Universitatea Tehnică "Gheorghe Asachi" din Iași Facultatea de Automatică și Calculatoare Prelucrarea Imaginilor - Proiect

**Prelucrarea imaginilor-Proiect**

**Mașini autonome-detecție pietoni**

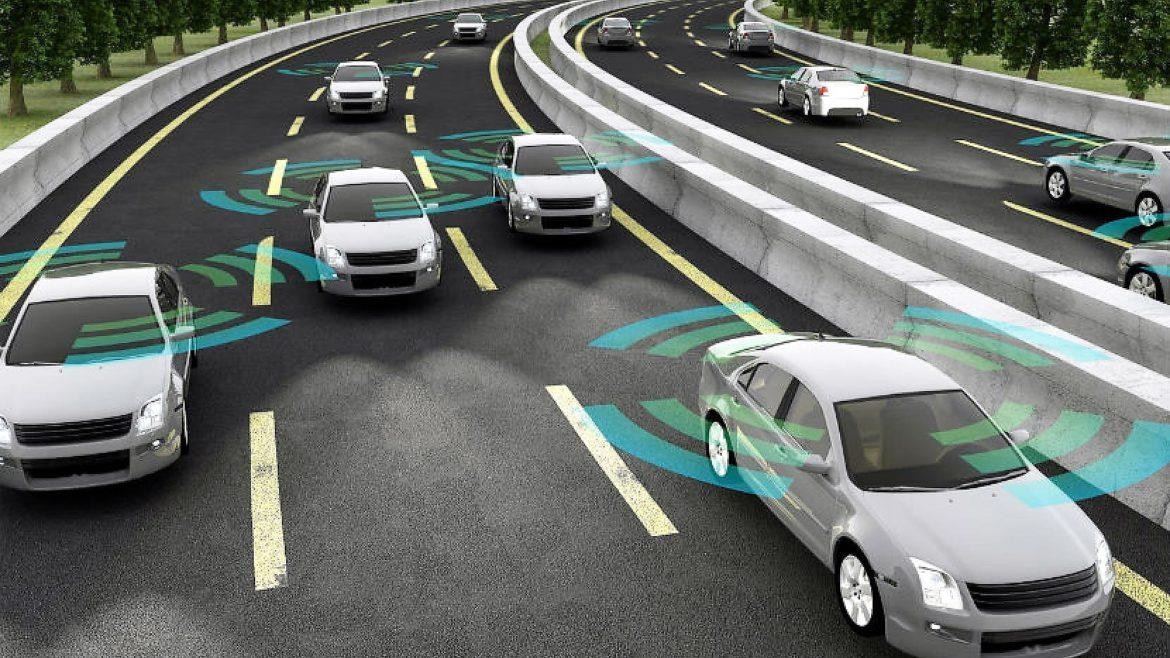
ECHIPĂ:E4

Cristea Vlad-Gabriel Grupa 1306A

Constantin Karina-Florina

Grupa 1306A

## Ce sunt mașinile autonome?



*Engleză: Autonomous Car, Self-driving Car, Autonomous Vehicle, Self Driving vehicle*

**Mașinile autonome** sunt vehicule capabile să execute total funcțiile de rulare, adaptându-se la condițiile de trafic, de infrastructură și de vreme.

Aceste mașini utilizează o gamă largă de senzori, camere, radare și radare laser (Lidar) pentru a-și construi o reprezentare fidelă a mediului înconjurător. În versiunile lor avansate, mașinile autonome pot comunica între ele și pot primi informații dinspre elementele de infrastructură (semafoare, indicatoare, etc).

## De ce o mașină autonomă?

Avantajele majore ale mașinilor autonome sunt reprezentate de potențialul major de limitare a accidentelor, de îmbunătățirea fluidității traficului, de îmbunătățirea calității vieții, de limitarea poluării și de ajutorul oferit persoanelor cu mobilitate redusă.

## Clasificarea mașinilor autonome

Mașinile autonome sunt clasificate în 6 niveluri de autonomie universal recunoscute la nivel mondial:

-**nivel 0**-fără automatizare

-**nivel 1**-asistență pentru șofer

-**nivel 2**-automatizare parțială

-**nivel 3**-automatizare condiționată

-**nivel 4**-automatizare înaltă

-**nivel 5**-automatizare completă

## Detecția pietonilor

**Ce ințelegem prin detecția de pietoni?**

**Câte tipuri de detecție există?**

Detectarea pietonilor (pedestrian detection) este o sarcină esențială și semnificativă în orice sistem inteligent de supraveghere video, deoarece oferă informații fundamentale pentru înțelegerea semantică a înregistrărilor video.

În ciuda provocărilor, detectarea pietonilor rămâne în continuare un domeniu activ de cercetare în viziunea computerizată în ultimii ani. Au fost propuse numeroase abordări:

### -detectare holistică

**-detecție bazată pe piese**

**-detectare bazată pe patch-uri**

**-detecție bazată pe mișcare**

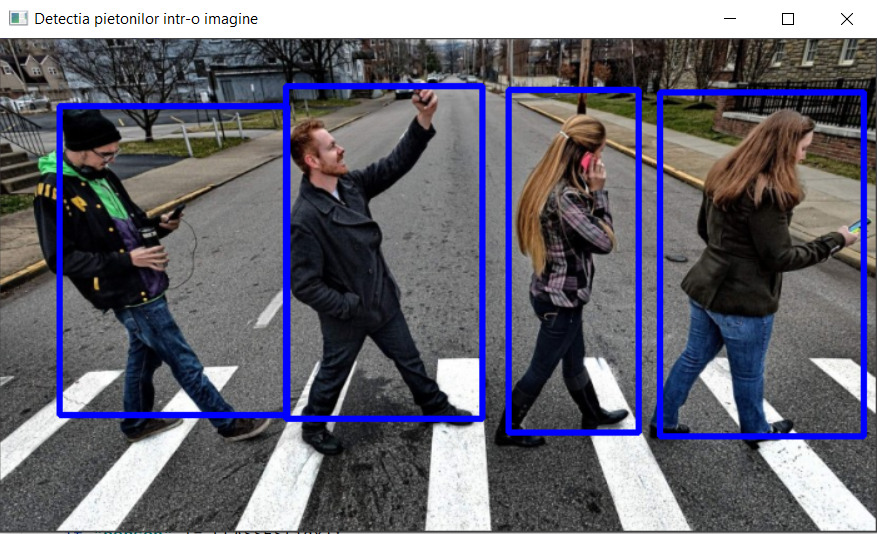
**-detectare folosind mai multe camere**

Pentru acest proiect,noi am ales să abordăm detecția de pietoni bazată pe mișcare.

## La ce se referă detecția bazată pe mișcare?

Detecția pietonilor bazată pe mișcare este un caz special deoarece aceasta se realizează în anumite condiții:utilizând o cameră fixă în condiții bune de iluminare.

Scăderea fundalului clasifică pixelii fluxurilor video fie ca fundal, în care nu este detectată nicio mișcare, fie în prim-plan, unde este detectată mișcarea. Această procedură evidențiază siluetele (componentele conectate din prim-plan) ale fiecărui element în mișcare din scenă, inclusiv al persoanelor.



## Modalitatea de lucru propusă

## Proiectul nostru a fost realizat în python și contine următoarele script-uri:

## - Detecția pietonilor într-un video.py

## - Detecția pietonilor într-o imagine.py

## - Detecția feței într-un video.py

## -Timer.py

## -CentroidTraker.py

## -Tracking-pietoni.py

## Primul script (detectia pietonilor intr-un video) detectează și încadrează persoanele într-un chenar dreptunghiular.Acest lucru a presupus citirea unui video folosind OpenCV. După citire au urmat operații de redimensionare și filtrare a video-ului pentru a obține o imagine mai clară, iar în final detecția propriu-zisă a persoanelor, folosind funcția cv2.dnn.readNetFromCaffe(prototxt=protopath,caffeModel=modelpath) din OpenCV în care:

## protopath= "MobileNetSSD\_deploy.prototxt"

## modelpath = "MobileNetSSD\_deploy.caffemodel"

## Functia cv2.dnn.readNetFromCaffe citește un model de rețea stocat în formatul cadrului Caffe.Parametrii functiei protopath si modelpath reprezinta :

## -calea prototxt către fișierul .prototxt cu descrierea textului arhitecturii rețelei.

## -calea caffeModel către fișierul .caffemodel cu rețeaua învățată.

## Funcția intoarce un obiect de tip net. După aceasta, vom converti ‘img-ul’ nostru intr-un blob astfel: blob=cv2.dnn.blobFromImage(frame, 0.007843, (w, h), 127.5) pe care îl vom trimite către detector : detector.setInput(blob).

## Parametrul ‘personfound‘ conține toate detecțiile gasite pana la momentul actual.Vom parcurge aceste detecții ,iar în momentul în care detecția noastră actuală este de tip ‘person‘(if "person" != CLASSES[idx]:) o vom încadra într-un chenar dreptunghic (cv2.rectangle(frame, (startX, startY), (endX, endY), (255, 255, 0), 2)).

## Calculăm si afișăm numărul de frame-uri pe secundă (FPS):

## FPS = (frames / time.seconds)

## text = "FPS: {:.2f}".format(FPS)

## cv2.putText(frame, text, (5, 30), cv2.FONT\_ITALIC, 1, (0, 255, 255), 2).

## Pe scurt, pasii facuti in detectia persoanelor intr-un video sunt urmatorii:

## -mai intai am citit un video cu ajutorul functiei cv2.VideoCapture()

## -apoi intr-o bucla repetitiva am impartit videoclipul in frame-uri si am aplicat acelasi algoritm ca si la detectia persoanelor intr-o imagine,la final afisand frame-ul cu persoanele marcate in imagine.

## Același algoritm se regăsește la Detecția feței într-un video.

## Detecția pietonilor într-o imagine se realizează în același mod ca și detecția pietonilor intr-un video,fără calculul și afișarea FPS-ului (frames per second).

## Scriptul Timer calculează timpul petrecut de o persoană într-un cadru. Este un bun exemplu de calcul al timpului total în care o persoană a fost prezentă în cadru.

## Centroidtracker: acest script ajută la urmărirea oricărui obiect dintr-un cadru. Am folosit acest lucru în scriptul tracking-pietoni.py pentru a urmări persoanele din cadru.

## Tracking-pietoni: Acest script detectează persoana și continuă să o urmărească în cadru. Atribuie un ID unic fiecărei persoane detectate.

## 

## Task-urile planificate și alocarea acestora pe membrii echipei Identificarea și alocarea task-urilor

## La documentație,primele definiții au fost atașate de mine iar colegul meu s-a axat pe descrierea,în mare,a fiecărui script.

## Pentru partea de detecție,am cautat mai multe video-uri de tip mp4 și le-am asociat fiecărei detecții în parte pentru a evidenția mai bine efectul scripturilor noastre.

## Codul pentru detecții a fost realizat de mine și de colegul meu.

## Pentru partea de tracking,am preluat de pe internet codul NMS(Non-Maximum Suppression) de aici: <https://www.pyimagesearch.com/2015/02/16/faster-non-maximum-suppression-python/>. Acest algoritm ne ajută să scăpăm de detecțiile greșite (zgomot).

### Git repository: https://github.com/karinaflorinac/PIM-P

**Referințe**

C. Papageorgiou and T. Poggio, "A Trainable Pedestrian Detection system", International Journal of Computer Vision (IJCV), pages 1:15–33, 2000

N. Dalal, B. Triggs, “Histograms of oriented gradients for human detection”, IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), pages 1:886–893, 2005

Bo Wu and Ram Nevatia, "Detection of Multiple, Partially Occluded Humans in a Single Image by Bayesian Combination of Edgelet Part Detectors", IEEE International Conference on Computer Vision (ICCV), pages 1:90–97, 2005

Mikolajczyk, K. and Schmid, C. and Zisserman, A. "Human detection based on a probabilistic assembly of robust part detectors", The European Conference on Computer Vision (ECCV), volume 3021/2004, pages 69–82, 2005

Hyunggi Cho, Paul E. Rybski, Aharon Bar-Hillel and Wende Zhang "Real-time Pedestrian Detection with Deformable Part Models"

B.Leibe, E. Seemann, and B. Schiele. "Pedestrian detection in crowded scenes" IEEE Conference on Computer Vision and Pattern Recognition(CVPR), pages 1:878–885, 2005

O. Barnich, S. Jodogne, and M. Van Droogenbroeck. "Robust analysis of silhouettes by morphological size distributions" Advanced Concepts for Intelligent Vision Systems(ACIVS), pages 734–745, 2006

S. Piérard, A. Lejeune, and M. Van Droogenbroeck. "A probabilistic pixel-based approach to detect humans in video streams" IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP), pages 921–924, 2011

S. Piérard, A. Lejeune, and M. Van Droogenbroeck. "3D information is valuable for the detection of humans in video streams" Proceedings of 3D Stereo MEDIA, pages 1–4, 2010

F. Fleuret, J. Berclaz, R. Lengagne and P. Fua, Multi-Camera People Tracking with a Probabilistic Occupancy Map, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 30, Nr. 2, pp. 267–282, February 2008.