Chapter11: Raspberry Pi color recognition

In this lesson , we will test object is a yellow table tennis. We know that the yellow value is ([156,43,46], [180,255,255]),

First, we should convert RGB to hsv by cv2.cvtColor(frame,cv2.COLOR_BGR2HSV),

Then, we can construct a mask according to the threshold value. After morphological treatment by expansion corrosion, perform the bitwise operation of the mask and the original image. When the color is found, draw a circle on the outline of the color to mark it.

The source code of the program is located at:

/home/pi/yahboom/color_tracking/color_tracking

The code as shown in the figure below.

```python
1 #!/usr/bin/env python2
2 # -*- coding: utf-8 -*-
3 """
4 * @par Copyright (C): 2010-2019, Shenzhen Yahboom Tech
5 * @file          basic_writeAndrRead
6 * @version       V1.0
7 * @details
8 * @par History
9 * @author        LongfuSun
10 """
11
12 from __future__ import division
13 import cv2
14 import time
15 import numpy as np
16
17 cap=cv2.VideoCapture(0)
18
19 #Set the camera resolution to (480,320)
20 #If you feel that the image is severely damaged, you can reduce it to (320, 240
21 cap.set(3,480)
22 cap.set(4,320)
23
24 #Set yellow value
25 yellow_lower=np.array([156,43,46])
26 yellow_upper=np.array([180,255,255])
27
28 time.sleep(1)
29
30 while 1:
31     #ret is whether to find the image, frame is the frame itself
32     ret,frame=cap.read()
33
34     frame=cv2.GaussianBlur(frame,(5,5),0)
35     hsv=cv2.cvtColor(frame,cv2.COLOR_BGR2HSV)
36     mask=cv2.inRange(hsv,yellow_lower,yellow_upper)
37
38     #Morphological operation
39     mask=cv2.erode(mask,None,iterations=2)
40     mask=cv2.dilate(mask,None,iterations=2)
41     mask=cv2.GaussianBlur(mask,(3,3),0)
```

```
42    res=cv2.bitwise_and(frame,frame,mask=mask)
43    cnts=cv2.findContours(mask.copy(),cv2.RETR_EXTERNAL,
44                           cv2.CHAIN_APPROX_SIMPLE)[-2]
45    if len(cnts)>0:
46        cnt = max (cnts,key=cv2.contourArea)
47        (x,y),radius=cv2.minEnclosingCircle(cnt)
48        cv2.circle(frame,(int(x),int(y)),int(radius)*2,
49                      (255,0,255),2)
50        print('x:',x,'y:',y)
51    cv2.imshow('capture',frame)
52    if cv2.waitKey(1)==119:
53        break
54 cap.release()
55 cv2.destroyAllWindows()
56
```

The result is as shown in the figure below. A blue contour with real-time changes is generated in the contour line of yellow table tennis.

Figure1-1