

OpenCV技巧| 二值圖孔洞填充方法與實現(附源碼)

原創 Color Space OpenCV與AI深度學習 今天

收錄於話題

#OpenCV 35 #圖像處理 5



OpenCV與AI深度學習

致力於OpenCV、機器視覺、深度學習和人工智能相關內容的學習、交流、分享！

110篇原創內容



公眾號

點擊上方

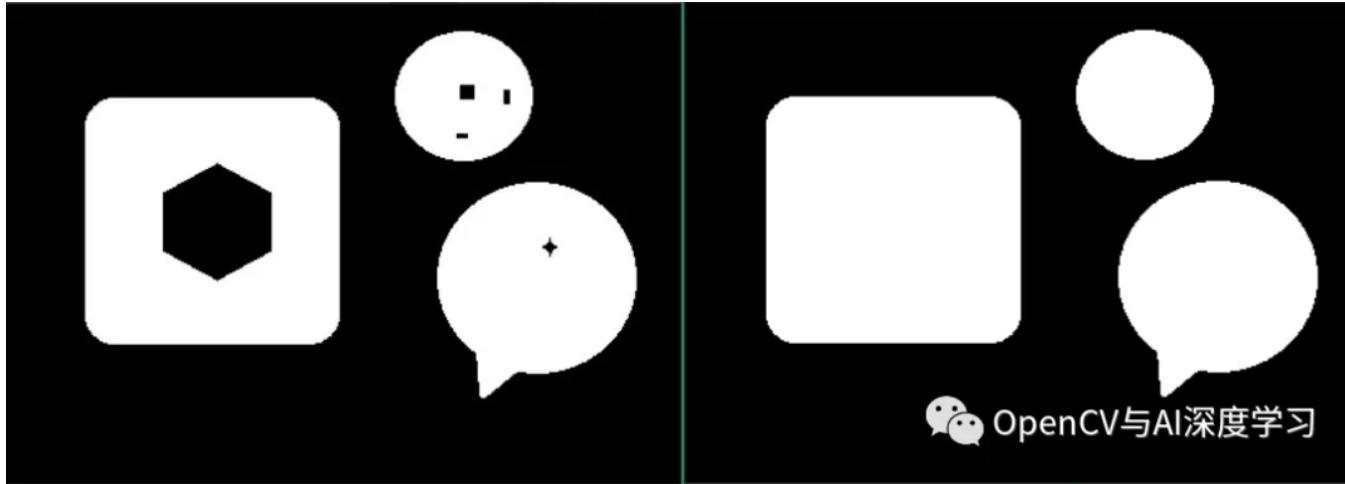
重磅乾貨 · 第一時間送達

導讀

本文主要介紹使用OpenCV對二值圖做孔洞填充的方法與實現

背景介紹

為什麼要做孔洞填充？因為在部分情況下，二值圖內部的孔洞和外部輪廓是一個整體，填充孔洞可以方便後續處理，減少干擾。



OpenCV孔洞填充的常用方法有2種：

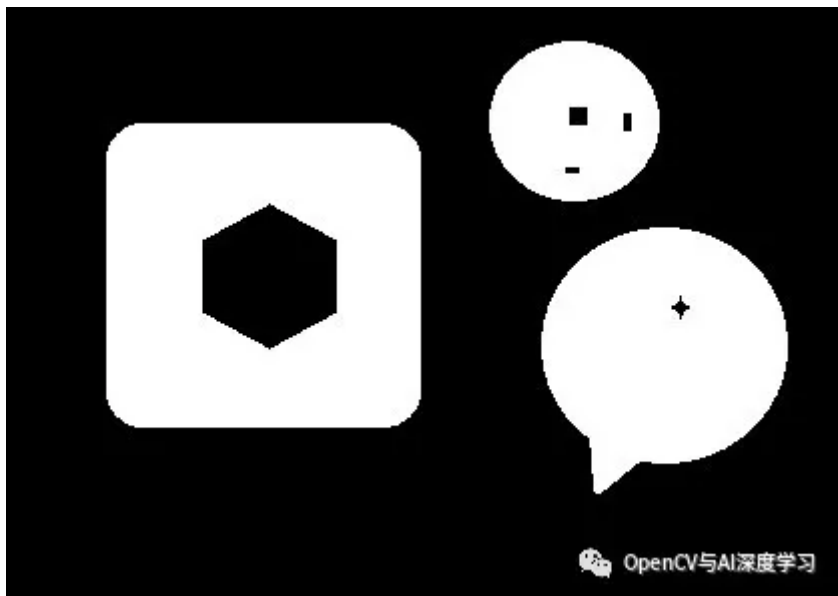
- ① 形態學閉運算。閉運算是先膨脹後腐蝕操作，如上圖，先膨脹白色高亮區域增加，孔洞會被填充。但使用閉運算針對不同的圖片不能準確知道需要的核大小是多少，所以並不能通用；
 - ② 輪廓繪製方法。通過輪廓繪製drawContours函數設置繪製線寬為-1即可填充繪製。但查找輪廓一般是在二值圖處理之後才會使用，這樣會造成重複操作。
- 盡量在二值圖時完成孔洞填充，不影響後續操作順序才是最終目標！

實現步驟與代碼

實現步驟：

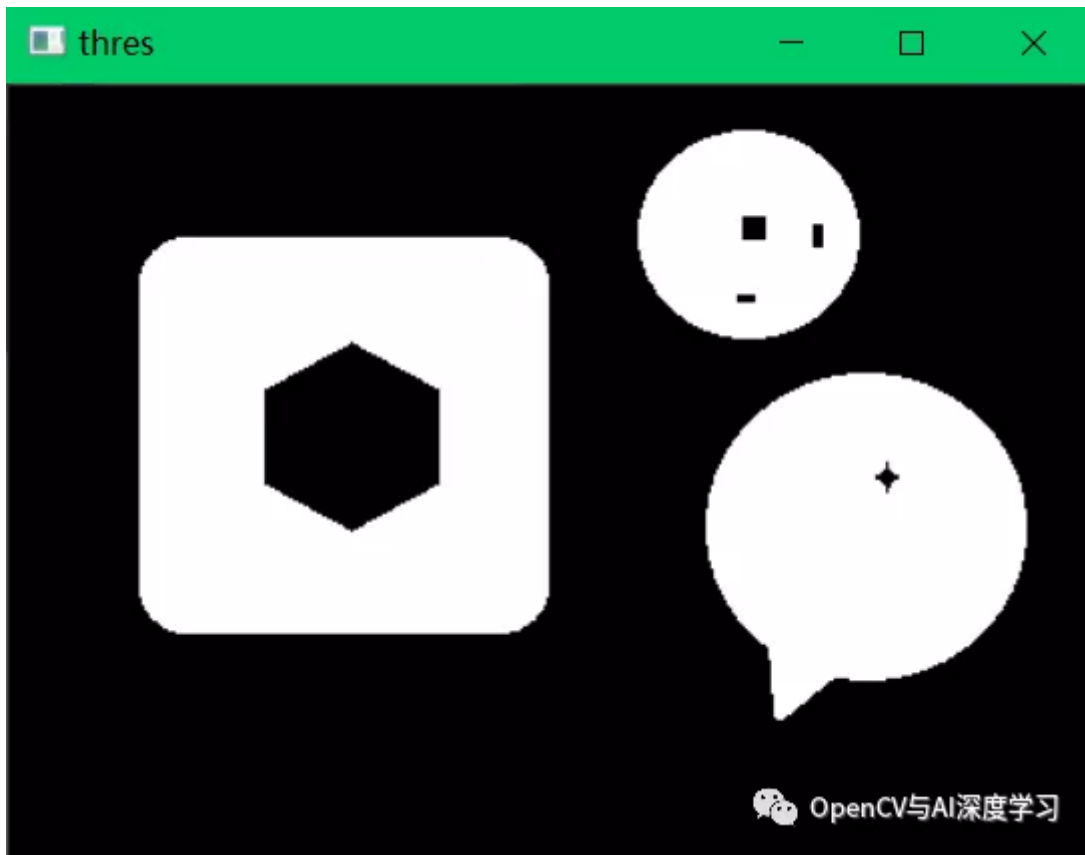
- ① 通過二值化或其他方法得到二值圖；
- ② 使用floodFill從(0,0)點開始執行漫水填充算法；
- ③ 漫水填充結果圖取反；
- ④ 取反後的結果與二值圖求並集。

實現代碼與逐步演示：



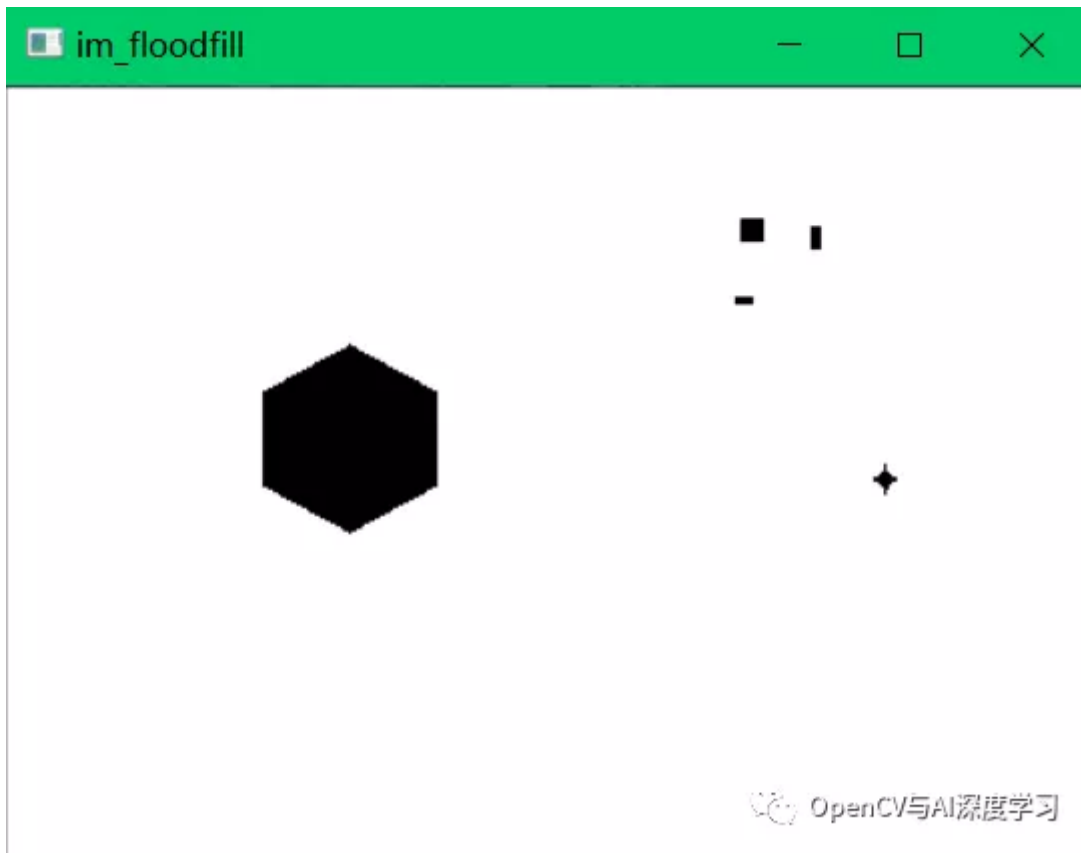
① 得到二值圖：

```
1 gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
2 _, thres = cv2.threshold(gray, 120, 255, cv2.THRESH_BINARY)
3 cv2.imshow("thres", thres)
```



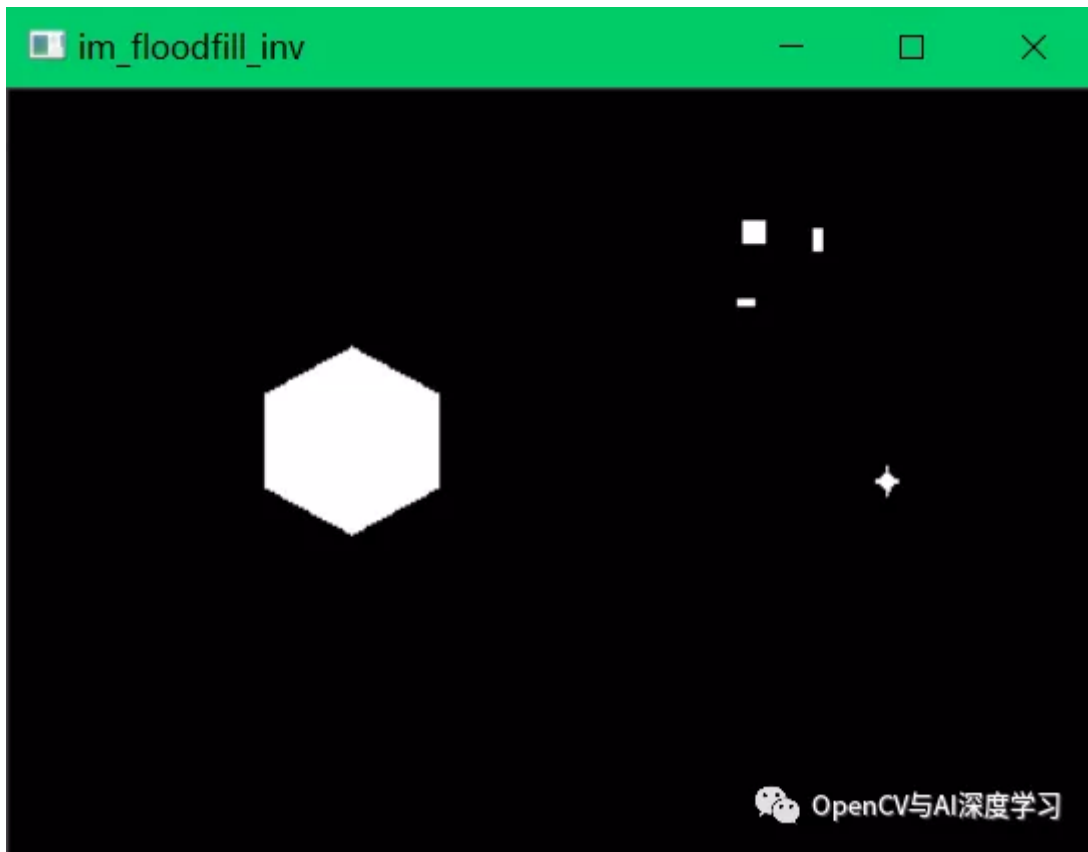
② 執行漫水填充算法

```
1 #执行漫水填充方法。
2 #注意图像大小比原图宽高都大2。
3 h, w = gray.shape[:2]
4 mask = np.zeros((h+2, w+2), np.uint8)
5
6 #漫水填充从(0, 0)点开始
7 cv2.floodFill(im_floodfill, mask, (0,0), 255)
8 cv2.imshow("im_floodfill", im_floodfill)
```



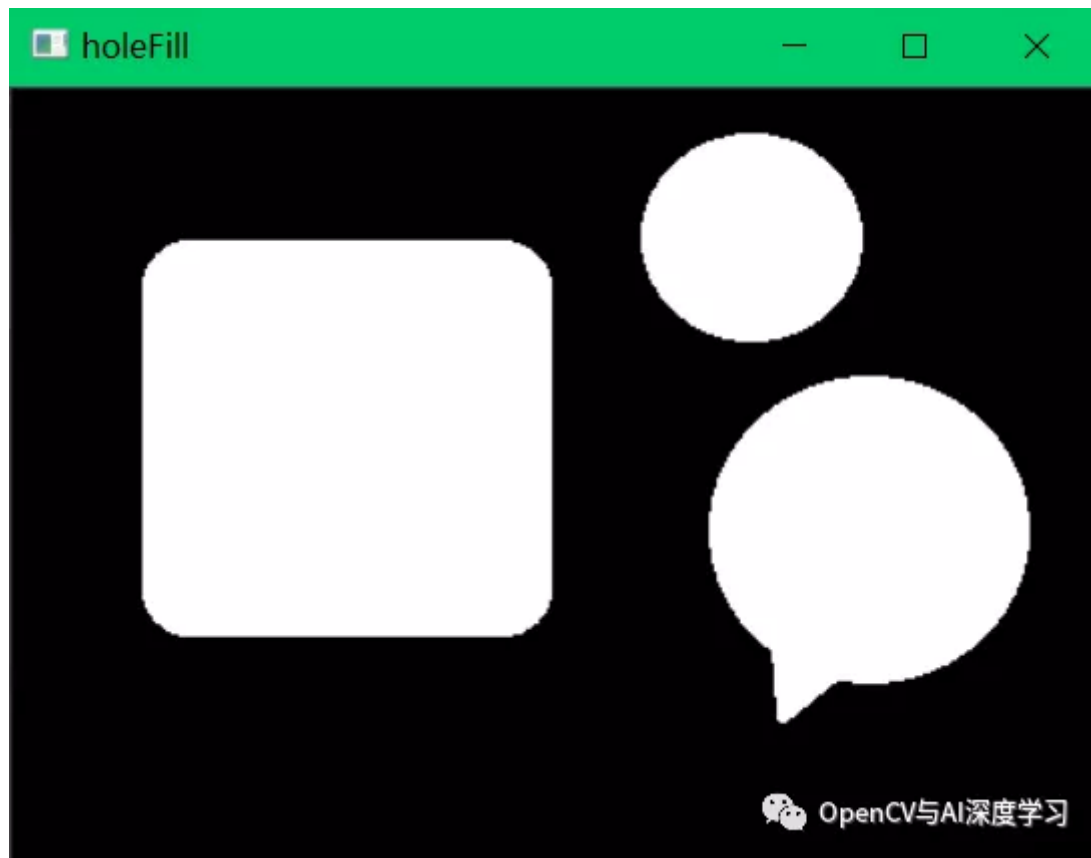
③ 漫水填充圖像取反

```
1 #反转漫水填充图像
2 im_floodfill_inv = cv2.bitwise_not(im_floodfill)
3 cv2.imshow("im_floodfill_inv", im_floodfill_inv)
```

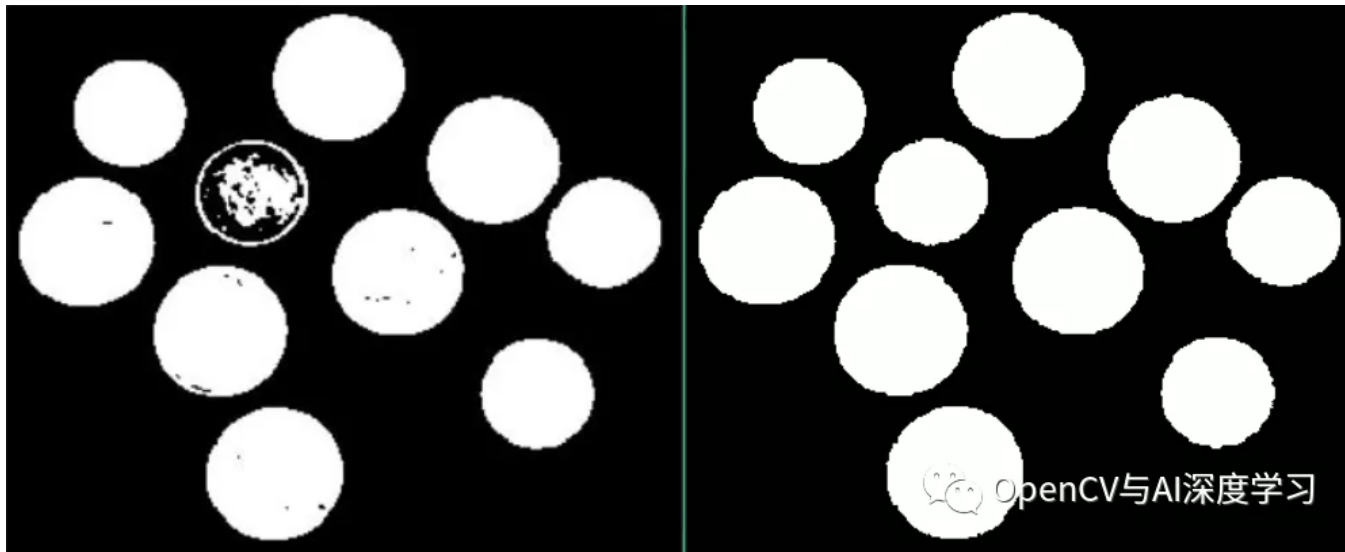


④ 取反圖像與二值圖求並集

```
1 #将二值图与上一步图像求并集
2 im_out = thres | im_floodfill_inv
3 cv2.imshow("holeFill", im_out
```

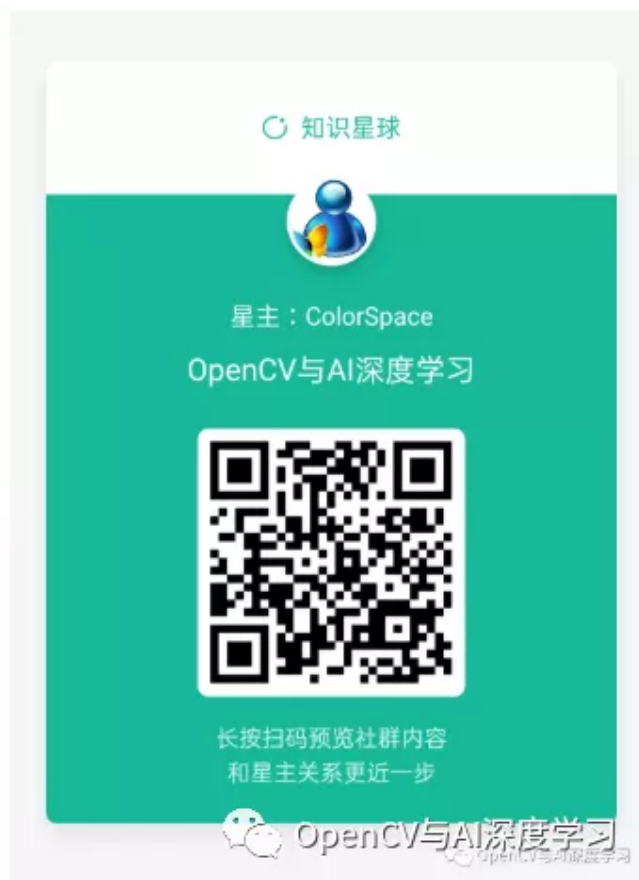


換張圖片測試同樣可以，此方法具有通用性，可自己封裝成函數使用。



參考鏈接：<https://learnopencv.com/filling-holes-in-an-image-using-opencv-python-c/>

對應的C++/C#源碼如有需要可加入下面知識星球獲取。



OpenCV學習歡迎加入QQ交流群：



後台回復：OpenCV145

即可獲取145個OpenCV實例應用代碼(Python/C++雙語言實現)

更多視覺圖像處理相關內容，請長按關注：



OpenCV與AI深度學習

致力於OpenCV、機器視覺、深度學習和人工智能相關內容的學習、交流、分享！

110篇原創內容



公眾號

覺得有用，麻煩給個贊和在看 

收錄於話題·35個 >

下一篇 · OpenCV應用| 計算兩條曲線之間的最小距離(附源碼) >

喜歡此內容的人還喜歡

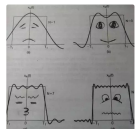
快速指南：使用OpenCV預處理神經網絡中的面部圖像

小白學視覺



我們為什麼要進行傅里葉變換？它的意義是什麼

新機器視覺



11. 圖像合成與圖像融合

計算攝影學

