

# DETECȚIA ȘI URMĂRIREA UNUI OBIECT DUPĂ CULOAREA SA CU UN ROBOT CONTROLAT PRIN BLUETOOTH

Alin PETRESCU\*

Universitatea Politehnica Timișoara, Facultatea de Inginerie Hunedoara, Departamentul de Inginerie Electrică și Informatică Industrială, grupa Tehnici informatice în ingineria electrică, an I, Hunedoara, Romania

\*Autorul corespondent: Petrescu Alin, [alinalin\\_p@yahoo.com](mailto:alinalin_p@yahoo.com)

**Rezumat.** *Lucrarea prezintă achiziția în timp real a unei transmisii video și procesarea ei prin intermediul bibliotecii OpenCV. Algoritmul folosit pentru detectarea și urmărirea obiectelor este MeanShift. După procesarea datelor cu poziția obiectului se transmite la robot comenzi de mișcare. Comunicarea se face prin bluetooth. În funcție de poziția obiectului detectat robotul se deplasează către el.*

**Cuvinte cheie:** *arduino, bluetooth car, video detectie, python, openCV, meanshift.*

## Introducere

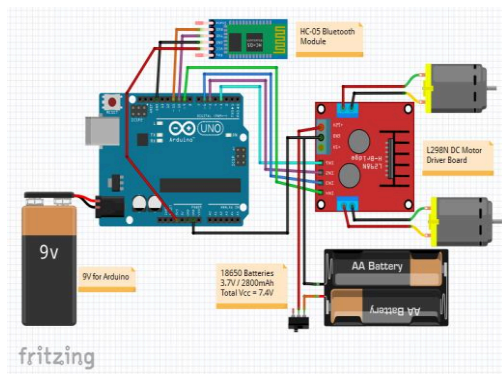
Istoria inteligenței artificiale așa cum este ea percepută în ziua de astăzi – tehnologie care gândește ca un om – începe, însă, în a doua parte a secolului trecut. Odată cu invenția computerului digital în anii '40, ideea construcției unei minți artificiale, electronice, a prins tot mai mult contur.

Una din ramurile inteligenței artificiale este viziunea computerizată. Prin viziunea computerizată se înțelege cum calculatoarele pot dobândi înțelegere din imagini și videoclipuri digitale. Aplicațiile AI de viziune computerizată de astăzi sunt peste tot, inclusiv vehicule autonome, telefoane inteligente, camere de supraveghere, camere de consum, AR / VR, robotică și aplicații industriale.

## Descrierea componentelor hardware utilizate

Componente hardware utilizate la asamblarea robotului sunt:

- placa de dezvoltare compatibilă cu Arduino Uno
- driver dual motor L298N
- două motoare de curent continuu
- bluetooth HC-05
- smartphone
- laptop



a) Schema hardware

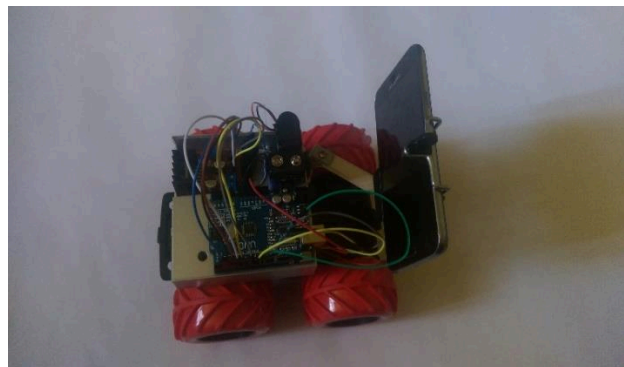


Figura 1.  
b) Robot controlat prin bluetooth



Figura 2.

a) Arduino Uno

b) L298N

c) Motor c.c.

d) HC-05

### Arduino Uno - Figura 2. a)

Arduino Uno este o placă de dezvoltare ideală pentru proiecte creative în domeniul electronicii. Ea utilizează microcontroller-ul ATmega328P și convertorul USB Serial CH340. Placa poate fi alimentată fie prin portul USB, fie dintr-o sursă externă prin conectorul de alimentare. Procesorul este capabil să ruleze un cod scris într-un limbaj de programare care este foarte similar cu limbajul C++.

### Driverul de motoare L298N - Figura 2. b)

Driver-ul conține și un limitator de tensiune liniar, astfel că atunci când tensiunea de alimentare a motoarelor este  $>7V$ , nu este nevoie să alimentăm separat partea de logică. Driver-ul este unul dual, putând să controleze două motoare. Chiar dacă are dimensiuni mai mari, este util prin faptul că beneficiază de un radiator destul de mare și disipă o cantitate mare de căldură. Dimensiuni: 43 x 43 x 27 mm.

### Motor curent continuu - Figura 2. c)

Motor electric metalic fara cutie de viteze. Diametrul axului este de 2mm. Tensiune maxima de alimentare este 6 V (se roteste incepand cu 3 V). Viteza fara sarcina : 2200 rot / minut.

### Bluetooth HC-05 - Figura 2. d)

Modulul bluetooth HC-05 este unul de înaltă performanță și consumă foarte puțină energie. De asemenea, dimensiunile acestuia sunt mici. Produsul poate fi utilizat în proiecte inovative unde este nevoie să fie transmise foarte ușor date, dar nu prin cablu, la distanțe rezonabile, de până la 10m. Acest modul comunică prin intermediul interfeței USART la o rata baud de 9600. De asemenea se pot configura valorile predefinite, folosind comenzi specifice.

### Aplicatia android Camera Stream Wifi

Aplicatia Camera Stream Wifi face posibila transmiterea wireless de imagini video de la smartphone la alte dispozitive. Pentru folosirea aplicatiei este necesara conectarea la o retea wireless. Dupa pornirea aplicatiei pe ecran apare imaginea de la camera video al smartphone-ului. Jos in coltul din stanga apare un ip si port care trebuie introdus in browser-ul diverselor dispozitive conectate la aceeasi retea wireless.

### Descrierea aplicației de prelucrare video

Aplicatia a fost scrisa in limbajul de programare python. Ea foloseste urmatoarele librarii:

```
from collections import deque
import numpy as np
import imutils
import cv2
import serial
import time
```

Libraria *time* este folosită pentru transmiterea de două comenzi pe secundă. Libraria *serial* este folosită pentru comunicarea bluetooth al dispozitivelor. Libraria *numpy* este folosită pentru operații matematice. Libraria principală care este folosită pentru prelucrare de imagini este OpenCV. Preluarea de imagine se face de un IP.

```
camera = cv2.VideoCapture("http://192.168.43.1:8080")
```

Algoritmul *meanshift* este folosit pentru detectarea de grup sau roi de tip de date asemănătoare. Acesta atribuie coordonate datelor detectate și face un contur în jurul lor. Pentru implementarea algoritmului, se localizează fereastra de căutare, cu ajutorul metodei *Meanshift*, în care se selectează componentele de nuanță din spațiul de culori HSV.

```
#culoare albastru inchis pana la albastru deschis  
inchis_blue = np.array([110, 60, 0]) #np.array([90, 60, 0])  
deschis_blue = np.array([121, 255, 255])
```

Următorul pas este calcularea funcției distribuție de probabilitate, prin intermediul unei histogramme de culori. Procesul este iterativ, *Meanshift* se folosește pentru a găsi centrul obiectului din imagine, iar noul punct din centru actualizează fereastra de căutare. Se desenează un contur de culoare galbenă în jurul obiectului și în centru se desenează un punct de culoare albă.

```
frame = imutils.resize(frame, width=600)  
hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)  
mask = cv2.inRange(hsv, inchis_blue, deschis_blue)  
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)  
    if radius > 10:  
        cv2.circle(frame, (int(x), int(y)), int(radius), (0, 255,  
255), 2)  
  
M = cv2.moments(c)  
cx = int(M["m10"] / M["m00"])  
cy = int(M["m01"] / M["m00"])  
cv2.circle(frame, (cx, cy), 7, (255, 255, 255), -1)
```



Figura 3. a) Transmisia de la smartphone

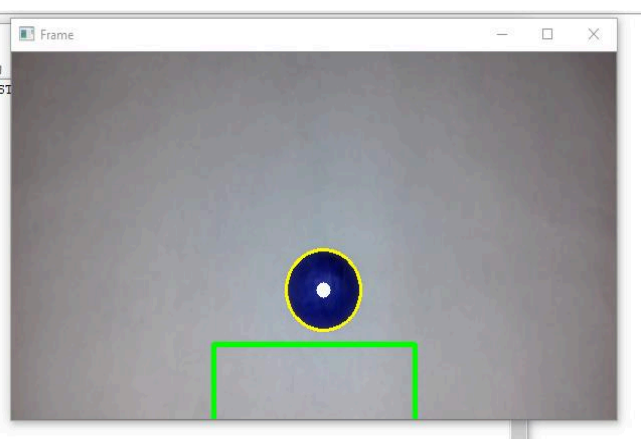


Figura 3. b) Imagine prelucrata

Dupa detectia coordonatelor (x,y) a obiectului se face verificare daca obiectul se afla in regiunea de interes marcata cu verde. Daca se afla in afara lui, se transmit comenzi de deplasare robotului pana cand centrul obiectul detectat se afla in interiorul regiunii de interes. In acest moment se trimite comanda STOP si robotul se opreste.

```
if curr_time - last_recorded_time >= 0.5:#adaugat pentru timp
    if ((cx>MijlocX-100) and (cx<MijlocX+100) and
(cy>MijlocY+100) and cy<MijlocY+220)):
        print("in patrat")
        bluetooth.write(bytes('o'+'\r\n',"utf8"))
        print("stop")
    if (cx<MijlocX-100):
        bluetooth.write(bytes('s'+'\r\n',"utf8"))
        print("stanga")
    elif (cx>MijlocX+100):
        bluetooth.write(bytes('d'+'\r\n',"utf8"))
        print("dreapta")
    elif (cy<MijlocY+100):
        print("sus")
        bluetooth.write(bytes('f'+'\r\n',"utf8"))
```

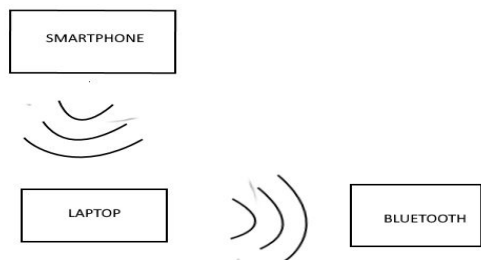


Figura 4. a) Comunicatie dispozitive

```
sus
sus
sus
sus
sus
sus
sus
sus
in patrat
stop
```

Figura 4. b) Date primite in consola Python

### Concluzii

Tehnicile de recunoaștere de imagini digitale folosesc rezultatele și metodele matematice din recunoașterea formelor, inteligența artificială, psiho-fiziologice, știința calculatoarelor, electronica și multe alte discipline științifice. Pentru a simplifica sarcina înțelegerii viziunii computerizate, putem distinge în lanțul algoritmic două nivele: procesarea de nivel scăzut a imaginii și înțelegerea de nivel înalt a imaginii utilizând algoritmi ce duc la un sfârșit scopul propus.

### Referințe

1. <https://www.pyimagesearch.com/2015/09/14/ball-tracking-with-opencv>
2. <https://opencv.org/about>
3. <https://www.youtube.com/watch?v=3tcn496oxnk>
4. <https://lastminuteengineers.com/1298n-dc-stepper-driver-arduino-tutorial>
5. <https://play.google.com/store/apps/details?id=ronakpatel1311.camerastreamer&hl=en>