THE SOCCER ROBOT

# DESCRIPTION

This robot is the result of a course that ….

# aaaa

AAAAAAA………..

# PROGRAM

import wiringpi as wp

from collections import deque

import imutils

import cv2

import numpy as np

import cv2.cv as cv

import time

import sys

wp.wiringPiSetupGpio()

#function definitions:

def Motor(x,y,pwm):

wp.pinMode(x,1)

wp.pinMode(y,1)

wp.pinMode(pwm,1)

wp.softPwmCreate(pwm,0,200)

return x,y,pwm

def forward(wheel,speed):

(x,y,pwm)=wheel

if wheel==right\_wheel:

wp.digitalWrite(x,0)

wp.digitalWrite(y,1)

else:

wp.digitalWrite(x,1)

wp.digitalWrite(y,0)

wp.softPwmWrite(pwm,speed)

def backward(wheel,speed):

(x,y,pwm)=wheel

if wheel==left\_wheel:

wp.digitalWrite(x,0)

wp.digitalWrite(y,1)

else:

wp.digitalWrite(x,1)

wp.digitalWrite(y,0)

wp.softPwmWrite(pwm,speed)

def stop(motor):

(x,y,pwm)=motor

wp.digitalWrite(x,0)

wp.digitalWrite(y,0)

def move\_dist(dist):

time\_move=1200\*dist/circum\_wheel

forward(left\_wheel,37)

forward(right\_wheel,52)

wp.delay(int(round(time\_move)))

stop(left\_wheel)

stop(right\_wheel)

def rotate(speed,dir='counter\_clock'):

if dir=='counter\_clock':

forward(right\_wheel,speed)

backward(left\_wheel,speed)

else:

backward(right\_wheel,speed)

forward(left\_wheel,speed)

def left\_turn(speed):

rotate(speed,'counter\_clock')

def right\_turn(speed):

rotate(speed,'clock')

def straight():

forward(left\_wheel,37)

forward(right\_wheel,52)

def stop\_bot():

stop(left\_wheel)

stop(right\_wheel)

def Go\_to\_Location(place):

camera = cv2.VideoCapture(0)

try:

Lower=place[0]

Upper=place[1]

check\_loc=True

caught=False

while check\_loc:

if not caught:

rotate(set\_speed)

(grabbed, frame) = camera.read()

frame = imutils.resize(frame, width=200,height=200)

hsv = cv2.cvtColor(frame, cv2.COLOR\_BGR2HSV)

mask = cv2.inRange(hsv,Lower,Upper)

mask = cv2.erode(mask, None, iterations=2)

mask= cv2.dilate(mask, None, iterations=2)

cnts =cv2.findContours(mask.copy(), cv2.RETR\_EXTERNAL, cv2.CHAIN\_APPROX\_SIMPLE)[-2]

center = None

if len(cnts)>0:

c = max(cnts, key=cv2.contourArea)

((x,y),radius) = cv2.minEnclosingCircle(c)

M = cv2.moments(c)

center = (int(M["m10"]/M["m00"]), int(M["m01"]/M["m00"]))

if center[0]<90:

print 'turn right'

caught=True

right\_turn(set\_speed)

elif center[0]>110:

print 'turn left'

caught=True

left\_turn(set\_speed)

else :

caught=True

print 'straight \t', radius

straight()

if 47<radius:

check\_loc=False

stop\_bot()

move\_dist(8)

print 'Moved Front successfuly'

except KeyboardInterrupt:

print 'stopped'

stop\_bot()

sys.exit()

def Servo(pin):

wp.softPwmCreate(pin,0,100)

return pin

def Sweep(servo,dir,delay,angle):

pin=servo

if dir=='down' or dir=='close':

for i in range(0,int(angle+1),1):

wp.softPwmWrite(pin,i)

wp.delay(delay)

else:

for i in range(int(angle),-1,-1):

wp.softPwmWrite(pin,i)

wp.delay(delay)

def Collect\_ball():

try:

Sweep(main\_arm,'down',200,14)

wp.delay(200)

Sweep(collector\_arm,'close',200,13)

wp.delay(200)

Sweep(main\_arm,'up',200,14)

except :

wp.softPwmWrite(main\_arm,0)

wp.softPwmWrite(collector\_arm,0)

def Shoot\_ball():

try:

Sweep(main\_arm,'down',200,8)

wp.delay(200)

Sweep(collector\_arm,'open',200,12)

wp.delay(2000)

Sweep(main\_arm,'up',200,8)

wp.delay(200)

Sweep(collector\_arm,'close',200,12)

except :

wp.softPwmWrite(main\_arm,0)

wp.softPwmWrite(collector\_arm,0)

right\_wheel=Motor(23,24,25)

left\_wheel=Motor(17,27,22)

main\_arm=Servo(12)

collector\_arm=Servo(26)

circum\_wheel=17

dist\_between\_wheels=17.9

set\_speed=25

orange\_Lower = (0,114,215)

orange\_Upper = (34,255,255)

pink\_Lower = (0,114,215)

pink\_Upper = (34,255,255)

ball=(orange\_Lower,orange\_Upper)

goal=(pink\_Lower,pink\_Upper)

Go\_to\_Location(ball)

Collect\_ball()

Go\_to\_Location(goal)

Shoot\_ball()