

實戰| 使用圖像處理改善OCR結果(詳細步驟+ 源碼)

原創 Color Space OpenCV與AI深度學習 2022-12-27 09:01 發表於重慶

收錄於合集

#圖像處理 189 #計算機視覺 218 #Blob分析 3 #OCR 15 #文字識別 3

點擊下方**卡片**，關注“**OpenCV與AI深度學習**”

視覺/圖像重磅乾貨，第一時間送達！



OpenCV與AI深度學習

專注計算機視覺、深度學習和人工智能領域乾貨、應用、行業資訊的分享交流！

184篇原創內容

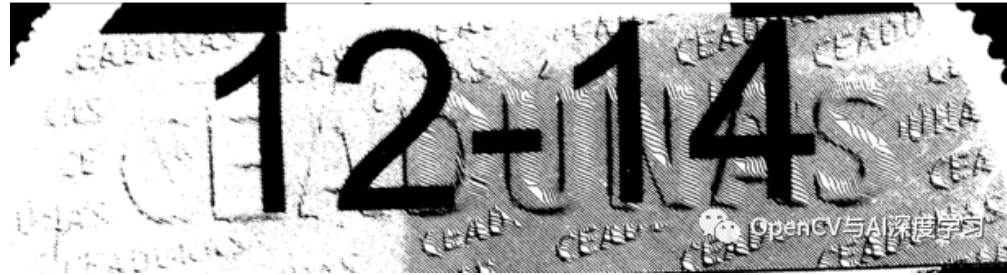
公眾號

導 讀

本文主要介紹一個通過圖像處理改善OCR識別結果的實例，並給出詳細步驟和源碼。

背景介紹

在很多情況下，文字識別會遇到困難。比如非單一的背景、雜訊干擾、文字部分缺失等。如下圖所示：



我們希望識別圖中的黑色文字(12-14),但背景較複雜且存在其他干擾，如果直接用Tesseract識別(代碼如下)，識別結果為空。

```
1  # -*- coding:utf-8 -*-
2  import pytesseract
3  from PIL import Image
4
5  # 打开图像
6  image = Image.open('0.png')
7
8  # OCR 识别：Lang 默认英文
9  text = pytesseract.image_to_string(image)
10
11 # 打印识别后的文本
12 print(text)
```

對這種複雜情況的文字識別，直接去識別很容易失敗。思考：可不可以通過圖像處理將我們需要的部分分割或凸顯出來再做識別？本文將以此為例做演示說明。

詳細實現步驟

【1】OTSU二值化

```
1 image = cv2.imread('0.png')
2 gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
3
4 _,thresh = cv2.threshold(gray, 0, 255, cv2.THRESH_BINARY_INV | cv2.THRESH_C
5 cv2.imshow("Otsu", thresh)
```



【2】距離變化+ 歸一化

```
1 dist = cv2.distanceTransform(thresh, cv2.DIST_L2, 5)
2 dist = cv2.normalize(dist, dist, 0, 1.0, cv2.NORM_MINMAX)
3 dist = (dist * 255).astype("uint8")
4 cv2.imshow("Dist", dist)
```



【3】對距離變換結果圖做OTSU二值化

```

1 _,dist = cv2.threshold(dist, 0, 255, cv2.THRESH_BINARY | cv2.THRESH_OTSU)
2 cv2.imshow("Dist Otsu", dist)

```

processing/

(2) <https://stackoverflow.com/questions/33881175/remove-background-noise-from-image-to-make-text-more-clear-for-ocr>



份Pytorch函數常用手冊,除包括Tensors介紹、基礎函數介紹、數據處理函數、優化函數、

C

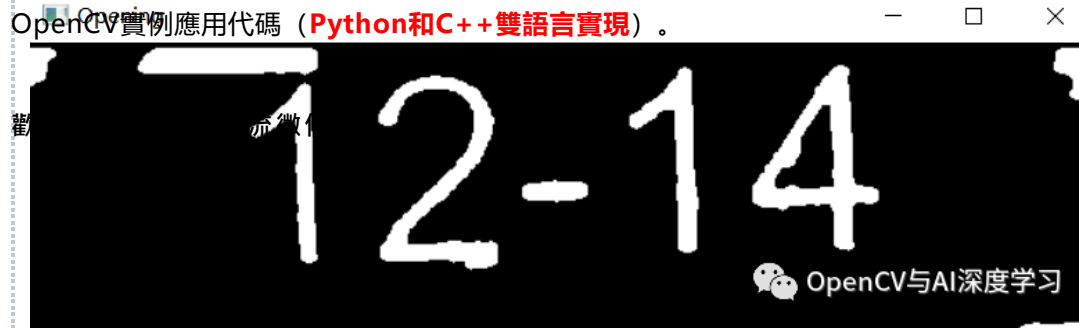
```

1 kernel = cv2.getStructuringElement(cv2.MORPH_ELLIPSE, (7, 7))
2 opening = cv2.morphologyEx(dist, cv2.MORPH_OPEN, kernel)
3 cv2.imshow("Opening", opening)

```

下

在, [OpenCV實例應用代碼 \(Python和C++雙語言實現\)](#)。



【5】輪廓篩選·找出文字區域

```

1 black_img = cv2.cvtColor(opening, cv2.COLOR_GRAY2BGR)
2
3 cnts = cv2.findContours(opening.copy(), cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX)
4 cnts = imutils.grab_contours(cnts)
5 chars = []
6 # Loop over the contours

```

```
7 for c in cnts:
8     # compute the bounding box of the contour
9     (x, y, w, h) = cv2.boundingRect(c)
10    if w >= 35 and h >= 100:
11        chars.append(c)
12
13    cv2.drawContours(black_img, chars, -1, (0, 255, 0), 2)
14    cv2.imshow("chars", black_img)
```



收錄於合集#圖像處理 189

上一篇

手把手教你使用YOLOV5 訓練目標檢測模型

下一篇

抗原試劑需求大爆發！機器視覺助力新冠抗原檢測試劑盒生產

喜歡此內容的人還喜歡

Spring參數自動綁定問題

小豹講安全



文件整理小工具

XData Analysis VBA



IP偽造插件FakeIP



