

# Mașină inteligentă Raspberry Pi cu recunoaștere în timp real a semnelor și semnalelor de trafic

Robert-Gabriel Gaube  
Dep. Calculatoare și Tehnologia  
Informației  
Universitatea Politehnică Timișoara  
Timișoara, România  
robert-gabriel.gaube@student.upt.ro

Patricia-Maria-Francesca Fermandjîn  
Dep. Calculatoare și Tehnologia  
Informației  
Universitatea Politehnică Timișoara  
Timișoara, România  
patricia.fermandjin@student.upt.ro

Marius-Andrei Murariu  
Dep. Calculatoare și Tehnologia  
Informației  
Universitatea Politehnică Timișoara  
Timișoara, România  
marius.murariu@student.upt.ro

**Abstract**—Această lucrare prezintă un sistem automatizat de procesare a imaginii montat pe o mașină inteligentă, proiectat pentru a recunoaște semnele și semnalele de trafic utilizând un Raspberry Pi și o cameră atașată. Mașina va lua măsurile corespunzătoare la detectarea diferitelor semne de trafic.

**Keywords**— *Smart car, Raspberry Pi, YOLOv8, real-time detection, traffic signs, autonomous vehicles.*

## I. INTRODUCERE

Vehiculele autonome devin din ce în ce mai mult un obiectiv al cercetării și dezvoltării datorită potențialului lor de a revoluționa sistemele de transport. Un component critic al acestor sisteme este capacitatea de a recunoaște și de a reacționa la semnele și semnalele de trafic. Acest proiect își propune să dezvolte un sistem automatizat de procesare a imaginilor montat pe o mașină inteligentă pentru a recunoaște semnele și semnalele de trafic folosind un Raspberry Pi și o cameră atașată. Mașina va lua măsurile necesare la întâlnirea diferitelor semne suportate, cum ar fi oprirea la semnul „Stop” sau la culoarea roșie a semaforului, sau ajustarea vitezei pe baza semnelor de limitare a vitezei.

## II. ANALIZA DE DOMENIU

Mașinile inteligente sunt autonome în sensul că sunt prevăzute cu opțiuni precum: recunoașterea mediului înconjurător, luarea rapidă și la timp a deciziilor, navigarea fără intervenția umană pe drum, menținerea unor modele de mobilitate sigure, efectuarea tuturor tipurilor de manevre și controlul vitezei de croazieră etc. Apariția și evoluția mașinilor autonome sunt rezultatul unor cercetări remarcabile provenite din domeniile comunicațiilor wireless, sistemelor încorporate, navigației, tehnologiilor de senzori și rețele ad-hoc, achiziției și diseminării datelor, și analizei datelor [1].

În zilele noastre, numărul de vehicule care se folosesc de sistemele de asistență și siguranță pentru șoferi (ADAS) este în continuă creștere, amplificând autonomia unei mașini în ceea ce privește funcțiile de conducere. Aceste sisteme variază prin ajutorul acordat șoferului, având diferite grade de autonomie (semi-autonome: transmisie automată, respectiv autonome: active cruise control) [2].

Ideea mașinilor autonome a început cu "mașinile fantomă" în anii 1920, unde mașina era controlată printr-un dispozitiv de telecomandă [3]. În anii 1980, am asistat la apariția mașinilor autonome autosuficiente și auto-gestionate. Un contribuitor major la domeniul mașinilor autonome a fost NavLab de la Universitatea Carnegie Mellon, unde cercetătorii au dezvoltat Vehiculul Autonom Terestru (ALV) [4]. În același deceniu, proiectul "Prometheus", sponsorizat de Mercedes în 1987 [5], a realizat un rezultat major cu designul primei lor mașini robotice capabile să urmărească marcasele de bandă și alte vehicule (cu toate acestea, din motive de

siguranță, era necesară intervenția umană). Deși nu era complet autonomă la acea vreme, abilitatea de a schimba automat benzile a fost o descoperire majoră. În secolul 21, interesul crescut pentru mașinile autonome a fost alimentat în principal de tehnologiile de înaltă performanță și de costurile reduse din diverse domenii.

Pe parcursul timpului au existat diferite metode de detectare și segmentare a imaginilor, câteva dintre acestea sunt abordate în următoarele paragrafe.

O tehnică flexibilă bazată pe un model de amestec gaussian cu o metodă automată de divizare și fuziune a fost introdusă ca un prim pas pentru a simplifica recunoașterea semnelor de circulație, în special în condiții degradate din lumea reală. Se recomandă utilizarea unui CNN (rețea neuronală convoluțională) pe mai multe scări cu reducerea dimensionalității pentru a determina semnificația simbolurilor semnelor de circulație [6].

A fost propusă o abordare Single Shot Detector (SSD) care combină și optimizează mai multe caracteristici pentru a îmbunătăți recunoașterea semnelor de circulație, în special pentru obiectele mici. Această metodă combină caracteristicile de nivel scăzut și de nivel înalt pentru a îmbunătăți performanța de detectare [7][8].

Cercetătorii au investigat ulterior diverse metode de extragere a datelor și algoritmi de învățare automată pentru a învăța cum să distingă în mod obișnuit activitățile și intențiile pietonilor în traficul urban [9].

A fost creată o rețea nouă de detectare și recunoaștere a semnelor de trafic (TSDR) care utilizează o operațiune multiscalară și o convoluție separată în funcție de adâncime pentru a îmbunătăți viteza, precizia și generalizabilitatea [10].

## III. DETECȚIE ÎN TIMP REAL A SEMNELOR ȘI SEMNALIZĂRILOR DE TRAFIC

Detecția în timp real a semnalelor de trafic se referă la capacitatea unui sistem de a identifica și localiza în mod automat semnalele de trafic într-un flux video în direct sau într-o intrare de cameră. Aceasta este o capacitate importantă pentru diverse aplicații, cum ar fi:

Monitorizarea și gestionarea traficului: Detectarea în timp real a semnalelor de trafic poate fi utilizată în cadrul sistemelor de transport inteligente pentru a monitoriza fluxul de trafic și pentru a optimiza sincronizarea semafoarelor, reducând congestia și îmbunătățind eficiența generală.

Principalele abordări pentru detectarea semnalelor de trafic în timp real implică, de obicei, utilizarea viziunii computerizate și a algoritmilor de învățare profundă. Aceste metode utilizează adesea tehnici precum detectarea obiectelor, segmentarea și clasificarea imaginilor pentru a identifica

prezența, locația și starea (de exemplu, roșu, galben, verde) semnalelor de trafic în datele de intrare.

Unele dintre provocările cheie în detectarea semnalelor de trafic în timp real includ:

Condiții de mediu variabile: Semafoarele de trafic trebuie detectate cu precizie în diferite condiții de iluminare, condiții meteorologice și unghiuri ale camerei.

Ocluzie și dezordine: Semnalele de trafic pot fi parțial ascunse de alte obiecte, cum ar fi vehicule, pietoni sau alte infrastructuri, ceea ce face ca detectarea să fie mai dificilă.

Performanță în timp real: Algoritmul de detectare trebuie să funcționeze la o frecvență de cadre ridicată pentru a furniza informații în timp util aplicației, cum ar fi ADAS sau vehiculele autonome.

Generalizare: Modelele de detecție trebuie să poată generaliza bine la diferite modele de semnale de trafic, locații și configurații rutiere.

#### A. Computer Vision

Computer Vision este un domeniu al inteligenței artificiale (AI) care permite calculatoarelor să proceseze intrări vizuale, cum ar fi imagini digitale și videoclipuri. Acest lucru este realizat prin analizarea automată a unor cantități mari de date, recunoașterea anumitor tipare și distincții și, în final, obținerea unui model capabil să extragă și să clasifice diverse caracteristici.

Capacitatea modelului de a distinge diferite obiecte este obținută prin tehnologii precum învățarea profundă (un subset al învățării automate) și rețele neuronale convoluționale.

O rețea neuronală convoluțională (CNN) este un algoritm de învățare profundă care primește o imagine ca intrare și diferențiază diferite aspecte ale imaginii atribuind grade de importanță prin greutatea și bias-uri învățabile. Acest proces implică, de obicei, mai multe straturi de convoluție care identifică dependențele dintre pixeli pentru a crea treptat reprezentări mai semnificative ale datelor. O CNN poate la început să traducă pixelii în linii, care sunt apoi combinate pentru a forma caracteristici, cum ar fi vizorul unui semafor, și în final combinate pentru a crea capul de semnal mai complex.

#### B. YOLO

YOLO (You Only Look Once) este un model de învățare profundă bazat pe rețele neuronale convoluționale (CNN) proiectat pentru detectarea obiectelor în timp real în aplicațiile de viziune computerizată. Practic, o rețea convoluțională prezice simultan multiple casete de delimitare și probabilități de clasă pentru acele casete. De asemenea, YOLO se antrenează pe imagini întregi și optimizează direct performanța de detectare.

YOLOv8 încorporează diverse metode de detectare a obiectelor, inclusiv clasificare, detectare de obiecte și segmentare a imaginilor, oferind informații la nivel de pixel despre fiecare obiect. Arhitectura modelului cuprinde un backbone, un neck și un head. Backbone-ul se concentrează pe extragerea caracteristicilor, neck-ul agregă și rafinează caracteristicile, în timp ce head-ul produce casetele de delimitare, probabilitățile de clasă și scorurile de încredere.

Arhitectura YOLOv8 (Fig. 1) utilizează o rețea de piramidă de caracteristici pentru a reduce treptat rezoluția

spațială a imaginii de intrare în timp ce crește numărul de canale de caracteristici, permițând detectarea obiectelor de diferite dimensiuni și rezoluții.

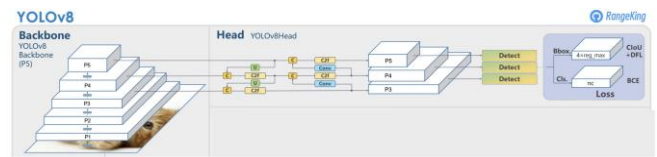


Fig. 1. Arhitectura YOLOv8.

#### C. Antrenarea modelului

Scopul aplicației noastre de viziune computerizată este de a identifica corect semnele și semnalele de trafic într-un flux video colectat în timp real de la camera Raspberry Pi. Pentru a satisface aceste cerințe, am antrenat un model personalizat de învățare profundă, obținând astfel o soluție fiabilă de detectare a semnelor rutiere folosind YOLOv8.

Setul de date pe care am antrenat și testat modelul este disponibil la <https://shorturl.at/fuvBL> și constă din 5771 imagini și etichetele asociate acestora. Pentru scopuri de antrenare, setul de date a fost împărțit în următorul mod: 3520 imagini pentru antrenare (61,15%), 1603 imagini pentru validare (27,77%) și 638 imagini pentru testare (6,37%), câteva dintre acestea sunt prezentate în figura 2.

După o analiză comparativă a performanței modelului în diferite scenarii, am obținut o rată satisfăcătoare de precizie prin antrenarea modelului în 100 de epoci cu loturi de 64 de imagini. În plus, hiperparametrii determinați experimental sunt o rată de învățare inițială de 0.0001 și o rată de abandon de 15%. Modelul YOLOv8 rezultat este compus din 168 de straturi și conține 3.008.573 de parametri.

Obiectele recunoscute în prezent de model sunt: lumină roșie, lumină verde, semn de oprire, limită de viteză 10 km/h, limită de viteză 20 km/h, limită de viteză 30 km/h, limită de viteză 40 km/h, limită de viteză 50 km/h, limită de viteză 60 km/h, limită de viteză 70 km/h, limită de viteză 80 km/h, limită de viteză 90 km/h, limită de viteză 100 km/h, limită de viteză 110 km/h și limită de viteză 120 km/h.



Fig. 2. Imagini extrase din setul de antrenare.

#### D. Testarea modelului

Pentru a evalua eficacitatea modelului nostru de învățare profundă în detectarea semnelor și semnalizărilor de trafic în timp real, am efectuat o serie de teste riguroase folosind un set de date diversificat. Scopul principal al acestei secțiuni este de a prezenta metodologia de testare, de a analiza performanța modelului în scopul evaluării preciziei și fiabilitatea modelului.

Pe baza rezultatelor testării, precizia generală a modelului este de 96,1213%. În figura 3 putem observa câteva metrici folosite pentru a măsura performanța modelului pe parcursul antrenării:

- **loss\_bbox** (Bounding Box Error) – o funcție de pierdere care măsoară în considerare de "strânse" sunt bounding boxes prezise față de obiectul de referință.
- **loss\_cls** (Cross Entropy Loss) - o funcție de pierdere care măsoară corectitudinea clasificării fiecărei casete de delimitare prezise.
- **dfl\_loss** (Dual Focal Loss) - o funcție de pierdere care combină două componente: clasificarea focală, care se concentrează pe exemplele dificile, și localizarea focală, care măsoară precizia casetelor de delimitare.
- **mAP** (Mean Average Precision) - este egală cu media preciziei medii pentru fiecare clasă care aparține modelului.
- **IoU (Intersection over Union)** – cuantifică gradul de suprapunere între două cutii de delimitare: una reprezentând „adevărul de bază” (locația reală a unui obiect) și cealaltă reprezentând „predicția” modelului pentru același obiect. Acesta măsoară cât de bine se aliniază un obiect prezis cu adnotarea reală a obiectului. Un scor IoU mai mare implică o predicție mai precisă.

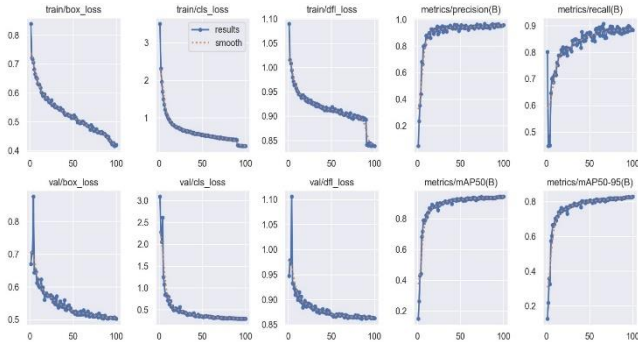


Fig. 3. Valorile de precizie și pierdere ale modelului pe seturile de antrenare și validare în timpul antrenării modelului.

Măsurile de performanță bazate pe matricea de confuzie sunt utilizate frecvent pentru a evalua performanța modelelor. Matricea de confuzie este o reprezentare tabelară a rezultatelor de clasificare prezise și reale și include patru metrici: Adevărat Pozitiv (TP), Fals Pozitiv (FP), Adevărat Negativ (TN) și Fals Negativ (FN).

**Acuratețea:** Acuratețea este raportul dintre numărul de predicții corecte și numărul total de predicții realizate de model, prezentată în ecuația 1.

$$\text{Acuratețe} = \frac{(TP+TN)}{(TP+FP+TN+FN)} \quad (1)$$

**Precizia** este raportul dintre adevăratele pozitive și numărul total de predicții optimiste realizate de model, prezentată în ecuația 2.

$$\text{Precizie} = \frac{TP}{(TP+FP)} \quad (2)$$

**Recall-ul** este raportul dintre numărul de adevărate pozitive și numărul total de instanțe pozitive reale, prezentată în ecuația 3.

$$\text{Recall} = \frac{TP}{(TP+FN)} \quad (3)$$

**Curba de încredere F1**, prevăzută în figura 4, este o reprezentare grafică a performanței unui model de învățare automată, în special în contextul clasificării binare sau multi-clasă. Aceasta curba ilustrează relația dintre două măsuri esențiale de performanță: precizia și recall-ul, combinate într-o singură metrică denumită scorul F1, prezentată în ecuația 4.

$$\text{Scor F1} = 2 * \frac{(\text{precizie} * \text{recall})}{(\text{precizie} + \text{recall})} \quad (4)$$

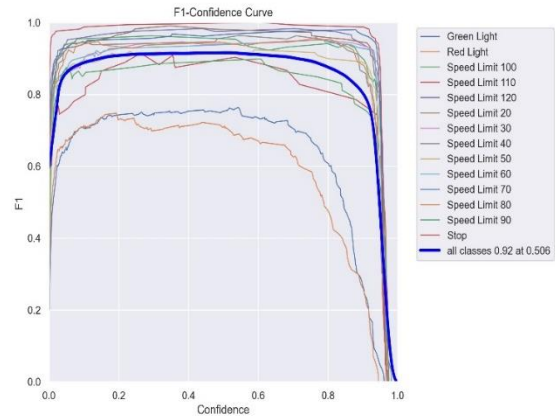


Fig. 4. Curba de încredere F1. Reprezentare grafică a performanței unui model ilustrând relația dintre 2 metrici importante: recall și precizie.

**Curba Precizie-Încredere**, prezentată în figura 5, este o reprezentare grafică utilizată pentru a evalua performanța unui model de clasificare. Aceasta ilustrează relația dintre precizia modelului și scorul de încredere la diferite praguri.

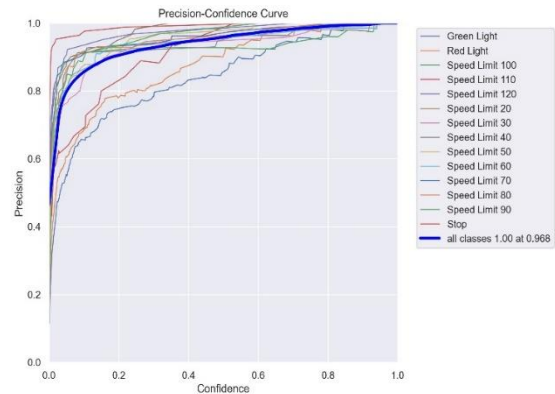


Fig. 5. Curba precizie-încredere. Reprezentare grafică a performanței unui model ilustrând relația dintre 2 metrici importante: precizia și scorul de încredere.



#### IV. DISPOZITIVUL EXPERIMENTAL

Mașina inteligentă Raspberry Pi este compusă dintr-un șasiu cu două straturi, 4 roți, 4 motoare DC, 2 drivere de motor cu punte H duală L298N, un suport pentru 4 baterii AA, un Raspberry Pi 4B, o cameră Raspberry Pi V2 de 8 megapixeli și un power bank de 10000 mAh după cum se observă în figura 6.

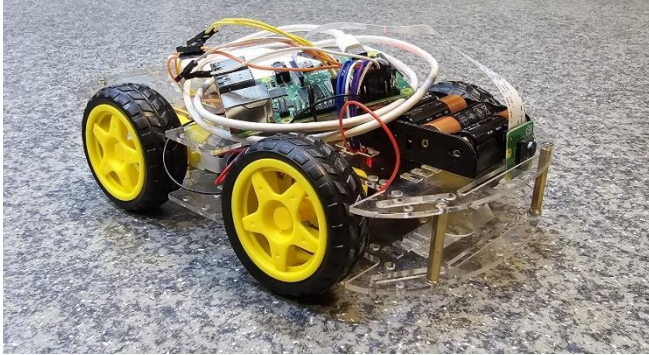


Fig. 6. Mașina inteligentă.

În schema din figura 7 putem observa schema sistemului:

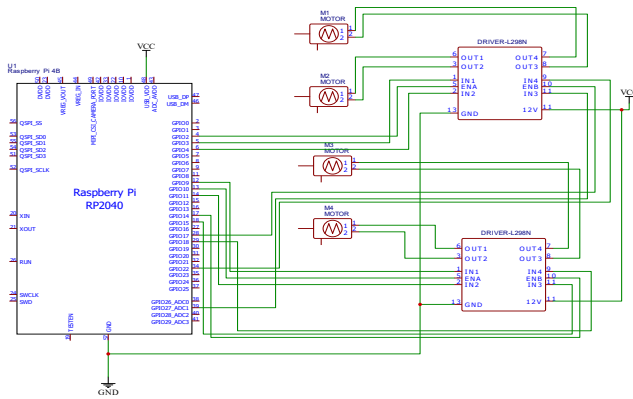


Fig. 7. Schema hardware a sistemului. Microcontroller-ul Raspberry Pi 4B este conectat la 2 punți L298n care controlează direcția de mișcare a motoarelor, fiecare dintre acestea din urmă este conectată la 2 motoare de curent continuu prevăzute cu o cutie de viteze de plastic.

Pentru fiecare semn de trafic sau culoare a semaforului recunoscută, un mesaj corespunzător va fi afișat în terminal și se va efectua acțiunea adecvată:

- Semnul Stop – Mașina inteligentă se va opri definitiv la întâlnirea acestui semn. Prevăzut în scenariul de test prezentat în figura 8.
- Semnul Limitare 50 km/h – Mașina inteligentă va reduce viteza la întâlnirea acestui semn.
- Semnul Limitare 70 km/h - Mașina inteligentă va crește viteza la întâlnirea acestui semn. Prevăzut în scenariul de test prezentat în figura 8.
- Semafor de culoare roșie – Mașina inteligentă se va opri și va aștepta apariția culorii verzi pentru a-și continua deplasarea.
- Semafor de culoare verde – Mașina inteligentă își va continua deplasarea la întâlnirea acestui semnal.



Fig. 8. Testarea sistemului. Mașina inteligentă detectează semnele de circulație (Limitare de viteză 70 km/h și Stop).

Codul sursă aferent antrenării și testării modelului, precum și a operării mașinii inteligente este disponibil la: <https://github.com/MurariuMarius/SCSS-Smart-Car.git>.

#### V. CONCLUZII

Sistemul creat de noi, bazat pe modelul inovator de învățare profundă YOLOv8, detectează cu succes semnele și semnalele de circulație în timp real, rezultând un produs sigur și fiabil.

În continuare există posibilitatea extinderii sistemului pentru a detecta alți participanți la trafic precum pietoni, alte vehicule, animale, obstacole de altă natură și a marcajelor de pe carosabil.

#### REFERENCES

- [1] R. Hussain and S. Zeadally, "Autonomous Cars: Research Results, Issues, and Future Challenges," in *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 1275-1313, Secondquarter 2019.
- [2] Rödel, C., Stadler, S., Meschtscherjakov, A., & Tscheligi, M. (2014). "Towards Autonomous Cars: The Effect of Autonomy Levels on Acceptance and User Experience". Proceedings of the 6th International Conference on Automotive User Interfaces and Interactive Vehicular Applications.
- [3] A. Lafrance, "our grandmother's driverless car." <https://www.theatlantic.com/technology/archive/2016/06/beep-beep/489029/>, 2016.
- [4] T. Kanade, C. Thorpe, and W. Whittaker, "Autonomous land vehicle project at cmu," in Proceedings of the 1986 ACM Fourteenth Annual Conference on Computer Science, CSC '86, (New York, NY, USA), pp. 71-80, ACM, 1986.
- [5] J. Schmidhuber, "Robot car history." <http://people.idsia.ch/~juergen/robotcars.html>.
- [6] A. Mannan, et al., Classification of degraded traffic signs using flexible mixture model and transfer learning, IEEe Access 7 (2019) 148800-148813.
- [7] Y. Jin, et al., Multi-feature fusion and enhancement single shot detector for traffic sign recognition, IEEe Access. 8 (2020) 38931-38940.
- [8] C. Gamez ´ Serna, Y Ruichek, Classification of traffic signs: the european dataset, IEEe Access. 6 (2018) 78136-78148.
- [9] D. Varytimidis, et al., Action and intention recognition of pedestrians in urban traffic, in: Proceedings of the 2018 14th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), 2018, pp. 676-682.
- [10] X. Bangquan, W. Xiao Xiong, Real-time embedded traffic sign recognition using efficient convolutional neural network, IEEe Access. 7 (2019) 53330 - 53346.
- [11] J. Redmon, S. Divvala, R. Girshick and A. Farhadi. You Only Look Once: Unified, Real-Time Object Detection, IEEE Xplore, 2016.

- [12] D. Reis, J. Kupec, J. Hong, A. Daoudi, Real-Time Flying Object Detection with YOLOv8, Journal of Imaging, 2023.
- [13] S. Saha, A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way, Medium, 2018. Available: <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>. [Accessed: 22/04/2024]
- [14] What is computer vision?, IBM. Available: <https://www.ibm.com/topics/computer-vision>. [Accessed: 22/04/2024]
- [15] J. Sawtell-Rickson, What Is Computer Vision, BuiltIn, 2022. Available: <https://builtin.com/machine-learning/computer-vision>. [Accessed: 22/04/2024]
- [16] Under the Hood: YOLOv8 Architecture Explained, Keylabs, 2023. Available: <https://keylabs.ai/blog/under-the-hood-yolov8architecture-explained> [Accessed: 22/04/2024]