

[首頁](#)[MySQL教學](#)[網站技巧](#)[網路程式設計](#)[軟體程式設計](#)[資料庫](#)[作業系統](#)[其它](#)[指令碼專欄](#)[變時空廣告](#)[首頁](#)[科技](#)[程式語言](#)

Python使用OpenCV進行標定

指令碼專欄 · 發表 2018-05-08

這篇文章主要介紹了Python使用OpenCV進行標定,小編覺得挺不錯的,現在分享給大家,也給大家做個參考。一起跟隨小



php使用curl偽造來源ip和refer的方法示例

iOS 簡約日曆控制元件EBCalendarView的實現程式碼

編過來看看吧

本文結合OpenCV官方樣例，對官方樣例中的程式碼進行修改，使其能夠正常執行，並對自己採集的資料進行實驗和講解。

一、準備

OpenCV使用棋盤格板進行標定，如下圖所示。為了標定相機，我們需要輸入一系列三維點和它們對應的二維影象點。在黑白相間的棋盤格上，二維影象點很容易通過角點檢測找到。而對於真實世界中的三維點呢？由於我們採集中，是將相機放在一個地方，而將棋盤格定標板進行移動變換不同的位置，然後對其進行拍攝。所以我們需要知道(X,Y,Z)的值。但是簡單來說，我們定義棋盤格所在平面為XY平面，即Z=0。對於定標板來說，我們可以知道棋盤格的方塊尺寸，例如30mm，這樣我們就可以把棋盤格上的角點座標定義為(0,0,0)，(30,0,0)，(60,0,0)，...，這個結果的單位是mm。

3D點稱為object points，2D影象點稱為image points。

iOS使用UICollectionView實現列表頭部拉伸效果

React Navigation 使用中遇到的問題小結

Python 批量合併多個txt檔案的例項講解

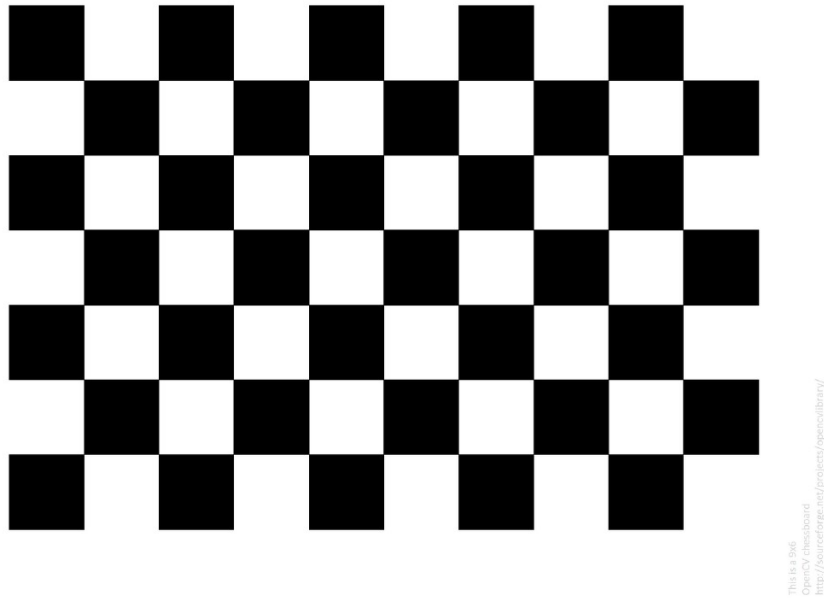
Spring中property-placeholder的使用與解析詳解

numpy 進行陣列拼接,分別在行和列上合併的例項

iOS tableView實現頂部圖片拉伸效果

Android 讀取txt,按行讀取的例項講解

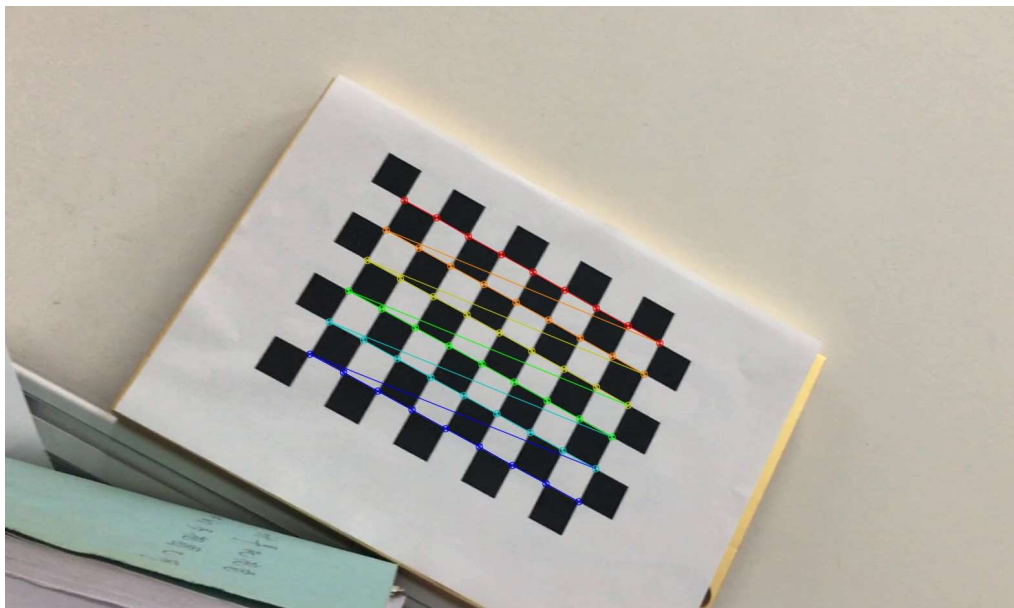
iOS tableView實現下拉圖片放大效果



二、檢測棋盤格角點

為了找到棋盤格模板，我們使用openCV中的函式 `cv2.findChessboardCorners()`。我們也需要告訴程式我們使用的模板是什麼規格的，例如8*8的棋盤格或者5*5棋盤格等，建議使用x方向和y方向個數不相等的棋盤格模板。下面實驗中，我們使用的是10*7的棋盤格，每個方格邊長是20mm，即含有9*6的內部角點。這個函式如果檢測到模板，會返回對應的角點，並返回 `true`。當然不一定所有的影象都能找到需要的模板，所以我們可以使用多幅影象進行定標。除了使用棋盤格，我們還可以使用圓點陣，對應的函式為 `cv2.findCirclesGrid()`。

找到角點後，我們可以使用`cv2.cornerSubPix()`可以得到更為準確的角點畫素座標。我們也可以使用`cv2.drawChessboardCorners()`將角點繪製到影象上顯示。如下圖所示：



三、標定

通過上面的步驟，我們得到了用於標定的三維點和與其對應的影象上的二維點對。我們使用`cv2.calibrateCamera()`進行標定，這個函式會返回標定結果、相機的內參數矩陣、畸變係數、旋轉矩陣和平移向量。

四、去畸變

第三步我們已經得到了相機內參和畸變係數，在將影像去畸變之前，我們還可以使用`cv.getOptimalNewCameraMatrix()`優化內參數和畸變係數，通過設定自由比例因子`alpha`。當`alpha`設為0的時候，將會返回一個剪裁過的將去畸變後不想要的畫素去掉的內參數和畸變係數；當`alpha`設為1的時候，將會返回一個包含額外黑色畫素點的內參數和畸變係數，並返回一個ROI用於將其剪裁掉。

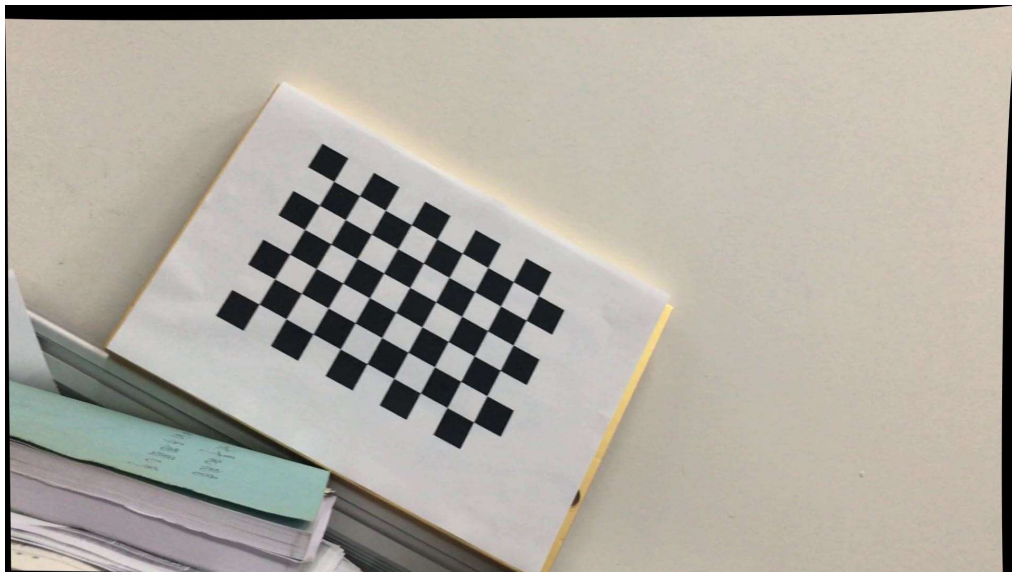
然後我們就可以使用新得到的內參數矩陣和畸變係數對影像進行去畸變了。有兩種方法進行去畸變：

(1)使用`cv2.undistort()`

這是一個最直接的辦法，只用直接呼叫函式就可以得到去畸變的影像，使用上面的ROI可以對其進行剪裁。程式碼如下：

```
1  # undistort
2  dst = cv2.undistort(img, mtx, dist, None, n
3
4  # crop the image
5  x,y,w,h = roi
6  dst = dst[y:y+h, x:x+w]
7  cv2.imwrite('calibresult.png',dst)
```

下圖顯示將一張圖片去畸變後，保留黑色畫素的結果：



(2) 使用remmapping

這是一個分兩步的方法，首先計算一個從畸變影像到非畸變影像的對映，然後使用這個對映關係對影像進行去畸變。

程式碼如下：

```
1 | # undistort
2 | mapx,mapy = cv2.initUndistortRectifyMap(mtx
3 | dst = cv2.remap(img,mapx,mapy,cv2.INTER_LIN
4 |
5 | # crop the image
6 | x,y,w,h = roi
7 | dst = dst[y:y+h, x:x+w]
8 | cv2.imwrite('calibresult.png',dst)
```

五、反投影誤差

通過反投影誤差，我們可以來評估結果的好壞。越接近0，說明結果越理想。通過之前計算的內參數矩陣、畸變係數、旋轉矩陣和平移向量，使用`cv2.projectPoints()`計算三維點到二維影像的投影，然後計算反投影得到的點與影像上檢測到的點的誤差，最後計算一個對於所有標定影像的平均誤差，這個值就是反投影誤差。

程式碼

所有步驟的程式碼如下所示：

```
1  #coding:utf-8
2  import cv2
3  import numpy as np
4  import glob
5
6  # 找棋盤格角點
7  # 閾值
8  criteria = (cv2.TERM_CRITERIA_EPS + cv2.TERM_CRITERIA_MAX_ITER, 10, 0.001)
9  #棋盤格模板規格
10 w = 9
11 h = 6
12 # 世界座標系中的棋盤格點,例如(0,0,0), (1,0,0)
13 objp = np.zeros((w*h,3), np.float32)
14 objp[:, :2] = np.mgrid[0:w,0:h].T.reshape(-1,2)
15 # 儲存棋盤格角點的世界座標和影像座標對
16 objpoints = [] # 在世界座標系中的三維點
17 imgpoints = [] # 在影像平面的二維點
18
19 images = glob.glob('calib/*.png')
20 for fname in images:
21     img = cv2.imread(fname)
22     gray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
23     # 找到棋盤格角點
```

```
24 ret, corners = cv2.findChessboardCorners(  
25 # 如果找到足夠點對，將其儲存起來  
26 if ret == True:  
27     cv2.cornerSubPix(gray, corners, (11,11), (-  
28     objpoints.append(objp)  
29     imgpoints.append(corners)  
30     # 將角點在影象上顯示  
31     cv2.drawChessboardCorners(img, (w,h), co  
32     cv2.imshow('findCorners',img)  
33     cv2.waitKey(1)  
34 cv2.destroyAllWindows()  
35  
36 # 標定  
37 ret, mtx, dist, rvecs, tvecs = cv2.calibra  
38  
39 # 去畸變  
40 img2 = cv2.imread('calib/00169.png')  
41 h, w = img2.shape[:2]  
42 newcameramt, roi=cv2.getOptimalNewCameraM  
43 dst = cv2.undistort(img2, mtx, dist, None,  
44 # 根據前面ROI區域裁剪圖片  
45 #x,y,w,h = roi  
46 #dst = dst[y:y+h, x:x+w]  
47 cv2.imwrite('calibresult.png',dst)  
48  
49 # 反投影誤差  
50 total_error = 0  
51 for i in xrange(len(objpoints)):  
52     imgpoints2, _ = cv2.projectPoints(objpoin  
53     error = cv2.norm(imgpoints[i],imgpoints2,  
54     total_error += error  
55 print "total error: ", total_error/len(obj
```

以上就是本文的全部內容，希望對大家的學習有所幫助，也希望大家多多支援itread01.com。

[python使用opencv按一定間隔擷取視訊幀](#)

[vue2.0設定proxyTable使用axios進行跨域請求的方法](#)

[使用rpcLib進行Python網路程式設計時的註釋問題](#)

[python使用opencv驅動攝像頭的方法
詳解如何使用babel進行es6檔案的編譯](#)

[spring cloud 使用Eureka 進行服務治理方法](#)

[Python程式設計使用NLTK進行自然語言處理詳解](#)

[在 Ubuntu 中使用 NTP 進行時間同步設定](#)

[微信小程序使用gitee進行版本管理
使用FormData進行Ajax請求上傳檔案的例項程式碼](#)

[使用pytorch進行影像的順序讀取方法
Django 使用Ajax進行前後臺互動的示例講解](#)

[首頁](#)

[Python教學](#)



ITREAD01.COM © 2018. 版權所有。



標籤： 指令碼專欄 python

👍 您可能也會喜歡...

[python使用opencv進行人臉識別](#)

[python使用TensorFlow進行影像處理的方法](#)

[python使用pil進行影像處理\(等比例壓縮、裁剪\)例項程式碼](#)

[python使用PyCharm進行遠端開發和除錯](#)

[python使用socket進行簡單網路連線的方法](#)

[Android Studio中使用jni進行opencv開發的環境配置方法](#)

[使用sklearn進行對資料標準化、歸一化以及將資料還原的方法](#)

[Python中使用Counter進行字典建立以及key數量統計的方法](#)