

OpenCV應用| 計算兩條曲線之間的最小距離(附源碼)

原創 色彩空間 OpenCV與AI深度學習 今天

收錄於話題

#OpenCV 34 #圖像處理 4



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！

109篇原創內容



公眾號

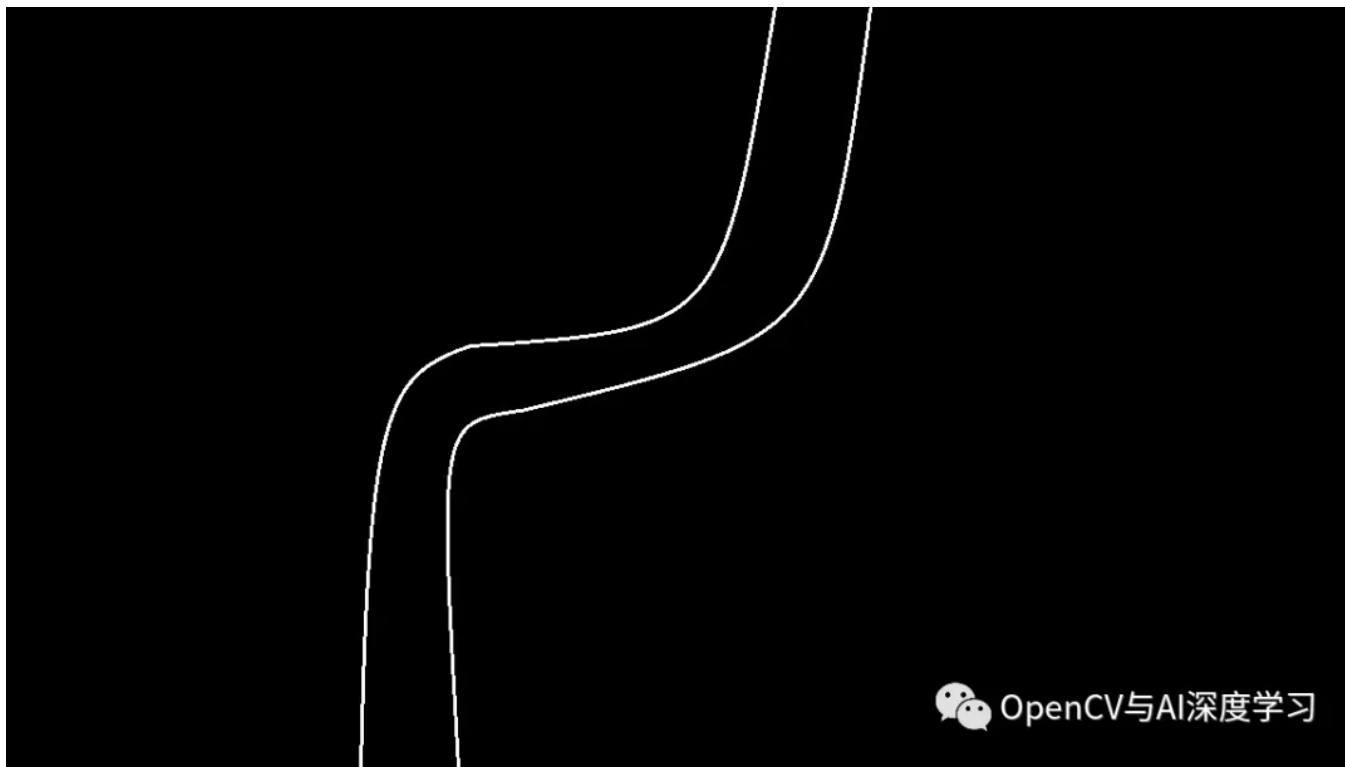
導讀

本文主要介紹使用OpenCV計算兩條曲線之間的最小距離

實現目標

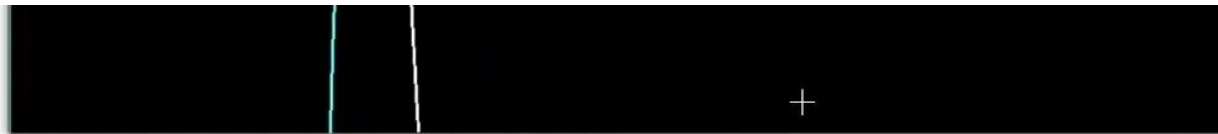
【

測試圖像：



動態效果演示：





實現步驟與代碼

實現步驟：

- ① 分別找到兩條曲線的輪廓；
- ② 計算每條曲線輪廓點與另一條輪廓上點的距離；
- ③ 記錄距離最小值與對應最小值時的點坐標；
- ④ 繪製結果。

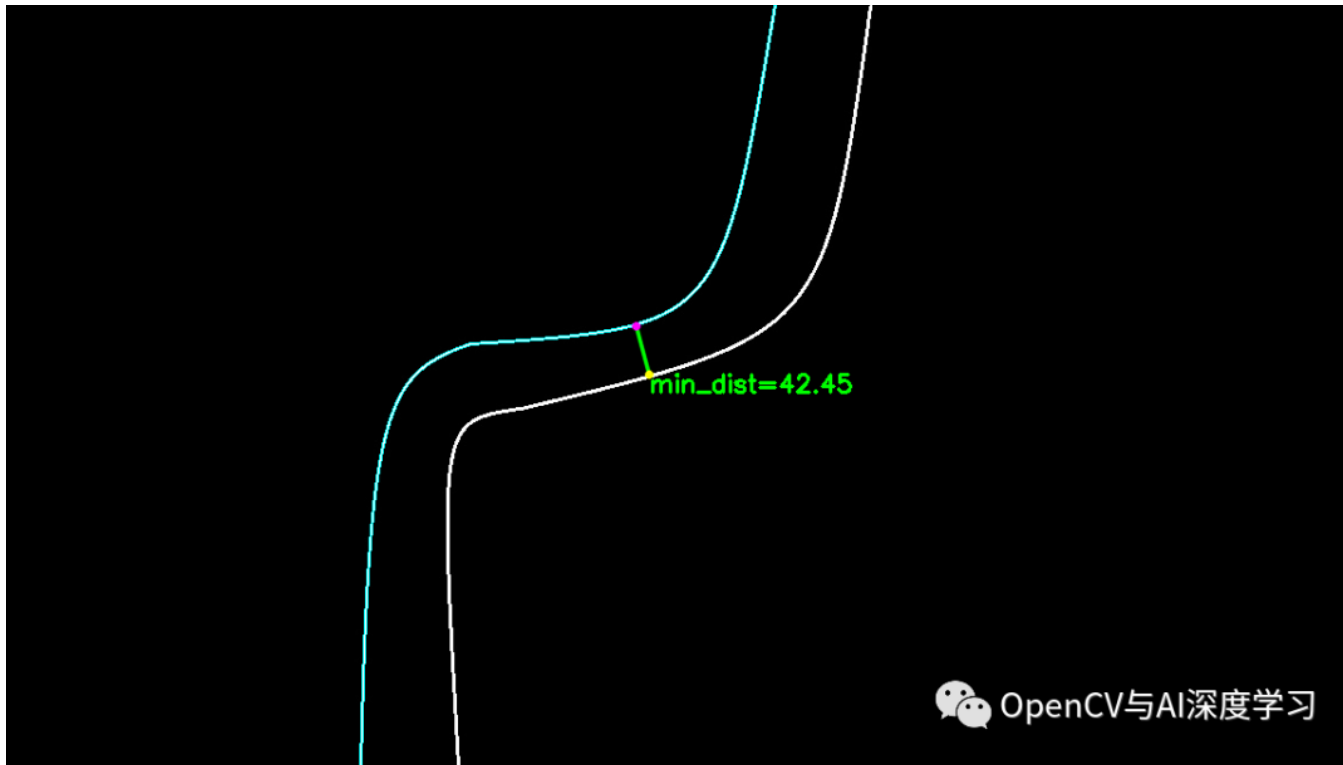
代碼實現與演示：

```
1  # -*- coding: cp936 -*-
2
3  import numpy as np
4  import math
5  import cv2
6
7  def cal_pt_distance(pt1, pt2):
8      dist = math.sqrt(pow(pt1[0]-pt2[0],2) + pow(pt1[1]-pt2[1],2))
9      return dist
10
11  font = cv2.FONT_HERSHEY_SIMPLEX
12  img = cv2.imread('test.jpg')
13  cv2.imshow('src',img)
```

```
14 gray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
15 gray = cv2.GaussianBlur(gray, (3,3), 0)
16
17 ret,thresh = cv2.threshold(gray, 150, 255, cv2.THRESH_BINARY)
18 contours,hierarchy = cv2.findContours(thresh, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_NONE)
19 #thresh,contours,hierarchy = cv2.findContours(thresh, cv2.RETR_LIST, cv2.CHAIN_APPROX_NONE)
20
21 flag = False
22 minDist = 10000
23 minPt0 = (0,0)
24 minPt1 = (0,0)
25 for i in range(0,len(contours[1])):#遍历所有轮廓
26     pt = tuple(contours[1][i][0])
27     #print(pt)
28     min_dis = 10000
29     min_pt = (0,0)
30     #distance = cv2.pointPolygonTest(contours[1], pt, False)
31     for j in range(0,len(contours[0])):
32         pt2 = tuple(contours[0][j][0])
33         distance = cal_pt_distance(pt, pt2)
34         #print(distance)
35         if distance < min_dis:
36             min_dis = distance
37             min_pt = pt2
38             min_point = pt
39     if min_dis < minDist:
40         minDist = min_dis
```

```
41     minPt0 = min_point
42     minPt1 = min_pt
43     temp = img.copy()
44     cv2.drawContours(img, contours, 1, (255, 255, 0), 1)
45     cv2.line(temp, pt, min_pt, (0, 255, 0), 2, cv2.LINE_AA)
46     cv2.circle(temp, pt, 5, (255, 0, 255), -1, cv2.LINE_AA)
47     cv2.circle(temp, min_pt, 5, (0, 255, 255), -1, cv2.LINE_AA)
48     cv2.imshow("img", temp)
49     if cv2.waitKey(1) & 0xFF == 27: #按下Esc键退出
50         flag = True
51         break
52     if flag:
53         break
54
55     cv2.line(img, minPt0, minPt1, (0, 255, 0), 2, cv2.LINE_AA)
56     cv2.circle(img, minPt0, 3, (255, 0, 255), -1, cv2.LINE_AA)
57     cv2.circle(img, minPt1, 3, (0, 255, 255), -1, cv2.LINE_AA)
58     cv2.putText(img, ("min_dist=%0.2f"%minDist), (minPt1[0], minPt1[1]+15), font, 0.7, (0, 255, 0), 2)
59
60     cv2.imshow('result', img)
61     cv2.imwrite('result.png', img)
62     cv2.waitKey(0)
63     cv2.destroyAllWindows()
```

最終結果：



OpenCV學習歡迎加入QQ交流群：



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！

109篇原創內容



公眾號

覺得有用 · 麻煩給個贊和在看 

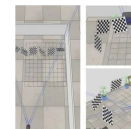
收錄於話題·34個 >

下一篇 · OpenCV技巧| 常用格式圖片保存為透明背景圖片(附Python源碼)-教你輕鬆製作Logo >

喜歡此內容的人還喜歡

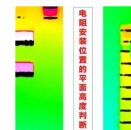
重磅！超多乾貨的相機標定、結構光、機械臂抓取等課程

計算機視覺工坊



3D視覺應用於PCB缺陷檢測

機器視覺課堂



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

