

Teme de lucrări de licențe/proiecte de diplomă realizate de absolvenții din anii anteriori

IMAGINI

Lucrare de licență

CĂRȚI DE JOC ȘI JOCURI DE CĂRȚI

DETECȚIA CĂRȚILOR ÎN IMAGINI ȘI ÎNVĂȚAREA

AUTOMATĂ DE STRATEGII PENTRU ANUMITE JOCURI

Absolvent

Dragoș-Constantin Tânțaru

³⁷**Coordonator științific**

Conf.dr. Bogdan Alexe

1	Introducere	9
2	Noțiuni Teoretice	11
2.1	Faster R-CNN	11
2.1.1	R-CNN	11
2.1.2	Fast R-CNN	12
2.1.3	Faster R-CNN	12
2.2	Resnet50-FPN	14
2.3	Mobilenetv3	15
2.3.1	Mobilenet	15
2.3.2	Mobilenetv2	16
2.3.3	Mobilenetv3	18
2.4	Monte-Carlo control	19
2.5	Double DQN	20
2.5.1	Deep Q-Network (DQN)	20
2.5.2	Double Q-learning	21
2.5.3	Double DQN	22
2.5.4	Aproximare Polyak	22
3	Recunoașterea cărților într-o imagine dată	23
3.1	Abordarea propusă	23
3.2	Generarea datelor de antrenare	24
3.3	Antrenarea rețelelor	31
3.3.1	Probleme de convergență	32
3.3.2	Optimizări la timpul de antrenare	32
3.3.3	Antrenarea unui Mobilenetv3	33
3.4	Prelucrarea detectiilor pentru fiecare joc	33
3.5	Performanță în scenarii reale și probleme ce pot apărea	35
4	Învățarea Automată a unor jocuri de cărți	36
4.1	Jocurile studiate și metodele folosite	36

Lucrare de licență

CĂRȚI DE JOC ȘI JOCURI DE CĂRȚI

DETECȚIA CĂRȚILOR ÎN IMAGINI ȘI ÎNVĂȚAREA

AUTOMATĂ DE STRATEGII PENTRU ANUMITE JOCURI

Absolvent

Dragoș-Constantin Tânțaru

³⁷**Coordonator științific**

Conf.dr. Bogdan Alexe

4.2	Blackjack cu 4 acțiuni	36
4.3	Macao	40
4.3.1	Codificarea stărilor și arhitectura rețelei	42
4.3.2	Antrenarea rețelei	43
4.3.3	Rezultate	44
4.4	Şeptică	44
4.4.1	Codificarea stărilor și arhitectura rețelei	45
4.4.2	Antrenare adversarială în contextul unui joc de Şeptică	46
4.4.3	Antrenarea rețelei	47
4.4.4	Rezultate	47
5	Integrarea celor două componente într-o singură aplicație	49
5.1	Biblioteci și tehnologii folosite	49
5.2	Prezentare a GUI-ului	50
5.2.1	Fereastra principală	50
5.2.2	Fereastra de detectie	51
5.3	Programarea GUI-ului	52
5.3.1	Capturarea imaginilor din Windows	53
5.3.2	Redimensionarea imaginilor pentru a ocupa tot viewport-ul aplicației	53
5.4	Comunicarea dintre cele două modele cu ajutorul automatelor finite	53
5.4.1	Engine-ul de Blackjack	54
5.4.2	Engine-ul de Şeptică	54
	Concluzii	56
	Bibliografie	58

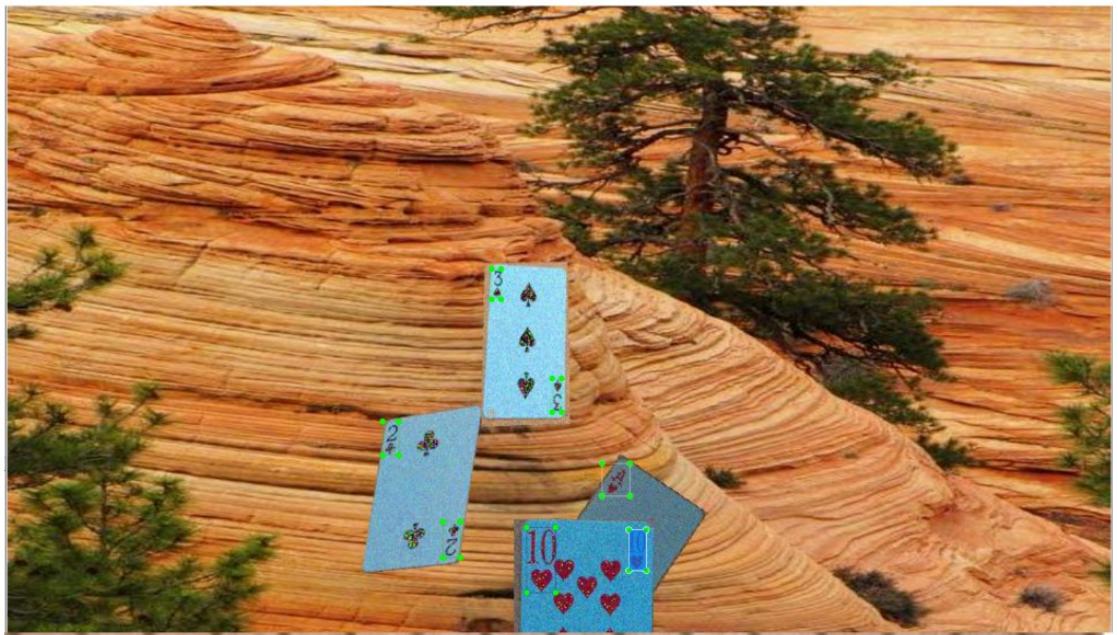


Figura 3.4: Exemplu de imagine de antrenare cu zgomot



Figura 3.5: Exemplu de imagine de antrenare cu tăieturi

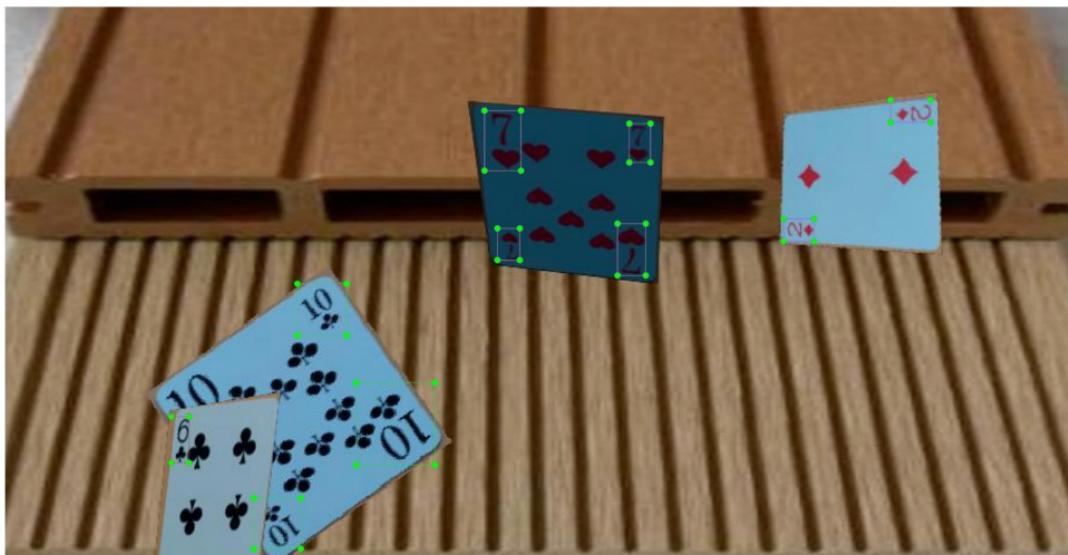


Figura 3.6: Imagine de antrenare cu transformări de întindere



(a) Detectie corecta pe un pachet diferit de cele folosite la antrenare



(b) Altă detectie corecta, de data asta și cu o mică rotație în plan

[Incepe un joc nou](#)

careu 9x9 cu restricții.

3. Word Twister
Descoperă cu literele date cuvinte românești.

4. X-Ball
Elimină grupuri de bile în ordinea corectă.

5. Slider
Mută bile pentru a forma cinci în rând.

6. Rebus
Cuvinte încruzișate.

Reguli
Jocul **Septica** are următoarele reguli:

- pachetul conține 32 de cărți: opt figuri, 7, 8, 9, 10, J, Q, K și A, fiecare figură având 4 cărți
- fiecare jucător primește inițial patru cărți, alese la întâmplare
- jucătorii pun însă o carte, alternativ

24 cărți 0 puncte

(c) Detectie corecta, insă...

[Incepe un joc nou](#)

celebrul joc de cărți.

2. Sudoku
Complează cu cifre un careu 9x9 cu restricții.

3. Word Twister
Descoperă cu literele date cuvinte românești.

4. X-Ball
Elimină grupuri de bile în ordinea corectă.

5. Slider
Mută bile pentru a forma cinci în rând.

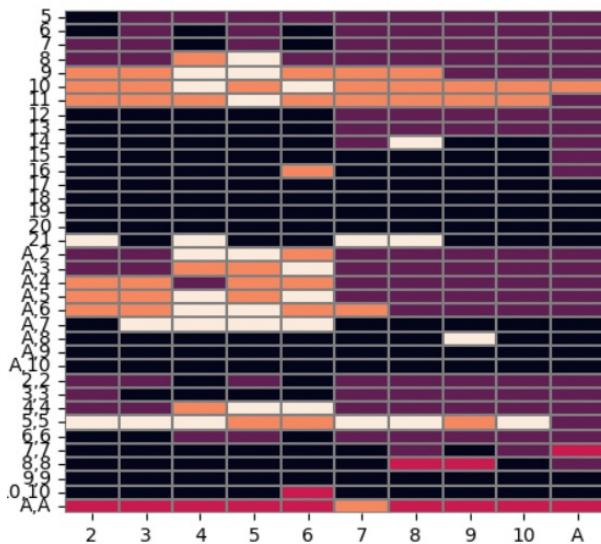
Reguli
Jocul **Septica** are următoarele reguli:

24 cărți 0 puncte

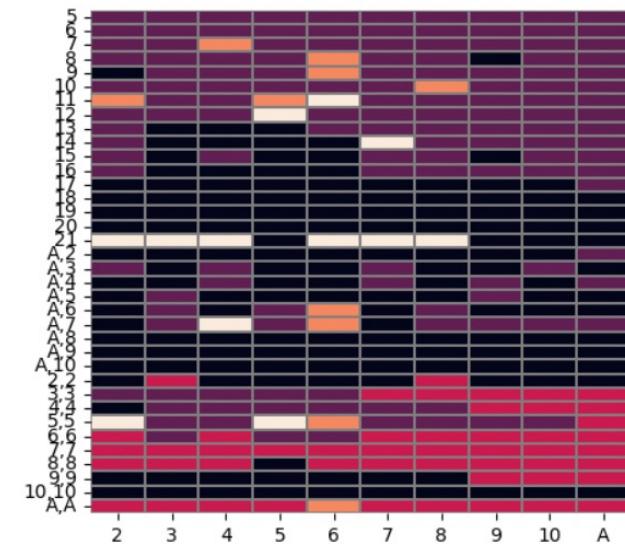
(d) Dacă dăm puțin scroll up, detectia devine proastă, deci tot mai apar mici probleme uneori

Your hand	2	3	4	5	6	7	8	9	10	A	Dealer's card
4-8	H	H	H	H	H	H	H	H	H	H	H
9	Dh	Dh	Dh	Dh	Dh	H	H	H	H	H	H
10	Dh	Dh	Dh	Dh	Dh	Dh	Ds	Ds	H	H	H
11	Dh	Dh	Dh	Dh	Dh	Dh	Ds	Ds	Ds	Ds	Ds
12	H	H	S	S	S	H	H	H	H	H	H
13	S	S	S	S	S	H	H	H	H	H	H
14	S	S	S	S	S	H	H	H	H	H	H
15	S	S	S	S	S	H	H	H	Rh	Rh	
16	S	S	S	S	S	H	H	H	Rh	Rh	
17	S	S	S	S	S	S	S	S	S	S	Rs
18-21	S	S	S	S	S	S	S	S	S	S	S
A,2	H	H	H	Dn	Dn	H	H	H	H	H	H
A,3	H	H	Dn	Dn	Dn	H	H	H	H	H	H
A,4	H	H	Dn	Dn	Dn	H	H	H	H	H	H
A,5	H	H	Dn	Dn	Dn	H	H	H	H	H	H
A,6	H	Dn	Dn	Dn	Dn	H	H	H	H	H	H
A,7	Ds	Ds	Ds	Ds	Ds	S	S	H	H	H	H
A,8	S	S	S	S	Ds	S	S	S	S	S	S
A,9-10	S	S	S	S	S	S	S	S	S	S	S
2,2	Ph	Ph	P	P	P	P	H	H	H	H	H
3,3	Ph	Ph	P	P	P	H	H	H	H	H	H
4,4	H	H	H	Ph	Ph	H	H	H	H	H	H
6,6	P	P	P	P	P	Ph	H	H	H	H	H
7,7	P	P	P	P	P	Ph	H	H	H	H	H
8,8	P	P	P	P	P	P	P	P	P	Rp	
9,9	P	P	P	P	S	P	P	S	S	S	
A,A	P	P	P	P	P	P	P	P	P	P	

www.wizardofodds.com



(a) Strategia Monte-Carlo



(b) Strategia Double Q-learning

H	Hit
S	Stand
C	Double if allowed, otherwise hit
Ds	Double if allowed, otherwise stand
P	Split
Ph	Split if double after split is allowed, otherwise hit
Rh	Surrender if allowed, otherwise hit
Rs	Surrender if allowed, otherwise stand
Rp	Surrender if allowed and double after split not allowed, otherwise split

(c) Strategia Optimă [36]

Figura 4.1: Comparație între cele trei strategii

Lucrare de licență

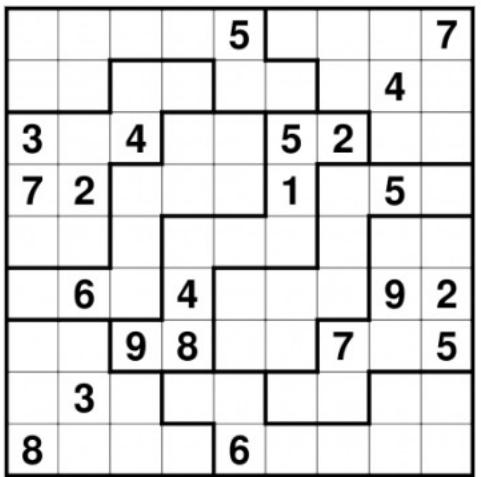
SOLUȚIE PENTRU SUDOKU 3D, BAZATĂ PE COMPUTER VISION

Absolvent

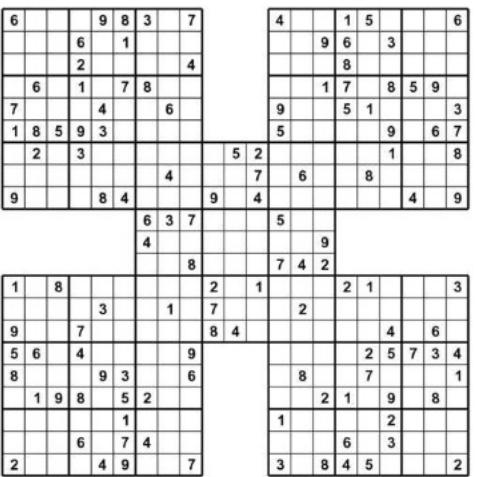
Rusu Andrei-Cristian

¹³**Coordonator științific**
Conf. Dr. Alexe Bogdan

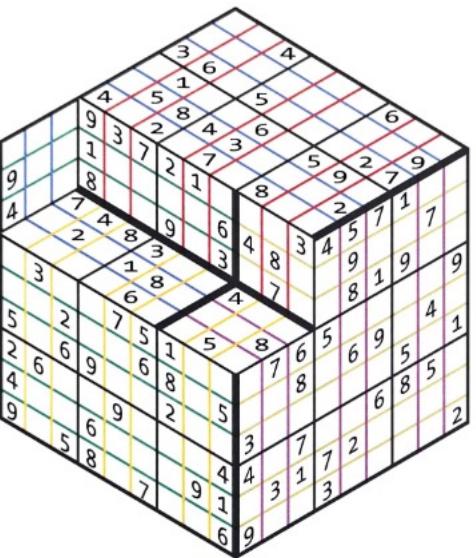
1 Introducere	5
1.1 Informații istorice	5
1.2 Sudoku 3D	6
2 Baza de date și preprocesarea datelor	8
3 Terminologie și convenții	11
3.1 Axele	11
3.2 Notația punctelor fețelor	11
3.3 Segmentele îngroșate	12
4 Extragerea informațiilor din imagine	13
4.1 Calcularea coordonatelor vârfurilor cubului	13
4.2 Identificarea segmentelor îngroșate	14
4.3 Extragerea vectorilor de deplasare a fețelor	15
4.4 Determinarea existenței unei fețe definite de patru puncte și extragerea acesteia	15
4.5 Extragerea drumurilor folosind segmentele îngroșate	18
4.5.1 Extragerea punctelor de început ale fețelor	18
4.5.2 Algoritmul	21
4.5.3 Formulele pentru calculul punctelor fețelor din drumuri	21
4.6 Extragerea cifrelor din imagini cu fețe	27
4.6.1 Preprocesarea imaginii	27
4.6.2 Extragerea informației din celule	28
4.6.3 Clasificarea cifrelor din imagini	29
5 Generarea soluției	31
5.1 Algoritmul	31
5.2 Verificarea validității unei configurații	33
5.3 Rezultate	34



(a)



(b)



(c)

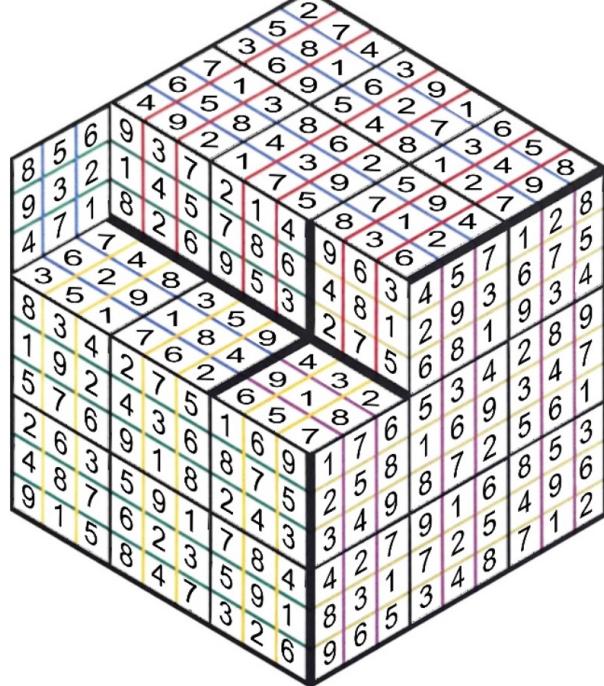
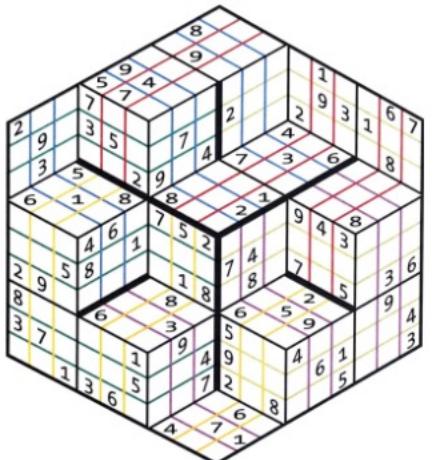
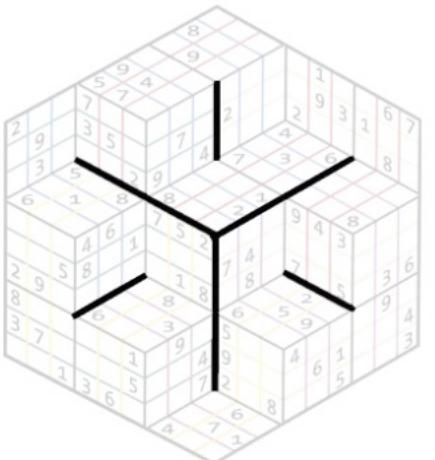


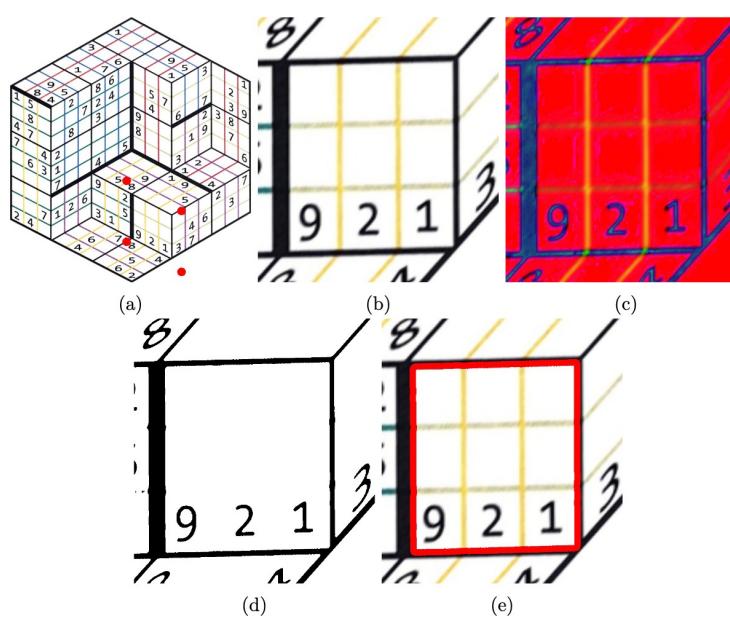
Figura 1.4: Soluția pentru configurația din Figura 1.3c



(a)



(b)



(d)

(e)

1. Introducere

1.1. Descrierea problemei	1
¹³ 1.2. Motivarea alegerii temei	2
1.3. Structura lucrării	3

Lucrare de licență

DETECȚIA LITERELOR DIN LIMBAJUL SEMNELOR

Absolvent

Costea Andrei

⁶**Coordonator științific**

Conf. Dr. Bogdan-Dumitru Alexe

2. Setul de date

2.1. Prezentarea și preprocesarea setului de date de bază	4
2.2. Colectarea setului de date pentru fine-tuning	7

3. Rețeaua neuronală

3.1. Procesarea datelor	9
3.2. Arhitectura rețelei	10
3.2.1. Extragerea trăsăturilor imaginilor	10
3.2.2. Ramura de clasificare	13
3.2.3. Ramura de localizare	14
3.2.4. Funcția de pierdere. Compilarea și summarizarea modelului	15
3.3. Antrenarea rețelei	16
3.4. Evaluarea rețelei	18
3.5. Fine-tuning. Evaluarea finală a rețelei	20
3.6. Observații	23

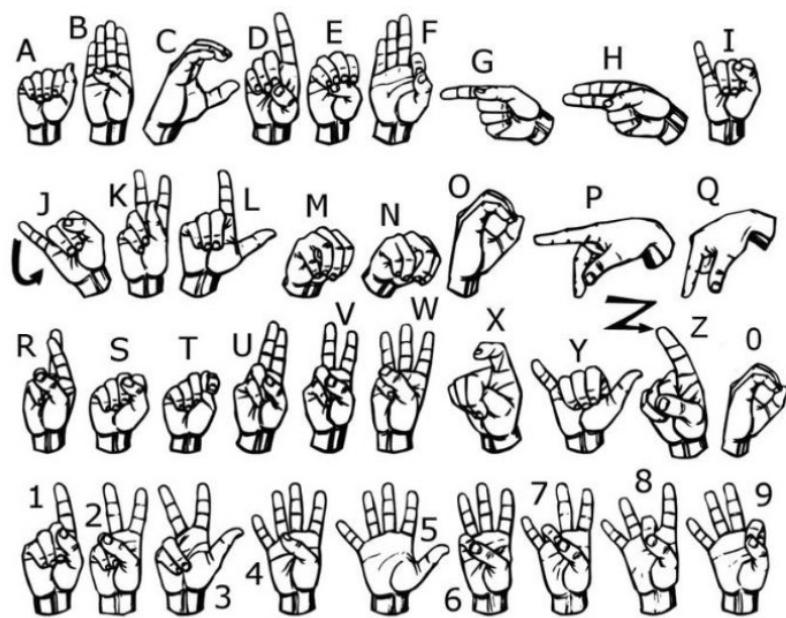


Figura 1.2 - alfabetul semnelor în folosință

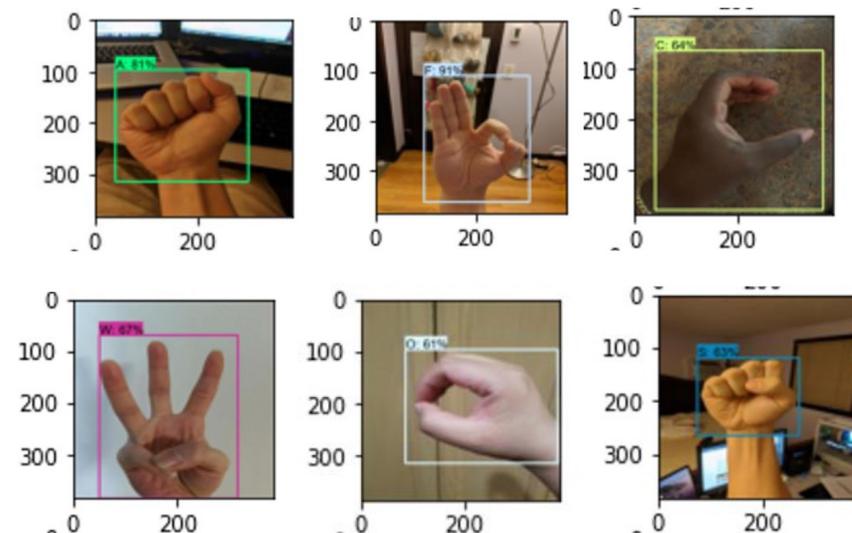
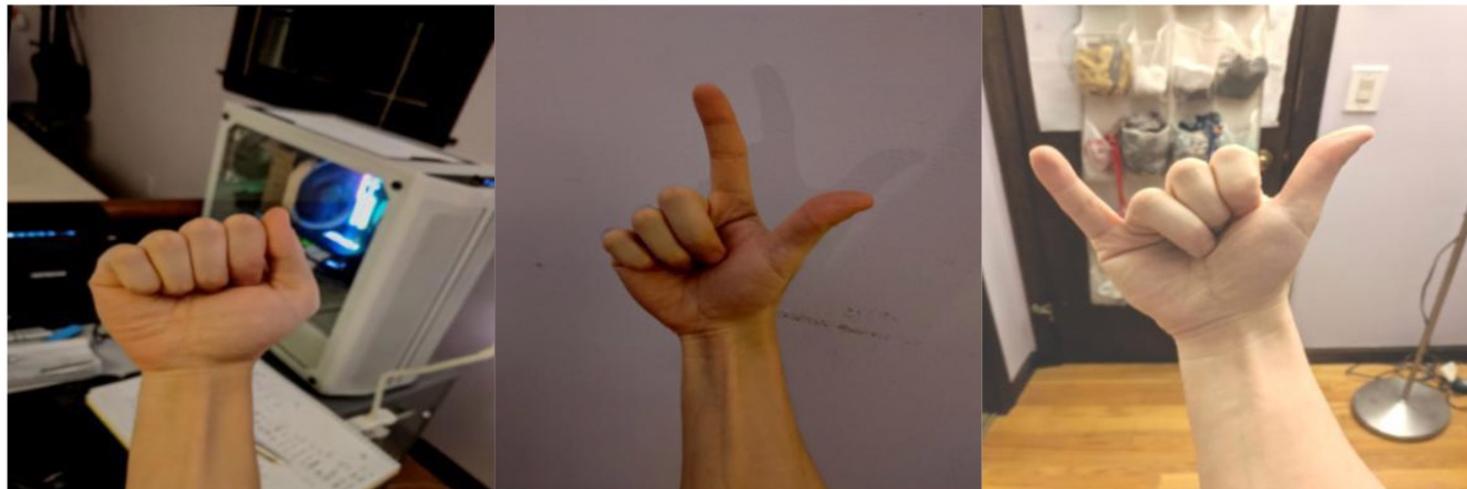


Figura 4.6 – Detectii corecte ale modelului



37
Figura 2.1 Exemple de imagini din setul de date

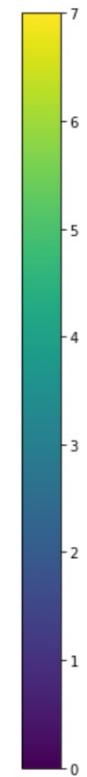
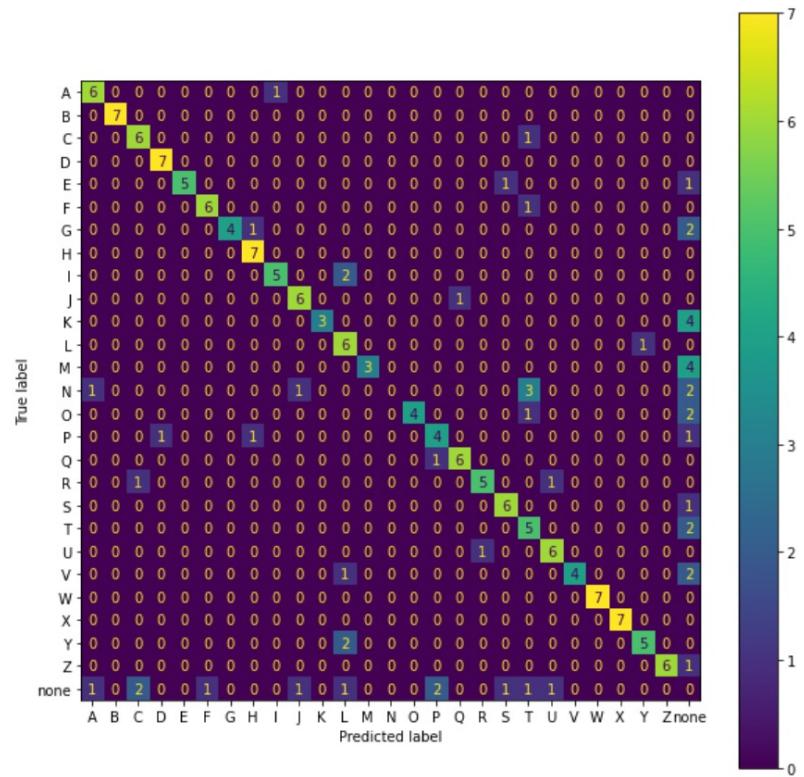


Figura 4.6 – Detectii corecte ale modelului

Lucrare de licență

ESTIMATOR DE PREȚURI PENTRU PIATA IMOBILIARĂ

Absolvent

Savu Ioan-Daniel

45 Coordonator științific

Conf. Dr. Alexe Bogdan

1 Introducere	6
2 Baza de date	8
2.1 Colectarea datelor	8
2.2 Analiza datelor	11
2.3 Preprocesarea datelor	14
3 Modele ce folosesc informații vizuale	16
3.1 Extragere de caracteristici și mașini cu vectori suport	16
3.1.1 Motivație	16
3.1.2 Extragere de Caracteristici. ResNet18	17
3.1.3 Regresie cu vectori suport	18
3.1.4 Implementare	19
3.1.5 Rezultate	21
3.2 Curățarea datelor	24
3.2.1 Implementare	24
3.2.2 Rezultate	25
3.3 Regresie folosind Rețele Neuronale	26
3.3.1 Implementare	26
3.3.2 Rezultate	27
3.3.3 Exemple	30
4 Modele ce folosesc informații text	32
4.1 Naive Bayes	32
4.1.1 Motivație	32
4.1.2 Provocările implementării	32
4.1.3 Implementarea folosind scikit-learn	33
4.1.4 Rezultatele implementării din scikit-learn	34
4.1.5 Implementarea proprie	35
4.1.6 Rezultatele implementării propriei	38
4.1.7 Exemple	38



Lucrare de licență

ESTIMATOR DE PREȚURI PENTRU PIATA IMOBILIARĂ

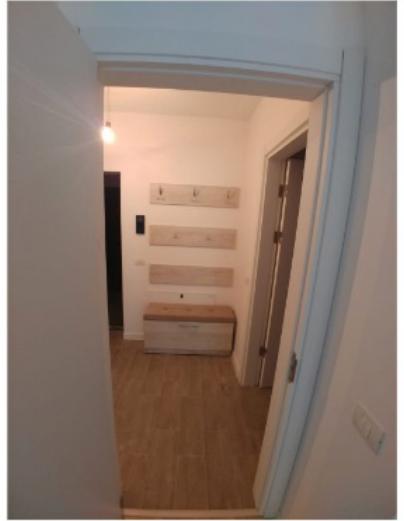
Absolvent

Savu Ioan-Daniel

45 Coordonator științific

Conf. Dr. Alexe Bogdan

4.2 Rețele neuronale	40
4.2.1 Implementare	40
4.2.2 Rezultate	41
5 Modele ce folosesc informații vizuale și text	43
5.1 Combinăție liniară între modele	43
5.1.1 Ecuații Normale	44
5.1.2 Implementare	45
5.1.3 Rezultate	46
5.1.4 Exemple	47
5.2 Rețele neuronale	48
5.2.1 Implementare	48
5.2.2 Rezultate	49
6 Dezvoltarea aplicației	51
6.1 Componenta de backend	51
6.1.1 Framework-ul Flask	52
6.1.2 Adaptarea estimatorului bazat pe Naive-Bayes	53
6.2 Componenta de frontend	54
6.2.1 React	54
6.2.2 Design-ul aplicației	54
7 Concluzii	57
Bibliografie	59



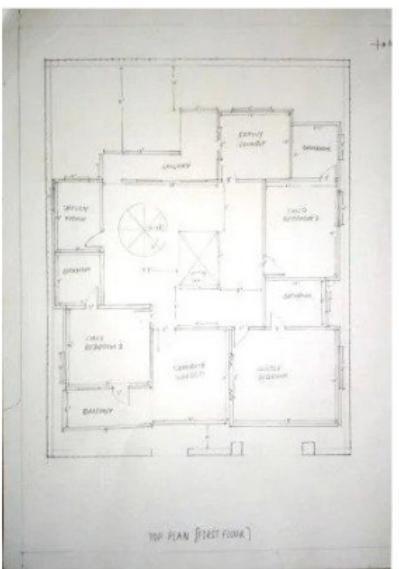
(a)



(b)



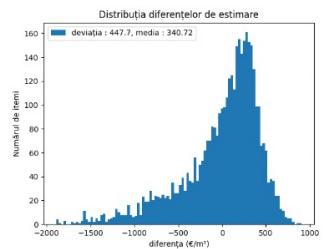
(c)



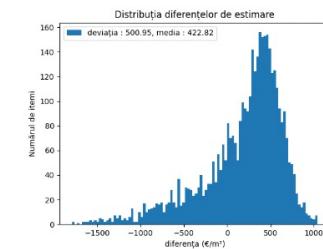
(d)

13 Figura 3.6: Exemple de imagini informative

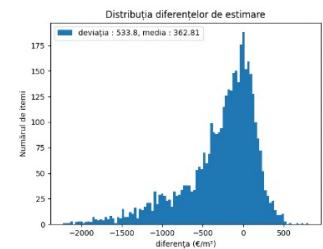
Figura 3.13: Imagini cu diverse camere dintr-un apartament a) baia, b) bucătăria, c) dormitorul, d) holul



(a)



(b)



(c)

Figura 4.1: distribuția diferențelor de estimare pe datele de testare pentru : a) metoda 1, b) metoda 2, c) metoda 3

Metoda folosită	Root Mean Squared Error ($\text{€}/\text{m}^2$)
Fără netezire	329,00
Laplace Smoothing cu $\alpha=1$	325,54
Laplace Smoothing cu $\alpha=0,5$	326,84
Gaussian Smoothing cu $N = 1$	321,01

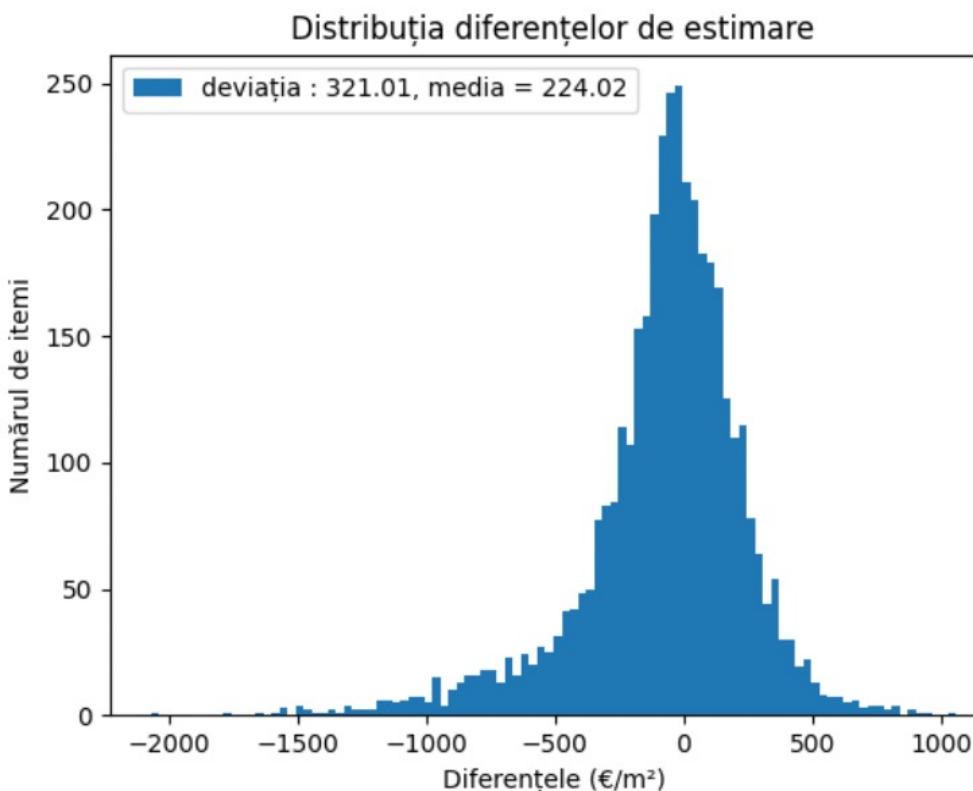


Figura 4.2: Distribuția diferențelor pentru netezire Gauss

IMAGINI

Jocuri pe tablă

Proiect de dimplomă

AGENT INTELIGENT PENTRU JOCUL ȚINTAR

Absolvent
Preda Alexandru-Florin

Coordonatori științifici:
Conf. dr. Bogdan Alexe
Asist. drd. Alexandra Diaconu

2 Jocul Țintar	10
2.1 Reguli și desfășurare	10
2.1.1 Faza 1: Punerea pieselor	11
2.1.2 Faza 2: Mutarea pieselor	12
2.1.3 Stări finale	13
2.2 Strategii	14
3 Recunoașterea configurației curente pe tabla de joc	15
3.1 Alinierea tablei folosind descriptori SIFT	15
3.1.1 Construirea spațiului scalar	16
3.1.2 Localizarea punctelor de interes	19
3.1.3 Stabilirea orientării	19
3.1.4 Descriptor de puncte de interes	19
3.2 Extragerea patch-urilor	20
3.3 Clasificarea patch-urilor folosind LinearSVC	20
3.3.1 Generarea datelor de antrenare și testare	20
3.3.2 Antrenarea modelului și testarea lui	20
3.4 Rezultatul final	22
4 Găsirea celei mai bune mutări	23
4.1 Algoritmul min-max	23
4.2 Algoritmul alpha-beta	24
4.3 Funcții euristică	26
4.4 Algoritmul bazat pe paradigma învățării prin recompense	27
4.4.1 Arhitectura modelului	27
4.4.2 Seturile de antrenare și evaluare	28
4.4.3 Integrare în aplicație	29
5 Evaluări experimentale	30
5.1 Modelul de clasificare	30
5.2 Cea mai bună euristică	31
5.3 Performanța rețelei	32

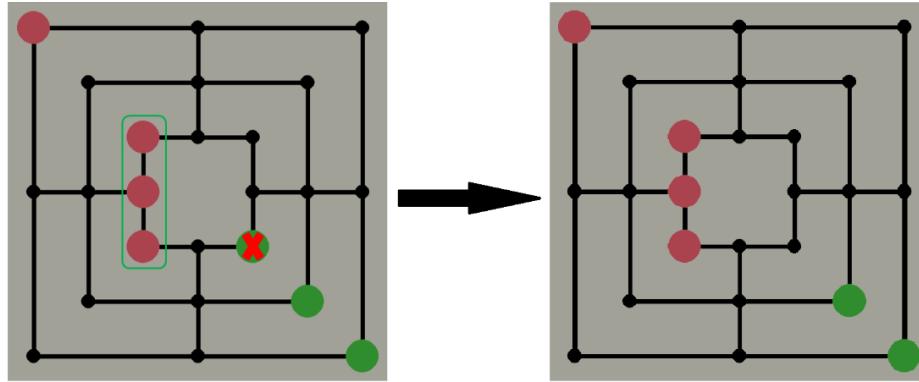


Figura 2.3: Exemplu moară validă și îndepărțare piesă.

2.1.2 Faza 2: Mutarea pieselor

După plasarea celor 9 piese pe tablă de către ambii jucători, jocul intră în faza a două. În această etapă, jucătorii nu mai poziționează piesele, ci le mută pe cele deja plasate pe tablă, până când jocul atinge o stare finală.

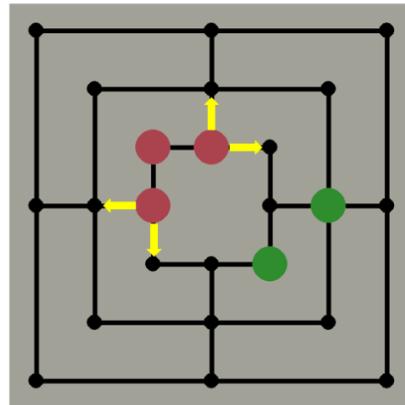


Figura 2.4: Exemplu mutări valide din faza 2.

Figura 2.4 ilustrează că o mutare este considerată validă atunci când piesa este mutată într-o poziție imediat adiacentă. Similar fazei precedente, jucătorii trebuie să formeze morii pentru a elimina piesele adversarilor, până când jocul ajunge în una dintre cele două stări finale posibile.

În Figura 2.5 se observă o mutare efectuată de jucătorul cu piesele portocalii în care piesa mutată de acesta formează moară încercuită cu verde. După formarea morii în

3.4 Rezultatul final

În Figura 3.6 sunt ilustrați pașii transformării imaginii de intrare în lista reprezentativă configurației tablăi. Imaginea de intrare este aliniată cu imaginea de referință folosind descriptori SIFT, iar din imaginea aliniată sunt extrase patch-urile ce conțin poziții valide ale pieselor, salvate într-o listă. Ulterior, fiecare patch este înlocuit, unul câte unul, cu predicția modelului de clasificare, 0 reprezentând o poziție liberă, 1 piesele verzi și -1 piesele portocalii.

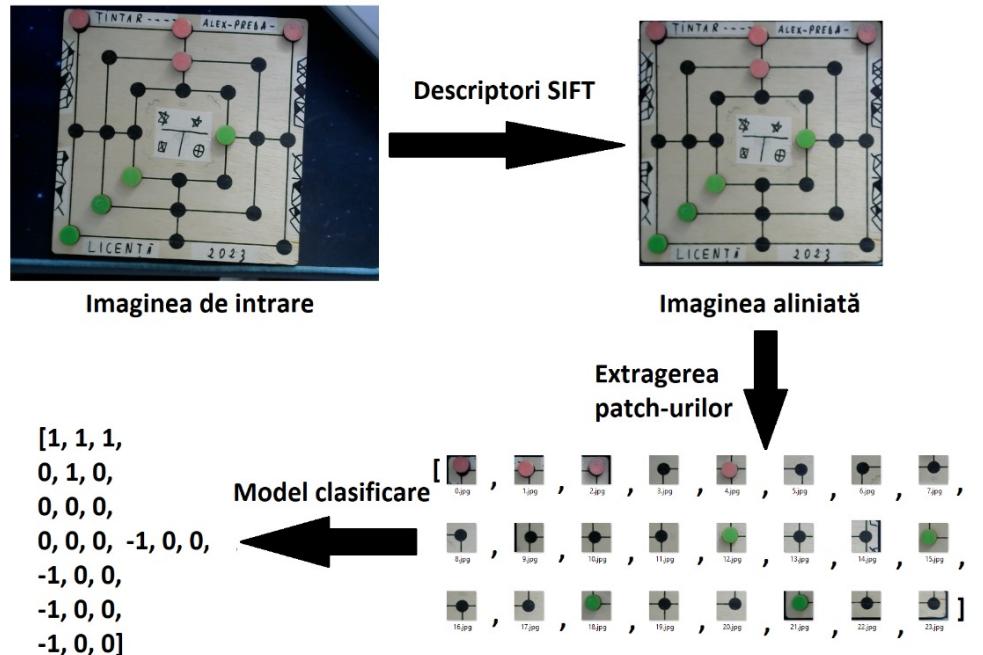


Figura 3.6: Extragere configurație tablă

Lucrare de licență

AGENT INTELIGENT PENTRU JOCUL PATRU ÎN LINIE

Absolvent

Borș Andrei

10 Coordonator științific

Conf. Dr. Bogdan Alexe

Cuprins

1 Introducere	5
1.1 Descrierea lucrării	5
1.2 Structura lucrării	6
1.3 Regulile jocului "Patru în linie"	6
2 Interpretarea informației vizuale din tabla de joc	8
2.1 Transformarea vizuală a tablei de joc	8
2.2 Transformări de contururi și algoritmul Kuhn-Munkres	11
3 Agenții inteligenți	13
3.1 Agenții bazați pe Algoritmul MinMax	13
3.1.1 Prezentarea generală a MinMax și Alfa-Beta	13
3.1.2 Tabele de transpoziție și Zobrist Hashing	17
3.1.3 Evaluarea euristică implementată de MinMax Simplu	19
3.1.4 Evaluarea euristică implementată de MinMax Avansat și conceptul de Zugzwang	20
3.2 Jucătorul bazat pe învățare pe bază de recompense	22
4 Evaluare experimentală și idei de îmbunătățire	25
4.1 Sistemul de interpretare vizuală	25
4.2 Agenții cu inteligență artificială	25
5 Structura aplicației	29

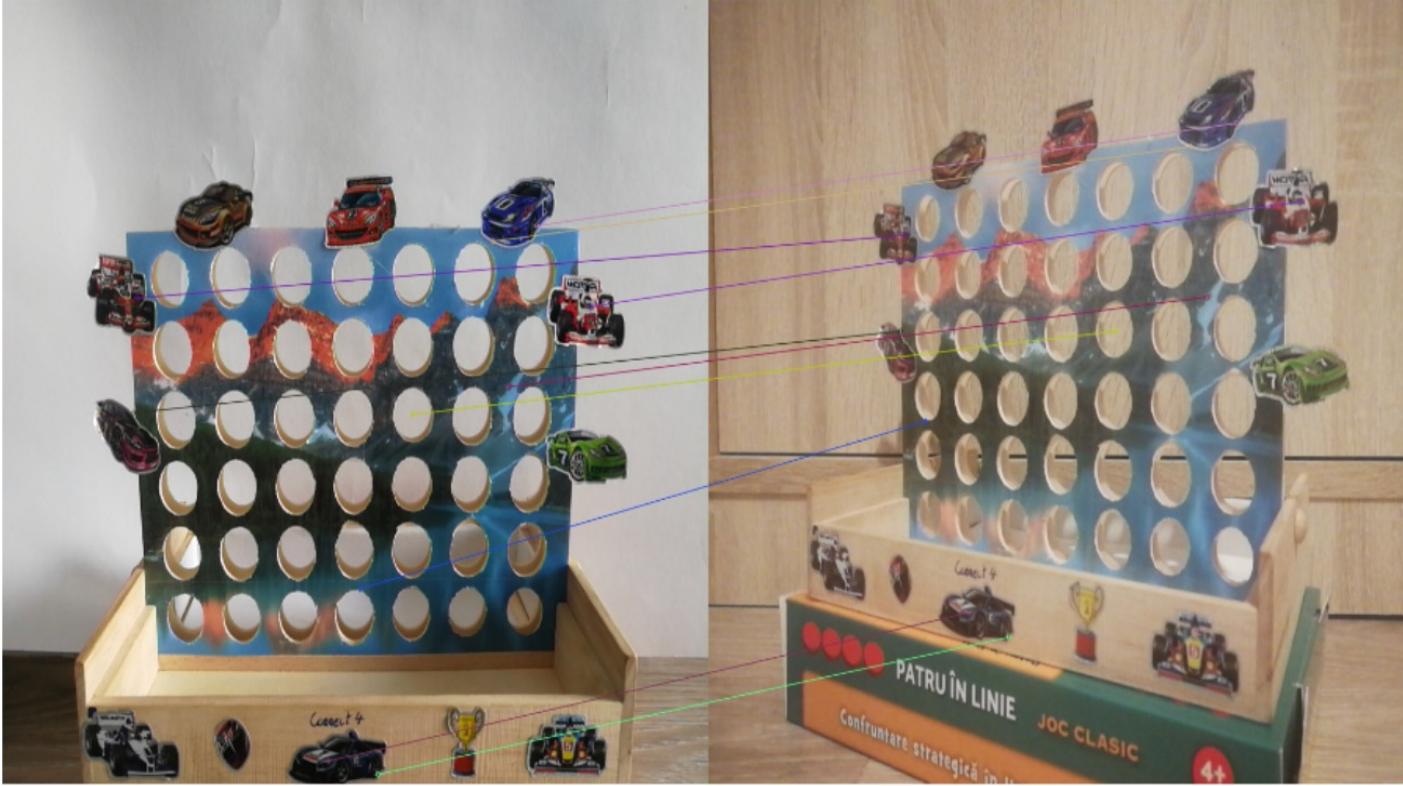


Figura 2.2: Imagine cu corespondentele găsite

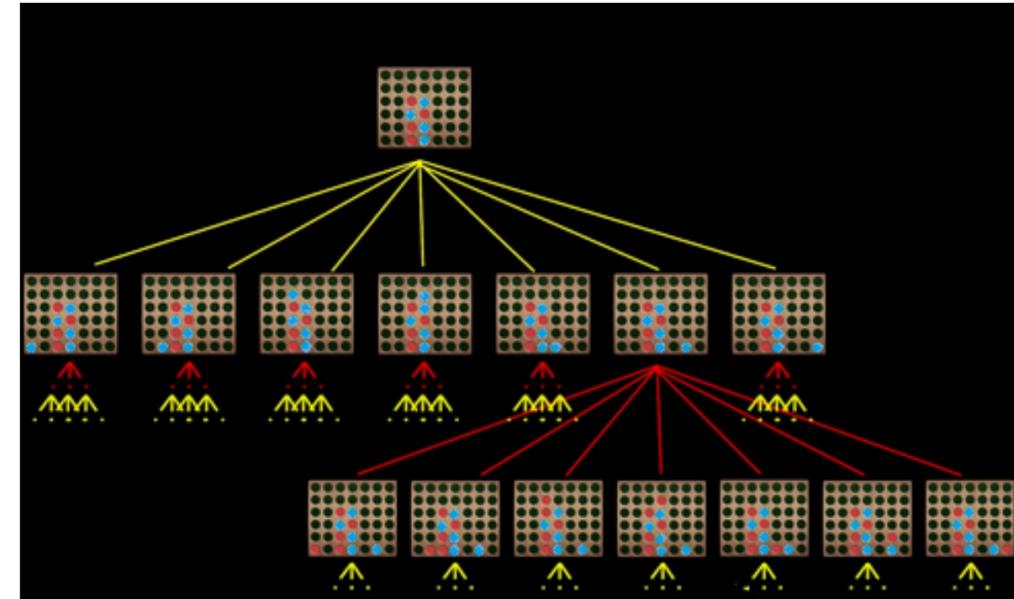


Figura 3.1: Fragment în parcurgea recursivă a algoritmului MinMax

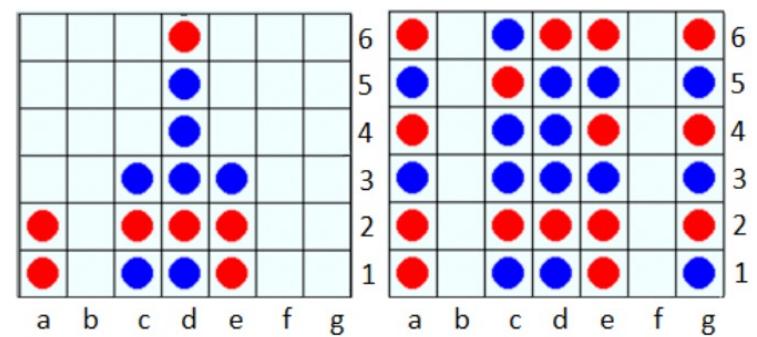


Figura 3.7: Exemplu de Zugzwang inițial și incipient.

Proiect de diplomă

AGENT INTELIGENT PENTRU JOCUL QWIRKLE

Absvent

Ionescu Alexandru-Theodor

Coordonator științific

Conf. Dr. Alexe Bogdan

Asist. drd. Diaconu Alexandra

1	Introducere	8
2	Descrierea Jocului Qwirkle	10
2.1	Prezentarea jocului	10
2.2	Regulile jocului	11
2.3	Modalitatea de punctare	13
2	Recunoașterea configurației curente de pe tabla de joc	15
3.1	Extragerea datelor jocului	15
3.2	Algoritmi utilizați	19
3.2.1	Descriptorii SIFT	19
3.2.2	Homografia	27
3.2.3	ResNet-18 și SVM liniar	29
3.3	Abordare alternativă	30
4	Implementarea unui jucător automat	35
4.1	Codificarea jocului	35
4.1.1	Agentul determinist bazat pe algoritmul Greedy	36
4.1.2	Aspectul interfeței grafice	37
4.2	Algoritmi de inteligență artificială	40
5	Evaluare experimentală	43
5.1	Rezultatele componente de vedere artificială	43
5.2	Rezultatele performanței algoritmului Greedy	46
6	Concluzii	48
	Bibliografie	50

Score Player1: 9

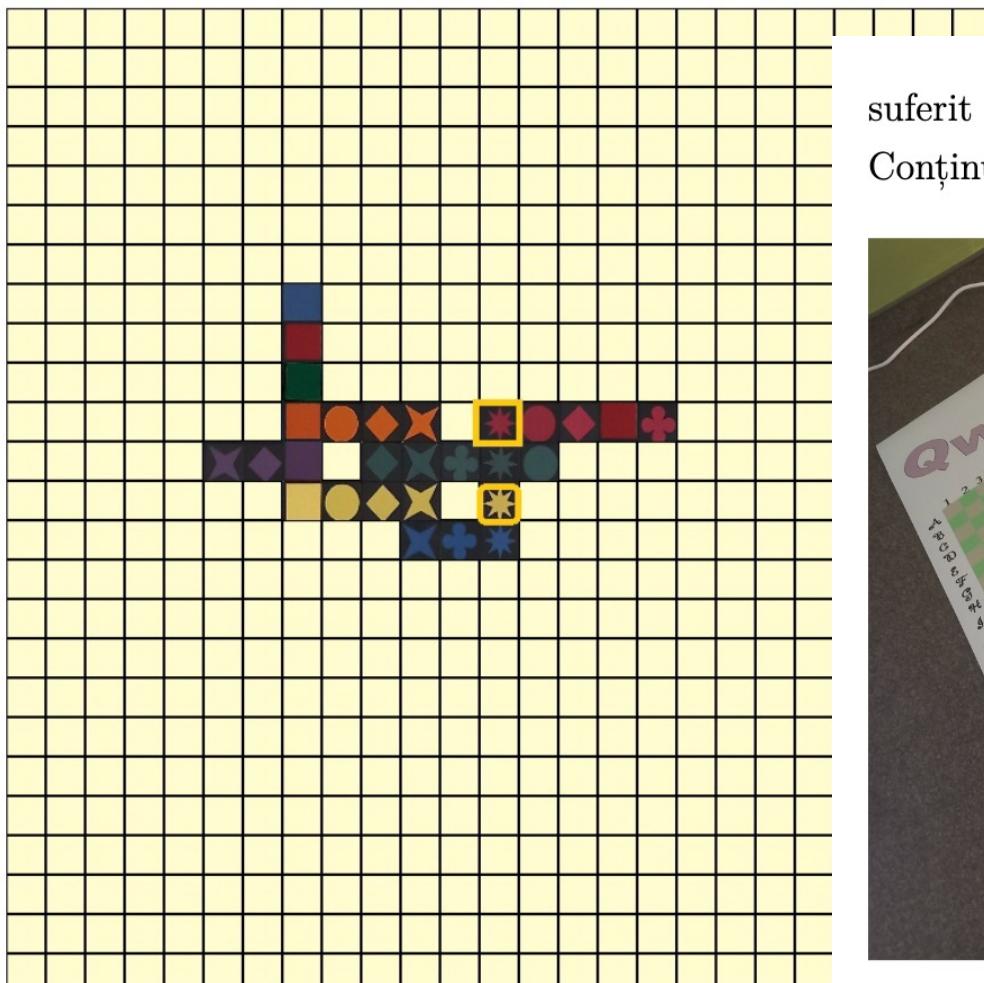


Figura 4.3: Tabla după mutare



Figura 5.2: Schimbarea perspectivei într-o imagine cu unghi mare

IMAGINI

Grafică + Învățare automată

Proiect de diplomă

PARKMATE: VALET INTELIGENT PENTRU PARCARE

Absolvent

Nicoleșcu Robert Alexandru

Coordonator științific

Conf. dr. Dumitru Bogdan Alexe

Asis. drd. Alexandra Diaconu

1	Introducere	6
1.1	Prezentarea problemei	6
1.2	Structura lucrării	7
1.3	Etapele algoritmului	8
1.4	Definiții	9
2	Generarea scenei	10
3	Caracteristicile mașinii autonome	14
3.1	Inițierea mașinii	14
3.2	Dotarea mașinii	16
4	Autonomia mașinii	21
4.1	Clasa "CarEngine"	21
4.1.1	Atribute	21
4.1.2	Metode	24
5	Antrenarea modelului	30
5.1	Generarea imaginilor de antrenare	30
5.2	Antrenarea folosind PyTorch	33
5.2.1	PyTorch	33
5.2.2	ResNet18	34
5.2.3	Antrenarea modelului	34
5.3	Conversia modelului	35
5.3.1	ONNX	35
5.3.2	Convertirea modelului la formatul ONNX?	36
5.3.3	Soluții alternative care nu necesită conversia modelului la ONNX	36



Figura 2.1: Exemplul 1 de scenă generată



Figura 2.2: Exemplul 2 de scenă generată



Figura 5.4: Exemplu negativ de imagine "Busy"

SUNET

Translatarea vorbirii în text

Proiect de diplomă

COMPARAREA UNOR MODELE DE TRANSLATARE A VORBIRII ÎN TEXT PENTRU LIMBA ROMÂNĂ

Absolvent

Vînaga Mădălina

³Coordonator științific

Conf. dr. Alexe Bogdan-Dumitru

Asist. drd. Diaconu Alexandra

1	Introducere	8
1.1	Obiective și scop	8
1.2	Structura lucrării de diplomă	9
2	Preliminarii	11
2.1	Sunetul	11
2.2	Spectrograme	13
2.3	Transformator	15
3	¹⁸ Seturi de date	18
3.1	Seturi de date existente	18
3.2	Setul de date ProTV News	21
3.3	Setul de date Romanian Kids Songs - RKS	27
4	Modele pentru translatarea vorbirii în text	28
4.1	Descrierea modelelor existente	28
4.2	Modelul pentru antrenare cu noile seturi de date	31
5	Evaluarea modelelor	33
5.1	Metrica de validare	33
5.2	Rezultate obținute pe setul de date ProTV News	35
5.2.1	Testare pe setul de date de la Știrile ProTv	35
5.2.2	Rafinarea modelului Whisper	38
5.2.3	Testare pe setul de date din filme și seriale românești	42
5.3	Rezultate obținute pe setul de date Romanian Kids Songs	44
5.3.1	Testare pe setul de date Romanian Kids Songs	44
5.3.2	Testare pe setul de date cu melodii din diverse genuri muzicale	45
5.4	Specificații tehnice	46
6	Concluzii	47

Tabelul 3.2: Informații despre setul de date ProTv News

Explicăm în ceea ce urmează structura site-ului pentru a fi mai ușor de înteles extragerea videoclipurilor și a textelor aferente. În secțiunea *video* de pe site-ul știrilor ProTV, Figura 3.3, se găsește galeria cu videoclipuri. Aceasta apare ⁶⁵ în partea de jos a paginii și conține pe o pagină 12 video-uri, cu titlurile aferente acestora.

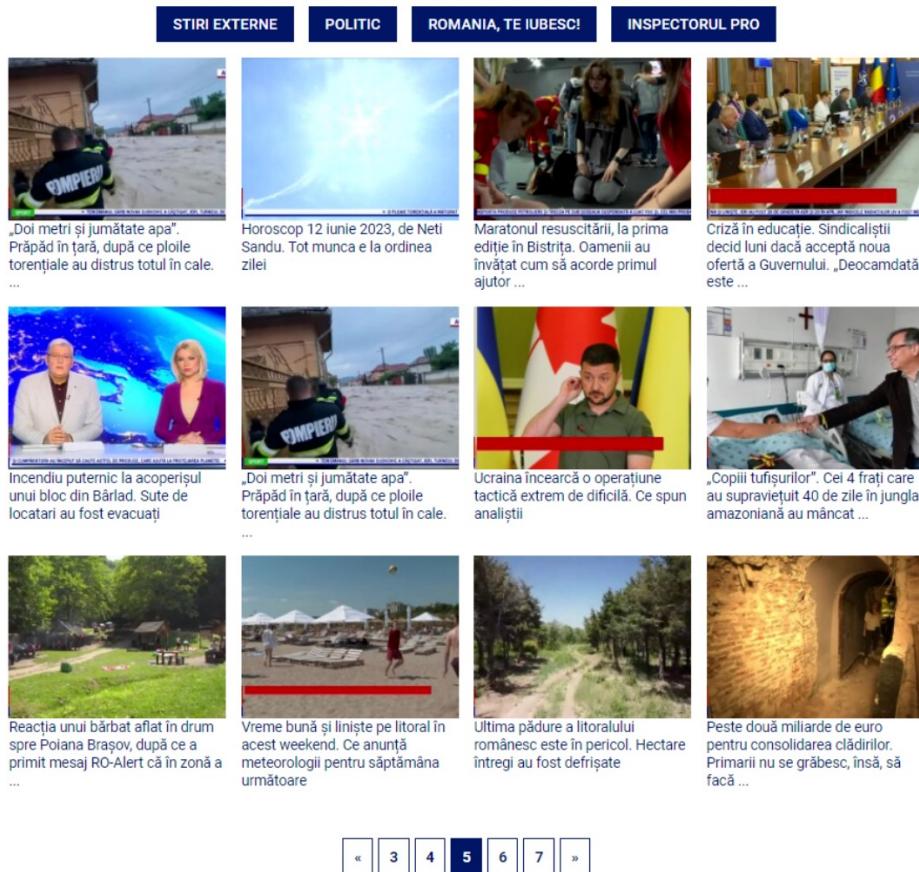


Figura 3.3: Galeria video de pe site-ul Știrile ProTv

Model	Similaritate
Vatis	97.82%
Racai	88.23%
Whisper Small	77.43%
Microsoft Transcribe	93.68%

: *Performanța aplicațiilor existente pe setul de test ProTv News*

Model	Similaritate
Vatis	83.96%
Racai	17.72%
Microsoft Transcribe	57.63%
Whisper Small Original	90.11%
Whisper Small + ProTV News	90.84%
Whisper Small + ProTV News + RKS	94.38%

Tabelul 5.11: Prezentarea similarității pentru setul de test *Romanian Kids Songs*

În Tabelul 5.12 se află câteva exemple din transcrierile modelelor *Vatis*, *Whisper Small* și modelul propriu pentru melodii. Se pot observa îmbunătățiri semnificative, în special la nivelul interjecțiilor și onomatopeelor. Înregistrările cu melodii pentru copii au contribuit la creșterea performanței modelului în transcrierea corectă a acestora.

Original	Vatis	Whisper Small Original	Whisper Small + ProTV News + RKS
oușoarele de Paști	douoarele de paști	două ușoarele de paști	oușoarele de paști
Sprinteni și ocheșei	sprinten fiocheșei	Sprint pensii ocheșei	sprintem și o cheșei
I i i veniți copii	i menis copil	Ii ii veni scopi	i i i veni scopii
Ham ham ham lângă bordei	amamlână bordei	Ham ham ham lângă borde	ham ham ham lângă bordei

Tabelul 5.12: Exemple de transcrieri pentru diferite modele

IMAGINI

Web + Învățare automată

Aplicație Web Pentru Compararea Prețurilor Din Magazinele Farmaceutice

Absolvent

Kayed Amar

Coordonator științific
Conf. dr. Bogdan Alexe

Introducere	5
1.1 Motivație	5
1.2 Aplicații similare și piața aplicațiilor de comparare a prețurilor	5
1.3 Cu ce se diferențiază aplicația propusă	7
1.4 Structura Lucrării	7
1.4.1 Extragerea datelor	7
1.4.2 Arhitectura UI și Back-End	8
Prezentarea tehnologiilor folosite	10
2 Python	10
2.1.1 Flask	10
2.1.2 Requests	10
2.1.3 BeautifulSoup	11
2.1.4 Selenium	11
2.1.5 PIL	11
2.2 ²³ ASP.NET Core 6 Web API	12
2.2.1 Entity Framework Core	12
2.2.2 Identity JSON Web Tokens	13
2.3 Angular	14
2.3.1 Angular Material	14
2.3.2 SCSS	14
2.3.3 Bootstrap	15
2.3.4 Firebase	15
Extragerea și analiza datelor	17
3 Metode de extragere a produselor	17
3.1 Web Scraping	18
3.1.1 Common Scraper	22
3.2 Web Crawling	29
3.3 Asocierea Datelor	30



E-shop Despre Dr. Max Branduri Farmacii Articole Ghid de vară Responsabilitate socială Ofertele lunii

Produse Dr. Max Medicamente fără rețetă Suplimente alimentare Frumusete și îngrijire Dermato-cosmetice Mama și copilul Dieta și nutriție Remedii naturiste Subiecte tabu

Farmacie > Medicamente OTC > Coldrex Lemon, 10 plicuri, Perrigo



Medicamente fără rețetă

Coldrex Lemon, 10 plicuri, Perrigo

Brand: PERRIGO

În stoc

19,49 Lei



* Pretul afișat este cu TVA și este valabil doar pentru comenzi c

☰ Produse

Campanii Promovaționale | Cariere | Farmacii | Info Tei | Noutăți

Farmacia Tei > Medicamente OTC > Sistemul Respirator > Ameliorare Simptome Racelașă Si Grăpa > Coldrex Lemon, 10 plicuri, Perrigo

Coldrex Lemon, 10 plicuri, Perrigo

paracetamol, acid ascorbic, chlorhidrat de fenilefrină

Cod produs: 018653004962

Brand: PERRIGO OTC

Data expirării: 30-06-2025

Vândut de: Farmacia Tei SRL

Document: Prospect

OPȚIUNI LIVRARE

Ridicare rapidă de la casa

Click&Collect

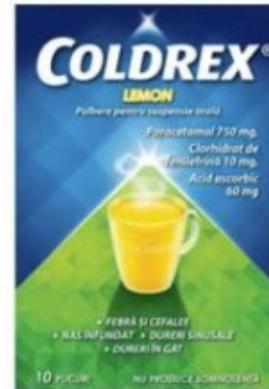
Taxa ambalare: 0,50 LEI

Alege farmacia:

Alege...

Livrare doar în București și Constanța

Pret: de la 10,00 LEI



Disponibilitate și preț în locațiile Farmacia Tei

3.3 Asocierea Datelor

Odată ce am extras datele folosind crawler-ul anterior menționat, se ridică problema asocierii acelorași produse din cadrul unor farmacii diferite. Se pot pune următoarele probleme:

- Cum am putea știi ca un anumit produs dintr-o farmacie este exact același cu alt produs din altă farmacie?
- Sub ce criteriu putem asocia produsele ce se aseamănă?
- Cum putem identifica un produs unic?

Există o multitudine de abordări care ar adresa problemele anterior enumerate. Se pot folosi combinații de algoritmi de similaritate de sir de caractere, modele de inteligență artificială pentru extragerea caracteristicilor din numele produsului, modele de vedere artificială pentru calcularea procentului de similaritate între imaginile a două produse, etc.

Soluția ideală ar fi să existe un identificator unic pe care să îl folosim drept cheie primară în cadrul bazei de date. Aceasta cheie primară poate fi reprezentată chiar de codul de bare al fiecărui produs, mai exact codul EAN-13.

EAN-13 este o versiune a sistemului EAN (European Article Numbering) și este recunoscut internațional de către organizația GS1. Acesta constă într-un număr de 13 cifre, care este structurat în trei părți distincte. Primele două sau trei cifre reprezintă codul

Aplicatie pe mobil pentru detectia de Alzheimer prin Inteligenta Artificiala

Lucrare de licenta

⁴³Universitatea din Bucuresti,
Facultatea de Matematica si
Informatica

Autor: Delia Ioana Dumitrescu
Bucuresti, Romania

Coordonator științific: Conf. dr. Alexe Bogdan
Bucuresti, Romania
2022

Lista tablele	4
Lista figuri	5
1 Introducere	6
2 Model	10
2.1 Retele neuronale convolutionale	10
2.2 EfficientNet	10
2.2.1 Metoda de scalare	10
2.2.2 Arhitectura	12
2.3 Utilizare	15
2.3.1 Set de date	15
2.3.2 Preprocesarea datelor	16
2.3.3 Antrenare	17
2.3.4 Rezultate	18
3 Interpretabilitatea modelelor	24
3.1 Maparea activarii clasei (CAM)	25
3.2 Maparea activarii clasei pin gradient(Grad-CAM)	27
3.3 Rezultate	28
4 Aplicatie pe mobil	30
4.1 Functionalitati	30
5 Backend	33
5.1 Prezentare	33
5.2 Comunicare prin HTTPS	34
5.3 Autentificare prin JWT	35
5.4 Persistarea datelor	38
6 Related work	40
7 Concluzii	42

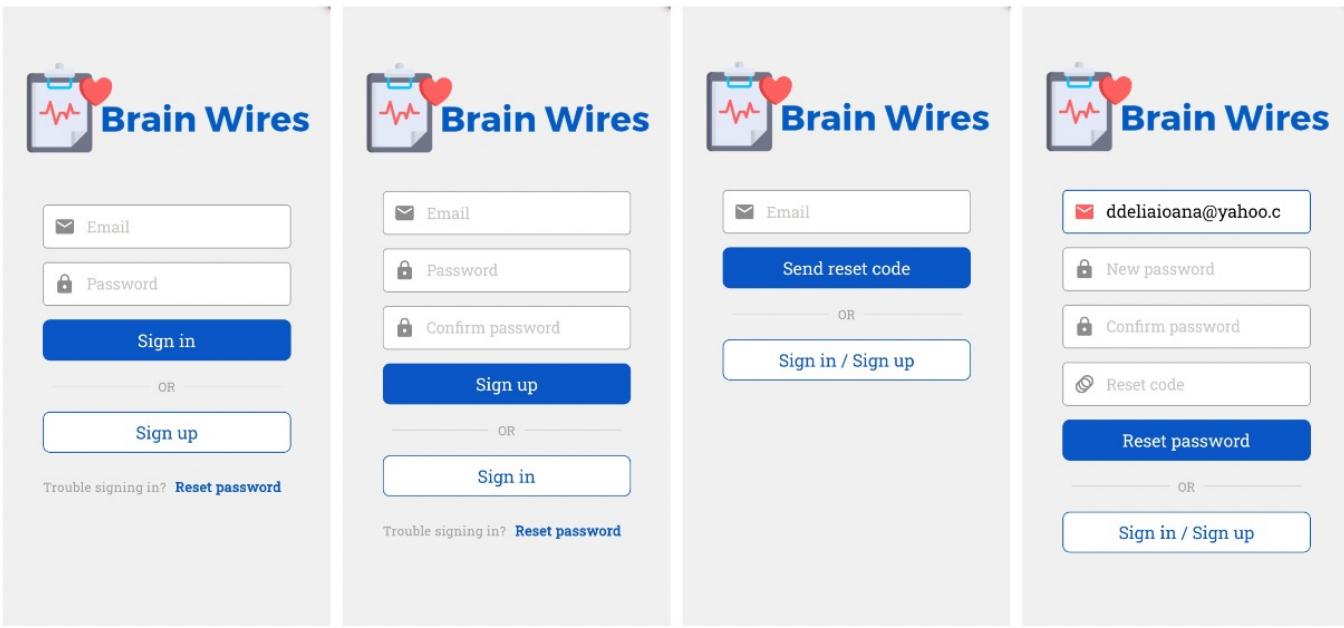


Figure 4.1: Aplicatie pe mobil - Meniu conectare

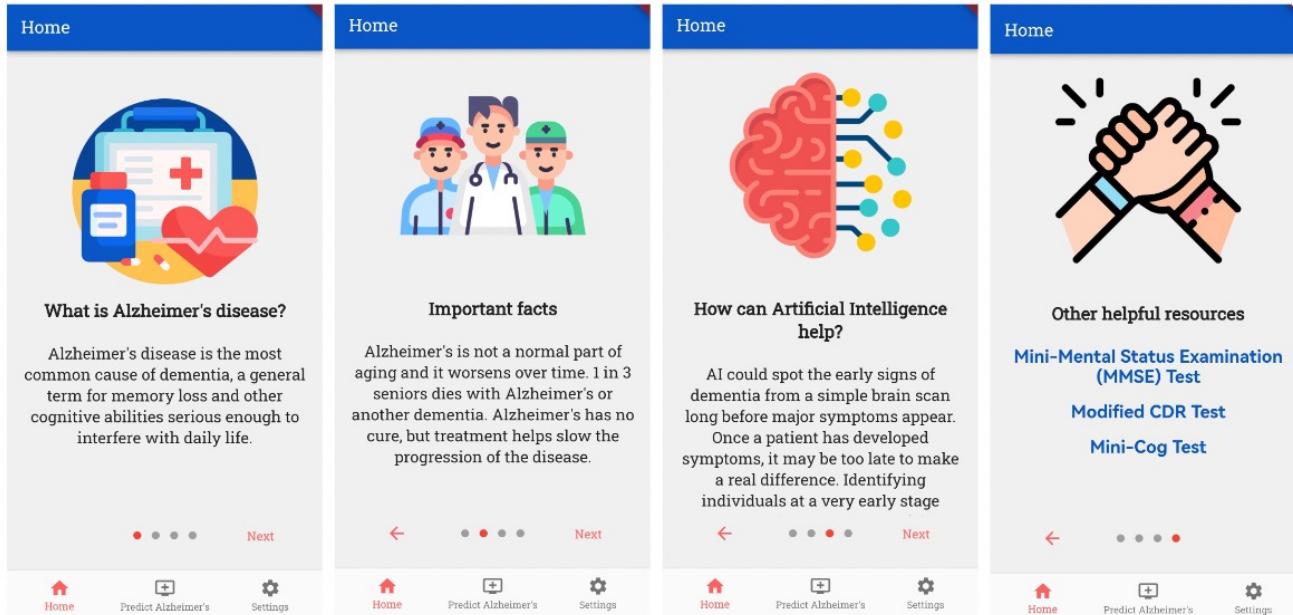


Figure 4.2: Aplicatie pe mobil - Sectiune "Acasa"

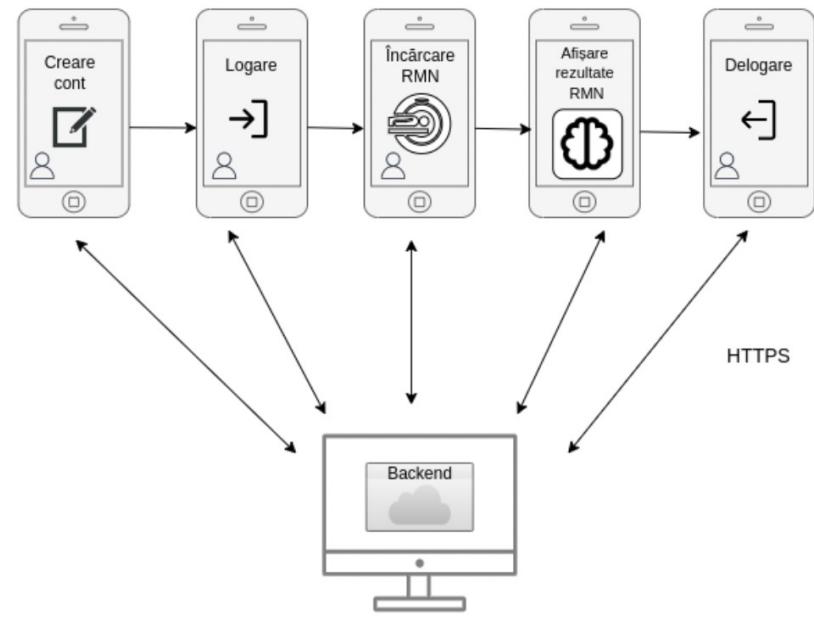


Figure 5.1: Fluxul principal de utilizare al aplicatiei in raport cu partea de backend

Interfața grafică a sistemului dezvoltat este reprezentată de o aplicatie pe mobil, iar fluxul de detectie de Alzheimer din interiorul ei este ilustrat mai jos. Utilizatorul încarcă o imagine RMN, modelul de Inteligență Artificială o procesează și sunt ilustrate stadiile de Alzheimer în care se incadrează și interpretarea deciziei.

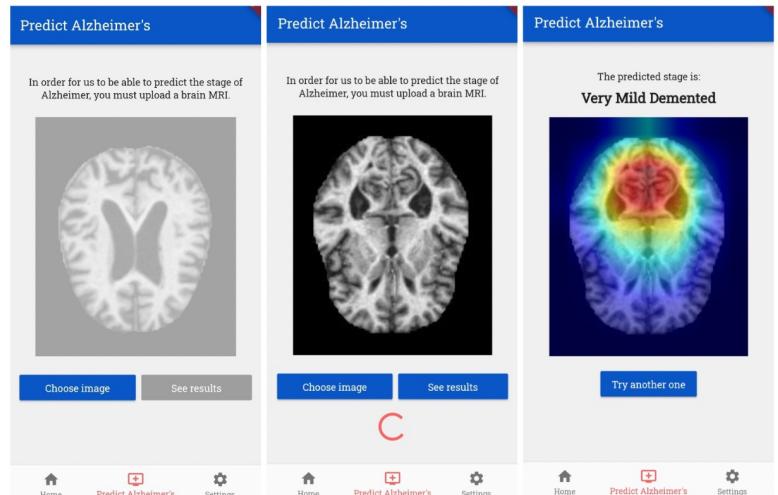


Figure 5.2: Flux detectie de Alzheimer pe mobil

Table 2.3: Acuratete retele antrenate

Model	Acuratete pe validare	Acuratete pe testare
EfficientNet-B0	98.93%	97.71%
EfficientNet-B1	99.75%	98.85%
EfficientNet-B2	99.42%	98.44%
EfficientNet-B3	99.75%	99.42%
EfficientNet-B4	99.42%	98.28%

Se poate observa mai jos evolutia erorii pe datele de antrenare si pe cele de validare. Pe masura ce reteaua este antrenata timp de mai multe epoci, atat eroarea de antrenare cat si cea de validare scad, ajungand aproape de 0 la finalul antrenarii.

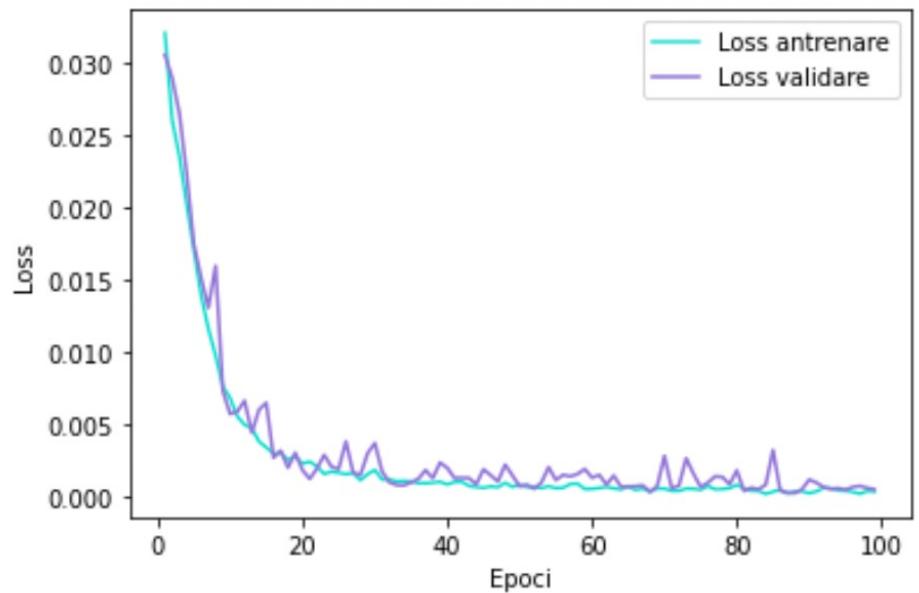


Figure 2.4: Evolutia erorii pe datele de antrenare si de validare

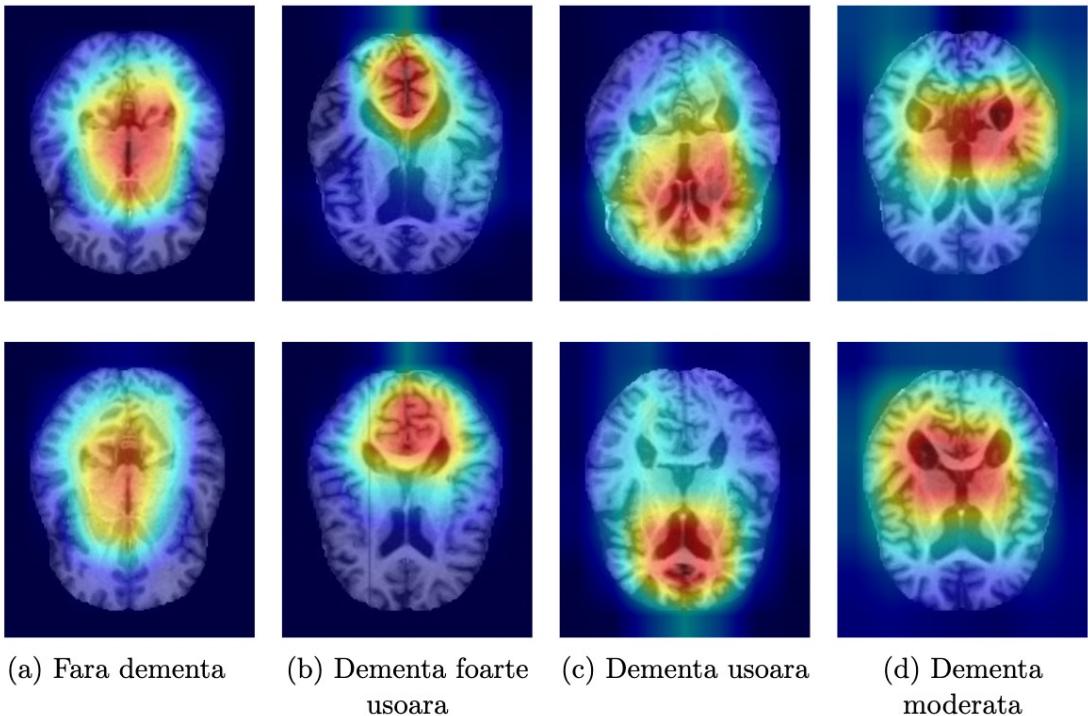


Figure 3.5: Exemplificare aplicare Grad-CAM pe RMN-uri

TEXT

Proiect de diplomă

**AGENT INTELIGENT PENTRU
AUTOMATIZAREA SERVICIULUI DE
SUPORT CLIENTI**

Absolvent
Bălmău Dragoș-Constantin

⁶²Coordonator științific
Conf. Dr. Alexe Dumitru-Bogdan
Asist. drd. Diaconu Alexandra

1	Introducere	15
1.1	Prezentarea problemei	15
1.2	Structura tezei	17
2	Fundamente teoretice	19
2.1	Rețele neuronale	19
2.1.1	Rețele neuronale adânci	20
2.2	Modele de încorporare a propozițiilor	20
2.2.1	Codificator universal pentru propoziții	21
2.3	Similaritate	23
2.3.1	Similaritatea cosinus	23
2.3.2	Coeficientul de corelație Pearson	24
3	Setul de date	27
3.1	Documentația oficială	27
3.2	Generarea datelor de testare și validare	29
4	Implementare și rezultate	33
4.1	Metode de implementare	33
4.2	Metode de testare și evaluare	34
4.3	Scenarii și modalități de testare	35
4.3.1	Modelul data2vec-text-base	36
4.3.2	Modelul all-mpnet-base-v2	37
4.3.3	Modelul USE v4	38
4.4	Fluxul de lucru și interfața	40

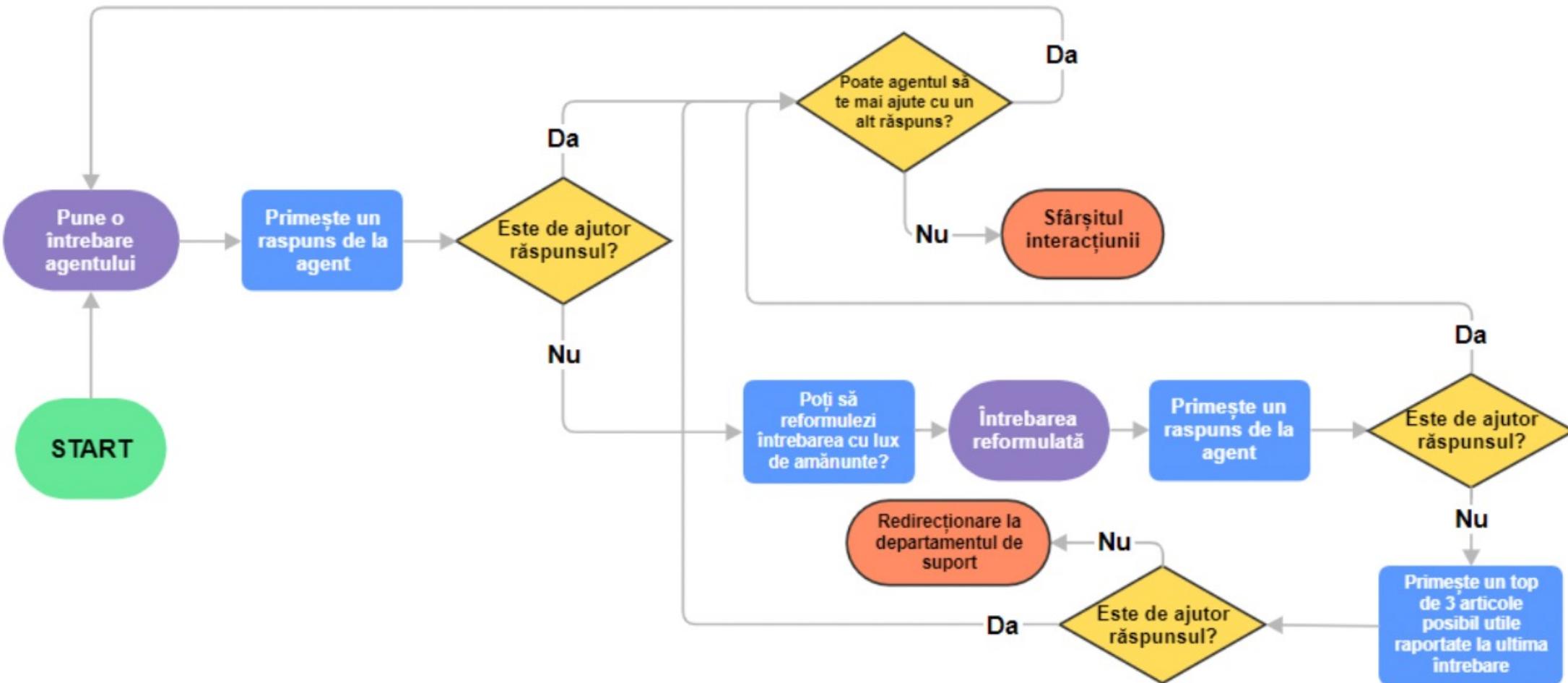


Figura 1.1: Fluxul de interacțiune cu agentul inteligent

Metode		Scenariul 1	Scenariul 2	Scenariul 3	Scenariul 4	Scenariul 5	Scenariul 6	Formula de calcul a similarității
Răspunsul se găsește în top K similarități	Top 1	0.2423	0.2421	0.2708	0.2651	0.3013	0.2917	Similaritatea cosinus
	Top 3	0.2525	0.2672	0.2766	0.2818	0.3101	0.3161	
	Top 5	0.2649	0.2698	0.2877	0.3091	0.3437	0.3552	
	Top 7	0.2764	0.2844	0.3319	0.3100	0.3437	0.3700	
	Top 9	0.2896	0.2860	0.3497	0.3206	0.3583	0.4162	
Metoda k-NN	3-NN			0.1301			0.1866	Coef. de corelație Pearson
	5-NN			0.1666			0.2189	
	7-NN			0.1761			0.2241	
	9-NN			0.2006			0.2245	
Răspunsul se găsește în top K similarități	Top 1	0.2077	0.2444	0.5116	0.2718	0.3370	0.4979	Coef. de corelație Pearson
	Top 3	0.2569	0.2566	0.5231	0.2823	0.3427	0.6197	
	Top 5	0.2744	0.2602	0.5573	0.3255	0.3488	0.6705	
	Top 7	0.2958	0.2881	0.6190	0.3602	0.3702	0.6981	
	Top 9	0.3043	0.2904	0.6335	0.3729	0.3953	0.7116	
Metoda k-NN	3-NN			0.1459			0.1996	
	5-NN			0.1604			0.2048	
	7-NN			0.1735			0.2377	
	9-NN			0.1882			0.2557	

Tabelul 4.4: Rezultatele obținute în urma scenarilor și metodelor de evaluare pentru modelul USE v4, în cazul setului de date pentru validare.

Robotică

Lucrare de licență

VEHICUL SEMIAUTONOM CU BRAȚ ROBOTIC

Absolvent

Surdu Bob Alexandru

Coordonatori științifici

¹³**Conf. Dr. Bogdan Alexe**

Drd. Andrei Dumitriu

1	Introducere	4
2	Noțiuni preliminare	7
16	2.1 ROS 2 - Robot Operating System v2	7
	2.2 Algoritmul de control PID	7
	2.3 Cartografiere și localizare simultană - SLAM	8
3	Componenta hardware	9
	3.1 Componentele hardware ale robotului	9
	3.2 Conexiunile dintre componente	15
	3.3 Asamblarea hardware a robotului	16
4	Componenta software	18
	4.1 Arhitectura Software	18
	4.2 Software-ul pentru Nvidia Jetson Orin Nano	19
	4.3 Software-ul care rulează pe microcontrolerul Arduino Mega	29
5	Concluzii	31
6	Anexă - Listă de acronime	33



Figure 1.1: Robotul asamblat în forma finală

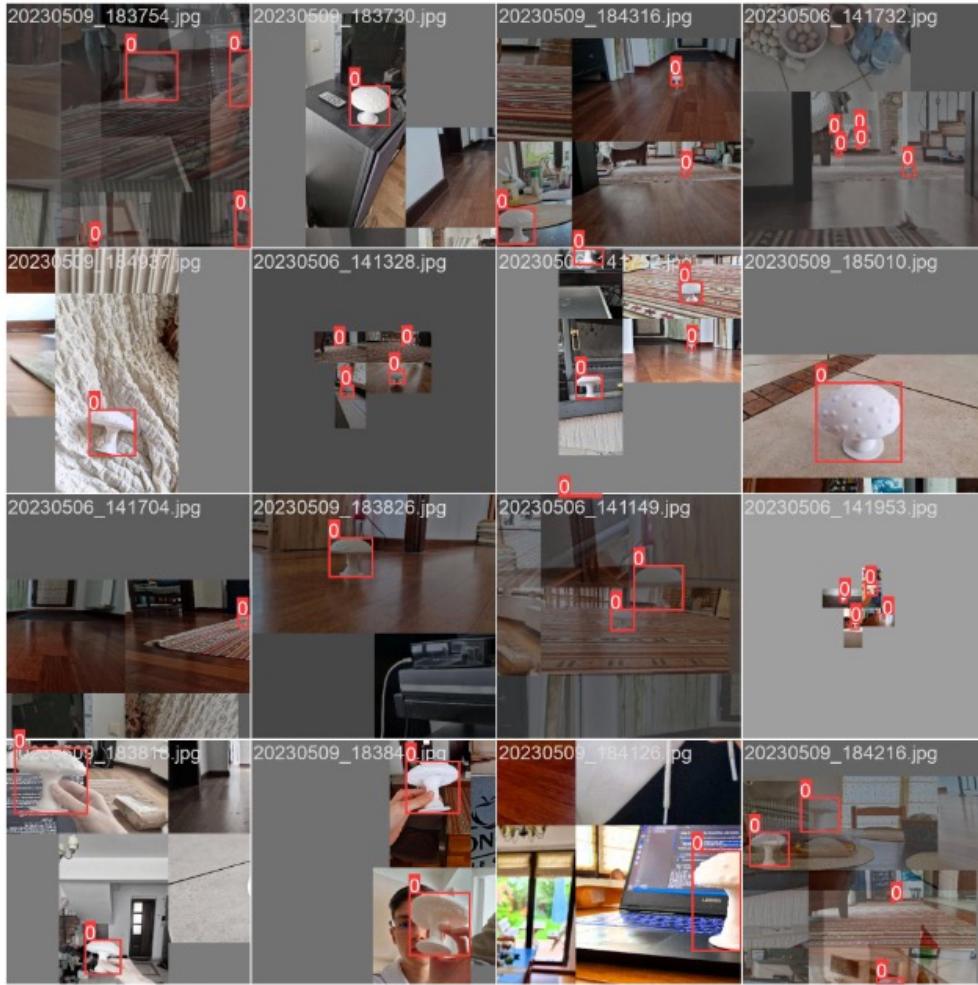


Figure 4.6: Augmentarea datelor de antrenare

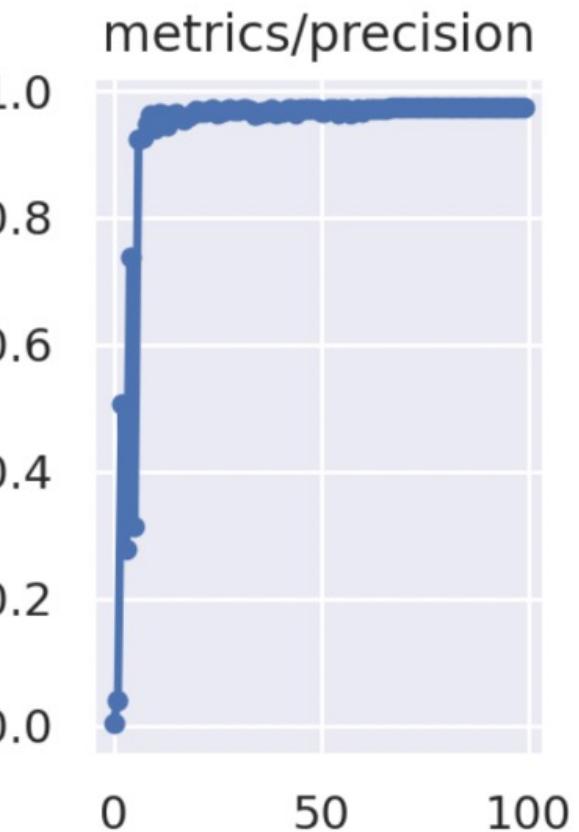


Figure 4.7: Evolutia preciziei în timpul antrenării

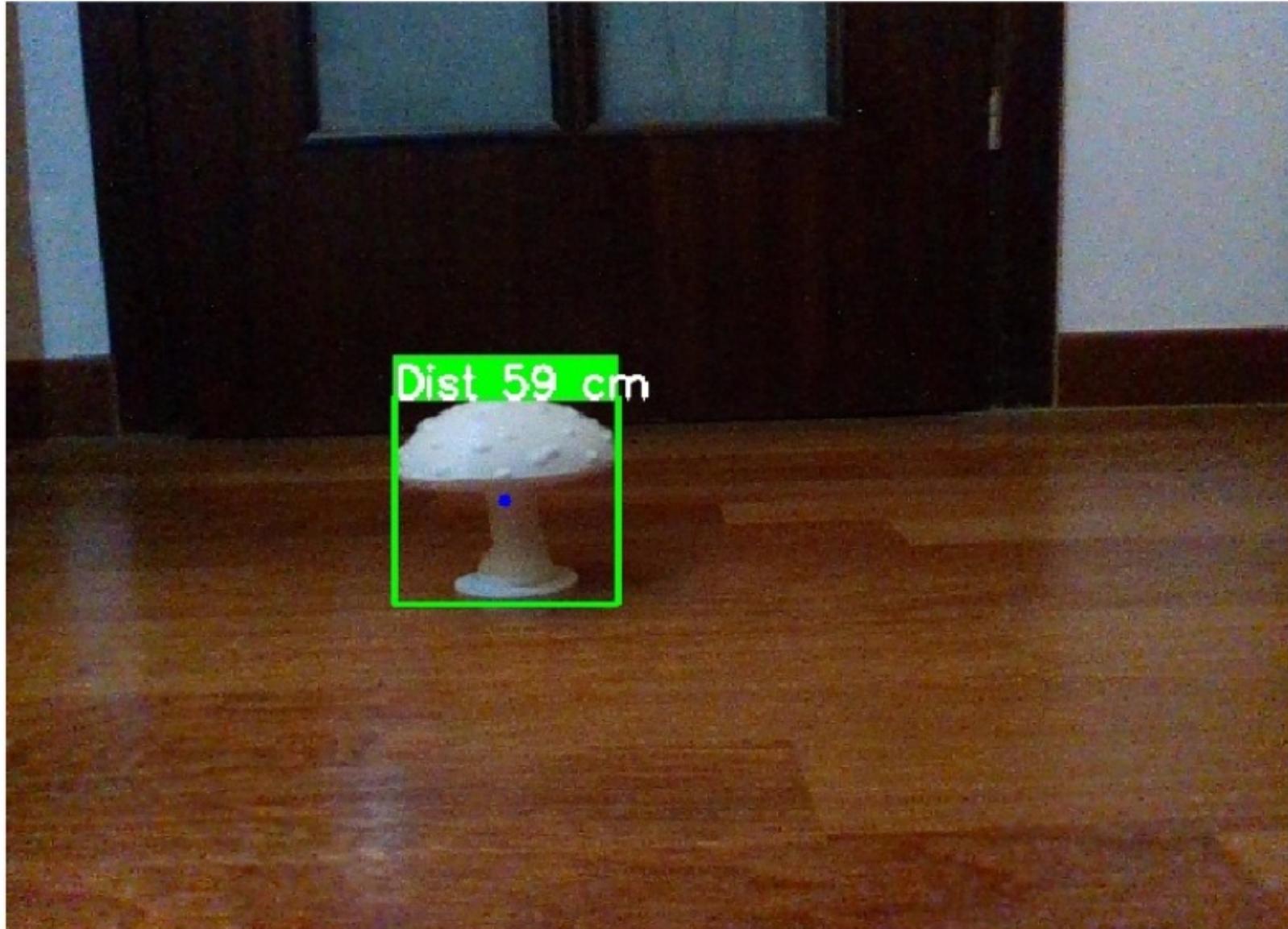


Figure 4.8: Obiect detectat folosind YOLO

Lucrare de licență

Frigiderul Intelligent

Absvent

Alexandra Buruiană

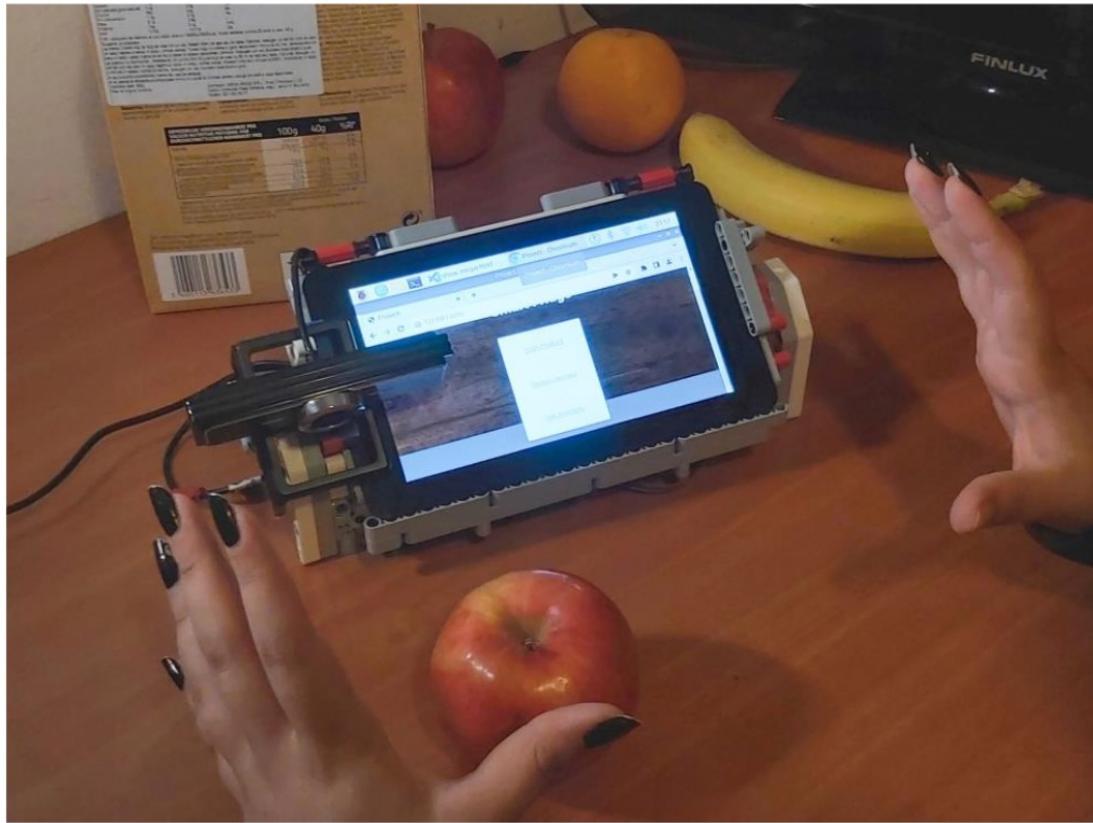
Coordonatori Științifici

Conf. Dr. Bogdan Alexe

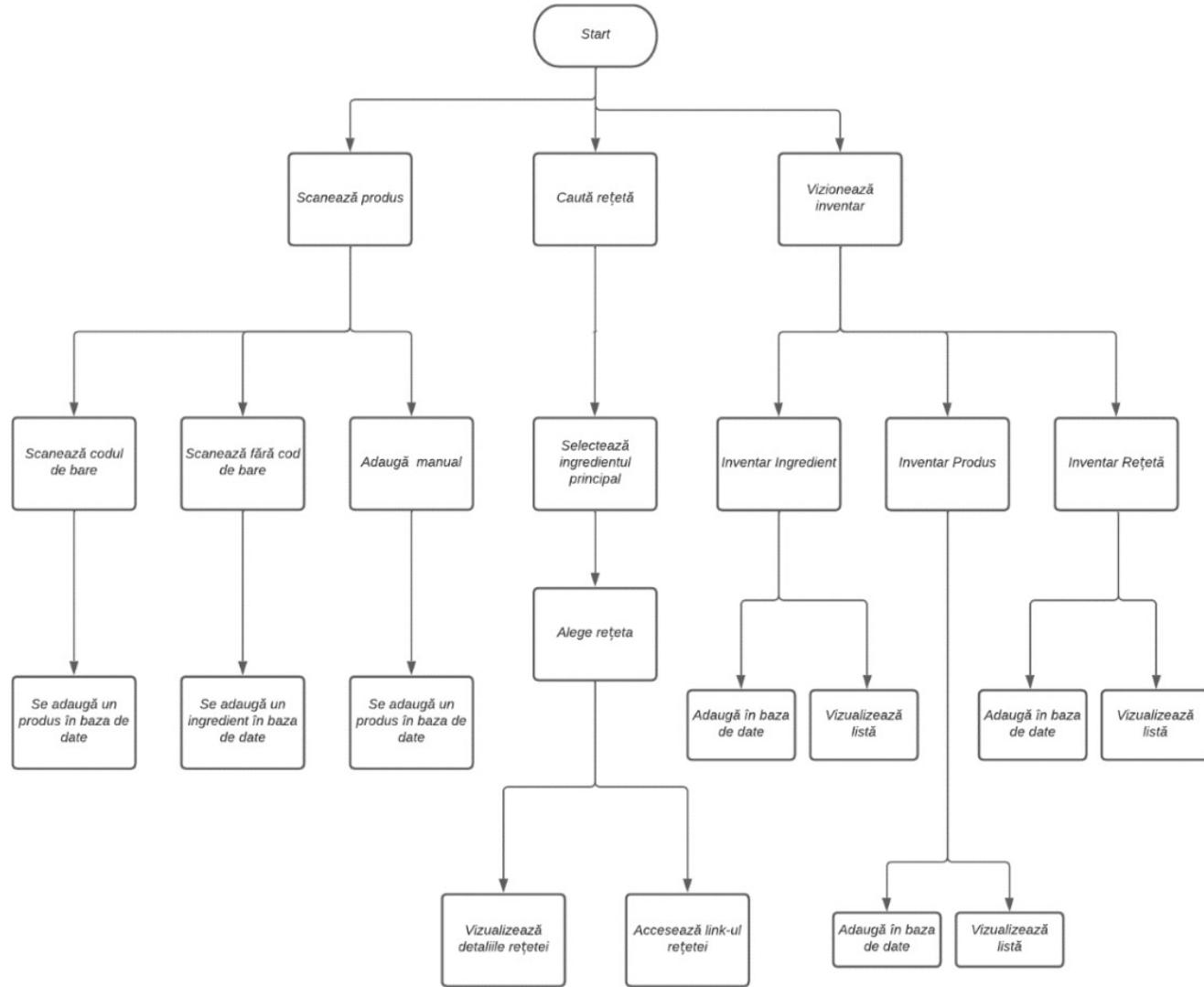
Prof. Andrei Dumitriu

Listă Figuri	5
Listă Abrevieri.....	6
Capitolul I. Introducere	7
6.1 Scopul Lucrării	8
I.2 Structura Lucrării	9
Capitolul II. Componenta Hardware	10
II.1 Minicalculatorul Raspberry Pi	10
II.2 Ecranul Touch-screen	11
II.3 Camera web	11
II.4 Conectarea componentelor.....	12
II.5 Instrumente software folosite.....	13
Capitolul III. Componenta Software	14
III. 1 Interpretarea de imagini	14
III.1.1 Interpretarea de coduri de bare.....	14
III.1.2 Detectarea de obiecte folosind vedere computerizată	15
III. 2 Accesarea și interpretarea paginilor web	17
III.2.1 Requests	17
III.2.2 Beautiful Soup	17
III. 3 Tehnologii de dezvoltare web.....	17
III.3.1 Django	17
III.3.2 HTML, CSS și Javascript	18
III.3.3 Bootstrap	18
Capitolul IV. Prezentarea aplicației	19
IV.1. Schema bazei de date	19
IV.2. Resurse externe.....	20
IV.3 Funcționalitățile aplicației	21
IV.3.1 Vizionarea inventarului.....	21





Figură 1.1 Frigiderul Inteligenț - imagine de ansamblu a componentelor de hardware și software



Figură 4.14 Fluxul acțiunilor – diagrama de desfășurare a acțiunilor utilizatorului

ALTE DATE

Lucrare de licență

ÎNVĂȚARE AUTOMATĂ ÎN PREDICTION REZULTATELOR MECIURILOR DE FOTBAL

Absolvent

Sebastian-Ioan ENE

14 Coordonator științific

Conf. Dr. Bogdan ALEXE

1 Introducere	5
1.1 Motivație	5
1.2 Problema abordată	6
1.3 Structura lucrării de licență	8
2 Construcția setului de date	9
2.1 Date colectate folosind platforma Understat.com	9
2.2 Date colectate folosind platforma Eurosport.com	11
2.3 Date colectate folosind platforma Football-data.co.uk	12
3 Descrierea modelului	14
3.1 Derivarea coeficienților de regresie liniară specifici per echipă	17
3.2 Derivarea coeficienților de regresie liniară globali	20
3.3 Optimizarea timpului de rulare a codului	21
4 Evaluare experimentală	23
4.1 Evaluarea modelului folosind cote	23
4.2 Strategia cu rezultate aleatoare	24
4.3 Strategia cu produs supraunitar pentru casa de pariuri Bet365	25
4.4 Rezultatele folosind caracteristica xG cu coeficienți per echipă	26
4.5 Rezultatele folosind caracteristica xG cu coeficienți globali	27
4.6 Rezultatele folosind caracteristica goluri înscrise cu coeficienți per echipă .	28
4.7 Rezultatele folosind caracteristica goluri înscrise cu coeficienți globali . .	29
4.8 Interpretarea rezultatelor	30
5 Posibile îmbunătățiri	32
6 Concluzii	35

În figura 3.2 este redat calculul acestor probabilități folosind Pycharm/Python.

Probabilitatea scorului Bayern Munich- Hoffenheim

Probabilitatea A Hoffenheim

Probabilitatea H Bayern Munich	# 0	# 1	# 2	# 3	# 4	# 5	# 6	# 7	# 8	# 9
0	0.00502	0.01565	0.02440	0.02536	0.01976	0.01232	0.00640	0.00285	0.00111	0.00039
1	0.01093	0.03407	0.05311	0.05519	0.04302	0.02682	0.01394	0.00621	0.00242	0.00084
2	0.01189	0.03708	0.05780	0.06006	0.04681	0.02919	0.01517	0.00676	0.00263	0.00091
3	0.00863	0.02690	0.04193	0.04357	0.03396	0.02118	0.01100	0.00490	0.00191	0.00066
4	0.00469	0.01464	0.02281	0.02371	0.01848	0.01152	0.00599	0.00267	0.00104	0.00036
5	0.00204	0.00637	0.00993	0.01032	0.00804	0.00502	0.00261	0.00116	0.00045	0.00016
6	0.00074	0.00231	0.00360	0.00374	0.00292	0.00182	0.00095	0.00042	0.00016	0.00006
7	0.00023	0.00072	0.00112	0.00116	0.00091	0.00057	0.00029	0.00013	0.00005	0.00002
8	0.00006	0.00020	0.00030	0.00032	0.00025	0.00015	0.00008	0.00004	0.00001	0.00000
9	0.00002	0.00005	0.00007	0.00008	0.00006	0.00004	0.00002	0.00001	0.00000	0.00000

Figura 3.2: Calculul probabilităților de scor Bayern Munich- Hoffenheim în Python

Indice	Liga	Sezon	Rnd[%]	RndMax[%]	Max%	Max2[%]	B365[%]	B365_2[%]
0	EPL	2019/2020	-1.96	4.66	20.65	-0.95	11.15	-13.32
1	EPL	2020/2021	-0.14	5.53	30.61	22.19	25.94	23.12
2	EPL	2021/2022	-7.87	-2.42	2.83	4.07	-4.27	-0.84
3	La_liga	2019/2020	-6.72	-1.55	-2.69	3.45	-12.03	-1.02
4	La_liga	2020/2021	-5.12	0.53	-2.24	0.93	-6.38	-4.3
5	La_liga	2021/2022	-6.03	-0.25	0.32	-6.28	-3.95	-17.38
6	Bundesliga	2019/2020	-3.74	2.12	6.75	25.43	0.49	18.35
7	Bundesliga	2020/2021	-3.15	2.42	12.73	3.86	6.83	17.58
8	Bundesliga	2021/2022	-2.9	3.94	2.6	-0.24	-6.49	-1.14
9	SerieA	2019/2020	-5.59	0.36	19.07	-8.7	6.96	-9.82
10	SerieA	2020/2021	-9.67	-3.72	-16.19	17.48	-19.76	-11.06
11	SerieA	2021/2022	-4.71	2.1	23.54	-4.01	13.57	-8.27
12	Ligue1	2019/2020	-5.74	-0.79	9.79	5.82	-0.34	-13.91
13	Ligue1	2020/2021	-4.81	2.49	25.22	-20.44	15.47	-35.06
14	Ligue1	2021/2022	-6	-0.31	3.27	13.2	-1.87	-6.8
Procente_mieie[%]			-4.94	1.01	9.08	3.72	1.69	-4.26

Tabelul 4.5: Câștigul/pierderea în procente pentru toate ligile considerate având la bază caracteristica goluri înscrise cu coeficienți globali