OpenCV技巧 | 二值图孔洞填充方法与实现(附源码)

发布于 2021-09-15 15:31:08

5.1K0

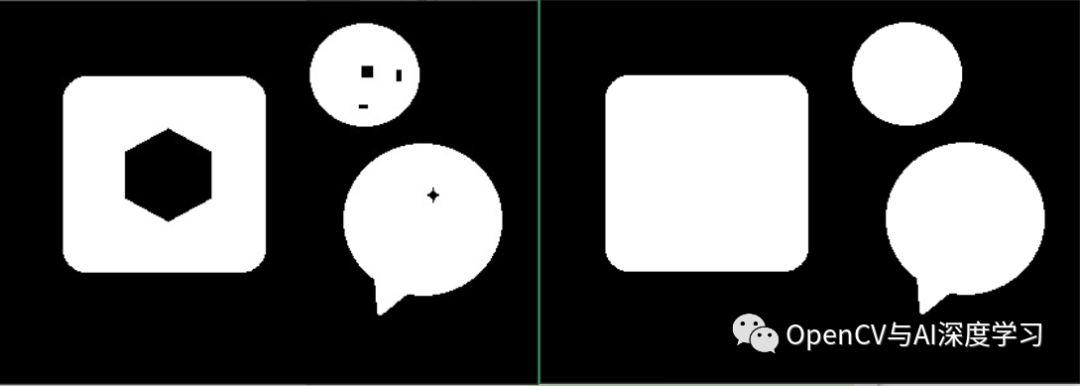
举报

**导读**

本文主要介绍使用OpenCV对二值图做孔洞填充的方法与实现。

**背景介绍**

为什么要做孔洞填充？因为在部分情况下，二值图内部的孔洞和外部轮廓是一个整体，填充孔洞可以方便后续处理，减少干扰。



OpenCV孔洞填充的常用方法有2种：

① 形态学闭运算。闭运算是先膨胀后腐蚀操作，如上图，先膨胀白色高亮区域增加，孔洞会被填充。但使用闭运算针对不同的图片不能准确知道需要的核大小是多少，所以并不能通用；

② 轮廓绘制方法。通过轮廓绘制drawContours函数设置绘制线宽为-1即可填充绘制。但查找轮廓一般是在二值图处理之后才会使用，这样会造成重复操作。

尽量在二值图时完成孔洞填充，不影响后续操作顺序才是最终目标！

**实现步骤与代码**

实现步骤：

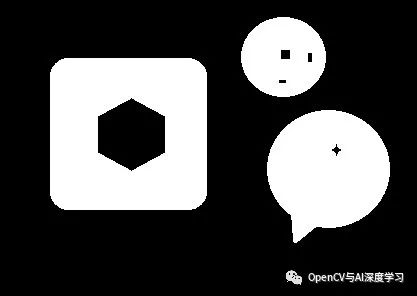
① 通过二值化或其他方法得到二值图；

② 使用floodFill从(0,0)点开始执行漫水填充算法；

③ 漫水填充结果图取反；

④ 取反后的结果与二值图求并集。

实现代码与逐步演示：



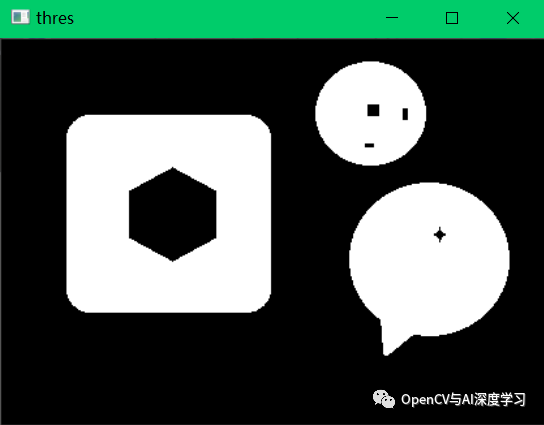
① 得到二值图：

gray = cv2.cvtColor(img, cv2.COLOR\_BGR2GRAY)

\_, thres = cv2.threshold(gray, 120, 255, cv2.THRESH\_BINARY)

cv2.imshow("thres", thres)

复制



② 执行漫水填充算法：

#执行漫水填充方法.

#注意图像大小比原图宽高都大2.

h, w = gray.shape[:2]

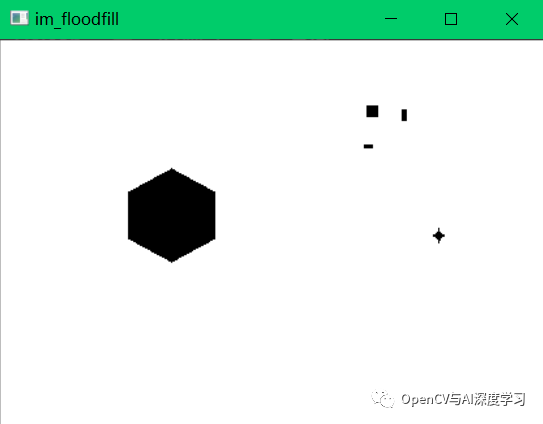
mask = np.zeros((h+2, w+2), np.uint8)

#漫水填充从(0, 0)点开始

cv2.floodFill(im\_floodfill, mask, (0,0), 255)

cv2.imshow("im\_floodfill", im\_floodfill)

复制



③ 漫水填充图像取反：

#反转漫水填充图像

im\_floodfill\_inv = cv2.bitwise\_not(im\_floodfill)

cv2.imshow("im\_floodfill\_inv", im\_floodfill\_inv)

复制



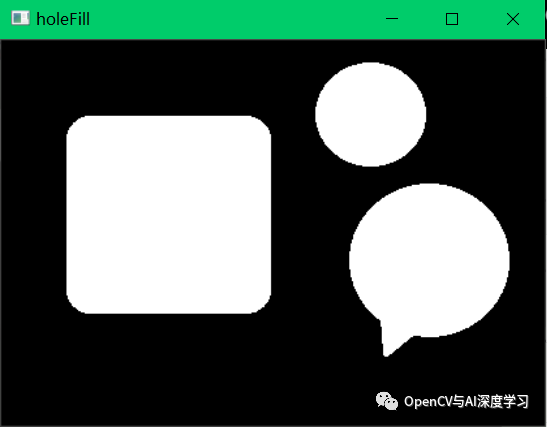
④ 取反图像与二值图求并集：

#将二值图与上一步图像求并集

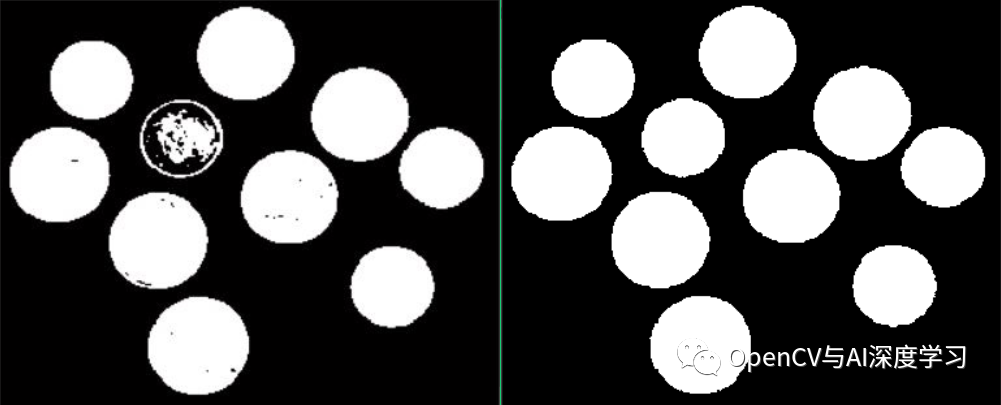
im\_out = thres | im\_floodfill\_inv

cv2.imshow("holeFill", im\_out

复制



**换张图片测试同样可以，此方法具有通用性，可自己封装成函数使用。**



参考链接：https://learnopencv.com/filling-holes-in-an-image-using-opencv-python-c/