```
import numpy as np
import cv2
import math
import struct
import time
import serial

ser1=serial.Serial('com7',9600)

time.sleep(2)

detect_cascade=cv2.CascadeClassifier('haarcascade_a.xml')
camR=cv2.VideoCapture(2)
camR.set(cv2.CAP_PROP_FRAME_WIDTH,720)
camR.set(cv2.CAP_PROP_FRAME_HEIGHT,405)
camL=cv2.VideoCapture(1)
camL.set(cv2.CAP_PROP_FRAME_WIDTH,720)
camL.set(cv2.CAP_PROP_FRAME_HEIGHT,405)

X=0
Y=0
Z=0

XX=0
YY=0
ZZ=0

c=0
c1=0
c2=0
c3=0
c4=0
c5=0
c6=0
c7=0
c8=0
c9=0
c10=0
c11=0

while(True):
    a1=2000 #right
    a2=2000 #left
    tfR,frameR=camR.read()
    gray = cv2.cvtColor(frameR, cv2.COLOR_BGR2GRAY)
    detect = detect_cascade.detectMultiScale(gray, 1.3, 5)
    for (xrr,yrr,wr,hr) in detect:
        cv2.rectangle(frameR,(xrr,yrr),(xrr+wr,yrr+hr),(255,0,0),2)
```

```
        cv2.putText(frameR, "A",(xrr,yrr-50), cv2.FONT_HERSHEY_SIMPLEX,
0.75,(255,0,0),2);
        cv2.putText(frameR, "X=" + `X`,(xrr,yrr-30), cv2.FONT_HERSHEY_SIMPLEX,
0.5,(255,0,0),2);
        cv2.putText(frameR, "Y=" + `Y`,(xrr,yrr-15), cv2.FONT_HERSHEY_SIMPLEX,
0.5,(255,0,0),2);
        cv2.putText(frameR, "Z=" + `Z`,(xrr,yrr), cv2.FONT_HERSHEY_SIMPLEX,
0.5,(255,0,0),2);

        cv2.putText(frameR, "XX=" + `XX`,(xrr,(yrr+hr)+15),
cv2.FONT_HERSHEY_SIMPLEX, 0.5,(255,0,0),2);
        cv2.putText(frameR, "YY=" + `YY`,(xrr,(yrr+hr)+30),
cv2.FONT_HERSHEY_SIMPLEX, 0.5,(255,0,0),2);
        cv2.putText(frameR, "ZZ=" + `ZZ`,(xrr,(yrr+hr)+45),
cv2.FONT_HERSHEY_SIMPLEX, 0.5,(255,0,0),2);

        xr=xrr+(wr/2)
        yr=yrr+(hr/2)
        a1=(50*math.tan(32.5*3.14/180)*(xr-360)/360)-3.4
        hr=(50*math.tan(17*3.14/180)*(yr-202.5)/202.5)


    cv2.imshow('frame_Right',frameR)

    tfL,frameL=camL.read()
    gray = cv2.cvtColor(frameL, cv2.COLOR_BGR2GRAY)
    detect = detect_cascade.detectMultiScale(gray, 1.3, 5)
    for (xll,yll,wl,hl) in detect:
        cv2.rectangle(frameL,(xll,yll),(xll+wl,yll+hl),(255,0,0),2)
        cv2.putText(frameL, "A",(xll,yll-50), cv2.FONT_HERSHEY_SIMPLEX,
0.75,(255,0,0),2);
        cv2.putText(frameL, "X=" + `X`,(xll,yll-30), cv2.FONT_HERSHEY_SIMPLEX,
0.5,(255,0,0),2);
        cv2.putText(frameL, "Y=" + `Y`,(xll,yll-15), cv2.FONT_HERSHEY_SIMPLEX,
0.5,(255,0,0),2);
        cv2.putText(frameL, "Z=" + `Z`,(xll,yll), cv2.FONT_HERSHEY_SIMPLEX,
0.5,(255,0,0),2);

        cv2.putText(frameL, "XX=" + `XX`,(xll,(yll+hl)+15),
cv2.FONT_HERSHEY_SIMPLEX, 0.5,(255,0,0),2);
        cv2.putText(frameL, "YY=" + `YY`,(xll,(yll+hl)+30),
cv2.FONT_HERSHEY_SIMPLEX, 0.5,(255,0,0),2);
        cv2.putText(frameL, "ZZ=" + `ZZ`,(xll,(yll+hl)+45),
cv2.FONT_HERSHEY_SIMPLEX, 0.5,(255,0,0),2);

        xl=xll+(wl/2)
        yl=yll+(hl/2)
        a2=(50*math.tan(32.5*3.14/180)*(xl-360)/360)+3.4
```

```
    hl=(50*math.tan(17*3.14/180)*(yl-202.5)/202.5)


cv2.imshow('frame_Left',frameL)

if (a1<1000):
    if (a2<1000):
        x=((3.4*((-a2)-a1))/((-6.8)+a2-a1))
        y=(50-(340/(6.8-a2+a1)))
        X=-x
        Y=-(50-y)
        z=-(Y*hr)/50
        Z=z

        ZZ=Z+13
        XX=38-(Y+1)
        YY=X+28

        print 'X='
        print X
        print 'Y='
        print Y
        print 'Z='
        print Z


if c11==1:
    c10=c9
    c9=c8
    c8=c7
    c7=c6
    c6=c5
    c5=c4
    c4=c3
    c3=c2
    c2=c1
    c1=c
    c=XX+YY+ZZ

    if c10>0:
        if ((c10-c9)*(c10-c9))<=4:
            if ((c10-c8)*(c10-c8))<=4:
                if ((c10-c7)*(c10-c7))<=4:
                    if ((c10-c6)*(c10-c6))<=4:
                        if ((c10-c5)*(c10-c5))<=4:
                            if ((c10-c4)*(c10-c4))<=4:
                                if ((c10-c3)*(c10-c3))<=4:
                                    if ((c10-c2)*(c10-c2))<=4:
```

```
                                    if ((c10-c1)*(c10-c1))<=4:
                                        if ((c10-c)*(c10-c))<=4:

ser1.write(struct.pack('>BBBB',XX,YY,ZZ,TT))
                                                    c11=0

    key=cv2.waitKey(1)

    if key==ord('r'):
        c11=1


    if key==ord('k'):
            break

camR.release()
camL.release()
cv2.destroyAllWindows()
```