```
import cv2
import numpy as np
import time
import matplotlib.pyplot as plt
import serial
import math

t=0
x=[]
y=[]
cap = cv2.VideoCapture(0)
lx2 =0
ly2=0
t2=0
t2=0
p=0
tq=0
tw=0
x1=[]
y1=[]
ser=serial.Serial("COM9")
while(1):

# Take each frame

  _, frame = cap.read()
  tq=time.time()

# Convert BGR to HSV
  hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)
  lg = np.array([50,50,50])
```

```
   ug= np.array([70,255,255])
# Threshold the HSV image to get only blue colors
   blur = cv2.GaussianBlur(hsv,(5,5),0)


   mask = cv2.inRange(blur, lg, ug)
   ret3,mask = cv2.threshold(mask,0,255,cv2.THRESH_BINARY+cv2.THRESH_OTSU)
   kernel = np.ones((3,3),np.uint8)
   erosion = cv2.erode(mask,kernel,iterations = 1)
   dilation = cv2.dilate(erosion,kernel,iterations = 3)
   dilation[1,1]=1
   image, contours, hierarchy =
cv2.findContours(dilation,cv2.RETR_TREE,cv2.CHAIN_APPROX_SIMPLE)
# Bitwise-AND mask and original image
   res = cv2.bitwise_and(frame,frame, mask= mask)
   cnt=contours[0]


   l1,l2 = tuple(cnt[cnt[:,:,0].argmin()][0])
   r1,r2 = tuple(cnt[cnt[:,:,0].argmax()][0])
   t1,t2 = tuple(cnt[cnt[:,:,1].argmin()][0])
   b1,b2 = tuple(cnt[cnt[:,:,1].argmax()][0])
   cv2.imshow('frame',frame)
   cv2.imshow('mask',mask)
   cv2.imshow('res',res)
   cv2.imshow("dilation",dilation)
   lx=(l1+r1+t1+b1)/4
   ly=(l2+r2+t2+b2)/4
   print lx,ly
   x.append(lx)
   y.append(ly)


   vel=math.sqrt((lx2-lx)*(lx2-lx)+(ly2-ly)*(ly2-ly))
```

```
#find velocity
vel=vel/(tq-tw)
print "velocity",vel
print "t2-t1",tq-tw
vel=int(vel)
a=vel%10
vel=vel/10
b=vel%10
vel=vel/10
c=vel%10
vel=vel/10
d=vel%10
vel=vel/10
e=vel%10
print "edcba",e,d,c,b,a



if e==0:
  ser.write("0")

elif e==1:
  ser.write("1")
elif e==2:
  ser.write("2")
elif e==3:
  ser.write("3")
elif e==4:
  ser.write("4")
elif e==5:
  ser.write("5")
```

```
elif e==6:
  ser.write("6")
elif e==7:
  ser.write("7")
elif e==8:
  ser.write("8")
elif e==9:
  ser.write("9")

if d==0:
  ser.write("0")
elif d==1:
  ser.write("1")
elif d==2:
  ser.write("2")
elif d==3:
  ser.write("3")
elif d==4:
  ser.write("4")
elif d==5:
  ser.write("5")
elif d==6:
  ser.write("6")
elif d==7:
  ser.write("7")
elif d==8:
  ser.write("8")
elif d==9:
  ser.write("9")

if c==0:
```

```
  ser.write("0")
elif c==1:
  ser.write("1")
elif c==2:
  ser.write("2")
elif c==3:
  ser.write("3")
elif c==4:
  ser.write("4")
elif c==5:
  ser.write("5")
elif c==6:
  ser.write("6")
elif c==7:
  ser.write("7")
elif c==8:
  ser.write("8")
elif c==9:
  ser.write("9")



if b==0:
  ser.write("0")

elif b==1:
  ser.write("1")
elif b==2:
  ser.write("2")
elif b==3:
  ser.write("3")
```

```
elif b==4:
  ser.write("4")
elif b==5:
  ser.write("5")
elif b==6:
  ser.write("6")
elif b==7:
  ser.write("7")
elif b==8:
  ser.write("8")
elif b==9:
  ser.write("9")

if a==0:
  ser.write("0")

elif a==1:
  ser.write("1")
elif a==2:
  ser.write("2")
elif a==3:
  ser.write("3")
elif a==4:
  ser.write("4")
elif a==5:
  ser.write("5")
elif a==6:
  ser.write("6")
elif a==7:
  ser.write("7")
elif a==8:
```

```
      ser.write("8")
    elif a==9:
      ser.write("9")
    ser.write("   ")
    lx2=lx
    ly2=ly
    t2=t1
    tw=tq

    k = cv2.waitKey(5) & 0xFF
    if k == 27:
      break

for i in x:
    t=t+1
p=0
print "t", t
for i in range (t):
    if x[i] != 1:
        if x[i+1]==1:
            x[i]=1
            y[i]=1
        elif x[i+2]==1:
            x[i]=1
            y[i]=1
        elif x[i+3]==1:
            x[i]=1
            y[i]=1
        elif x[i+4]==1:
            x[i]=1
            y[i]=1
```

```
for i in range (t):
   if x[i]!=1 and y[i]!=1:
      x1.append(x[i])
      y1.append(y[i])



plt.plot( x1, y1)
plt.xlabel('x coordinate')
plt.ylabel('y coordinate')
plt.title('trajectory')
plt.show ()

cv2.destroyAllWindows()
```