

OpenCV-python 学习笔记 OpenCV颜色空间转换

1. 转换色彩空间

在OpenCV中有超过150种进行色彩空间转换的方法。

在这里介绍其中的两种：

BGR↔Gray 与BGR↔HSV

使用函数为：`cv2.cvtColor(input_image,flag)`，其中 `flag` 是转换类型

对于BGR↔Gray：

使用 `flag` 为：`cv2.COLOR_BGR2GRAY`

对于BGR↔HSV：

使用 `flag` 为：`cv2.COLOR_BGR2HSV`

BGR:

与RGB类似，不过R在高位,G在中间,B在低位

HSV:

H（色彩/色度） S（饱和度） V（亮度）

Gray:

灰度空间

查询所有的flag:

```
1 import cv2
2 flags=[i for i in dir(cv2) if i.startswith('COLOR_')]
3 print (flags)
```

在 OpenCV 的 HSV 格式中，H（色彩/色度）的取值范围是 [0, 179]，S（饱和度）的取值范围 [0, 255]，V（亮度）的取值范围 [0, 255]。但是不同的软件使用的值可能不同。所以当你拿 OpenCV 的 HSV 值与其他软件的 HSV 值对比时，一定要记得归一化。

2. 物体跟踪

目的：提取某个特定颜色的物体

在 HSV 颜色空间中要比在 BGR 空间中更容易表示一个特定颜色。

例如：提取的是一个蓝色的物体

步骤：

- (1) 从视频中获取每一帧图像
- (2) 将图像转换到HSV空间
- (3) 设置HSV阈值到蓝色范围
- (4) 获取蓝色物体

例如在蓝色物体周围画圈：

```
1  import cv2
2  import numpy as np
3
4  cap = cv2.VideoCapture(0)
5
6  while(1):
7      #获取每一帧
8      ret, frame = cap.read()
9      #转换到HSV
10     hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)
11     #设定蓝色的阈值
12     lower_blue = np.array([110, 50, 50])
13     upper_blue = np.array([130, 255, 255])
14     #根据阈值构建掩模
15     mask = cv2.inRange(hsv, lower_blue, upper_blue)
16     #对原图和掩模进行位运算
17     res = cv2.bitwise_and(frame, frame, mask=mask)
18     #显示图像
19     cv2.imshow('frame', frame)
20     cv2.imshow('mask', mask)
21     cv2.imshow('res', res)
22     k = cv2.waitKey(5) & 0xFF
23     if k == 27:
24         break
25     #关闭窗口
26     cv2.destroyAllWindows()
```

3. 寻找跟踪对象的HSV值

利用函数 `cv2.cvtColor()`:

向函数传入RGB参数, 例如寻找绿色的HSV值:

```
1 import cv2
2 import numpy as np
3 green=np.uint8([0,255,0])
4 hsv_green=cv2.cvtColor(green,cv2.COLOR_BGR2HSV)
5
6 error: /builddir/build/BUILD/opencv-2.4.6.1/
modules/imgproc/src/color.cpp:3541:
7 error: (-215) (scn == 3 || scn == 4) && (depth == CV_8U ||
depth == CV_32F) in function.cvtColor
8
9 #scn (the number of channels of the source),
10 #i.e. self.img.channels(), is neither 3 nor 4.
11 # #depth (of the source),
12 #i.e. self.img.depth(), is neither CV_8U nor CV_32F.
13 # 所以不能用 [0,255,0] 而用 [[[0,255,0]]]
14 # 的三层括号应分别对应于 cvArray cvMat IplImage
15
16 green=np.uint8([[[0,255,0]]])
17 hsv_green=cv2.cvtColor(green,cv2.COLOR_BGR2HSV)
18 print (hsv_green )
19 [[[60 255 255]]]
```

现在你可以分别用 `[H-100, 100, 100]` 和 `[H+100, 255, 255]` 做上下阈值。除了个方法之外, 你可以使用任何其他图像编辑软件 (例如 GIMP) 或者在线换软件找到相应的 HSV 值, 但是后别忘了调节 HSV 的范围。