

ROBOCUPJUNIOR RESCUE LINE 2023

TEAM DESCRIPTION PAPER

HACKATRONICI

Abstract

Il nostro robot è un Lego Mindstorm EV3 31313 con kit di espansione Education. La coppia di sensori di colore, insieme ad un algoritmo di correzione di rotta permette un segui linea molto performante e preciso.

Un ulteriore sensore in modalità luce riflessa è stato impiegato per la detection dei punti cruciali della rescue line come l'imbocco di un incrocio o l'entrata nell'evacuation zone. L'impiego di questo ulteriore sensore ha migliorato di molto la reazione al verificarsi di tali eventi.

Un sensore ad ultrasuoni è stato impiegato per la detection di oggetti davanti al robot. Questo è stato impiegato negli scenari di aggiramento ostacolo e evacuation zone.

Nell'evacuation zone il sensore di luce riflessa cambia modalità in colore per la detection degli evacuation point.

Nella realizzazione si è puntato a massimizzare le performance sul tempo di percorrenza della rescue line, pur mantenendo un'ottima precisione. Questo ha consentito una maggiore disponibilità di tempo nello scenario dell'evacuation zone.

1. Introduction

a. Team

Il nostro Gruppo è composto da:

Alessandro Chiarulli: classe 4°, assemblaggio del robot. Programmazione rescue line.

Gabriele Montrone: classe 3°, calibrazione detection verde. Programmazione svolte.

Giuseppe Clemente: classe 3°. Programmazione evacuation zone.

Mario Recchia: classe 3°, calibrazione detection svolte, evacuation point, entrata e uscita evacuation point. Programmazione aggiramento ostacolo.

Esperienze passate dei membri del team:

- sul podio per le olimpiadi di problem solving
- ambasciatore globale digital education hackathon
- iscritti all'albo nazionale delle eccellenze
- progetti Arduino: vaso intelligente, prototipo di robot per la rescue line, centralina luci per dj set
- sviluppo software: applicazioni front end su web
- certificazioni conseguite: astropy
- corsi certificati: Cisco, Cambridge, EiPass

2. Project Planning

a. Overall Project Plan

L'obiettivo preposto in fase di progettazione è stato quello di massimizzare i punti nella fase del segu-linea, coprendo tutti i possibili scenari.

In fase di progettazione iniziale abbiamo stabilito di occuparci della fase di evacuation zone a best effort.

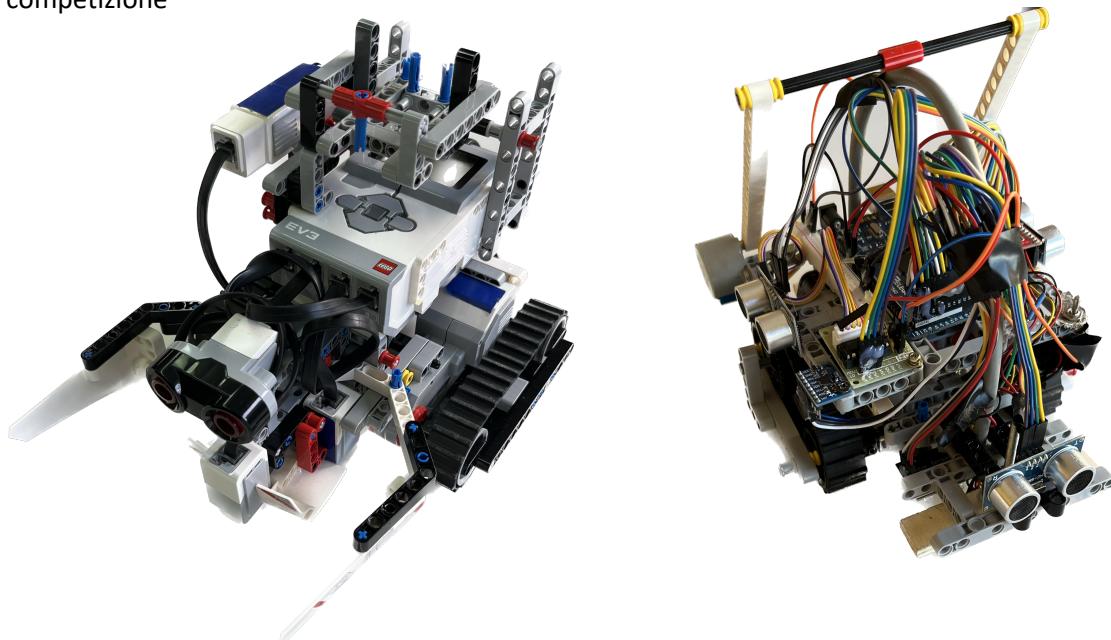
In effetti per questa fase è stato implementato un algoritmo semplice. Una strategia più complessa è stata implementata come prototipo, ma non l'abbiamo ritenuta sufficientemente stabile, per cui è stata scartata a favore dell'algoritmo più semplice.

Milestone

- Durata complessiva del progetto: 30 giorni, suddiviso in:
 - realizzazione robot: 4 giorni, in parallelo con le altre attività
 - programmazione segui linea: 3 giorni
 - programmazione svolte: 5 giorni, dopo il segui linea
 - programmazione detection palline: 3 giorni
 - programmazione detection evacuation point: 1 giorno
 - programmazione detection uscita evacuation zone: 1 giorno
 - programmazione rescue kit: 1 giorno
 - Test e raffinamento segui linea: 5 giorni
 - Test e raffinamento speed bump: 1 giorno
 - Test e raffinamento rampe: 1 giorno
 - Test e raffinamento rampa basculante: 1 giorno
 - Test e raffinamento evacuation zone: 8 giorni

Le attività sono state svolte in parallelo, indicativamente sulla base dei compiti assegnati in partenza.

Sono stati sviluppati due prototipi in parallelo, uno utilizzando Arduino, l'altro un Kit EV3. In fase di Test abbiamo optato per la soluzione EV3 perché si è dimostrata più stabile. La soluzione Arduino, seppur più complessa e con maggiori potenzialità, è rimasta in una fase prototipale, non stabile per affrontare la competizione



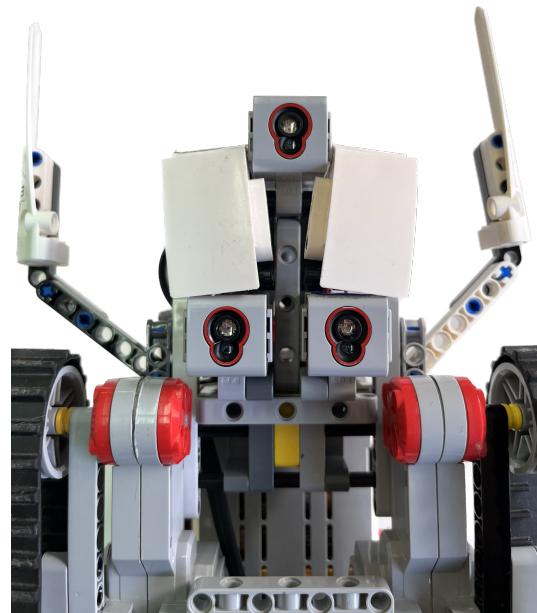
Gundam ✓

Frankie Brundle ✗

b. Integration Plan

Per il segui-linea abbiamo usato un sistema a tre sensori, due a livello dei motori e uno più avanti. I due sensori di dietro riconoscono la linea al centro ed eventuali verdi, mentre il sensore davanti è in modalità luce riflessa per riconoscere il foglio riflettente all'entrata dell'evacuation zone, ma viene anche sfruttato nel segui-linea per riconoscere gli incroci.

Per l'ostacolo e le pareti dell'evacuation zone abbiamo usato un sensore ad ultrasuoni. Per rilasciare il Rescue Kit abbiamo usato un motore medio, potente abbastanza da supportarne il peso



Oltre che sul lato hardware, l'integrazione è stata gestita anche a livello software. Per agevolare l'integrazione software ci siamo avvalsi di GitHub per il versioning e la gestione della revisione del codice. Inoltre, il sorgente è stato suddiviso in moduli per consentire lo sviluppo indipendente dopo aver definito le interfacce dei moduli.

```
mhackrobocup
|
|   .gitignore
|   README.md
|
|   --definitivo
|       040_sensor_distanza.ino
|       050_momimento.ino
|       060_aggira_ostacolo.ino
|       070_evacuation.ino
|       definitivo.ino
|       definizioni.h
|
|   --ev3
|       GaraVicenza.lmsp
|       Stanza.lmsp
```

3. Hardware

Sensori di colore

I sensori di colore sono in grado di riconoscere sette colori: nero, blu, verde, giallo, rosso, bianco, marrone.

Per il segui linea è stata necessaria una calibrazione ad hoc per gestire i casi di accavallamento tra nero e bianco, situazione che per il sensore rientra nella casistica "blu".

Sensore di luce riflessa

Il sensore di luce riflessa, invece, funziona emettendo una luce rossa e misurando l'intensità con cui viene riflessa. Lavora su una scala da 0 a 100, dove 0 è molto scuro e 100 è molto luminoso. Nel nostro caso, se la luce riflessa è maggiore di 95 vuol dire che ha trovato l'evacuation zone, mentre se è minore di 10 ha trovato la linea nera.

Una volta entrato nell'evacuation zone, il robot cambia ordine dei sensori, rendendo quello davanti di colore e quelli di dietro di luce riflessa. In questo modo possiamo riconoscere prima le zone sicure in cui riporre le vittime.

Sensore ad ultrasuoni

Il sensore ad ultrasuoni funziona emettendo delle onde sonore e misurandone l'eco di ritorno. In base al tempo in cui il suono impiega tornare al sensore è in grado di misurarne la distanza con un grado di precisione discretamente elevato.

a. Mechanical Design and Manufacturing

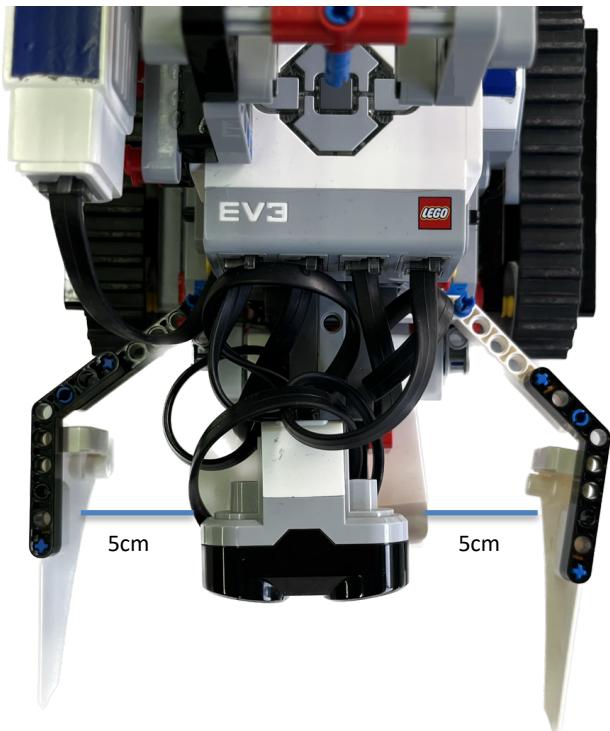
La struttura del robot è fatta con i pezzi del kit EV3. Abbiamo cercato di creare una struttura più efficiente possibile a livello di peso, tenendo in considerazione le rampe e il peso aggiuntivo del rescue kit. Per questo motivo il numero dei pezzi componenti il robot è ridotto al minimo, cercando il compromesso fra peso e solidità.

I motori grandi utilizzati per il movimento sono i più potenti del kit EV3, con un sensore di rotazione integrato che è in grado di fornire una precisione al grado. Sono sufficientemente potenti per supportare il peso del robot nel percorso e sulle rampe, considerando anche i casi in cui ci sia uno speed-bump sulla rampa.

Per il Rescue Kit abbiamo usato il motore medio, poiché meno potenza è richiesta per riporlo nella zona sicura. Anche questo ha un sensore di rotazione integrato, così da potergli fare effettuare una rotazione di 135 gradi per più volte per essere sicuri di rilasciare il rescue kit dal cestello.

Per il salvataggio abbiamo usato un meccanismo estremamente semplice: due bracci ai lati del sensore ad ultrasuoni, con uno spazio dai sensori di poco più di 5cm, per poter accogliere le palline e non farle sfuggire.

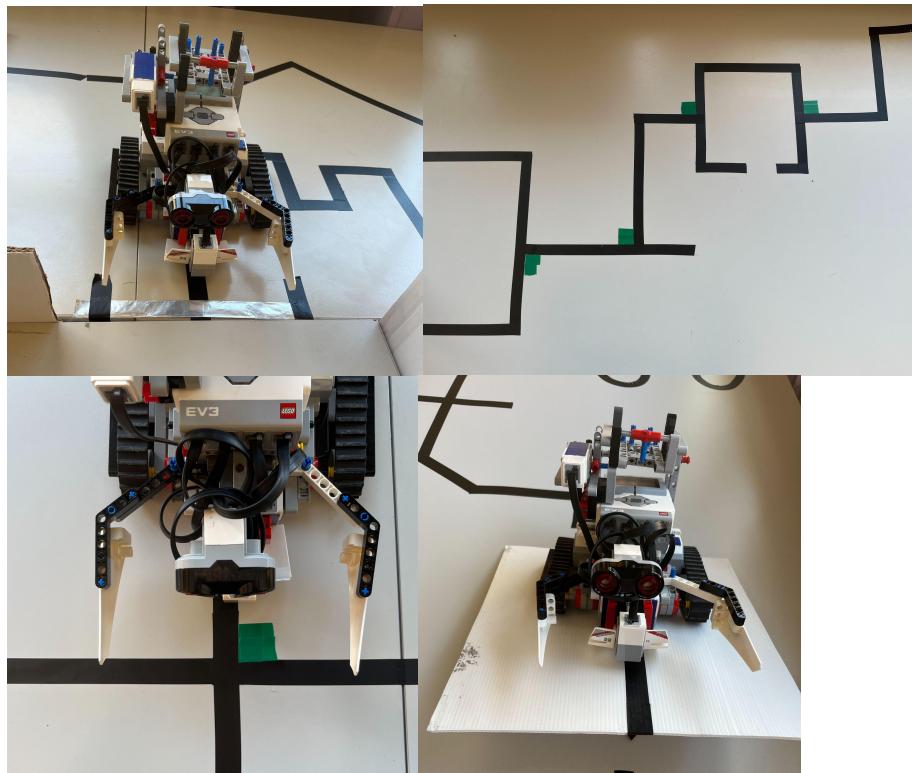
Una volta individuato il triangolo verde o rosso nella Evacuation Zone il robot si allontana repentinamente, per sganciare le palline, esegue una curva di 180° e torna indietro per posare il Rescue Kit.



Test procedures

Dal punto di vista meccanico sono stati previsti i seguenti test:

- avanzamento sulla rampa (test potenza / peso, attrito dei cingoli)
- rilascio del rescue kit (procedura di rilascio)
- raccolta palline



b. Electronic Design and Manufacturing

Del controllo dell'elettronica di bordo se ne occupa il brick principale dell'EV3 31313. Ha 8 porte disponibili, di cui 4 di input per il controllo dei sensori e 4 di output per i motori. Se il robot è connesso al computer possiamo avere in tempo reale i valori letti dai sensori, utili per la calibrazione e il controllo dei colori.

Il robot è alimentato da 6 batterie AA da 1,5V.



Procedure di test

Per ogni scenario da gestire (linea, svolta, bumper, rampa fissa e mobile, ostacolo, evacuation zone) è stata prevista una procedura di test indipendente, attivando di volta in volta nel software solo la parte di codice coinvolta.

Dopo aver testato le singole procedure, si è passati al **test di integrazione**, dove uno scenario completo è stato testato più volte, attivando questa volta tutto il software sviluppato.

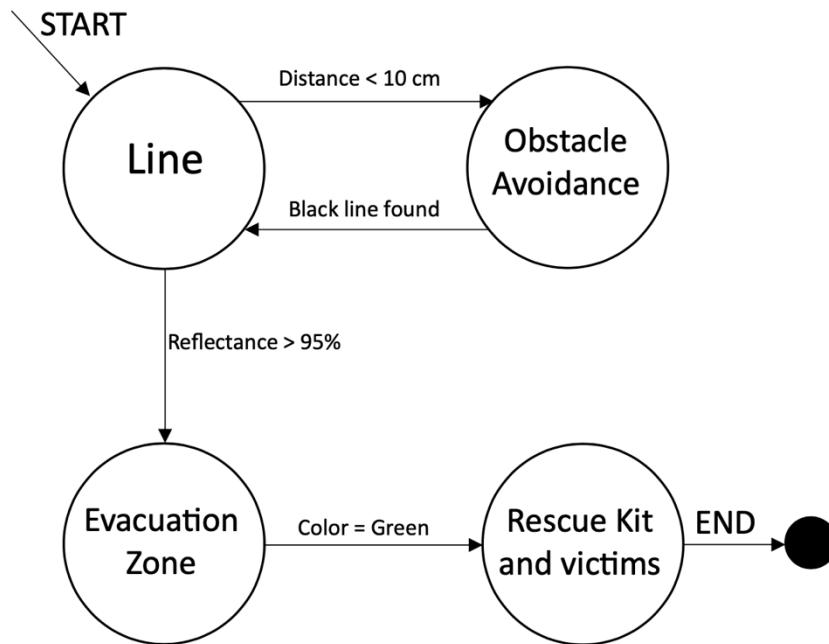
4. Software

Per il codice del robot abbiamo utilizzato l'ambiente di sviluppo integrato dell'EV3, con una programmazione a blocchi semplice ma che include tutti i blocchi necessari per effettuare un percorso completo.

Per agevolare lo sviluppo collaborativo, tenere traccia delle varie versioni sviluppate nel corso del tempo, etichettare le versioni testate e gestire i conflitti sul sorgente è stato usato github. Ogni membro del team ha lavorato indipendentemente sul suo modulo. Il software finale è stato poi finalizzato integrando i vari sorgenti

a. General software architecture

La struttura generale del nostro codice è la seguente:



Se il sensore davanti rileva la luce riflessa maggiore del 95% entra in modalità evacuation zone, altrimenti viene eseguito il segui-linea. Se viene rilevata dal sensore ad ultrasuoni una distanza minore di 10 cm entra in modalità aggira ostacolo. Se il robot è entrato in modalità stanza controlla prima da che lato si trova l'entrata, poi inizia a lavorare a “spazzaneve”, coprendo tutta la superficie disponibile fino a quando non trova il triangolo verde. Quando lo trova lascia il rescue kit

b. Innovative solutions

La funzionalità di cui andiamo più orgogliosi è il segui-linea. Trovare il giusto compromesso tra robustezza, precisione e velocità è stata una sfida. Per il tuning dei parametri e per tenere traccia dei risultati per ogni nuova configurazione ci siamo avvalsi di un foglio elettronico su cui annotare valori ed evidenze.

L'utilizzo di un simulatore come open roberta si è mostrato di grande utilità, in particolar modo ci ha consentito di partire con dei test preliminari quando il robot era indisponibile (durante la sua costruzione, modifica e miglioramento strutturale).

Anche la scomposizione in moduli e l'integrazione con github è stata una mossa vincente per il lavoro in team.

5. Performance evaluation

Abbiamo riscontrato una buona precisione del robot nel seguire la linea. Quasi tutti i casi di incrocio sono coperti e, a meno che di casi estremamente sfortunati, è sempre in grado di seguire la linea senza perdersi. Per l'evacuation zone abbiamo grossi margini di miglioramento per via della soluzione semplice adottata. Non avendo un meccanismo preciso per raccogliere le palline, diverse situazioni potrebbero mettere in difficoltà il robot. Generalmente, nel caso in cui le palline siano ben distribuite nella stanza, ci sono buone possibilità che il robot riesca a portare a termine i suoi compiti. Il rescue kit viene rilasciato correttamente con una buona probabilità di successo.

6. Conclusion

Per la partecipazione a queste gare abbiamo provato ad usare il 100% delle potenzialità del nostro robot, utilizzando tutte le funzioni presenti anche da parte del software. Abbiamo preparato il robot ad affrontare tutti i casi possibili presenti nei percorsi, con particolare attenzione al segui-linea. Nonostante le limitate possibilità da parte del kit Lego abbiamo provato a creare un robot capace di eseguire lo scopo principale della categoria.

References

- [1] [EV3 Site - Tutorial](#)
- [2] [Open Roberta](#)