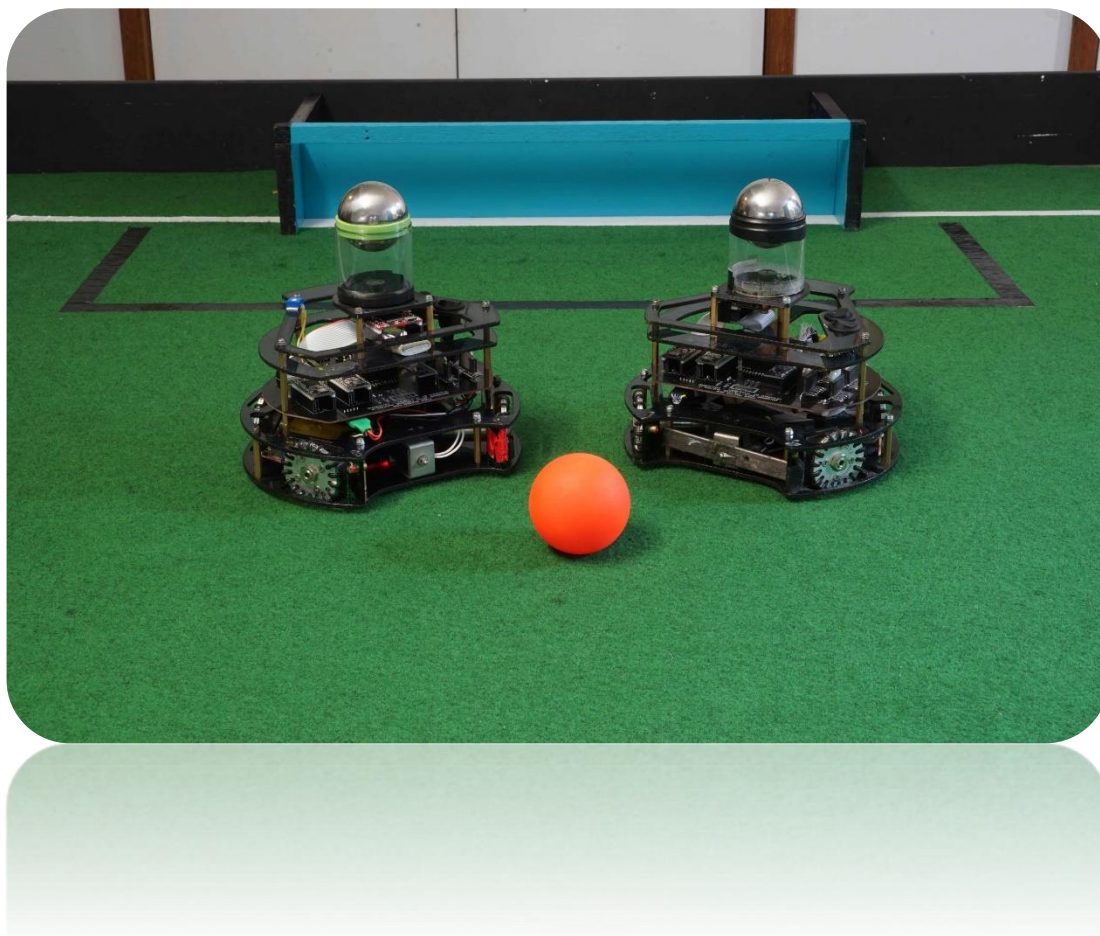


UNIVERSITATEA “POLITEHNICA” DIN TIMISOARA
FACULTATEA DE AUTOMATICĂȘI CALCULATOARE DEPARTAMENTUL
AUTOMATICĂ ȘI INFORMATICĂ APLICATĂ



Coordonator:

-Conf.dr.ing. Florin DRĂGAN

-As. ing. Emil VOIȘAN

Autori

-Trifaș Robert-Eugen

-Podean Benjamin-Daniel

-Brânzea Alexandru

Cuprins

1.Introducere.....	3
1.1 Context.....	3
1.2 Obiectiv.....	3
2.Prezentarea temei.....	4
3.Tehnologii utilizate.....	5
4.Ghidul programatorului.....	6
5.Ghidul utilizatorului.....	7
6.Rezultate.....	8
6.1. Generare lume pentru simulare.....	8
6.2. Ball detection & Ball finder.....	9
7.Concluzii.....	10
8.Bibliografie.....	11

Introducere

Robotică mobilă este un domeniu interdisciplinar cu aplicații variate, de la logistică la divertisment. Platforma TurtleBot3, modulară și accesibilă, împreună cu ROS și simulările în Gazebo, oferă un mediu ideal pentru dezvoltarea și testarea aplicațiilor robotice într-un mod sigur și eficient.

Proiectul propune programarea unui TurtleBot3 pentru a detecta, urmări și manipula o minge într-un teren virtual, direcționând-o într-o poartă. Acest scenariu imită un joc simplu de fotbal robotic și integrează procesarea imaginilor, controlul mișcării și simularea fizică.

Context (1.1)

Robotica mobilă implică percepție și interacțiune autonomă cu mediul. Simulările în Gazebo oferă un cadru reproductibil și economic pentru testarea funcționalităților complexe, eliminând riscurile testării fizice.

Obiectiv (1.2)

Scopul proiectului este dezvoltarea unui sistem robotic capabil să perceapă și să acționeze eficient. Activitățile includ:

1. Crearea unui teren virtual cu minge și poartă.
2. Dezvoltarea algoritmilor pentru detectarea mingii.
3. Implementarea controlului mișcării pentru direcționarea mingii.
4. Validarea performanței prin simulări.



Prezentarea temei

Proiectul urmărește dezvoltarea unui sistem robotic capabil să detecteze, urmărească și să manipuleze o minge într-un teren virtual, direcționând-o către o poartă. Scenariul combină percepția senzorială, procesarea imaginilor, controlul mișcării și interacțiunea robot-minge, reprezentând o provocare clasică în robotică mobilă.

Elemente cheie ale temei

1. **Terenul virtual:** Creat în **Gazebo**, include o suprafață de joc, margini și o poartă. Mingea este un obiect dinamic cu comportament fizic realist.
2. **Robotul TurtleBot3:** Echipat cu camera RGB-D, LiDAR și senzori IMU, configurat pentru detectarea și manipularea mingii.
3. **Procesarea imaginii:** Algoritmi dezvoltați în **Python** cu **OpenCV** permit detectarea mingii și estimarea poziției sale.
4. **Controlul mișcării:** Algoritmi de control (ex. PID) ajustează viteza și direcția robotului pentru urmărirea mingii.
5. **Interacțiunea robot-minge:** Robotul direcționează mingea prin manevre precise.
6. **Validarea performanțelor:** Teste în diverse configurații pentru evaluarea eficienței și timpului de atingere a obiectivului.

Relevanța proiectului

Proiectul oferă o bază pentru aplicații precum roboți autonomi pentru curățenie, jocuri interactive și platforme educaționale, demonstrând integrarea tehnologiilor avansate într-un scenariu practic.

Tehnologii utilizate

Acest proiect utilizează tehnologii avansate pentru integrarea eficientă a hardware-ului, software-ului și simulării:

1. **Robot Operating System (ROS):** Cadrul software open-source oferă:
 - Gestionarea comunicațiilor între modulele robotului.
 - Controlul hardware-ului pentru senzori și actuatori.
 - Planificarea mișcării pentru navigație autonomă.
2. **Gazebo:** Simulator 3D pentru:
 - Fizică avansată (gravitație, coliziuni).
 - Vizualizare 3D a interacțiunilor robotului în timp real.
 - Integrare completă cu ROS.
3. **TurtleBot3:** Platforma robotizată utilizată, echipată cu:
 - Camera RGB-D, LiDAR și IMU.
 - Design modular și compatibilitate nativă cu ROS.
4. **OpenCV:** Biblioteca de procesare a imaginilor, folosită pentru:
 - Detectarea mingii prin segmentarea culorilor.
 - Filtrarea zgomotului și detectarea contururilor.
 - Estimarea distanței între robot și minge.
5. **Control PID:** Algoritm de control pentru:
 - Ajustarea vitezei și traiectoriei robotului.
 - Corectarea erorilor prin componente proporționale, integrale și derivative.
6. **Simularea în Gazebo:** Permite testarea realistă prin:
 - Simularea gravitației și coliziunilor.
 - Definirea mingii ca obiect dinamic cu proprietăți fizice.
7. **Limbaje de programare:**
 - **Python:** Dezvoltarea algoritmilor de percepție și control.

Aceste tehnologii asigură un sistem robust și eficient pentru atingerea obiectivelor proiectului.

Ghidul programatorului

Actualizare sistem: Toate pachetele sunt la zi.

Instalare ROS 2 Humble: Instalare versiunea completă

Pachete TurtleBot3: Instalează driverele și resursele necesare pentru robot. Configurează variabila de mediu `TURTLEBOT3_MODEL` (ex: `burger`, `waffle`).

Simulator Gazebo: Instalează simulatorul și configurează un fișier `.world` cu:

- Teren de simulare (plan drept, margini).
- Mingi (obiect dinamic)
- Poartă (obiect fix).

Biblioteca OpenCV: Instalează pentru procesarea imaginilor.

Funcții ale proiectului:

- Detectarea și urmărirea mingii folosind camera RGB-D.
- Estimarea poziției și distanței mingii.
- Manipularea mingii cu algoritmi de control (PID).
- Direcționarea mingii către poartă.

Testare: Rulează simularea în Gazebo și validează comportamentul robotului.

Ghidul Utilizatorului

1. Pornirea simulării:

- Deschide Gazebo și încarcă fișierul `.world` care conține terenul, mingea și poarta.
- Asigură-te că TurtleBot3 este poziționat corect în simulare.

2. Funcționalități principale:

- **Detectarea mingii:** Robotul folosește camera RGB-D pentru a identifica mingea și poziția acesteia în teren.
- **Urmărirea mingii:** Robotul se deplasează spre minge, ajustând viteza și direcția folosind algoritmi de control.
- **Manipularea mingii:** Robotul împinge mingea în direcția dorită, evitând obstacolele și marginea terenului.
- **Marcarea golului:** Robotul îndrumă mingea în poartă.

3. Controlul robotului:

- Robotul funcționează autonom, dar poți monitoriza mișcările prin ROS 2.
- Datele senzorilor și starea robotului sunt vizibile în terminal.

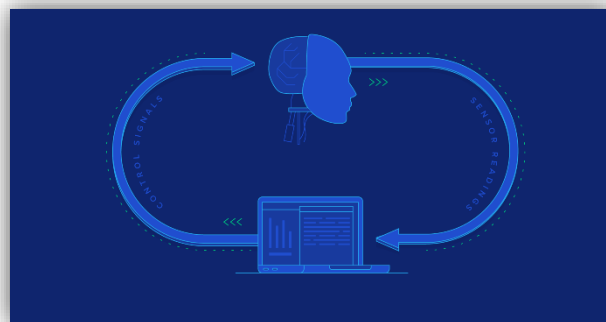
4. Testare și validare:

- Rulează scenarii diferite (poziții ale mingii și robotului).
- Verifică timpul necesar și precizia mișcărilor.

5. Sfaturi utile:

- Ajustează parametrii PID pentru un control mai precis.
- Folosește Rviz pentru vizualizarea traseului și a senzorilor.

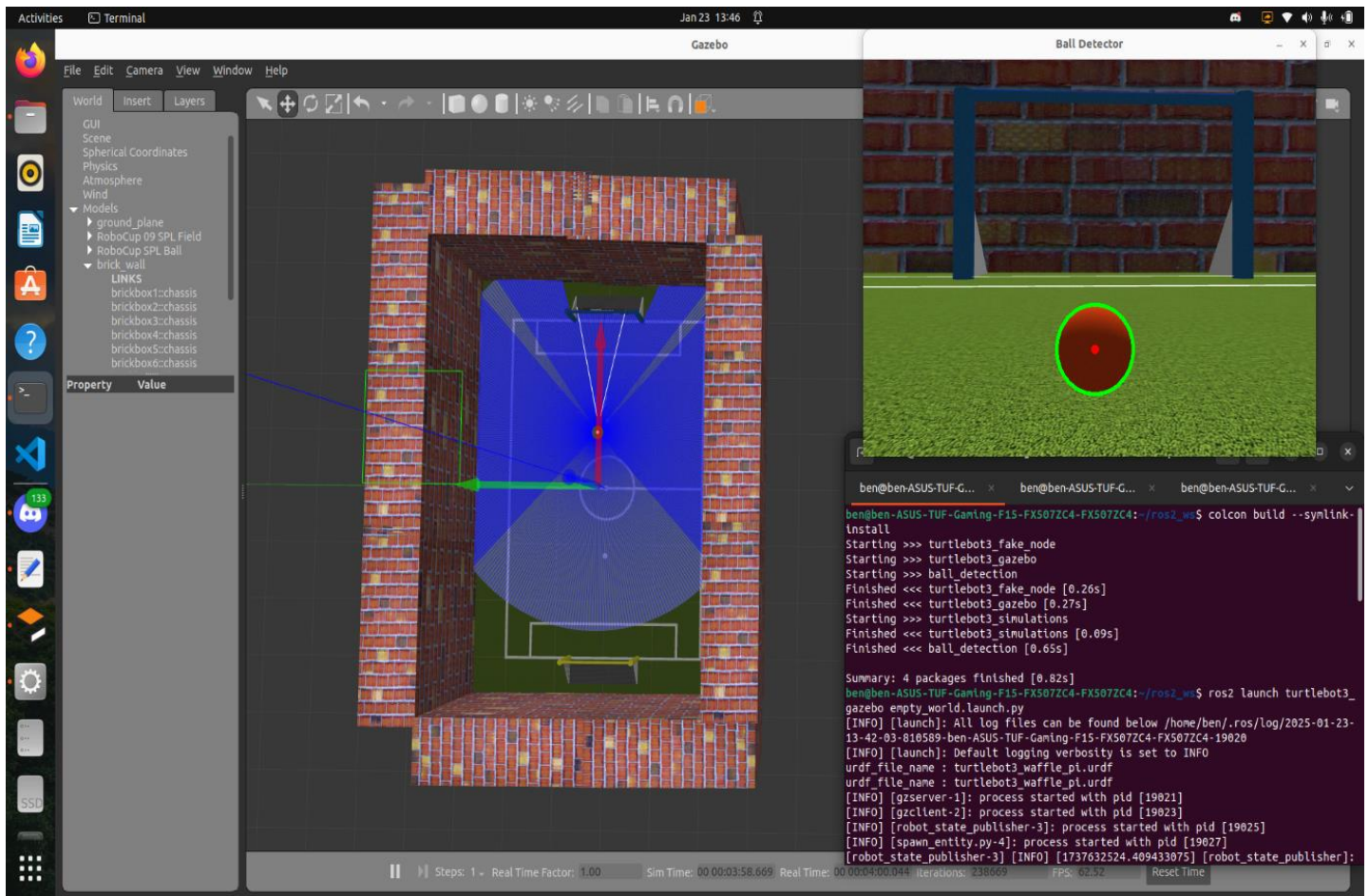
Ghidul permite utilizatorilor să pornească, să monitorizeze și să ajusteze funcționarea TurtleBot3.



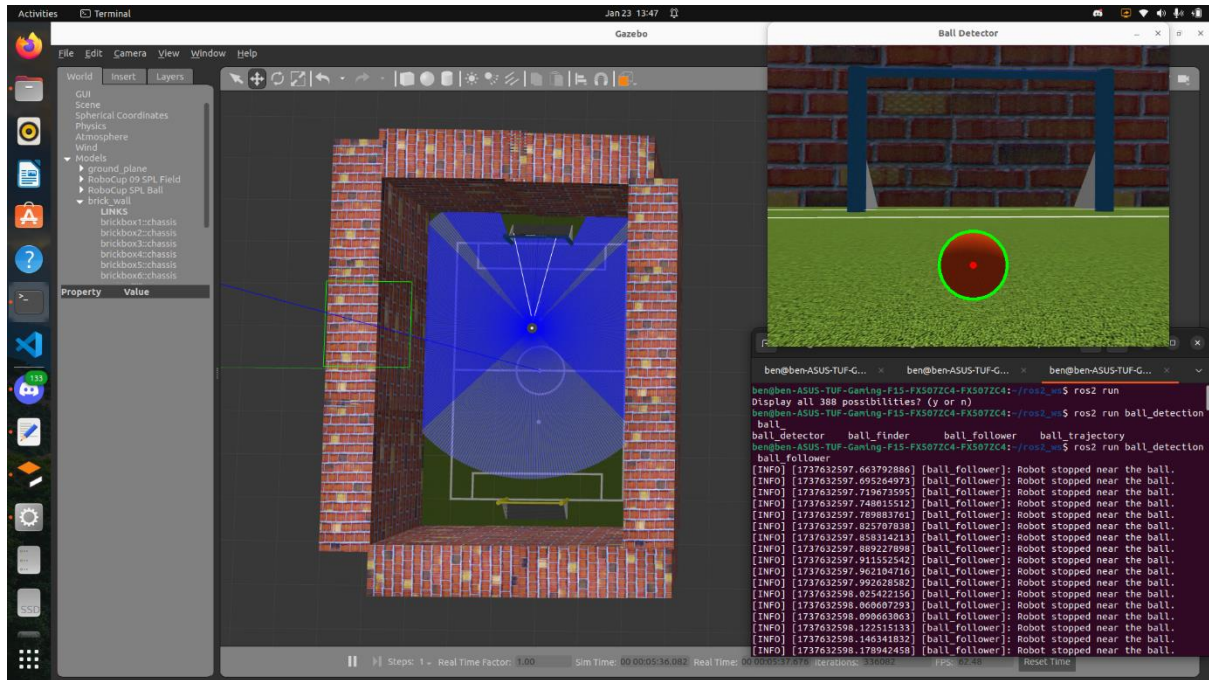
Rezultate

Generare Simulare + Robot se face cu comanda:

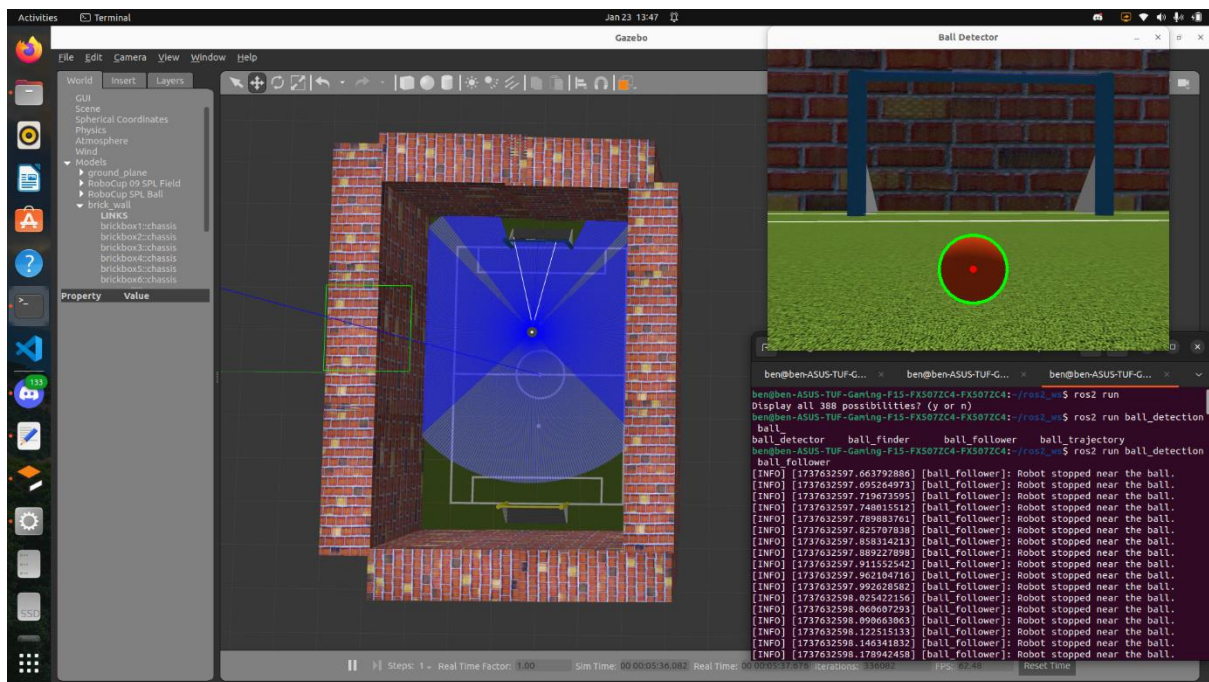
ros2 launch turtlebot3_gazebo empty_world.launch.py



Ball detection+Ball follower:



Ball detection command: `ros2 launch ball_detection ball.py`



Ball follower command: `ros2 run ball_detection ball_follower`

Concluzie

În dezvoltarea robotului au fost întâmpinate trei dificultăți majore:

- **Calibrarea sistemului:** Ajustările precise ale componentelor hardware și software au necesitat monitorizare continuă pentru asigurarea unei performanțe optime.
- **Probleme cu camera:** Detecția mingii a fost afectată de condițiile de iluminare și poziționarea camerei, ceea ce a necesitat îmbunătățiri ale algoritmilor de procesare a imaginii (masca imaginii detectează mingea). Algoritmul implementat pentru imagine este făcut în așa fel încât robotul să vadă doar mingea
- **Instabilitatea robotului la coliziuni:** Robotul s-a răsturnat frecvent la întâlnirea cu obstacole, ceea ce a impus modificarea designului și optimizarea algoritmilor de echilibru. (scriptul pentru cautarea mingii rula și după ce robotul era instabil astfel că nu își poate recăpăta forma inițială)

În concluzie, rezolvarea acestor dificultăți a contribuit la îmbunătățirea performanței generale a robotului și a aprofundat înțelegerea interacțiunii dintre componentele hardware și software.

Bibliografie

1. Siciliano, B., Sciavicco, L., Villani, L., & Oriolo, G. (2010). "Robotics: Modelling, Planning and Control." Springer.
2. Quigley, M., Gerkey, B., & Smart, W. D. (2015). "Programming Robots with ROS." O'Reilly Media.