```
cv.line(target,min_loc,max_loc,(0,255,255),2)
```

2

0

[关注](#) | [顶部](#) | [评论](#)





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---直方图反向投影  
» 下一篇： OpenCV---图像二值化

posted @ 2018-07-06 09:50 山上有风景 阅读(11534) 评论(1) 编辑 收藏

评论列表

#1楼 2018-12-14 18:45 walkingSun



怎么判断没有匹配到结果

支持(1) 反对(0)

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。

2

0

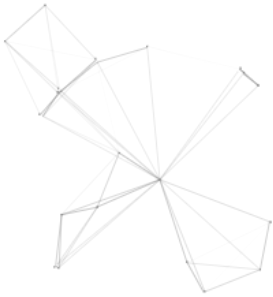
关注

顶部

评论

Copyright © 2019 山上有风景

Powered by .NET Core 3.1.0 on Linux



2

0

[关注](#) | [顶部](#) | [评论](#)