

Mașini autonome - detecție pietoni

1st Daniel Ilioi

Facultatea de Automatică

și Calculatoare

Iași, România

daniel.ilioi@student.tuiasi.ro

2nd Savin Damian

Facultatea de Automatică

și Calculatoare

Iași, România

damian.savin@student.tuiasi.ro

Abstract - Scopul proiectului constă în dezvoltarea unui sistem automatizat de detectare a pietonilor, folosind tehnici de viziune computerizată și învățare automată. Am utilizat camera Zed Mini pentru captarea imaginilor și am completat setul de date cu imagini de pe platforma Roboflow. Pentru antrenarea modelului, am folosit rețeaua YOLOv5, inițial cu un model pre-antrenat, apoi am creat propriul model pe Google Colab folosind varianta YOLOv5s. Modelul rezultat permite detectarea pietonilor în timp real și are potențialul de a îmbunătăți siguranța rutieră prin integrarea în sistemele de conducere autonomă.

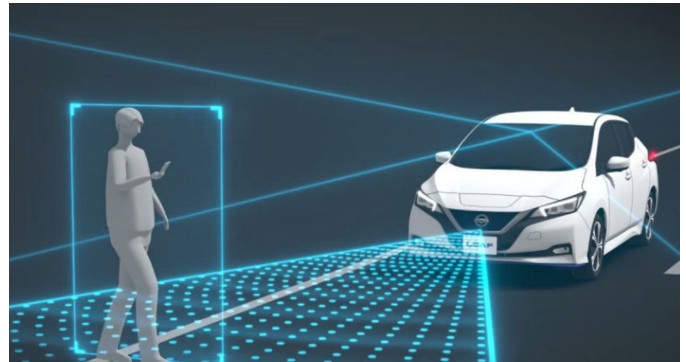


Fig. 1. Detecție pietoni

I. INTRODUCERE

În ultimii ani, societatea și-a îndreptat mai mult atenția și a încercat din ce în ce mai mult să crească siguranța oamenilor, atât în trafic cât și în viața de zi cu zi. Siguranța în trafic este un aspect foarte important, în special în țările dezvoltate sau în curs de dezvoltare. Creșterea constantă a numărului de mașini de pe străzi, atrage cu sine și creșterea riscului de accidente de pe șosele. Conform unei statistici a Uniunii Europene, în anul 2016 accidentele fatale în care sunt implicați pietoni constituie 21% din totalul accidentelor, procentul scăzând cu 35% în ultimii 10 ani. Aceasta scădere poate proveni și din echiparea mașinilor cu sisteme de frânare automată, introduse de companii precum: Lexus, Volvo, Mercedes-Bez și BMW (figura 1). Din acest punct de vedere, posibilitatea de detecție a pietonilor duce la scăderea procentului de pietoni implicați în accidentele rutiere. Detectarea pietonilor în contextul mașinilor autonome reprezintă procesul prin care vehiculele autonome folosesc tehnologii precum camere video, lidar, radar sau senzori AI pentru a identifica și urmări prezența și mișcarea pietonilor în apropierea vehiculului. Această tehnologie este crucială pentru siguranța rutieră, deoarece ajută vehiculele autonome să evite coliziunile cu pietonii și să adapteze comportamentul de conducere pentru a le oferi prioritate acestora. Detectarea pietonilor poate fi realizată prin analiza semnalelor și datelor din senzori pentru a identifica obiectele care corespund unui profil de pieton, iar apoi se pot lua măsuri, cum ar fi frânarea sau evitarea obstacolului, pentru a proteja pietonii de eventuale pericole.

II. UȘURINȚĂ ÎN UTILIZARE

ZED Mini este o cameră stereo care oferă imagini de înaltă definiție și măsoară cu exactitate adâncimea mediului. Aceasta a fost proiectată pentru cele mai dificile aplicații, inclusiv controlul autonom al vehiculului, cartografierea mobilă, cartografierea aeriană, securitatea și supravegherea.

Sistemul de detecție pentru identificarea pietonilor, utilizând o cameră stereo ZED Mini, a fost proiectat având la bază principiile simplității și clarității având doi senzori de imagine pentru a simula stereoviziunea umană. Această abordare permite capturarea unor imagini 3D și estimarea adâncimii într-un mod similar modului în care funcționează ochii umani. Dimensiunile reduse ale camerei, combinate cu rezoluția sa de imagine, permit capturarea detaliilor semnificative în timp real. Dispune de o structură compactă și prietenoasă pentru utilizator, facilitând interacțiunea și asigurând o prezentare directă a informațiilor legate de detectarea pietonilor.

III. STATE OF THE ART ȘI RELATED WORK

A. Detectarea în timp real a pietonilor folosind YOLO

Detectarea în timp real a pietonilor folosind YOLO (You Only Look Once) implică utilizarea unui model de rețea neurală convoluțională (CNN) specializat în detectarea obiectelor. YOLO este cunoscut pentru viteza sa ridicată și precizia rezonabilă, ceea ce îl face potrivit pentru aplicații în timp real, cum ar fi detectarea pietonilor în sisteme de asistență pentru conducere (ADAS), camere de securitate, roboți și alte aplicații unde este necesară detectarea rapidă și eficientă.

În ceea ce privește precizia și viteza, YOLO este extrem de rapid comparativ cu alte metode de detectare, însă acuratețea sa poate fi mai mică decât a altor metode mai lente. Cu privire la scalabilitate, YOLO prezintă avantajul că poate fi extins pentru a detecta alte tipuri de obiecte, nu doar pietoni, prin antrenarea modelului cu alte seturi de date. YOLO este o familie de modele de detectare a obiectelor, fiind renumită pentru performanța sa în timp real. Pe parcursul dezvoltării, au fost lansate diverse versiuni ale YOLO, fiecare aducând îmbunătățiri în ceea ce privește viteza, precizia și eficiența de calcul.

YOLOv2, o versiune îmbunătățită a YOLO (figura 2), este un model de detecție cu performanțe superioare aplicate sarcinilor generale de detecție. Ar putea oferi un compromis destul de bun între viteza și precizie, fiind capabil să depășească tehnicile avansate, cum ar fi R-CNN mai rapid, SSD și așa mai departe, dar încă rulează mai repede decât toate. Rețeaua YOLOv2 integrează extragerea casetelor candidate, extragerea caracteristicilor, clasificarea ținta și locația țintă într-o singură rețea profundă. Acest lucru permite antrenamentul end-to-end și transformă problema tradițională de detectare într-o problemă de regresie. Pentru a realiza o detecție eficientă și precisă a pietonilor, introducem detectorul general, YOLOv2, ca și cadru de bază al modelului nostru de detectare a pietonilor și apoi să facă unele modificări în structura și parametrii rețelei, adaptându-se mai bine pentru pieton. YOLO rămâne, totuși, o soluție pentru operațiuni în timp real datorită vitezei sale mai mari de detecție în comparație cu alte arhitecturi.

Name	Year	Type	Dataset	mAP	Inference rate (fps)
R-CNN	2014	2D	Pascal VOC	66%	0.02
Fast R-CNN	2015		Pascal VOC	68.80%	0.5
Faster R-CNN	2016		COCO	78.90%	7
YOLOv1	2016		Pascal VOC	63.40%	45
YOLOv2	2016		Pascal VOC	78.60%	67
SSD	2016		Pascal VOC	74.30%	59
RetinaNet	2018		COCO	61.10%	90
YOLOv3	2018		COCO	44.30%	95.2
YOLOv4	2020		COCO	65.70%	62
YOLOv5	2021		COCO	56.40%	140
YOLOR	2021	3D	COCO	74.30%	30
YOLOX	2021		COCO	51.20%	57.8
Complex-YOLO	2018		KITTI	64.00%	50.4
Complexer-YOLO	2019		KITTI	49.44%	100
Wen et al.	2021		KITTI	73.76%	17.8
RAANet	2021		NuScenes	62.00%	

Fig. 2. Comparație între diferite modele 2D-3D de detectare a obiectelor

YOLOv3 a reprezentat un salt semnificativ față de versiunile anterioare în ceea ce privește performanța și flexibilitatea. Este cunoscut pentru echilibrul său între viteză și acuratețe. Este alegerea de bază pentru proiecte unde se dorește un echilibru între viteză și acuratețe, dar fără a necesita cea mai recentă tehnologie.

YOLOv4 reprezintă o opțiune mai bună pentru aplicații unde acuratețea suplimentară este importantă, însă fără a sacrifica viteza.

YOLOv5 este recomandat pentru majoritatea aplicațiilor moderne, datorită integrării excelente cu PyTorch și performanțelor optimizate.

YOLOv9 conferă o mai bună acuratețe cu mai puține resurse, cercetătorii din spatele YOLOv9 concentrându-se pe îmbunătățirea eficienței pentru a obține performanță în timp real pe o gamă mai largă de dispozitive. Au fost introduse două inovații cheie: un nou model de arhitectură numit Rețea generală eficientă de agregare a straturilor (GELAN) care maximizează acuratețea minimizând parametrii și FLOP-urile; și o tehnică de antrenament numită Informații programabile de gradient (PGI) care oferă gradientele de învățare mai fiabile, în special pentru modelele mai mici.

B. Rețelele neuronale convoluționale (ConvNets) pentru detectarea pietonilor

Rețelele neuronale convoluționale (ConvNets) sunt folosite pe scară largă pentru detectarea pietonilor datorită abilității lor de a extrage automat caracteristici complexe din imagini și de a clasifica obiectele cu o precizie mare. Aceste rețele, o ramură a deep learning-ului, sunt deosebit de eficiente în prelucrarea datelor vizuale, precum imaginile și videoclipurile. ConvNets sunt compuse din multiple straturi, fiecare având un rol distinct în procesarea și interpretarea datelor.

În primul rând, straturile convoluționale sunt cele care aplică filtre (sau kerneluri) pe imagini în vederea identificării unor caracteristici locale, cum ar fi muchiile, colțurile și texturile. Pe măsură ce rețeaua progresează, filtrele devin capabile să detecteze caracteristici mai complexe și abstracte, cum ar fi formele și modelele specifice pietonilor.

În al doilea rând, straturile de pooling reduc dimensiunea spațială a caracteristicilor extrase, menținând esențialul informațiilor, ceea ce contribuie la reducerea complexității și a timpului de procesare, făcând modelul mai eficient.

În altă ordine de idei, straturile complet conectate sunt cele care, la finalul rețelei, combină toate caracteristicile extrase pentru a realiza predicția finală, determinând dacă regiunile imaginii conțin sau nu pietoni.

Detectarea pietonilor cu ConvNets implică două etape principale: localizarea și clasificarea. Prima etapă constă în generarea de propuneri de regiuni candidate (bounding boxes) care pot conține pietoni, în vederea identificării de regiuni care conțin pietoni. În ceea ce privește a doua etapă, pentru fiecare regiune propusă, ConvNet-ul clasifică dacă regiunea conține sau nu un pieton. Aceasta implică o trecere suplimentară prin rețea pentru a determina probabilitatea ca regiunea să conțină un pieton.

Există mai multe arhitecturi pentru detectarea pietonilor, printre care: Faster R-CNN, YOLO și SSD. Prima integrează un model ConvNet pentru extragerea caracteristicilor cu o "Region Proposal Network" (RPN), care sugerează regiuni candidate pentru detectarea pietonilor. Faster R-CNN este renumit pentru precizia sa înaltă, deși poate fi mai lent comparativ cu alte metode.

YOLO utilizează un singur model ConvNet pentru a efectua atât localizarea, cât și clasificarea într-o singură trecere prin

rețea, ceea ce îl face extrem de rapid și ideal pentru aplicații în timp real.

SSD (Single Shot Multibox Detector), similar cu YOLO, folosește mai multe straturi de detecție pentru a îmbunătăți acuratețea la diferite scări de obiecte. Este, de asemenea, foarte rapid și eficient în detectarea pietonilor.

ConvNets prezintă o serie de avantaje, unul dintre acestea fiind precizia, întrucât ele pot învăța și recunoaște caracteristici complexe ale pietonilor, oferind o acuratețe ridicată. ConvNets au o capacitate de generalizare, putând fi antrenate pe seturi mari de date și astfel pot generaliza bine pe imagini noi.

Un alt avantaj este viteza, întrucât anumite arhitecturi, cum ar fi YOLO și SSD, permit detectarea pietonilor în timp real, ceea ce este esențial pentru aplicațiile de siguranță rutieră și supraveghere.

ConvNets prezintă și o serie de provocări, deoarece necesită o putere mare de calcul, în special pentru antrenare, ceea ce poate limita utilizarea lor pe dispozitive cu resurse limitate. De asemenea, performanța ConvNets depinde de calitatea și diversitatea setului de date de antrenament. Datele inadecvate pot duce la rezultate nesatisfăcătoare în detectarea pietonilor.

IV. SETUL DE DATE

A. Achiziția de date

Setul de date pentru detectarea de pietoni a fost creat prin capturarea imaginilor în diverse condiții de iluminare, de vreme și mediu urban. Imaginile cuprind scene variate, precum intersecții aglomerate, treceri de pietoni, precum și condiții diferite de vreme, inclusiv zile însorite și zile ploioase, și diferite momente ale zilei, respectiv în timpul zilei și în timpul nopții. Fotografiele au fost realizate din diferite unghiuri și distanțe, simulând astfel situații realiste întâlnite de sistemele de detecție a pietonilor în medii urbane. Astfel, setul nostru de date cuprinde un număr de aproximativ 2000 de imagini, dintre care mai puțin de un sfert din numărul de imagini reprezintă un set propriu de date, creat prin capturarea de imagini cu ajutorul camerei Zed Mini. Restul de imagini din setul de date au fost preluate de pe platforma Roboflow, cunoscută pentru furnizarea de date pentru antrenarea modelelor de viziune computerizată. Aceste imagini au fost clonate din următoarele proiecte: 'aldydagy_kishiler' (utilizator 'person-cffpe') aproximativ 1400 de imagini, și din alte două proiecte intitulate 'person', aparținând utilizatorilor 'iot-7qo9b' și 'yaen', aproximativ 300 de imagini. Astfel, se oferă o varietate de scene urbane capturate în diverse condiții de iluminare, vreme și unghiuri, relevante pentru detectarea pietonilor. Dintre toate aceste imagini, 70% vor fi utilizate pentru antrenarea modelului, 20% pentru validare, iar restul de 10% pentru testare.

B. Prelucrarea datelor

Cu ajutorul platformei Roboflow, am aplicat tehnici de augmentare a datelor și preprocesare pentru a îmbunătăți acest set de date pentru detectarea de pietoni. Acest lucru a presupus redimensionarea, rescalarea, remodelarea, dar mai ales adnotarea imaginilor, în special aplicarea a așa numitelor

bounding boxes (casete de delimitare) (figura 3). Aceste casete

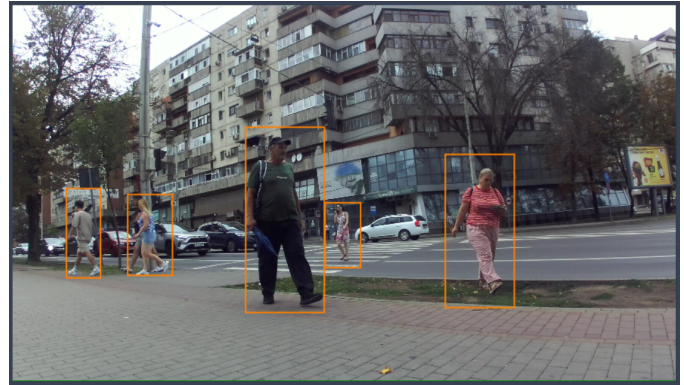


Fig. 3. Exemplu de aplicare a bounding boxes

sunt dreptunghiuri trasate în jurul obiectelor din imagini, în cazul nostru pietonii, indicând astfel locația și dimensiunea aproximativă a obiectului pe care modelul trebuie să îl recunoască. În plus, am ajustat parametrii de luminozitate și contrast pentru a îmbunătăți setul de date, contribuind astfel la antrenarea modelului nostru de detecție a pietonilor.

V. DESCRIEREA METODEI

Proiectul nostru de detecție a pietonilor combină tehnologia avansată a camerei ZED Mini cu algoritmi de detecție de obiecte bazate pe deep learning (YOLOv5), pentru a crea un sistem capabil să detecteze pietoni în timp real, în diverse condiții de iluminare, de vreme și mediu urban.

SDK-ul ZED utilizează inteligența artificială și rețele neuronale pentru a determina ce obiecte sunt prezente atât în imaginile din stânga, cât și în cele din dreapta, camera fiind dotată cu doi senzori de imagine. SDK-ul calculează apoi poziția 3D a fiecărui obiect, precum și caseta de delimitare a acestora, utilizând date din modulul de adâncime. Obiectele pot fi, de asemenea, urmărite în mediul înconjurător în timp, chiar dacă camera este în mișcare, datorită datelor din modulul de urmărire pozițională. SDK-ul ZED detectează toate obiectele prezente în imagini și calculează poziția și viteza lor 3D. Distanța obiectului față de cameră este exprimată în unități metrice (ex. metri) și calculată din partea din spate a senzorului stâng al camerei până la obiectul scenei.

Pe lângă utilizarea SDK-ului ZED pentru calcularea poziției și urmărirea obiectelor, am implementat un model de detecție bazat pe YOLOv5. YOLOv5s reprezintă versiunea "small" a YOLOv5, fiind optimizată pentru viteză și eficiență, cu un număr redus de parametri și operații de calcul comparativ cu alte versiuni ale YOLOv5, precum YOLOv5m (medium), YOLOv5l (large), sau YOLOv5x (extra-large). (figura 4).

YOLOv5s este o variantă populară și performantă a familiei de modele YOLO, specializată în detectarea obiectelor în imagini. YOLOv5s este un detector de obiecte cu o singură etapă, ceea ce înseamnă că detectează și clasifică obiectele într-o singură trecere prin rețeaua neuronală. Spre deosebire de alte metode cu două etape cum ar fi R-CNN, YOLOv5s nu






				
Nano YOLOv5n	Small YOLOv5s	Medium YOLOv5m	Large YOLOv5l	XLarge YOLOv5x
4 MB _{FP16} 6.3 ms _{V100} 28.4 mAP _{COCO}	14 MB _{FP16} 6.4 ms _{V100} 37.2 mAP _{COCO}	41 MB _{FP16} 8.2 ms _{V100} 45.2 mAP _{COCO}	89 MB _{FP16} 10.1 ms _{V100} 48.8 mAP _{COCO}	166 MB _{FP16} 12.1 ms _{V100} 50.7 mAP _{COCO}

Fig. 4. Comparație între versiunile YOLOv5

generează propuneri de regiuni de interes înainte de clasificare. Mai mult decât atât, YOLOv5s prezintă și numeroase avantaje precum viteza, fiind unul dintre cele mai rapide detectoare de obiecte, acuratețea, reușind să ofere o precizie ridicată, și flexibilitatea, care prezintă suport adaptării la diferite rezoluții de imagine.

A. Arhitectura YOLOv5s

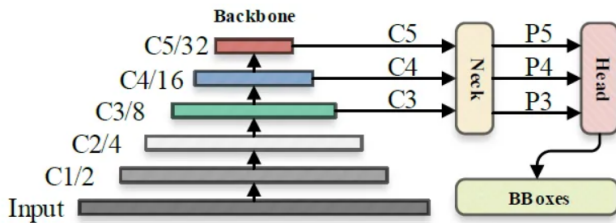


Fig. 5. Arhitectura YOLOv5

Arhitectura este împărțită în trei componente principale: Backbone (rețeaua de bază care extrage caracteristici importante din imagine), Neck (componenta care combină caracteristicile importante extrase la diferite niveluri de granularitate) și Head (componenta care face predicțiile de detectare a obiectelor). Stratul de intrare procesează imaginea care este trimisă către backbone pentru a extrage caracteristicile esențiale (figura 5). Astfel, rețeaua de bază (backbone) generează hărți de caracteristici de diferite dimensiuni, care sunt ulterior combinate de rețeaua de fuziune a caracteristicilor (neck) pentru a crea trei hărți de caracteristici finale: P3, P4 și P5, cu dimensiuni de 80x80, 40x40 și 20x20. Aceste hărți fiind folosite pentru a detecta obiecte de dimensiuni diferite în imagine. Ulterior, aceste hărți de caracteristici fiind transmise către componenta care face predicțiile de detectare (head), unde se realizează operațiile necesare, rezultatul fiind un tablou multidimensional (BBboxes) care conține diferite informații precum coordonatele cadrului, înălțimea și lățimea acestuia. Astfel, prin aplicarea unor praguri specifice pentru a elimina informațiile irelevante se obține informația finală de detectare.

B. Antrenarea modelului

Procesul de antrenare a modelului presupune finalizarea unui număr fixat de epoci (epochs) sau atingerea unui prag

de performanță dorit. Fiecare epocă constă în antrenarea modelului pe întregul set de date, eșantionat în volume egale (batch size). Inițial, am utilizat modelul de antrenare disponibil pe platforma Roboflow, bazat pe YOLOv5s. Am creat o nouă clasă numită "persoană" pentru a reprezenta bounding box-urile din imagini (figura 6), configurând astfel modelul pentru a recunoaște și localiza pietonii în imagini. Utilizând această platformă am generat o versiune preliminară a modelului nostru, adaptată specific la setul respectiv de date. Prin accesul la resurse optimizate și setări preconfigurate am reușit să accelerăm procesul de antrenare.

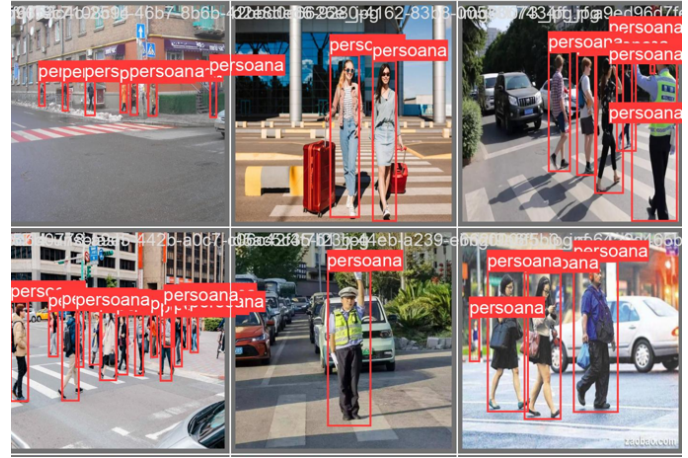


Fig. 6. Clasa persoana reprezentând bounding boxes

Pentru a obține îmbunătățiri în ceea ce privește performanța modelului, am continuat prin antrenarea unui model personalizat utilizând Google Colab, o platformă care oferă resurse GPU pentru a accelera procesul de antrenare. Astfel, utilizând datele etichetate pentru a antrena modelul, am implementat un script Python, implicând ajustarea hiperparamterilor, evaluarea performanței și iterarea asupra modelului pentru a îmbunătăți acuratețea.

VI. PREZENTAREA SUCCINTĂ A PROTOTIPULUI

Prototipul nostru de detectare a pietonilor constă într-un sistem care îmbină antrenarea modelului de detecție, achiziția datelor și implementarea unui cod eficient din punct de vedere al detecției persoanelor. Codul Python este proiectat pentru a oferi mai multe funcționalități esențiale. În primul rând, permite preluarea modelului de antrenare și aplicarea acestuia pentru a face predicții precise. De asemenea, integrează camera Zed Mini pentru a captura imagini în timp real, facilitând astfel obținerea datelor direct din sursa de imagine. Codul include și funcționalități pentru procesarea videoclipurilor și imaginilor, permițând manipularea atât a imaginilor, cât și a secvențelor video. În final, salvarea rezultatelor este gestionată în mod eficient, garantând că toate datele sunt conservate și disponibile pentru analiza ulterioară. Aceste funcționalități sunt implementate în următoarele funcții principale:

- "procesareFrame": pentru preluarea fiecărui cadru dintr-un flux video sau dintr-o sursă de imagini, aplicând modelul de detecție și returnând rezultatele detectării
- "zedYolo": pentru capturarea și procesarea imaginilor de la camera Zed
- "videoYolo": pentru procesarea videoclipurilor existente
- "imageYolo": pentru procesarea imaginilor statice
- "selectareTipFisier": pentru selecția fișierelor
- "salvare": pentru salvarea rezultatelor

Prin utilizarea camerei stereo ZED Mini și a modelului de învățare profundă YOLOv5, prototipul nostru oferă o soluție eficientă pentru detectarea pietonilor în medii complexe. Performanțele sale îl recomandă pentru o gamă largă de aplicații, de la vehicule autonome la sisteme inteligente de supraveghere.

VII. REZULTATE PRELIMINARE

În această primă etapă de antrenare a modelului am utilizat un set de date mai restrâns, un număr de 338 de imagini, față de setul de date curent. De asemenea, am folosit modelul de antrenare disponibil pe platforma Roboflow, bazat pe YOLOv5s. După cum se poate vizualiza și în figura 7, se

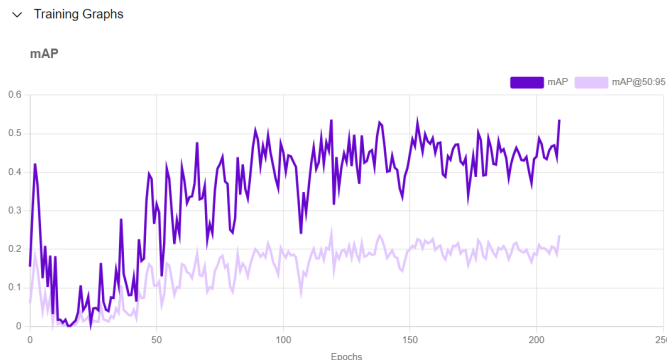


Fig. 7. Rezultate preliminare - mAP

obține o medie a preciziei medii (mAP) a modelului de detecție de aproximativ 53% pentru un număr de 209 epoci. De asemenea, celelalte metrici esențiale pentru evaluarea performanței modelului de detecție a persoanelor, precizia și recall-ul se pot observa în figura 8, reieșind o medie de 64%, respectiv 50%.

VIII. CONCLUZII PRELIMINARE

Rezultatele preliminare obținute în urma experimentelor indică o performanță moderată a modelului de detecție. Cu media preciziei medii (mAP) de doar 53%, se poate afirma că modelul are capacitate limitată de a localiza și a clasifica pietonii din imagini. Precizia de 64%, arată că aproximativ două treimi din totalul detecțiilor realizate sunt adevărate, iar restul de o treime reprezintă faptul că modelul a identificat în mod eronat pietonii. Recall-ul de 50%, indică faptul că modelul pierde multe obiecte relevante (false negative), ceea ce poate fi decisivă în ceea ce privește aplicațiile de siguranță.

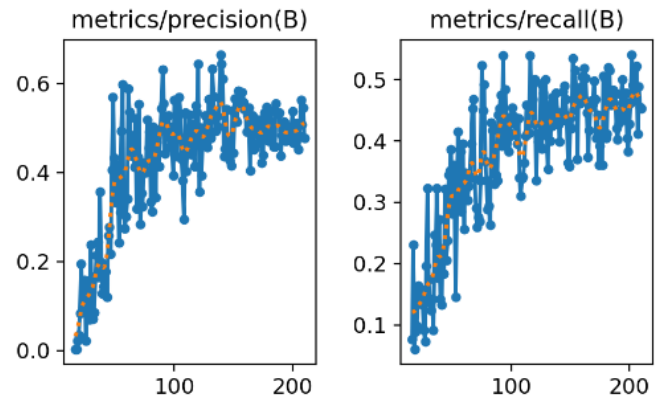


Fig. 8. Rezultate preliminare - precizie și recall

Prin urmare, există oportunități de îmbunătățire a acestor metrici, prin mărirea setului de date și prin crearea unui model personalizat de antrenare.

IX. REZULTATE FINALE

Pentru continuarea realizării detecției de pietoni, am trecut print-o multitudine de abordări pentru a ajunge la rezultatul final. În această etapă, am adus setul de date la un număr semnificativ mai mare față de setul de date anterior, aproximativ 2000 de imagini. Mai mult decât atât, am creat un model personalizat de antrenare a modelului cu scopul de a obține performanțe ridicate. Astfel, se poate observa o optimizare semnificativă a metricilor de evaluare a modelului, mAP crescând de la 53% la 90% (figura 9) pentru un număr de 121 epoci.

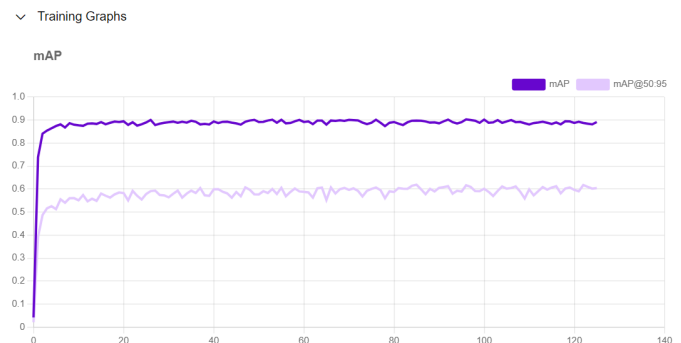


Fig. 9. Rezultate finale - mAP

De asemenea, se constată și creșterea în ceea ce privește precizia și recall-ul la 92%, respectiv 80% (figura 10).

X. CONCLUZII FINALE

Rezultatele finale ale proiectului, cu mAP de 90%, precizia de 92% și recall-ul de 80%, indică o remediere semnificativă a performanței modelului de detecție. O valoare ridicată a preciziei indică faptul că modelul generează foarte puține alarme false, de asemenea, un recall de 80% sugerează că modelul reușește să identifice o bună majoritate a pietonilor

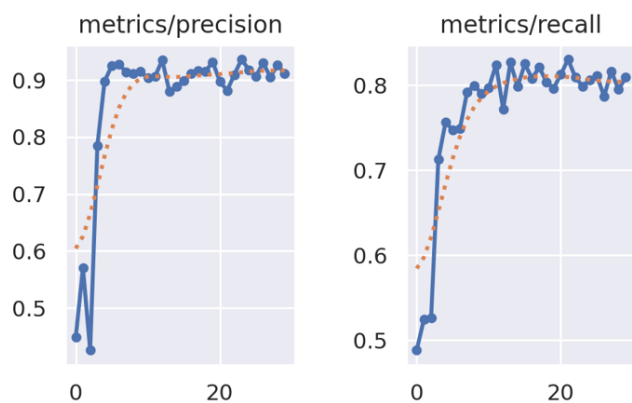


Fig. 10. Rezultate finale - precizie și recall

din imagini, astfel se minimizează cazurile de detecții eșuate. Cu aceste rezultate, proiectul își atinge scopul propus de a dezvolta un sistem de detectare a pietonilor eficient și fiabil. În concluzie, finalizarea unui proiect de detectare a pietonilor implică evaluarea performanței, rezolvarea problemelor, colectarea feedback-ului și documentarea adecvată a întregului proces. Este, de asemenea, un moment potrivit pentru a privi spre viitor și a identifica oportunitățile de îmbunătățire și dezvoltare ulterioară.

REFERENCES

- [1] V. Harisankar, R. Karthika, Real Time Pedestrian Detection Using Modified YOLO V2
- [2] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, Piotr Dollár, Focal Loss for Dense Object Detection
- [3] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified real-time object detection, 2015.
- [4] Joseph Redmon, Ali Farhadi, YOLOv3: An Incremental Improvement, <https://arxiv.org/abs/1804.02767>
- [5] Shanshan Zhang, Rodrigo Benenson, Bernt Schiele, CityPersons: A Diverse Dataset for Pedestrian Detection
- [6] GitHub: <https://github.com/ultralytics/yolov5>
- [7] Documentation: <https://docs.ultralytics.com/yolov5/>
- [8] RoboFlow: <https://app.roboflow.com/zed-mini/zed-mini/3>