**ROMÂNIA**

**MINISTERUL APĂRĂRII NAȚIONALE**

**ACADEMIA TEHNICĂ MILITARĂ ,,FERDINAND I”**

**FACULTATEA DE SISTEME ELECTRONICE ŞI INFORMATICE MILITARE**

**Specializare: Comunicații pentru apărare și securitate**



**SISTEM AUTONOM DE DEPLASARE PE BAZA RECUNOAȘTERII SEMNELOR DE CIRCULAȚIE CU RASPBERRY PI**

CONDUCĂTOR ȘTIINȚIFIC:

**Conf. univ. dr. ing Florin Roman ENACHE**

ABSOLVENT:

**Sd. sg. maj. Adriana-Victorița MIU**

Conţine \_\_\_\_\_\_\_\_\_\_\_ file

Inventariat sub nr. \_\_\_\_\_\_

Poziţia din indicator: \_\_\_\_

Termen de păstrare: \_\_\_\_\_

**BUCUREŞTI**

**2022**

# ABSTRACT

# REZUMAT

Acest proiect de diplomă își propune realizarea unui sistem autonom care să se deplaseze pe un traseu bine delimitat pe baza semnelor de circulație și demonstrarea posibilității de concretizare a unui software capabil să detecteze și să realizeze recunoașterea acestora în timp util. Software-ul este apt în a fi rulat pe sisteme embedded datorită dimensiunii mici și a vitezei mari,menținând în parametri optimi precizia detecției si a prelucrării imaginilor în timp real.

De-a lungul acestei lucrări sunt explicate bazele realizării proiectului, cât și procesele ce au condus la implementarea și configurarea software-ului și a compoziției hardware .

Limbajul de programare majoritar utilizat în cadrul proiectului este Python, un limbaj de nivel înalt, datorită dinamicității si a capabilității sale de a dezvolta aplicații de tip *Machine Learning*, sisteme capabile să învețe prin experiență în urma aplicării unor algoritmi.

Primul capitol are un rol introductiv și prezintă tema și scopul proiectului alături de noțiunile generale despre inteligența artificială și subdomeniul acesteia utilizat în cadrul proiectului, Deep Learning, precum și impactul atins de AI în dezvoltarea mașinilor autonome.

Al doilea capitol aprofundează domeniul Deep Learning-ului, în speță a învățarii supravegheate. Astfel, acest capitol detaliază teoretic algoritmii, arhitectura rețelei neuronale și procesul de obținere a unui model apt sa realizeze detecția și recunoașterea semnelor rutiere. Sunt discutate modul de funcționare al rețelelor neuronale convoluționale și procesele care sunt vizate în vederea antrenării unui model de detecție a obiectelor.

Al treielea capitol are rolul de a prezenta implementarea software și implementarea hardware ale proiectului. Implementarea software implică mediul de lucru, framework-ul de DL și structuralizarea arhitecturii rețelei neuronale utilizate și limitările impuse de factorii de mai sus. Impelementarea hardware urmărește procesul de realizare al platformei, componența acesteia, legăturile fizice între modulele externe și platforma de dezvoltare și limitările la nivel hardware. Sunt prezentate, de asemenea , și detalii despre senzorii de distanță, senzorii cu infraroșu, modul de lucru și schemele lor electrice

Al patrulea capitol are un rol experimental, prezentând rezultatele practice obținute și concluziile trase în urma acestora. Totodată, în acest capitol se realizeaza un debug general asupra întregului sistem realizat și se evidențiază limitările și problemele întălnite și posibilele viitoare îmbunătățiri ale acestora.

# CUPRINS

[ABSTRACT 5](#_Toc95937710)

[REZUMAT 5](#_Toc95937711)

[CUPRINS 7](#_Toc95937712)

[LISTĂ DE ABREVIERI 9](#_Toc95937713)

[TABELĂ FIGURI 10](#_Toc95937714)

[1. OBIECTIVE ȘI NOȚIUNI GENERALE 11](#_Toc95937715)

[1.1 Obiective,motivația alegerii temei și domeniile de aplicație 11](#_Toc95937716)

[1.2 Noțiuni generale 11](#_Toc95937717)

[1.2.1 Deep Supervised Learning (DSL) 14](#_Toc95937718)

[1.2.2 Deep Unsupervised Learning (DUL) 15](#_Toc95937719)

[1.3 Abordarea Deep Learning 16](#_Toc95937720)

[1.4 Provocări impuse de DL 17](#_Toc95937721)

[1.5 Deep Learning în comparație cu Machine Learning 17](#_Toc95937722)

[1.6 Deep Learning în contextul automatizării autovehiculelor 17](#_Toc95937723)

[1.6.1 Detecția drumului 17](#_Toc95937724)

[1.6.2 Navigarea 17](#_Toc95937725)

[1.6.3 Detecția și recunoașterea semnelor rutiere și a pietonilor 17](#_Toc95937726)

[2 NOȚIUNI TEORETICE 17](#_Toc95937727)

[2.1 Definiții 17](#_Toc95937728)

[2.2 Procesul de învățare 17](#_Toc95937729)

[2.2.1 Funcția de activare (activation function) 17](#_Toc95937730)

[2.2.2 Antrenarea 17](#_Toc95937731)

[2.2.3 Funcția de pierderi (loss function) 17](#_Toc95937732)

[2.2.4 Parametrii și hiperparametrii rețelelor neuronale convoluționale. Probleme impuse în urma modificării acestora 17](#_Toc95937733)

[2.3 Rețelele neuronale convoluționale (CNN) 17](#_Toc95937734)

[2.3.1 Arhitectura 17](#_Toc95937735)

[2.3.2 Convoluția 17](#_Toc95937736)

[2.3.3 Pooling 17](#_Toc95937737)

[2.3.4 Fully-Connected 17](#_Toc95937738)

[2.4 Modelul de antrenare ales 17](#_Toc95937739)

[2.5 Modalități de antrenare 17](#_Toc95937740)

[2.5.1 Fine Tuning 17](#_Toc95937741)

[2.5.2 Transfer Learning 17](#_Toc95937742)

[2.5.3 Learning from scratch 17](#_Toc95937743)

[2.6 Procesarea imaginilor în DL 17](#_Toc95937744)

[2.6.1 Adnotarea 17](#_Toc95937745)

[2.6.2 Augumentarea 17](#_Toc95937746)

[3 METODE DE IMPLEMENTAREA ȘI VERIFICARE 17](#_Toc95937747)

[3.1 Implementarea software 17](#_Toc95937748)

[3.1.1 Mediul de dezvoltare, limbajul de programare și librării utilizate 17](#_Toc95937749)

[3.1.2 Construirea setului de date 17](#_Toc95937750)

[3.1.3 Implementarea CNN pe baza principiului de Transfer Learning 17](#_Toc95937751)

[3.1.4 Modelul SSD MobileNet v2 320x320 17](#_Toc95937752)

[3.1.5 Procesorul grafic Nvidia 18](#_Toc95937753)

[3.1.6 CPU și GPU. Limitări impuse de placa grafică. 18](#_Toc95937754)

[3.1.7 Coral USB Accelerator (TPU Edge) 18](#_Toc95937755)

[3.1.8 Tensorflow. Framework-ul TFLite 18](#_Toc95937756)

[3.1.9 Tensorboard 18](#_Toc95937757)

[3.1.10 OpenCV 18](#_Toc95937758)

[3.1.11 Probleme de implementare și soluții 18](#_Toc95937759)

[3.2 Implementare hardware 18](#_Toc95937760)

[3.2.1 Platforma computațională Raspberry Pi 4B 18](#_Toc95937761)

[3.2.2 Driver-ul DRV8835 Dual Motor pentru Raspberry Pi 18](#_Toc95937762)

[3.2.3 Modulul cameră v2 Raspberry Pi 18](#_Toc95937763)

[3.2.4 Probleme hardware de implementare și soluții 18](#_Toc95937764)

[4 REZULTATE EXPERIMENTALE 18](#_Toc95937765)

[4.1 Prezentarea evoluției sistemului autonom analizat 18](#_Toc95937766)

[4.2 Rezultate obținute 18](#_Toc95937767)

[4.3 Analiza performanțelor sistemului obținut 18](#_Toc95937768)

[5 CONCLUZII ȘI PERSPECTIVE DE VIITOR 18](#_Toc95937769)

[6 BIBLIOGRAFIE 18](#_Toc95937770)

[7 ANEXE 18](#_Toc95937771)

# LISTĂ DE ABREVIERI

|  |  |  |
| --- | --- | --- |
|  | AV | Autonom Vechicle |
|  | AI | Artificial Intelligence |
|  | ADAS | Advanced Driver Assitance Systems |
|  | SVM | Support Vector Machine |
|  | ANN | Artificial Neural Network |
|  | DNN | Deep Neural Network |
|  | DL | Deep Learning |
|  | CV | Computer Vision |
|  | LiDAR | Light Detection And Ranging |
|  | RADARs | Radio Detection and Ranging |
|  | DRL | Deep Reinforcement Learning |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# TABELĂ FIGURI

# 1. OBIECTIVE ȘI NOȚIUNI GENERALE

## Obiective,motivația alegerii temei și domeniile de aplicație

Prezenta lucrare are ca obiectiv principal realizarea cu ajutorul Raspberry Pi a unui dispozitiv autonom ce se deplasează pe baza semnelor de circulație pe o machetă ce simulează un drum real.

Motivația pentru alegerea acestui proiect de diplomă îsi are originea în aspirația de a studia și de a aprofunda domeniul inteligenței artificiale, în speță aria rețelelor neuronale, cât și evoluția exponențială a cercetărilor în zona self-drivingului.

Conducerea autonomă este unul dintre vastele domenii de aplicare ale inteligenței artificiale și unul dintre cele mai complexe proiecte derulate la momentul actual. Companiile care dezvoltă sisteme AV se bazează majoritar pe AI, sub forma de deep learning și machine learning, pentru a procesa o cantitate mare de date în mod eficient și pentru a antrena și valida sistemele de conducere autonomă. Companii precum: Telsa, Cruise, Alphabet Inc’s, Waymo sau Aurora Innovation Inc dezvoltă tehnologii ce permit automatizarea completă a vehiculelor.

Totodată, lucrarea de față vizează modalități de aplicare a modeului de Deep Learning (“învățare profundă”) pe dispozitive cu resurse și costuri reduse, precum placa de dezvoltare Raspberry Pi, demonstrând astfel utilizarea facilă, disponibilitatea de lucru și o performanță bună spre foarte bună.

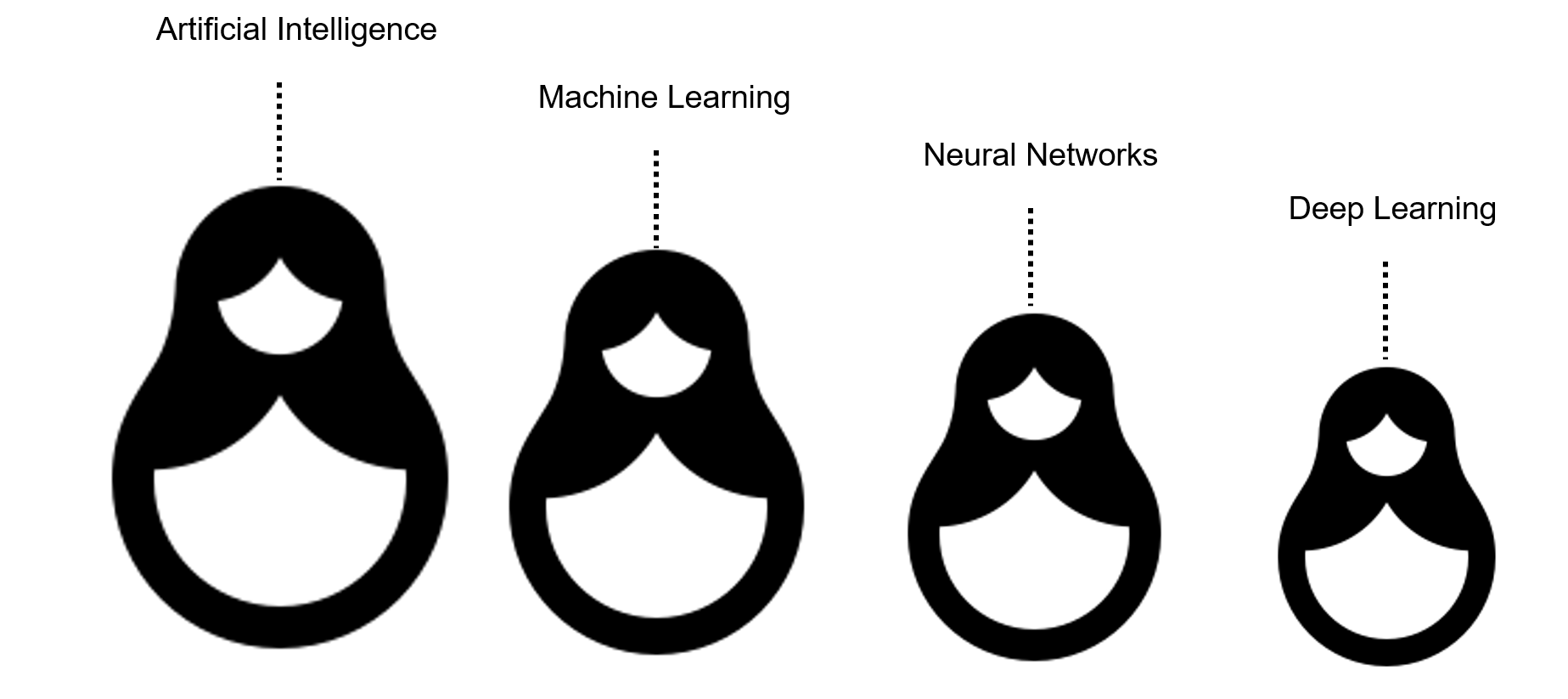
## Noțiuni generale

Siguranța este un aspect important al conducerii autonome. Factorul uman rămâne cea mai frecventă cauză a accidentelor rutiere. Astfel introducerea vechiculelor autonome va reduce considerabil o mare parte din aceste probleme, desigur cu un compromis reprezentat de o marjă infimă de risc. În cadrul dezvoltării vehiculelor autonome, în speță a sistemelor de asistență avansată în conducere (ADAS), inteligența artificială a câștigat mult teren, astfel cercetările s-au orientat în direcția exploatării informațiilor vizuale datorită utilității acesteia pentru detectarea semnelor rutiere, a drumurilor, a celorlalte vechicule și a pietonilor. Un mare interes îl prezintă sistemele de recunoaștere a semnelor rutiere care vizează detecția semnelor, interpretarea acestora și transmisia informațiilor către șofer sau în cazul dorit, transmisia informațiilor către vechicul care efectuează execuția fără să aibe nevoie de o decizie umană. Odată cu dezvoltarea domeniului Deep Learning, utlimii algoritmi și arhitecturi utilizate în dezvoltarea sistemelor de recunoaștere a semnelor rutiere vizează rețelele artificiale neuronale, precum SSD MobileNet V2 și Faster-RCNN abordate și în cadrul acestui proiect, acestea din urmă utilizează principiul rețelelor neuronale convoluționare.

În această lucrare, întregul sistem de recunoașterea a semnelor rutiere va fi împărțit într-o arii de lucru: dezvoltarea software, în speță zona ce presupune dezvoltarea modelului de recunoaștere a semnelor rutiere și implementarea hardware a sistemului mobil ce presupune asamblarea și programarea motoarelor, a senzorilor și a camerei video. Modelul de recunoaștere a semnelor rutiere este, de altfel, abordat în 2 etape, prima etapă o reprezintă detecția semnelor dintr-o imagine sau un frame al fluxului video, iar etapa a doua reprezintă procesul de clasificare în urma căruia semnul detectat în prima etapă este clasificat într-una din clasele de referință definite în dataset.

Deși inteligența artificială(AI), învățarea profundă (DL) și machine learning sunt folosite concomitent, acestea prezintă concepte și arii de interes diferite înrudite între ele. În cei mai simpli termeni, AI este o ramură a informaticii care acoperă procedeele de dezvoltare a dispozitivelor inteligente.

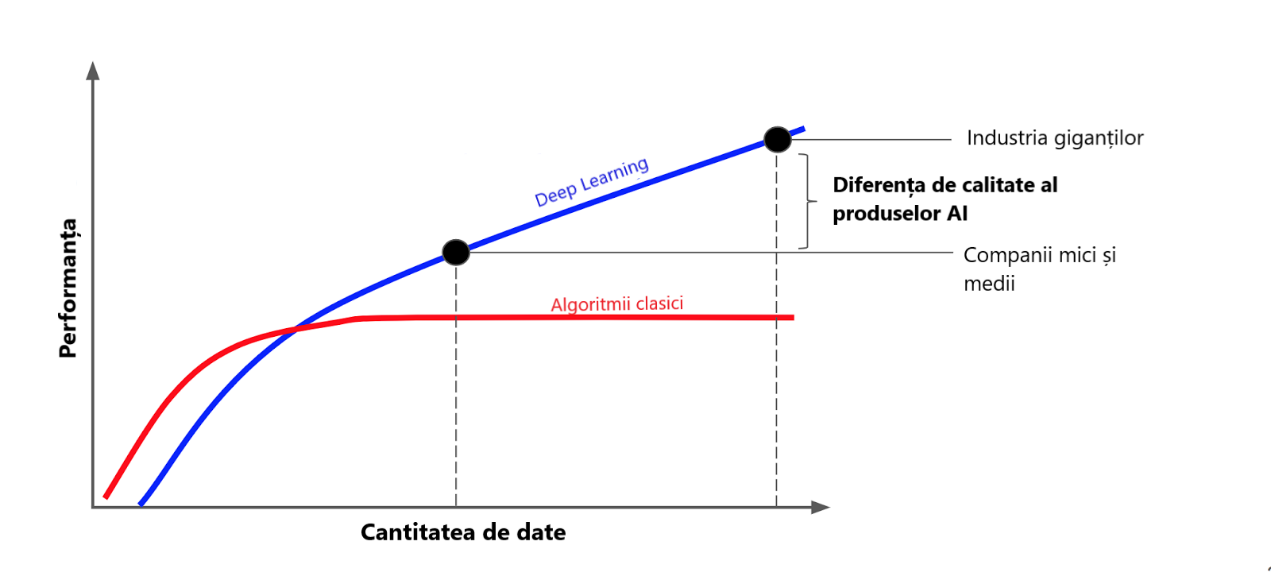
Probabil cel mai simplu mod de a ne gândi la inteligența artificială, Deep Learning și Machine Learning este, conform unui articol postat de IBM, vizualizarea ierarhică a acestora ca în cazul păpușilor rusești Matryoshka, fiecare fiind în esență o componentă a domeniului anterior.[1]



Structura ierarhică a subdoemeniilor inteligenței artificiale

Machine Learning este un subdomeniu al AI care oferă sistemelor capacitatea de a învăța și de a se îmbunătăți automat din experiență, fără a fi programate în mod explicit. Mașina este antrenată folosind cantități mari de date și algoritmi care îi oferă capacitatea de a învăța cum să îndeplinească sarcina. Abordările în domeniul Machine Learning includ învățarea arborelui de decizie, programarea logică inductivă, clustering, învățarea prin consolidare și rețelele bayesiene.

Deep Learning introduce o abordare extrem de sofisticată a învățării automate, fiind modelat pe baza rețelelor neuronale profunde. Acestea din urmă sunt inspirate din înțelegerea noastră a biologiei creierului uman cu toate interconexiunile între neuroni, dar spre desosebire de creierul uman în care orice neuron este conectat cu alt neuron, rețelele neuronale au straturi, conexiuni și direcții de propagare a datelor discrete. Astfel, DL a reprezentat un pas important în evoluția sistemelor automate și în general în domeniul inteligenței artificiale.



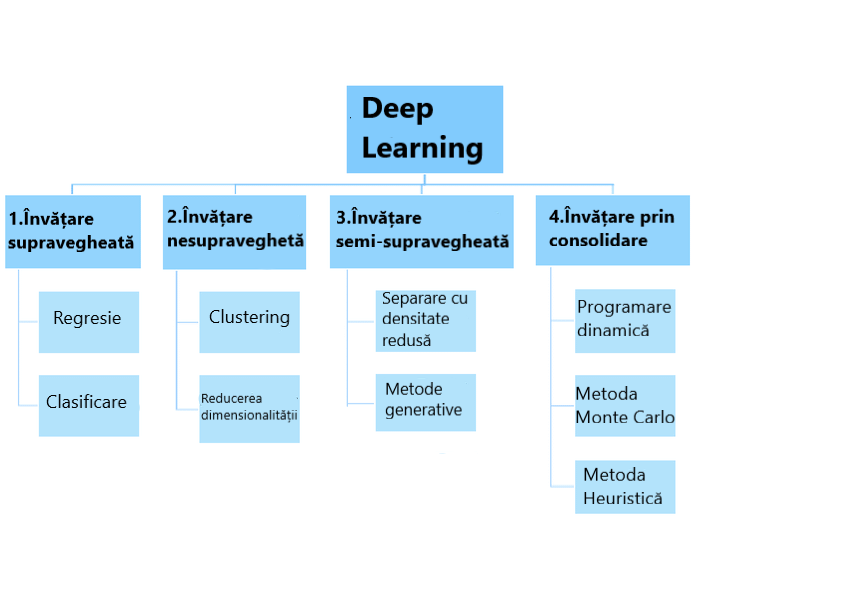
Performanțele DL în comparație cu abordările clasice ale ML

Graficul din figura 1.3 a fost realizat în urma unui studiu asupra comparației între metodele clasice de învățare în machine learning și domeniul deep learning.Se poate observa că pentru o cantitate mică de date algoritmii clasici prezintă o mai bună perfomanță în detrimentul deep learning, dar cum tehnologia evoluează exponențial și cantitatea de date utilizată crește proporțional, astfel domeniul de deep learning a câștigat mult teren în favoarea performanțelor pe care le acesta le oferă.

Deep Learning este o tehnologie cheie din spatele mașinilor fără șofer, permițându-le să recunoască un semn de oprire sau să distingă un pieton de un stâlp de iluminat. Reprezintă factorul esențial al controlului vocal în dispozitivele de consum, precum tabletele, telefoanele, televizoarele și difuzoarele hands-free.

În cadrul învățării profunde, un model de calculator învață să efectueze sarcini de clasificare direct din imagini, text sau sunet. Modelele de învățare profundă pot atinge o precizie uimitoare, depășind uneori performanța la nivel uman. Modelele sunt antrenate prin utilizarea unui set cât se poate de mare de date etichetate (dataset) și a unor arhitecturi de rețele neuronale care conțin mai multe straturi (layers) fără a fi necesară extragerea manuală a atributelor.[2]

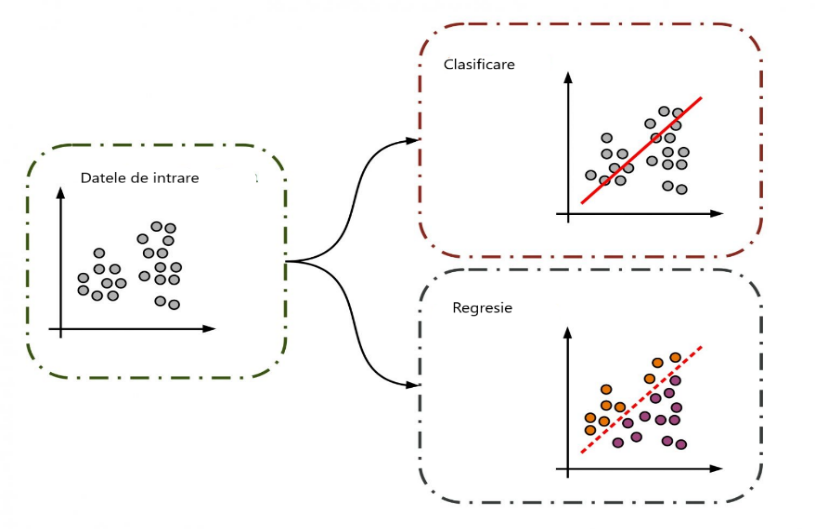
La momentul actual sunt cunoscute patru mari tehnici ale DL: învățare supravegheată (deep supervised learning), învățare nesupravegheată (deep unsupervised learning), învățarea semi-supraveghetă (deep semi-supervised learning) și învățare prin consolidare (deep reinforcement learning).



Structura tehnică a DL

### Deep Supervised Learning (DSL)

Învățarea supervizată este prezentă atunci când învățăm sau antrenăm mașina cu ajutorul unor date bine etichetate (labeled data), ceea ce înseamnă că anumite date sunt deja etichetate în mod corect. În învățarea supravegheată sunt variabilele de intrare (x) și variabila de ieșire (Y) și se utilizează un algoritm de învățare a funcției de mapare (Y=f(x)). Scopul final este aproximația cât mai bună a funcției de mapare, astfel încât la introducere de date noi de intrare (x), să se poată realiza prezicerea datei de ieșire (Y). Modelul este elaborat prin deducerea anumitor structuri existente în datele de intrare.



Metode de învățare supervizată

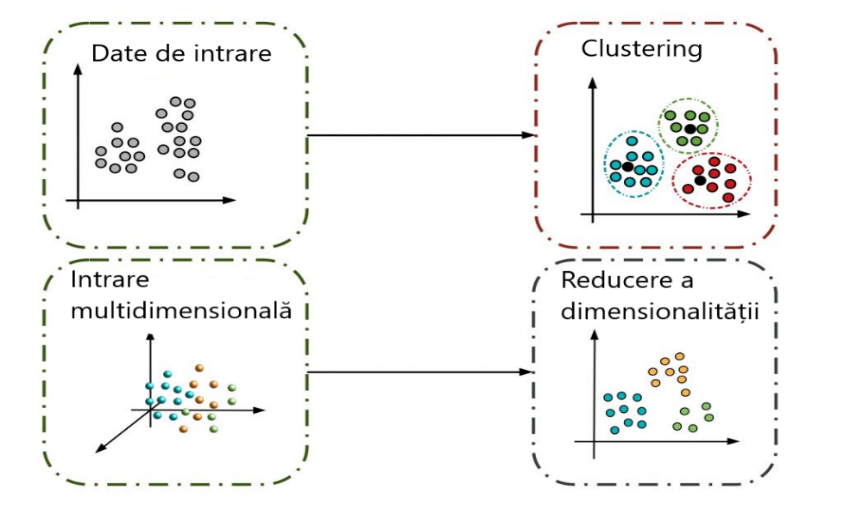
**Clasificarea** folosește un algoritm pentru a atribui cu precizie datele de testare în categorii specifice. Recunoaște anumite entități din setul de date și încearcă să tragă câteva concluzii cu privire la modul în care aceste entități ar trebui să fie etichetate sau definite.Exemple de metode de clasificare: clasificatorii liniari, mașina cu suport vectorial (SVM), arbori decizionali și pădure de arbori decizionali (random forest).

**Regresia** utilizează un algoritm pentru a înțelege relația dintre variabilele dependente și cele independente. Modelele de regresie sunt utile pentru prezicerea valorilor numerice pe baza diferitelor puncte de date. Unii algoritmi de regresie populari sunt regresia liniară, regresia logistică și regresia polinomială.

### Deep Unsupervised Learning (DUL)

Învățarea nesupervizată utilizează algoritmi de învățare automată pentru a analiza și a grupa seturi de date neetichetate. Acești algoritmi descoperă modele ascunse în date fără a fi nevoie de intervenția umană. Modelele de învățare nesupervizată sunt utilizate pentru trei scopuri esențiale: grupare(clustering), reducerea dimensionalității și învățarea regulilor de asociere. Deci sarcina mașinii este să grupeze informațiile nesortate în funcție de asemănări, modele și discrepanțe, fără nicio pregătire anterioară a datelor.

Exemple de algoritmi: algoritmul Apriori și k-Means.



Metode de învățare nesupraveghetă

**Clustering** reprezintă o modalitate de a extrage datele în scopul grupării acestora (neetichetate) pe baza asemănărilor sau diferențelor lor.

**Asocierea** utilizează reguli diferite pentru a găsi relații între variabilele dintr-un set de date prestabilit.

**Reducerea dimensionalitățiilor** este o tehnică de învățare atunci când numarul de dimensiuni dintr-un anumit set de date este prea mare. Reduce numărul de intrări de date la o dimensiune rezonabilă, păstrând integritatea lor.

### Deep Semi-Supervised Learning (DSSL)

În cazul învățării semi-supervizate datele de intrare reprezintă o mixare de date etichetate și neetichetate. În cazul acestui tip de învățare, există o anumită deficiență în ce privește predicția, însă modelul trebuie să învețe structura existentă în date pentru a le putea grupa și a realiza predicții.

## Abordarea Deep Learning

Deși învățarea profundă a fost teoretizată pentru prima dată în anii 1980, există două motive principale pentru care a devenit utilă abia în ultimii ani și anume:

* Învățarea profundă necesită cantități mari de date etichetate. De exemplu, progesul dezvoltării mașinilor autonome implică milioane de imagini și mii de ore de înregistrări video a traficului, a semnelor de circulație și tot ce implică interacțiunea umană cu circumstanțele din timpul șofatului.
* Învățarea profundă necesită o putere de calcul substanțială. Astfel, GPU-urile de înaltă performanță reprezintă o soluție fezabilă, acestea având o arhitectură paralelă ce poate realiza un număr considerabil de calcule simultan, fiind eficientă pentru învățarea profundă. Atunci când este combinată cu clusterele sau cu cloud computing-ul, aceasta permite echipelor de dezvoltare să reducă timpul de antrenare pentru o rețea de învățare profundă de la săptămâni la ore sau mai puțin, optimizând astfel sarcinile urmărite în acest domeniu de actualitate.

## Provocări impuse de Deep Learning

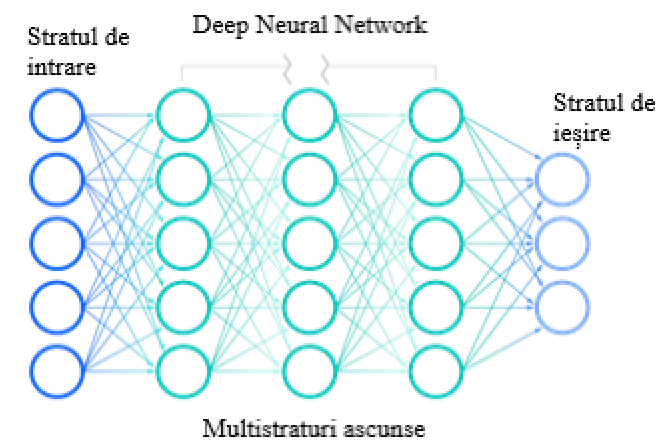
Rețelele de DL pot fi aplicate cu succes unor cantități mari de date în scopul descoperirii de noi informații, aplicarea informațiilor deja cunoscute cât și predicția bazată pe informații (pentru a dezvolta noi structuri). Altfel spus, DL poate fi un motor puternic pentru obținerea unor rezultate care pot fi puse în practică.[3]

Principalele avantaje ale utilizării DL:

* Utilizarea optimă a datelor nestructurate: procente uriașe din cantitatea de date ale unei organizații sunt nestructurate, deoarece o bună parte din acestea există în diferite tipuri de formate, cum ar fi texte, imagini, audio, video etc.
* Eliminarea nevoii de inginerie a caracteristicilor (feature engineering): ingineria caracteristicilor reprezintă procesul de utilizare a informațiilor deja cunoscute dintr-un domeniu pentru a selecta și transforma cele mai relevante variabile din datele brute.
* Capacitatea de a furniza rezultatele cu o acuratețe mare: un model de DL este capabil să îndeplinească mii de sarcini de rutină și repetitive într-o perioadă de timp mai scurtă.
* Reducerea costurilor inutile.
* Eliminarea necesității de etichetare a datelor: nevoia de date etichetate corespunzatoare se reduce, deoarece algoritmii excelează la învățarea fără supervizare.[3]
* Arhitectura DL este flexibilă, astfel încât sa poată fi adaptată la noi probleme în viitor.

## Deep Learning și Machine Learning. Diferențe.

Termenul “deep”din Deep Learning se referă la adâncimea straturilor dintr-o rețea neuronală. O rețea neuronală care constă din mai mult de trei straturi (care ar include intrările și ieșirile) poate fi considerată un algoritm DL. Acest lucru este reprezentat, cu ajutorul următoarei diagrame:

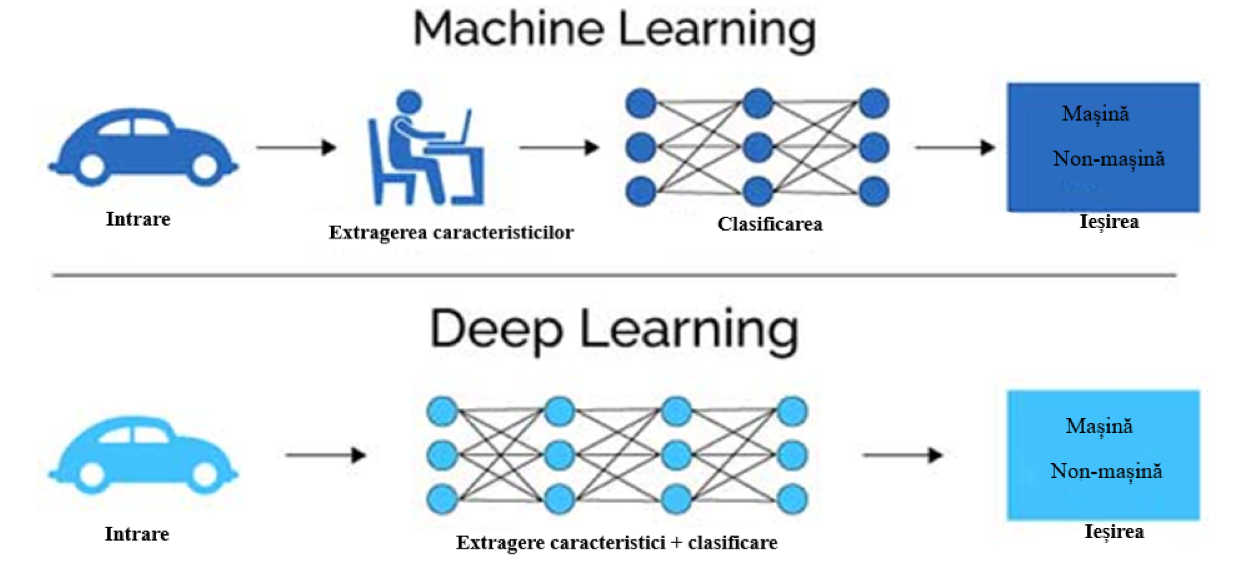


Rețea neuronală cu straturi multiple

Din figura se observă că fiecare componentă a unui strat este interconectată cu toate componentele stratului ulterior.

Machine Learning sau învățarea automată a fost definită în anii 1950 de către pionierul AI Arthur Samuel ca fiind “domeniul de studiu care oferă computerelor capacitatea de a învăța fără a fi programate în mod explicit.”.

* + Principala diferență dintre DL și ML constă în modul în care fiecare algoritm al acestor domenii învață, dupa cum este exemplificat și în figura, și în cantitatea de date pe care fiecare tip de algoritm o utilizează:
* DL automatizează o mare parte din procesul de extragere a caracteristicilor, reducând intervenția umană manuală necesară. De asemenea, permite utilizarea unor seturi mari de date, ceea ce i-a adus titlul de "învățare mecanică scalabilă".
* ML sau învățarea "non-profundă" depinde mai mult de intervenția umană pentru a învăța. Experții umani determină ierarhia caracteristicilor pentru a înțelege diferențele dintre intrările de date, fiind nevoie de mai multe date structurate pentru a învăța.[1]



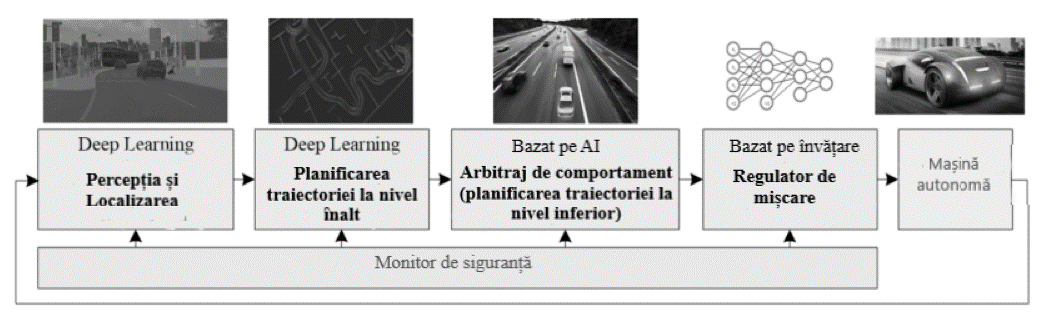
Deep Learning/Macine Learning

* + "Deep" Machine Learning poate utiliza seturile de date etichetate pentru a-și informa algoritmul, dar nu necesită neapărat un set de date etichetate. Aceasta poate asimila date nestructurate în forma lor brută (de exemplu: text, imagini) și poate determina automat setul de caracteristici care disting diverse clase.[4]
  + Prin observarea modelelor din date, un model de Deep Learning poate grupa intrările în mod corespunzător.Astfel, un model DL are nevoie de mai multe date pentru o îmbunătățire a acurateței, în timp ce un model ML se bazează pe mai puține date, având în vedere structura de bază a datelor.[1]
  + Algoritmii ML tind să fie mai puțin complexi decât algoritmii DL și pot fi rulați adesea pe computere convenționale, însă sistemele DL necesită resurse hardware mult mai puternice. Această cerere de putere a determinat o utilizare sporită a unităților de procesare grafică. GPU-urile sunt utile în cadrul aplicațiilor de AI, deoarece au lățime de bandă mare și capacitatea de a reduce latența (întârzierile) în transferul de memorie datorită paralelismului de tip thread (capacitatea mai multor operații de a rula eficient simultan).
  + ML este deja utilizată în serviciul de e-mail, în cadrul băncilor și în medicină. Tehnologia DL permite programe mai complexe și urmărește automatizarea acestora, cum ar fi mașinile autonome sau roboții care efectuează operații chirurgicale avansate.[5]

## Deep Learning în contextul automatizării autovehiculelor

Prima mașină parțial autonomă a fost inventată în 1989, fiind vorba despre vehiculul terestru automat în rețea neuronală (ALVINN). Acesta folosea rețele neuronale pentru a detecta liniile, a segmenta mediul, a se autodetecta și a conduce. A funcționat bine, dar a fost limitată de puterea de procesare lentă și de datele insuficiente la vremea respectivă.

Mașinile fără șofer sunt sisteme autonome de luare a deciziilor. Acestea pot procesa fluxuri de date provenite de la diverși senzori, cum ar fi camere, LiDAR, RADAR, GPS sau senzori de inerție. Aceste date sunt apoi modelate cu ajutorul unor algoritmi DL, care iau apoi decizii relevante pentru mediul în care se află mașina. În figura este ilustrată structura evenimentelor care se urmăresc în realizarea unor sisteme de deplasare complet autonome.



un pipeline modular de percepție-planificare-acțiune de funcționare a mașinilor autonome[6]

Pentru a înțelege cum funcționează mașinile autonome, trebuie să examinăm următoarele direcții:

* Percepția
  + Localizarea
  + Predicția
  + Luarea deciziilor
* Planificarea traiectoriei la nivel înalt
* Arbitrajul de comportament
* Regulator de mișcare

Detecția obiectelor este o problemă bine cunoscută în domeniul Computer Vision (CV) și al Deep Learning-ului. Există două componente într-un model de detecție a obiectelor, și anume, rețeaua neuronală de bază și rețeaua neuronală de detectare.

### Percepția

Percepția ajută mașina “să observe” lumea înconjurătoare, precum și să recunoască și să clasifice lucrurile pe care le vede. Pentru a lua decizii bune, mașina trebuie să recunoască instantaneu obiectele, un aspect ce impune provocăroi uriașe în asigurarea detecției real-time..

Detecția obiectelor a fost și este o problemă bine cunoscută în domeniul computer vision (CV) și al Deep Learning-ului. Există două componente într-un model de detectare a obiectelor, și anume, rețeaua neuronală de bază și rețeaua neuronală de detectare.

Astfel, mașina trebuie să vadă și să clasifice semafoarele, pietonii, semnele de circulație, trotuarele, locurile de parcare, benzile de circulație și multe altele. Totodată, trebuie să știe și distanța exactă dintre ea și obiectele din jur. Percepția este mai mult decât simpla vedere și clasificare, ea permite sistemului să evalueze distanța și să decidă fie dacă să încetinească, fie să frâneze.

Pentru a obține un nivel înalt al percepției, o mașină autonomă trebuie sa aibe în dotare următorii senzori:

* *Camera*: permite mai multe sarcini precum clasificarea, segmentarea și localizarea
* *LiDAR*: o metodă de măsurare a distanței dintre obiecte prin emiterea unui fascicul laser pentru a fi reflectat de obstacol
* *RADAR*: calculează distanța folosind unde radio, care pot funcționa în orice condiții față de fasciculele laser.[7]

### Localizarea

Algoritmii de localizare folosiți pentru mașinile autonome calculează poziția și orientarea vehiculului în timp ce acesta navighează (o știință cunoscută sub numele de odometrie vizuală, VO).

DL este utilizată în general pentru a îmbunătăți performanța VO și pentru a clasifica diferite obiecte. Rețelele neuronale, cum ar fi PoseNet și VLocNet++, sunt unele dintre modelele care utilizează date punctuale pentru a estima poziția și orientarea 3D.

### Predicția

Înțelegerea conducătorilor auto umani este o sarcină foarte complexă. Aceasta implică mai degrabă emoții decât logică, iar acestea sunt alimentate de reacții. Devine foarte nesigur care va fi următoarea acțiune a șoferilor sau a pietonilor din apropiere, astfel încât un sistem care poate prezice acțiunile altor utilizatori ai drumurilor poate fi foarte important pentru siguranța rutieră.

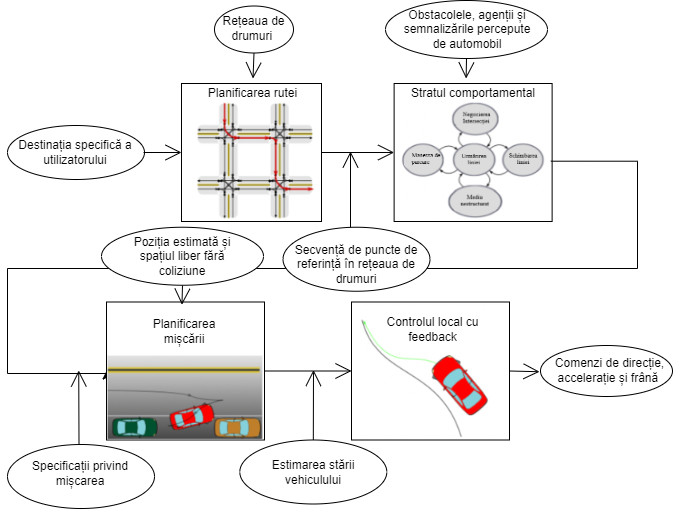
Automobilul are o vedere de 360 de grade asupra mediului înconjurător care îi permite să perceapă și să capteze toate informațiile și să le proceseze. Odată introdus în algoritmul DL, acesta poate să prezinte toate mișcările posibile pe care le-ar putea face ceilalți participanți la trafic.

Rolul DL este de a interpreta sarcinile complexe de viziune, de a se localiza în mediul înconjurător, de a îmbunătăți percepția și de a acționa manevrele cinematice în mașinile care se conduc singure.

### Luarea deciziilor[8]

Procesul decizional este vital în cazul mașinilor autonome. Acestea au nevoie de un sistem care să fie dinamic și precis într-un mediu incert. Acesta trebuie să țină cont de faptul că nu toate datele preluate de la senzori vor fi adevărate și că oamenii pot face alegeri imprevizibile în timpul conducerii.

În cazul luării deciziilor, se utilizează Deep Reinforcement Learning (DRL). Mai concret, în centrul DRL se află un algoritm de luare a deciziilor numit proces decizional Markov (MDP), utilizat pentru a prezice comportamentul viitor al șoferilor.



*Etapele parcurse de vehicul în luarea unei decizii*

Conform figurii , în luarea deciziilor se țin cont de următoarele:

* *Planificarea traseului sau a rutei*: automobilul trebuie să planifice cel mai bun traseu posibil de la poziția sa curentă la destinația solicitată.
* *Arbitrajul comportamentului*: incertitudine în ceea ce privește acțiunile celorlalți șoferi este rezolvată prin utilizarea algoritmilor de planificare probabilistică, cum ar fi MDP.
* *Planificarea mișcării*: include viteza vehiculului, schimbarea benzii de rulare și multe altele.
* *Controlul vehiculului*: este utilizat pentru a realiza traiectoria de referință din sistemul de planificare a mișcării.

# NOȚIUNI TEORETICE

Rețelele neuronale artificiale

Neuronii artificiali

Functii activare

Liniara

Sigmoid

Softmax

ReLu

## Definiții

## Procesul de învățare

### Funcția de activare (activation function)

### Antrenarea

### Funcția de pierderi (loss function)

### Parametrii și hiperparametrii rețelelor neuronale convoluționale. Probleme impuse în urma modificării acestora

## Rețelele neuronale convoluționale (CNN)

### Arhitectura

### Convoluția

### Pooling

### Fully-Connected

## Modelul de antrenare ales

## Modalități de antrenare

### Fine Tuning

### Transfer Learning

### Learning from scratch

## Procesarea imaginilor în DL

### Adnotarea

### Augumentarea

# METODE DE IMPLEMENTAREA ȘI VERIFICARE

## Implementarea software

### Mediul de dezvoltare, limbajul de programare și librării utilizate

### Construirea setului de date

### Implementarea CNN pe baza principiului de Transfer Learning

### Modelul SSD MobileNet v2 320x320

### Procesorul grafic Nvidia

### CPU și GPU. Limitări impuse de placa grafică.

### Coral USB Accelerator (TPU Edge)

### Tensorflow. Framework-ul TFLite

### Tensorboard

### OpenCV

### Probleme de implementare și soluții

## Implementare hardware

### Platforma computațională Raspberry Pi 4B

### Driver-ul DRV8835 Dual Motor pentru Raspberry Pi

### Modulul cameră v2 Raspberry Pi

### Probleme hardware de implementare și soluții

# REZULTATE EXPERIMENTALE

## Prezentarea evoluției sistemului autonom analizat

## Rezultate obținute

## Analiza performanțelor sistemului obținut

# CONCLUZII ȘI PERSPECTIVE DE VIITOR

# BIBLIOGRAFIE

# Bibliography

**There are no sources in the current document.**

# ANEXE

<https://course.elementsofai.com/ro/1/1>

