

Documentazione - Progetto di Robotica

Augello Andrea Castiglione Francesco Paolo La Martina Marco

Università degli Studi di Palermo

11 gennaio 2021



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

1 Introduzione

2 Gestione dei nodi ROS

3 Modello del moto e posizionamento

4 Object recognition

5 Posizione dei target

6 Modello probabilistico

7 Scheduling dei comportamenti

8 Possibili modifiche



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Section 1

Introduzione



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Introduzione

La pandemia del coronavirus SARS-CoV-2 ha dato una forte spinta alla ricerca sia nel campo sanitario che informatico, mettendo in evidenza forti carenze dal punto di vista infrastrutturale.

Al momento, considerando la limitata disponibilità del vaccino alle masse, uno dei migliori modi di evitare la contrazione del coronavirus è di evitarne l'esposizione. Il distanziamento sociale si configura di conseguenza come un prerequisito per una significativa riduzione del numero di infetti, come evidenziato da simulazioni di un sistema ad agenti [1]. Un problema chiave risulta essere il controllo del rispetto delle norme di distanziamento all'interno degli spazi chiusi.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Obbiettivo

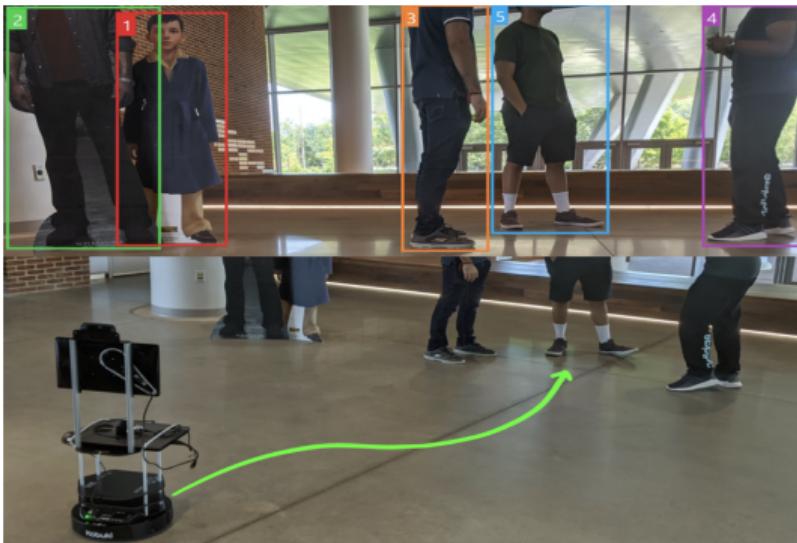
L'obbiettivo del progetto è quello di sviluppare un robot con lo scopo di **evitare assembramenti in ambienti indoor** e di invitare a **rispettare le norme sul distanziamento sociale**.

Nella dimostrazione presentata il nostro robot rileva le persone nella stanza e individua i possibili assembramenti. In seguito alla fase di rilevazione si sposterà verso l'assembramento evitando gli ostacoli e, arrivato, esorterà le persone al rispetto del distanziamento sociale.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Stato dell'arte |

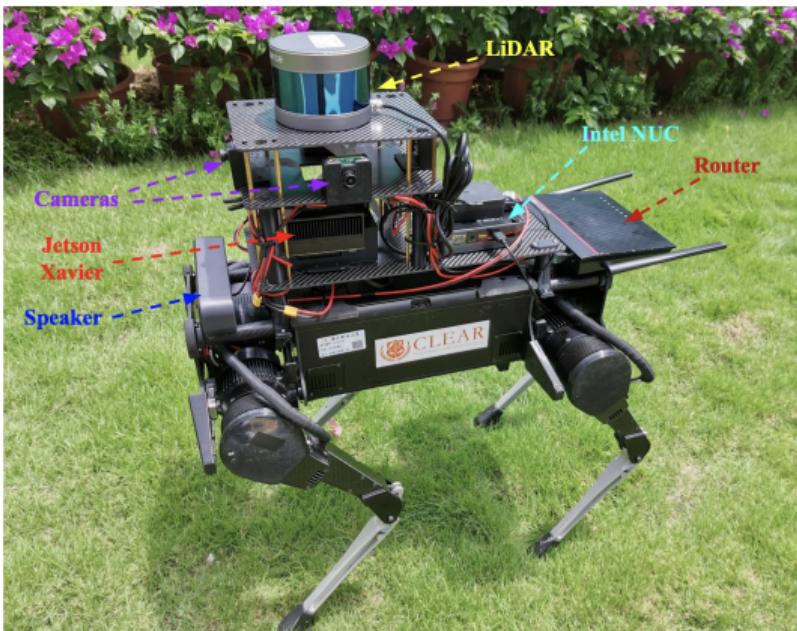


Un TurtleBot 2, con una camera RGB-D e CCTV per il rilevamento degli assembramenti, una camera termica FLIR C3 per rilevare la temperatura corporea e un lidar 2-D per evitare le collisioni. L'elaborazione delle immagini provenienti da CCTV è eseguita in un laptop con una CPU Intel i7 7th generation e una GPU Nvidia GTX1060, mentre il resto dell'elaborazione è eseguita su una CPU Intel i9 8th generation e una GPU Nvidia RTX2080 montate sul robot.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Stato dell'arte II



Un Laikago con 2 camere laterali e CCTV per il rilevamento degli assembramenti e un lidar 3-D per evitare le collisioni. L'elaborazione riguardante il modulo visivo è effettuata in una NVIDIA Jetson AGX Xavier, il resto è eseguito in una CPU Intel i5 8259U. Il rilevamento delle persone viene effettuato tramite la rete YOLO.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

| | |
|-------------------------|--|
| OS | Ubuntu 18.04 Ubuntu 20.04 Raspberry Pi OS Buster |
| ROS version | melodic noetic |
| Webots | R2020b revision 1 |
| OpenCV [2] | 4.x |
| Imutils [3] | 0.5.3 |
| Matplotlib [4] | 3.3.3 |
| Numpy [5] | 1.17.2 |
| Scikit-learn [6] | 0.21.3 |
| Raspberry | Raspberry Pi 3B+ |



TIAGo Iron



Il robot scelto per l'obiettivo proposto è il **TIAGo Iron**, un robot umanoide a due ruote con torso e testa ma senza braccia articolate [7]. Il datasheet del **TIAGo** [8] indica la presenza di speaker e display, non presenti nel modello Webots [9], che sono quindi stati aggiunti.

La camera del **TIAGo** è RGB-D ma il modello Webots ne è sprovvisto, di conseguenza è stata utilizzata una camera monoscopica RGB. L'IMU utilizzata ha 6 gradi di libertà.

Il modello Webots del **TIAGo** presenta un lidar (Hokuyo URG-04LX-UG01 [10]) che, conformemente al datasheet, ha un range di 5.6 m ed un FOV di 240° (agli estremi è parzialmente occluso).



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

1 Introduzione

2 Gestione dei nodi ROS

3 Modello del moto e posizionamento

4 Object recognition

5 Posizione dei target

6 Modello probabilistico

7 Scheduling dei comportamenti

8 Possibili modifiche



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

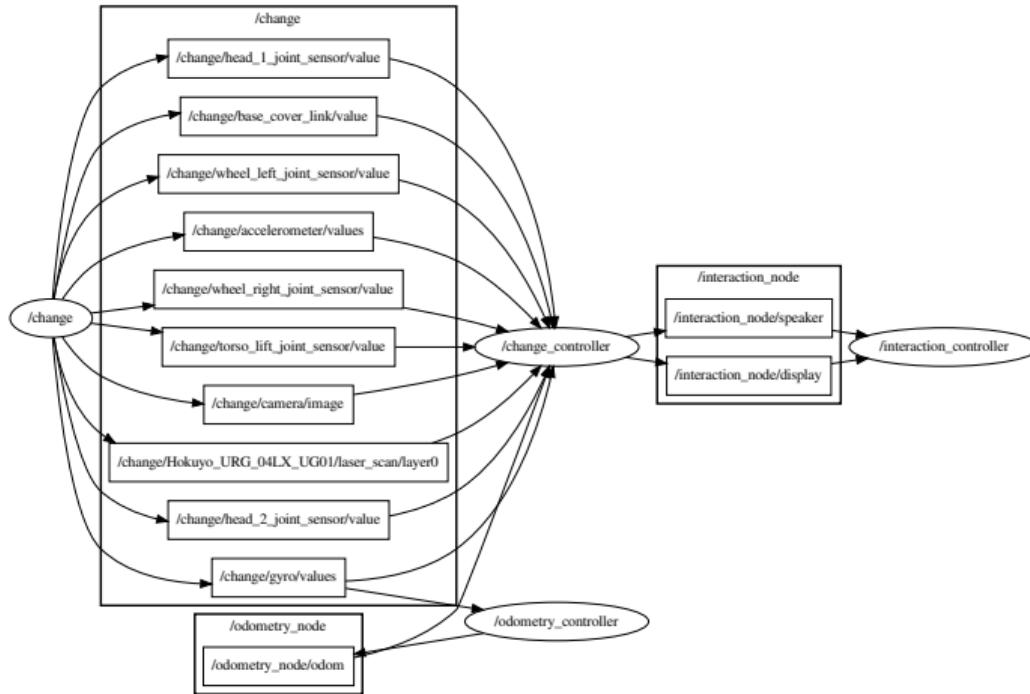
Section 2

Gestione dei nodi ROS



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

ROS nodes



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Webots node

Il nodo Webots si occupa solamente di lanciare Webots, e di impostare il valore del clock di ROS in base al tempo della simulazione, in modo da potere effettuare le integrazioni del tempo correttamente.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Change controller node

Il nodo Change controller si occupa della gran parte della elaborazione. Oltre ad arbitrare sui comportamenti da assumere, gestisce più moduli che si occupano di:



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Change controller node

Il nodo Change controller si occupa della gran parte della elaborazione. Oltre ad arbitrare sui comportamenti da assumere, gestisce più moduli che si occupano di:

- acquisire i dati dai sensori



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Change controller node

Il nodo Change controller si occupa della gran parte della elaborazione. Oltre ad arbitrare sui comportamenti da assumere, gestisce più moduli che si occupano di:

- acquisire i dati dai sensori
- mandare i comandi ai motori



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Change controller node

Il nodo Change controller si occupa della gran parte della elaborazione. Oltre ad arbitrare sui comportamenti da assumere, gestisce più moduli che si occupano di:

- acquisire i dati dai sensori
- mandare i comandi ai motori
- gestire il movimento, quindi rotazioni e traslazioni



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Change controller node

Il nodo Change controller si occupa della gran parte della elaborazione. Oltre ad arbitrare sui comportamenti da assumere, gestisce più moduli che si occupano di:

- acquisire i dati dai sensori
- mandare i comandi ai motori
- gestire il movimento, quindi rotazioni e traslazioni
- acquisire e analizzare le immagini dalla camera



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Interaction node

Il nodo Interaction ha il compito di gestire le interazioni audio/video. Ogni messaggio che viene riprodotto, prima in lingua italiana e poi inglese, viene anche visualizzato testualmente sullo schermo in italiano, inglese e cinese. I possibili comportamenti assunti dal robot sono:



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Interaction node

Il nodo Interaction ha il compito di gestire le interazioni audio/video. Ogni messaggio che viene riprodotto, prima in lingua italiana e poi inglese, viene anche visualizzato testualmente sullo schermo in italiano, inglese e cinese. I possibili comportamenti assunti dal robot sono:

- Salutare all'avvio



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Interaction node

Il nodo Interaction ha il compito di gestire le interazioni audio/video. Ogni messaggio che viene riprodotto, prima in lingua italiana e poi inglese, viene anche visualizzato testualmente sullo schermo in italiano, inglese e cinese. I possibili comportamenti assunti dal robot sono:

- Salutare all'avvio
- Mostrare sul display immagini che esortano a rispettare il distanziamento sociale



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Interaction node

Il nodo Interaction ha il compito di gestire le interazioni audio/video. Ogni messaggio che viene riprodotto, prima in lingua italiana e poi inglese, viene anche visualizzato testualmente sullo schermo in italiano, inglese e cinese. I possibili comportamenti assunti dal robot sono:

- Salutare all'avvio
- Mostrare sul display immagini che esortano a rispettare il distanziamento sociale
- Riprodurre un messaggio audio che invita a rispettare il distanziamento sociale quando rileva un assembramento o quando scansiona l'ambiente



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Odometry node

Il nodo Odometry si occupa di stimare la posizione del robot, come spiegato approfonditamente nella sezione 3. In generale ciò che fa è integrare costantemente i valori del giroscopio e della velocità delle ruote per condividere posizione e orientamento del robot.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

- 1** Introduzione
- 2** Gestione dei nodi ROS
- 3** Modello del moto e posizionamento
- 4** Object recognition
- 5** Posizione dei target
- 6** Modello probabilistico
- 7** Scheduling dei comportamenti
- 8** Possibili modifiche



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Section 3

Modello del moto e posizionamento



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Spostamento

Il modello del moto è caratterizzato da rotazioni e traslazioni. Per effettuare lo spostamento lineare utilizziamo il controllore PID (Proporzionale-Integrale-Derivativo) delle ruote fornito da Webots, che richiede un angolo di rotazione target per ogni ruota. Utilizziamo quindi l'angolo di rotazione corrente, e il diametro delle ruote per calcolare la posizione delle ruote necessaria al fine di ottenere lo spostamento desiderato (Eq. 1).

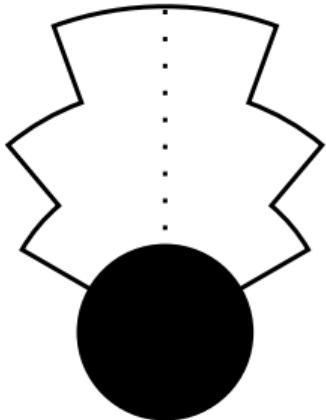
$$(1) \quad \text{targetAngle} = \text{currentAngle} + 2\pi \frac{\text{distance}}{2\pi \cdot \text{diameter}}$$

Noto l'angolo corrente e l'angolo target utilizziamo un controllore proporzionale per raggiungere l'angolo desiderato.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Collision avoidance



Il **TIAGo** è in grado di rilevare gli ostacoli grazie all'utilizzo di un sensore lidar.

Nell'immagine viene mostrata la zona nella quale, se viene indicata dal lidar la presenza di un ostacolo, il **TIAGo** si ferma per ragioni di sicurezza al fine di evitare danni a persone e/o oggetti.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Orientamento

Per le rotazioni ci basiamo sui dati forniti dal giroscopio, fondendoli con quelli provenienti dagli encoder delle ruote [11]. Calcoliamo quindi l'angolo di rotazione effettuando un'integrazione discreta dei campioni della velocità angolare con una interpolazione lineare del primo ordine (Eq. 2).

$$(2) \quad \theta_i = \sum_{j=1}^i \frac{\omega_{j-1} + \omega_j}{2} (t_j - t_{j-1})$$



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Orientamento

L'integrazione dei dati del giroscopio è soggetta ad un drift più significativo che l'uso dell'odometria attraverso gli encoder delle ruote.

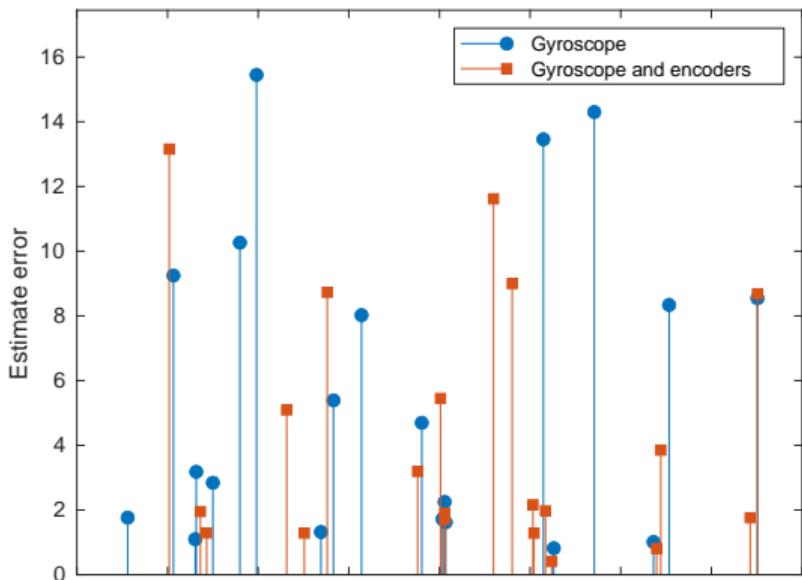
$$(3) \quad w_{odom} = \frac{R(u_{r,i} - u_{l,i})}{L}$$

L'errore nella stima odometrica però ha anche componenti non sistematiche difficili da stimare e correggere, per stimare la velocità di rotazione effettuiamo quindi una fusione dei dati forniti da encoder e giroscopio.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Orientamento



La fusione dei dati dei sensori riduce il MSE della stima da 55 (solo giroscopio) a 32.

Algorithm 1: Stima dell'angolo

Data: $\omega_{gyro,i}, \omega_{odom,i}, \omega_{i-1}, \theta_{i-1}$
Result: θ_i

```
 $\delta \leftarrow \omega_{gyro,i} - \omega_{odom,i}$ 
if  $|\delta/\omega_{gyro,i}| > thresh$  then
|  $\omega_i \leftarrow \omega_{gyro,i}$ 
else
|  $\omega_i \leftarrow \omega_{odom,i}$ 
end
 $\theta_i \leftarrow \theta_{i-1} + \frac{\omega_{i-1} + \omega_i}{2} (t_j - t_{j-1})$ 
```

return θ_i



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Posizionamento

Inizialmente al segnale dell'accelerometro veniva applicato un integrale doppio per ottenere lo spostamento lineare [12].

$$(4) \quad \begin{cases} \mathbf{v}_i &= \sum_{j=1}^i \frac{\mathbf{a}_{j-1} + \mathbf{a}_j}{2} (t_j - t_{j-1}) \\ \mathbf{s}_i &= \sum_{j=1}^i \frac{\mathbf{v}_{j-1} + \mathbf{v}_j}{2} (t_j - t_{j-1}) \end{cases}$$



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Posizionamento

Abbiamo ritenuto necessario cambiare approccio, decidendo di utilizzare gli encoders delle ruote per determinare gli spostamenti.

Integriamo la velocità lineare del robot, calcolata a partire dal raggio R e le velocità angolari u delle ruote.

$$(5) \quad v_i = \frac{R(u_{r,i} + u_{l,i})}{2}$$

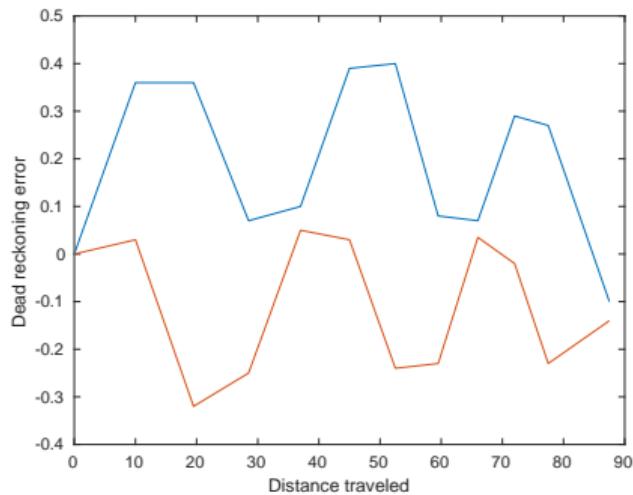
Questi valori sono aggiornati ad ogni intervallo di campionamento utilizzando la velocità lineare e la velocità angolare del robot [13].

$$(6) \quad \mathbf{P}_i = \sum_{j=1}^i \begin{bmatrix} v_j \cos(\theta_j) \\ v_j \sin(\theta_j) \end{bmatrix} \cdot (t_j - t_{j-1})$$

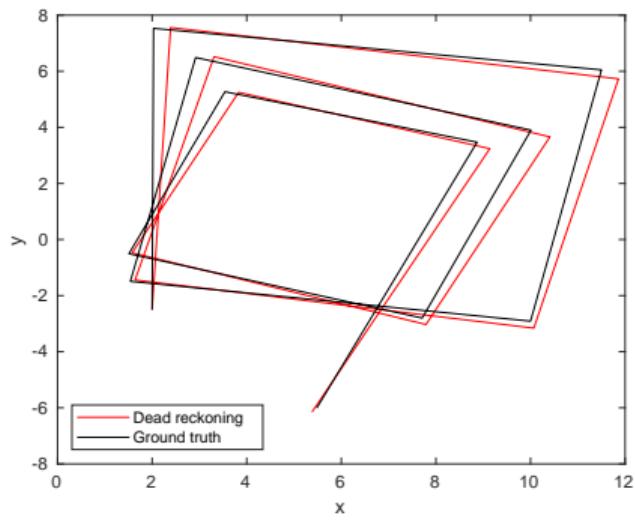


UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Posizionamento



(a) Errore nella stima della posizione



(b) Errore nella stima della traiettoria

Abbiamo misurato le performance della stima di posizione e i risultati sono ritenuti soddisfacenti per raggiungere l'obiettivo proposto.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

- 1** Introduzione
- 2** Gestione dei nodi ROS
- 3** Modello del moto e posizionamento
- 4** Object recognition
- 5** Posizione dei target
- 6** Modello probabilistico
- 7** Scheduling dei comportamenti
- 8** Possibili modifiche



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Section 4

Object recognition

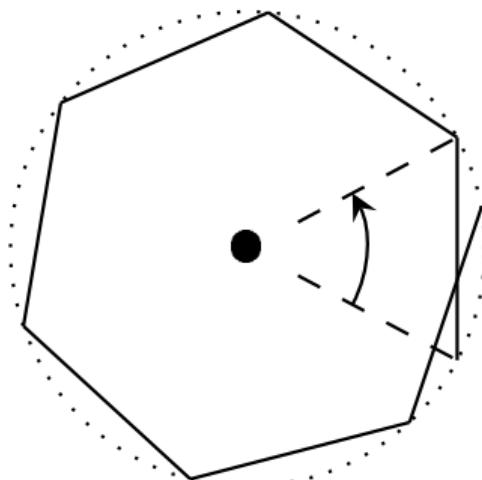


UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Campionamento delle immagini

Il FOV della camera è di 57° , quindi per ricoprire 360° è necessario effettuare 7 campionamenti. L'ultimo campionamento è sovrapposto al primo per 39° .

È possibile che un individuo si trovi in una zona di confine tra due campioni, e che quindi non sia correttamente identificabile in nessuna delle due immagini in cui appare parzialmente. Mitighiamo questo problema effettuiamo una rudimentale operazione di image mosaicing [14] e campioniamo l'immagine così ottenuta ad intervalli di 28° .



YOLO

Abbiamo valutato le performance di YOLOv3 (you only look once), YOLOv3-tiny, HoG, HoG + SVG + NMS . In seguito a vari test su HoG abbiamo ritenuto essere problematica la larghezza delle bounding boxes fornite. YOLOv3 fornisce risultati soddisfacenti. Considerando le caratteristiche hardware del robot mobile, abbiamo optato per l'uso di YOLOv3-tiny, il quale risulta essere significativamente più efficiente (approssimativamente del 442% [15]). Inoltre è rilevante in tal senso un paragone fra YOLOv3 e YOLOv3-tiny in termini di mAP (mean average precision) e FLOPS (floating-point operations per second) addestrate sul dataset COCO, come illustrato dalla tabella.

| Model | mAP | FLOPS | FPS |
|-------------|------|-----------|-----|
| YOLOv3-320 | 51.5 | 38.97 Bn | 45 |
| YOLOv3-416 | 55.3 | 65.86 Bn | 35 |
| YOLOv3-608 | 57.9 | 140.69 Bn | 20 |
| YOLOv3-tiny | 33.1 | 5.56 Bn | 220 |
| YOLOv3-spp | 60.6 | 141.45 Bn | 20 |



- 1** Introduzione
- 2** Gestione dei nodi ROS
- 3** Modello del moto e posizionamento
- 4** Object recognition
- 5** Posizione dei target
- 6** Modello probabilistico
- 7** Scheduling dei comportamenti
- 8** Possibili modifiche



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Section 5

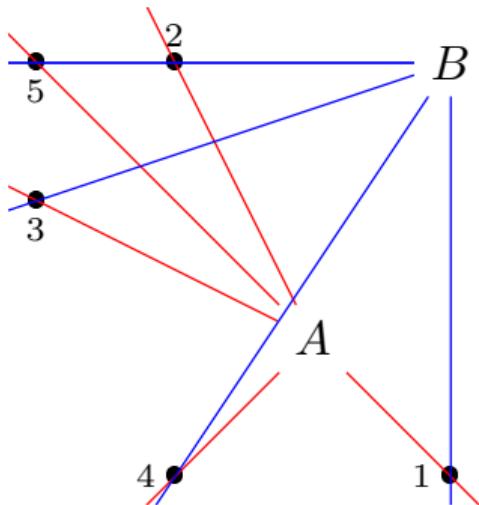
Posizione dei target



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Triangolazione

La triangolazione come metodo di individuazione delle persone presenta dei problemi nel nostro scenario:

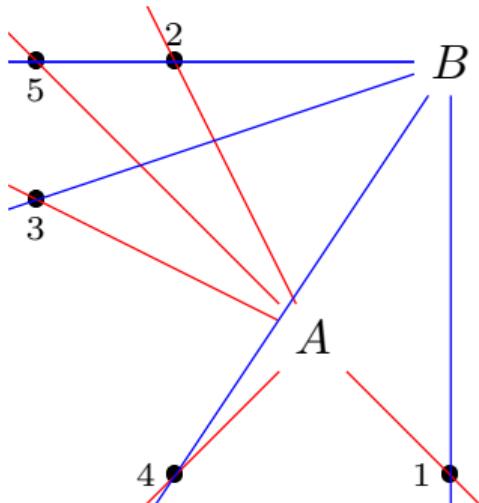


UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Triangolazione

La triangolazione come metodo di individuazione delle persone presenta dei problemi nel nostro scenario:

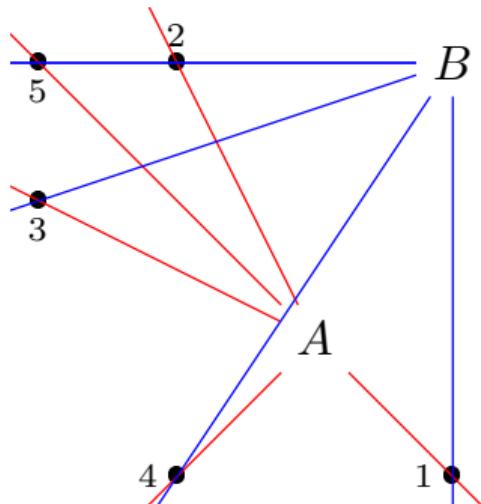
- Occlusione delle persone: se due persone (5 e 2) sono una dietro l'altra lungo una retta immaginaria che le congiunge al robot (B), quest'ultimo non sarà in grado di individuare la più distante.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Triangolazione

La triangolazione come metodo di individuazione delle persone presenta dei problemi nel nostro scenario:



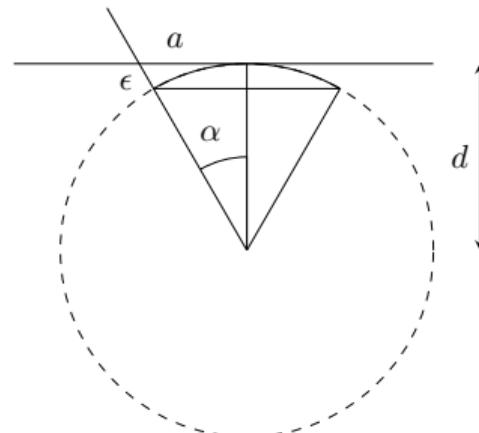
- Occlusione delle persone: se due persone (5 e 2) sono una dietro l'altra lungo una retta immaginaria che le congiunge al robot (B), quest'ultimo non sarà in grado di individuare la più distante.
- Imputazione delle osservazioni: quando il robot effettua scan successivi non è in grado di dedurre quali osservazioni derivano dalla stessa persona. Non è possibile determinare quali intersezioni corrispondono ad osservazioni reali e quali sono spurie.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Calcolo della distanza |

L'altezza percepita dell'oggetto non è un indicatore affidabile della sua distanza dal robot in quanto parte dell'oggetto potrebbe essere occlusa o non presente nel frame. La larghezza del torso è meno suscettibile a tali problemi, e non dipende dalla posizione. Ipotizzando che la camera abbia un FOV di 2α e sia distante d dall'oggetto, la massima distanza orizzontale che un punto dell'immagine potrebbe avere dal centro del piano dell'immagine sarebbe $a = d \tan \alpha$.



Calcolo della distanza II

Ignorare la prospettiva significa effettuare un'approssimazione lineare del primo ordine e trattare il punto come se si trovasse su una circonferenza di raggio d centrata sulla camera. Di conseguenza consideriamo il punto come se fosse più vicino di quanto non sia realmente, commettendo l'errore mostrato nell' Eq. 7. Con una camera con FOV di 1 radiante la sottostima è del 13.9%.

$$\begin{aligned} \epsilon &= \sqrt{a^2 + d^2} - d = \sqrt{(d \tan \theta)^2 + d^2} - d = \\ (7) \quad &= d \left(\sqrt{\frac{1}{\cos^2 \alpha}} - 1 \right) = d (\sec \alpha - 1) \end{aligned}$$



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Calcolo della distanza III

Poiché stiamo utilizzando un simulatore non è nota la larghezza del sensore da utilizzare per l'eq. 8. Abbiamo ovviato a tale problema posizionando il robot ed un oggetto dalle dimensioni note in posizioni note e abbiamo utilizzato questi dati insieme a delle misure in pixel nell' eq. 9. Abbiamo così stimato le dimensioni del sensore virtuale da utilizzare nei calcoli successivi.

$$(8) \quad object\ distance(m) = \frac{f(m) \times real\ width(m) \times image\ width(pixels)}{object\ width(pixels) \times sensor\ width(m)}$$

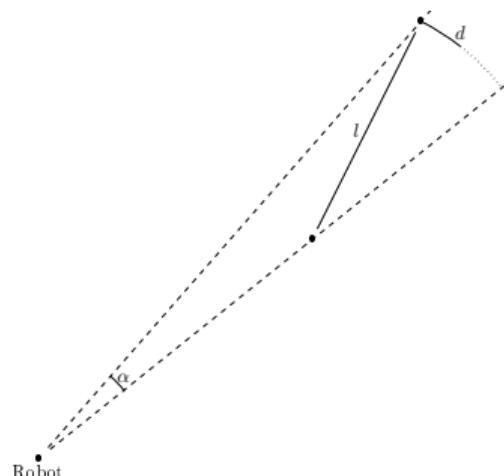
$$(9) \quad sensor\ width(m) = \frac{f(m) \times real\ width(m) \times image\ width(pixels)}{object\ width(pixels) \times object\ distance(m)}$$



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

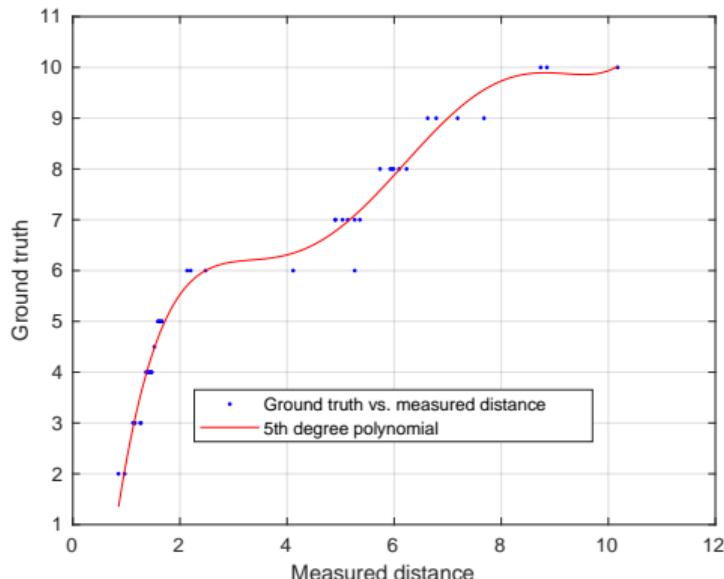
Scarto dei duplicati

È stato necessario effettuare una fase di clustering al fine di scartare le bounding box duplicate. L'algoritmo di clustering utilizzato è DBSCAN (Density based scan) [16], i cui parametri principali sono **eps**, ovvero la massima distanza fra due punti affinché vengano considerati appartenenti a un cluster , **min _ samples**, ovvero il numero minimo di punti affinché un cluster sia valido (nel nostro caso è uguale a 1 in quanto non vogliamo scartare ROI) ed infine la metrica di distanza.



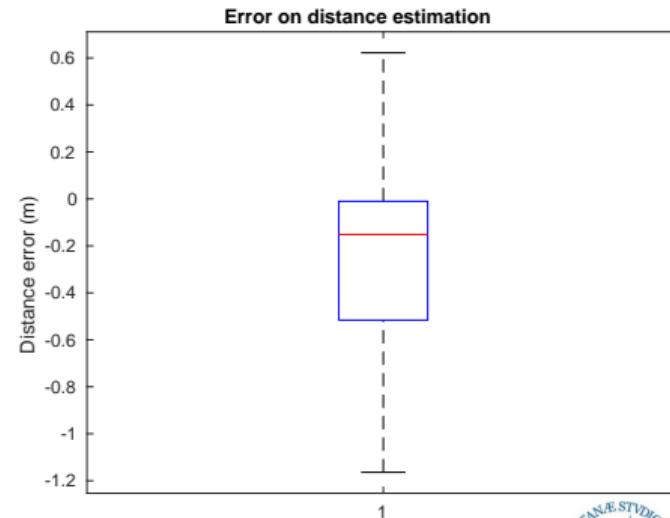
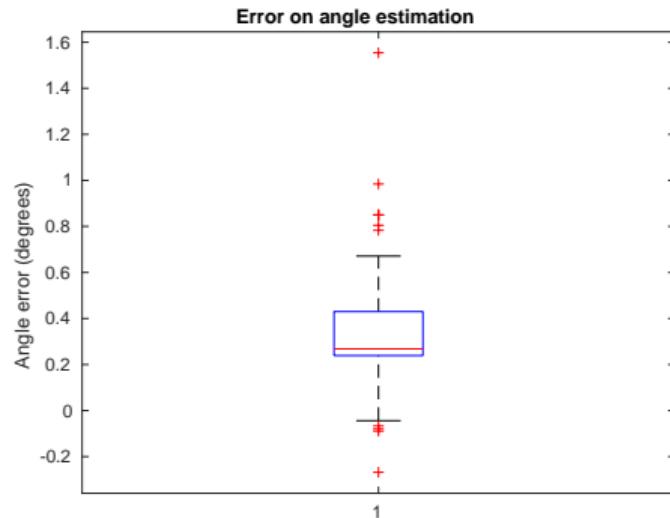
Correzione dei valori |

Per migliorare la stima sulla distanza abbiamo paragonato le reali distanze dei target con le stime effettuate dal sistema ottenendo il polinomio interpolante $0.003116 * x^5 - 0.09722 * x^4 + 1.124 * x^3 - 5.908 * x^2 + 14.5 * x - 7.367$, che approssima la funzione di correzione della stima.



Correzione dei valori II

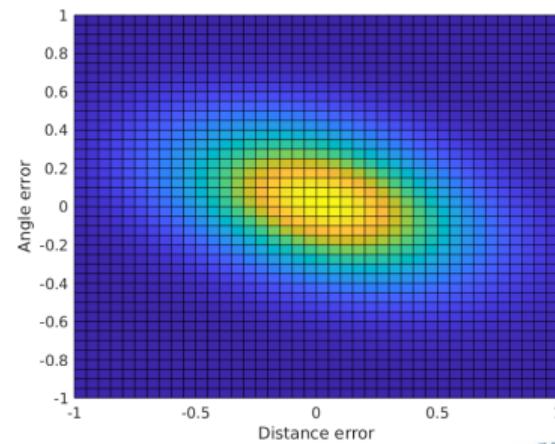
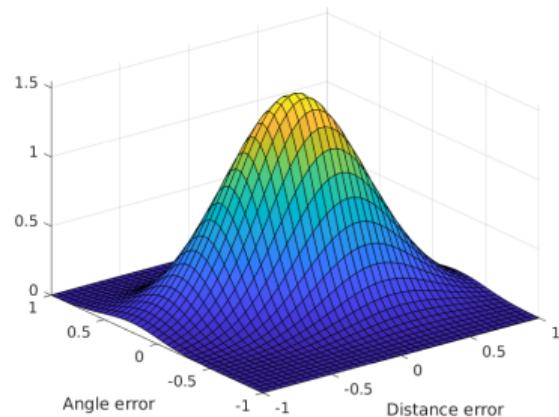
Le bontà della stima della distanza e dell'angolo si evince dalle figure :



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Correzione dei valori III

Nelle figure viene riportata la distribuzione gaussiana multivariata in 3 e 2 dimensioni con matrice di covarianza calcolata dai nostri test :



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

- 1** Introduzione
- 2** Gestione dei nodi ROS
- 3** Modello del moto e posizionamento
- 4** Object recognition
- 5** Posizione dei target
- 6** Modello probabilistico
- 7** Scheduling dei comportamenti
- 8** Possibili modifiche



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Section 6

Modello probabilistico



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Modello probabilistico

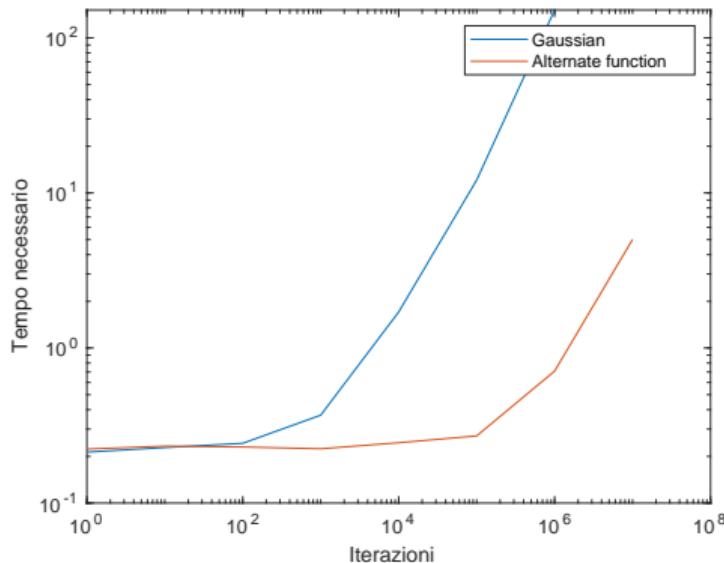
Nella nostra applicazione la distanza stimata dell'oggetto sarà soggetta ad errore non trascurabile. Per questa ragione e per i problemi legati alla triangolazione abbiamo abbandonato la rappresentazione basata su oggetti e abbiamo fatto ricorso ad un filtro di occupazione bayesiano (BOF) [17].

Nel filtro di occupazione bayesiano, il problema dell'associazione dei dati viene superato in quanto viene gestito da un livello di astrazione superiore. Il concetto di oggetto viene difatti riformulato da proprietà più utili quali occupazione o rischio, che vengono stimate direttamente per ogni cella utilizzando sia osservazioni dai sensori che conoscenze pregresse. Le caratteristiche di incertezza legate ai sensori vengono descritte, in questo modello, attraverso le probabilità di occupazione.



Funzione densità di probabilità |

Al fine di trasformare le osservazioni ottenute in una probabilità che le persone si trovino effettivamente nella posizione indicata è stato necessario definire una funzione densità di probabilità. La distribuzione normale è computazionalmente onerosa. Abbiamo quindi fatto ricorso ad una approssimazione.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Funzione densità di probabilità II

La distribuzione triangolare è spesso utilizzata per approssimare una gaussiana, tuttavia, presenta delle caratteristiche non volute per il nostro caso d'uso.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Funzione densità di probabilità II

La distribuzione triangolare è spesso utilizzata per approssimare una gaussiana, tuttavia, presenta delle caratteristiche non volute per il nostro caso d'uso.

- Non presenta le "fat tails"



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Funzione densità di probabilità II

La distribuzione triangolare è spesso utilizzata per approssimare una gaussiana, tuttavia, presenta delle caratteristiche non volute per il nostro caso d'uso.

- Non presenta le "fat tails"
- Dopo una prima osservazione non confermata successivamente la probabilità scende a zero.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Funzione densità di probabilità II

La distribuzione triangolare è spesso utilizzata per approssimare una gaussiana, tuttavia, presenta delle caratteristiche non volute per il nostro caso d'uso.

- Non presenta le "fat tails"
- Dopo una prima osservazione non confermata successivamente la probabilità scende a zero.
- Le occlusioni annullano la probabilità



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Funzione densità di probabilità II

La distribuzione triangolare è spesso utilizzata per approssimare una gaussiana, tuttavia, presenta delle caratteristiche non volute per il nostro caso d'uso.

- Non presenta le "fat tails"
- Dopo una prima osservazione non confermata successivamente la probabilità scende a zero.
- Le occlusioni annullano la probabilità
- I target in movimento non verrebbero rilevati



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Funzione densità di probabilità II

La distribuzione triangolare è spesso utilizzata per approssimare una gaussiana, tuttavia, presenta delle caratteristiche non volute per il nostro caso d'uso.

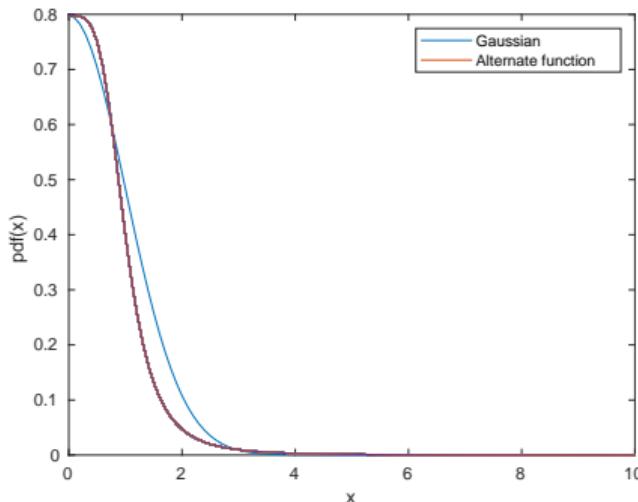
- Non presenta le "fat tails"
- Dopo una prima osservazione non confermata successivamente la probabilità scende a zero.
- Le occlusioni annullano la probabilità
- I target in movimento non verrebbero rilevati

Essendo la scansione una operazione estremamente costosa non è nemmeno possibile affidarsi esclusivamente all'aggiunta di rumore alla griglia di occupazione confidando in una eventuale convergenza.

Funzione densità di probabilità III

Al fine di ottenere un'approssimazione di una gaussiana adatta al nostro scenario, per modellare la probabilità che data l'occupazione della cella in posizione x si ottenga l'osservazione z , è stato utilizzato un funzionale ispirato al guadagno del filtro di Butterworth.

$$(10) \quad p(z|x) = \frac{K}{1 + d(x, z)^4}$$



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Parametri PDF |

La distanza euclidea nell'equazione 10 porterebbe a formare aree ad alta probabilità di forma circolare. Questo però non si adatterebbe bene al nostro modello di errore del sensore: l'angolo dell'oggetto rispetto al robot è noto con una precisione molto elevata, mentre la maggior parte dell'incertezza si concentra nella distanza. Calcoliamo quindi la distanza non come distanza euclidea tra le coordinate cartesiane della cella e dell'osservazione, ma come norma L^2 delle coordinate polari, opportunamente normalizzate per tenere conto della diversa incertezza sulla misura di angolo e distanza.

Parametri PDF II

L'incertezza sulla distanza è ricavata dalle misure di calibrazione effettuate. Per l'angolo è stato seguito un approccio differente: utilizzare una varianza calcolata a partire da una serie di osservazioni porterebbe ad assegnare maggiore probabilità ad oggetti lontani.

In un setup come quello in figura, con due target rilevati, rispettivamente a distanza d_1 e d_2 , le aree ad alta probabilità relative ad entrambi i punti avranno lunghezza l e ampiezze α_1 e α_2 .

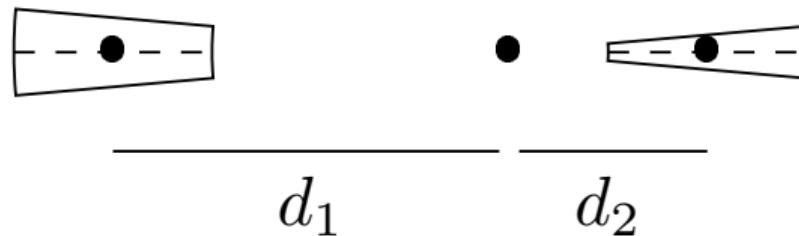


Figura: Area della regione probabile a seconda della distanza.

Parametri PDF III

La dimensione dell'area ad alta probabilità è direttamente proporzionale alla distanza e all'ampiezza dell'angolo. Se quindi α_1 coincidesse con α_2 ad oggetti distanti verrebbero associate aree molto più grandi.

$$(11) \quad \begin{aligned} A_1 &= \pi((d_1 + l/2)^2 - (d_1 - l/2)^2) \frac{\alpha_1}{2\pi} = \alpha_1 d_1 l \\ A_2 &= \pi((d_2 + l/2)^2 - (d_2 - l/2)^2) \frac{\alpha_2}{2\pi} = \alpha_2 d_2 l \\ \frac{A_1}{A_2} &= \frac{\alpha_1 d_1}{\alpha_2 d_2} \end{aligned}$$

Per evitare questo problema, dopo avere individuato un valore α^* appropriato questo viene scalato per un coefficiente inversamente proporzionale alla distanza della cella, quindi $\alpha_1 = \alpha^*/d_1$ e $\alpha_2 = \alpha^*/d_2$, da cui $A_1 = A_2$.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Aggiornamento bayesiano |

Dato un insieme di osservazioni ottenute da una scansione Z_i aggiorniamo il belief.

$$(12) \quad p(\mathbf{x}_i|Z_i) = p(\mathbf{x}_{i-1}) \cdot \sum_{\mathbf{z} \in Z_i} p(\mathbf{z}|\mathbf{x}_i)$$

Questa formula è valida sotto le seguenti assunzioni:



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Aggiornamento bayesiano |

Dato un insieme di osservazioni ottenute da una scansione Z_i aggiorniamo il belief.

$$(12) \quad p(\mathbf{x}_i|Z_i) = p(\mathbf{x}_{i-1}) \cdot \sum_{\mathbf{z} \in Z_i} p(\mathbf{z}|\mathbf{x}_i)$$

Questa formula è valida sotto le seguenti assunzioni:

- In uno stesso scan, date due osservazioni z_1 e z_2 queste non hanno intersezione, quindi $p(z_1 \cup z_2) = p(z_1) + p(z_2) - p(z_1 \cap z_2) = p(z_1) + p(z_2)$



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Aggiornamento bayesiano |

Dato un insieme di osservazioni ottenute da una scansione Z_i aggiorniamo il belief.

$$(12) \quad p(\mathbf{x}_i|Z_i) = p(\mathbf{x}_{i-1}) \cdot \sum_{\mathbf{z} \in Z_i} p(\mathbf{z}|\mathbf{x}_i)$$

Questa formula è valida sotto le seguenti assunzioni:

- In uno stesso scan, date due osservazioni z_1 e z_2 queste non hanno intersezione, quindi $p(z_1 \cup z_2) = p(z_1) + p(z_2) - p(z_1 \cap z_2) = p(z_1) + p(z_2)$
- In assenza di nuove osservazioni la stima dello stato del sistema rimane invariata: $\overline{bel}(x_i) = bel(x_{i-1})$



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Aggiornamento bayesiano II

In mancanza di nuovi dati si potrebbero tentare due approcci: resettare le stime o lasciarle del tutto invariate.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Aggiornamento bayesiano II

In mancanza di nuovi dati si potrebbero tentare due approcci: resettare le stime o lasciarle del tutto invariate.

- Lasciare invariato il belief a seguito di multiple scansioni senza successo porterebbe a non notare che tutte le persone nell'ambiente in cui ci si trova sono andate via, continuando a considerare valide tutte le posizioni precedenti.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Aggiornamento bayesiano II

In mancanza di nuovi dati si potrebbero tentare due approcci: resettare le stime o lasciarle del tutto invariate.

- Lasciare invariato il belief a seguito di multiple scansioni senza successo porterebbe a non notare che tutte le persone nell'ambiente in cui ci si trova sono andate via, continuando a considerare valide tutte le posizioni precedenti.
- Dall'altro lato, un approccio troppo drastico quale immediatamente scartare tutte le precedenti stime porterebbe a perdere informazioni utili a causa di occlusioni temporanee



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Aggiornamento bayesiano II

In mancanza di nuovi dati si potrebbero tentare due approcci: resettare le stime o lasciarle del tutto invariate.

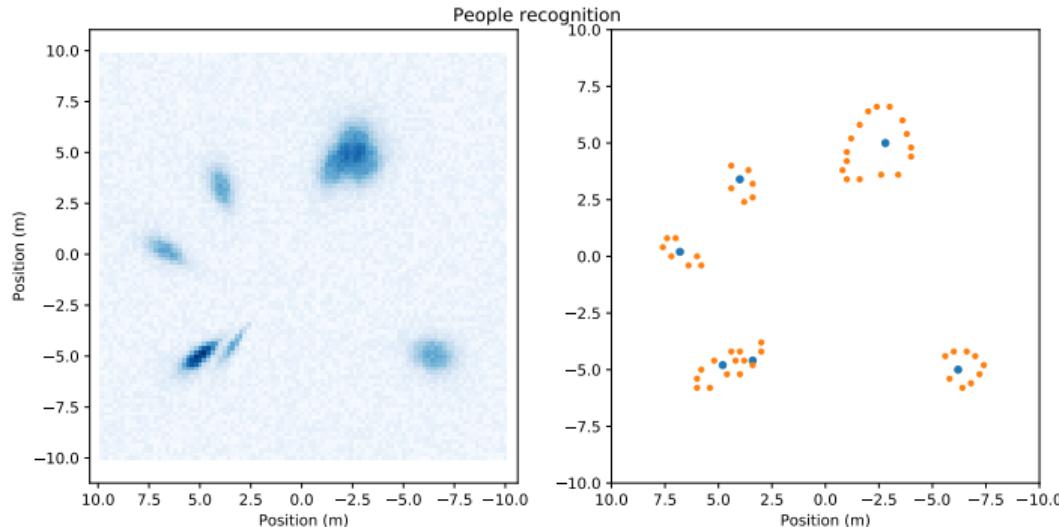
- Lasciare invariato il belief a seguito di multiple scansioni senza successo porterebbe a non notare che tutte le persone nell'ambiente in cui ci si trova sono andate via, continuando a considerare valide tutte le posizioni precedenti.
- Dall'altro lato, un approccio troppo drastico quale immediatamente scartare tutte le precedenti stime porterebbe a perdere informazioni utili a causa di occlusioni temporanee

Effettuiamo uno smoothing degli histogrammi prima di ogni update e aggiungiamo inoltre del rumore.

Individuazione dei cluster

Al fine di individuare le zone con alta probabilità di contenere persone, in primo luogo effettuiamo una sogliatura utilizzando il metodo Otsu [18], ottenendo così una mappa binaria.

Estraiamo le regioni ad alta probabilità contigue ed i loro contorni [19] e selezioniamo come centro il punto con maggiore probabilità.



- 1** Introduzione
- 2** Gestione dei nodi ROS
- 3** Modello del moto e posizionamento
- 4** Object recognition
- 5** Posizione dei target
- 6** Modello probabilistico
- 7** Scheduling dei comportamenti
- 8** Possibili modifiche



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Section 7

Scheduling dei comportamenti



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Modalità esplorazione

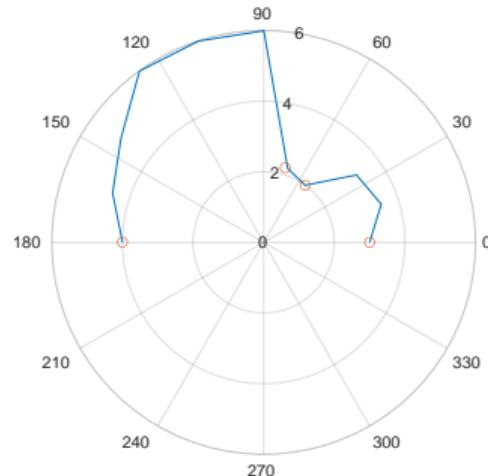
Quando il robot entra in modalità esplorazione il suo comportamento è di esplorazione casuale della mappa. Il robot continua ad esplorare ed effettuare scan periodici fino a quando non viene rilevato un obiettivo. In tal caso il robot entra in modalità campi di potenziale. Se il robot incontra un ostacolo nel suo cammino ruota di 90° nella direzione con più spazio e continua l'esplorazione casuale.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Bug mode

Quando il robot entra in modalità bug [20] effettuiamo uno scan lidar. Confrontiamo i valori ottenuti con una soglia al fine di individuare le discontinuità nel profilo degli ostacoli. Analizziamo successivamente le discontinuità del segnale, corrispondenti ai bordi degli ostacoli. Da tali punti viene calcolato l'angolo fra la retta che li congiunge con l'obbiettivo. Il robot si muove infine verso il punto al quale corrisponde l'angolo minore, che ci farà allontanare meno dall'obbiettivo.



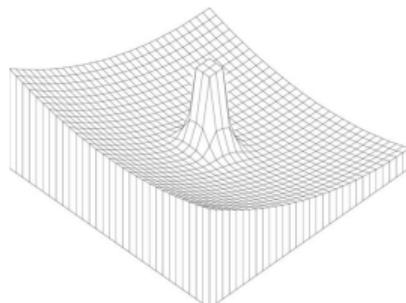
Campi di potenziale

Potenziale attrattivo

I modelli basati sui campi di potenziale richiedono la definizione di potenziali attrattivi e repulsivi.

Il potenziale attrattivo è centrato sul target, ed è modellato come un potenziale quadratico, la forza attrattiva è quindi lineare rispetto alla distanza (Eq. 13).

$$(13) \quad F_{att}(\mathbf{x}) = k_a \cdot |\mathbf{x} - \mathbf{x_d}|$$



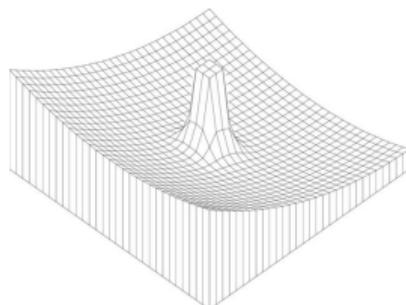
UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Campi di potenziale

Potenziale repulsivo

Gli ostacoli generano invece un potenziale repulsivo. Questo ha un raggio d'azione limitato, la forza repulsiva è massima per un ostacolo a distanza nulla, e decresce in modo monotono fino ad annullarsi se la distanza dell'ostacolo supera una soglia d_t con la legge descritta nell'equazione:

$$(14) \quad F_{rep}(distance) = \begin{cases} k_r \cdot \left(\frac{1}{distance^2} - \frac{1}{d_t^2} \right) & distance \leq d_t \\ 0 & d > d_{thresh} \end{cases}$$



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

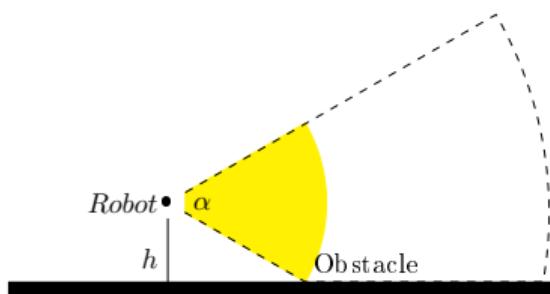
Campi di potenziale

Rilevamento dell'ostacolo

La maggior parte dei potenziali repulsivi descritti in letteratura dipendono esclusivamente dalla distanza dall'ostacolo, interferendo con il moto anche se questo avviene parallelamente all'ostacolo. Per ovviare a questo problema prendiamo in considerazione solo gli ostacoli situati in un FOV frontale al robot con ampiezza α calcolata per permettere di muoversi parallelamente ad una superficie mantenendo una distanza di sicurezza h :

(15)

$$\alpha = 2 \sin^{-1} \frac{h}{d_t}$$



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Campi di potenziale

Calcolo del movimento

L'approccio seguito consiste in due comportamenti calcolati indipendentemente: il comportamento traslatorio e il comportamento rotazionale.

La traslazione viene calcolata in funzione dello spazio di movimento a disposizione e dall'angolo rispetto al target secondo l'equazione 16, ottenuta per interpolazione del profilo desiderato. L'angolo viene invece determinato dalla somma vettoriale delle forze attrattive e repulsive. La rotazione viene inoltre sogliata per evitare cambi di direzione troppo bruschi(Eq. 17).

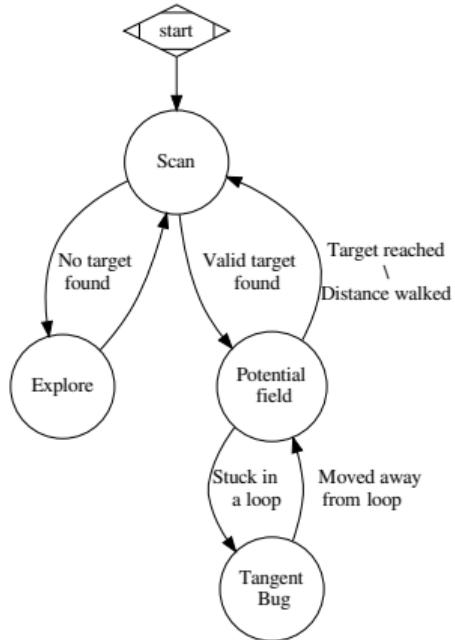
$$(16) \quad d = 0.5 + \frac{0.9 \cdot \text{distance} - 0.5}{1 + \left| \frac{6\omega}{\pi} \right|}$$

$$(17) \quad \omega = \begin{cases} \omega^* = \theta - \angle(\mathbf{F}_{att} - \mathbf{F}_{rep}) & -\Omega \leq \omega^* \leq \Omega \\ -\Omega & \omega^* < -\Omega \\ \Omega & \omega^* < \Omega \end{cases}$$



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Automa a stati finiti

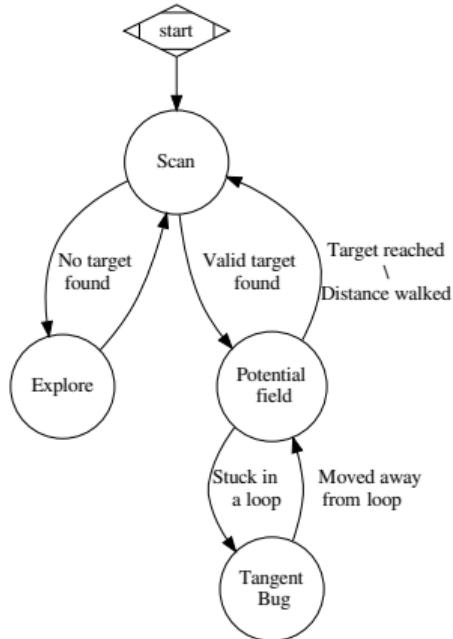


- All'avvio, **Change** effettua uno scan dell'ambiente.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Automa a stati finiti

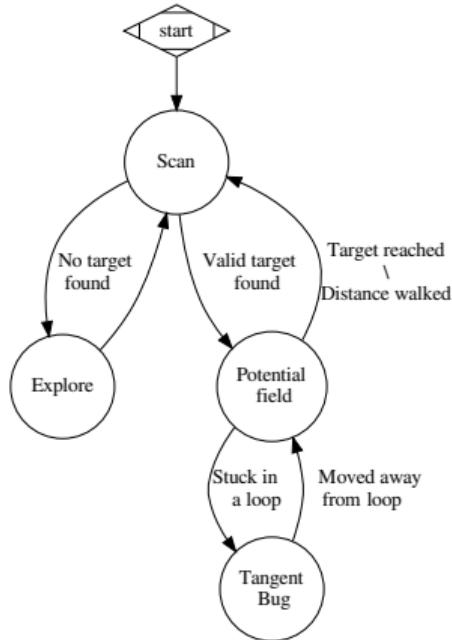


- All'avvio, **Change** effettua uno scan dell'ambiente.
- Se non vengono rilevati obiettivi entra in modalità esplorazione e, dopo aver percorso una determinata distanza, effettua un nuovo scan



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

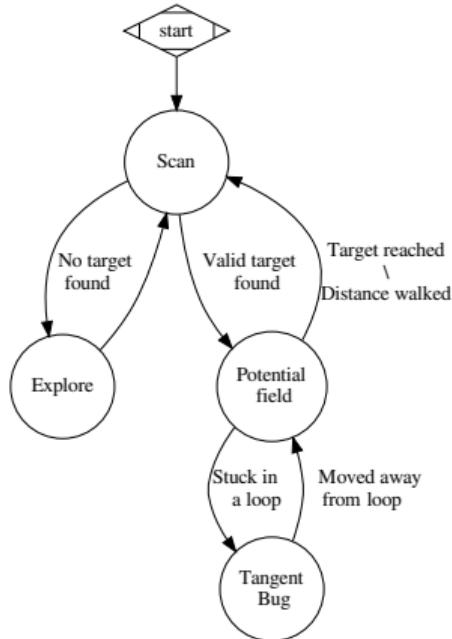
Automa a stati finiti



- All'avvio, **Change** effettua uno scan dell'ambiente.
- Se non vengono rilevati obiettivi entra in modalità esplorazione e, dopo aver percorso una determinata distanza, effettua un nuovo scan
- Se viene rilevato un obiettivo il robot entra in modalità campi di potenziale fermandosi una volta raggiunto l'obiettivo o percorso una determinata distanza.



Automa a stati finiti



- All'avvio, **Change** effettua uno scan dell'ambiente.
- Se non vengono rilevati obiettivi entra in modalità esplorazione e, dopo aver percorso una determinata distanza, effettua un nuovo scan
- Se viene rilevato un obiettivo il robot entra in modalità campi di potenziale fermandosi una volta raggiunto l'obiettivo o percorso una determinata distanza.
- Se, in modalità campi di potenziale, il **TIAGo** rimane bloccato in un loop, entra in modalità bug fino all'uscita dal loop, e ritorna nella modalità campi di potenziale.



- 1** Introduzione
- 2** Gestione dei nodi ROS
- 3** Modello del moto e posizionamento
- 4** Object recognition
- 5** Posizione dei target
- 6** Modello probabilistico
- 7** Scheduling dei comportamenti
- 8** Possibili modifiche



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Section 8

Possibili modifiche



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Possibili modifiche

Riportiamo di seguito possibili modifiche da apportare:

- Aggiungere un algoritmo per effettuare SLAM, in modo da poter pianificare il moto con algoritmi come A* [21] o RRT [22]



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Possibili modifiche

Riportiamo di seguito possibili modifiche da apportare:

- Aggiungere un algoritmo per effettuare SLAM, in modo da poter pianificare il moto con algoritmi come A* [21] o RRT [22]
- Utilizzare le informazioni ottenute dalla camera (e.g. flusso ottico [23]) anche per migliorare le stime sulla posizione.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Possibili modifiche

Riportiamo di seguito possibili modifiche da apportare:

- Aggiungere un algoritmo per effettuare SLAM, in modo da poter pianificare il moto con algoritmi come A* [21] o RRT [22]
- Utilizzare le informazioni ottenute dalla camera (e.g. flusso ottico [23]) anche per migliorare le stime sulla posizione.
- Utilizzare una depth-cam per stimare con più precisione la distanza delle persone



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Possibili modifiche

Riportiamo di seguito possibili modifiche da apportare:

- Aggiungere un algoritmo per effettuare SLAM, in modo da poter pianificare il moto con algoritmi come A* [21] o RRT [22]
- Utilizzare le informazioni ottenute dalla camera (e.g. flusso ottico [23]) anche per migliorare le stime sulla posizione.
- Utilizzare una depth-cam per stimare con più precisione la distanza delle persone
- Tracciamento delle persone in real-time



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Possibili modifiche

Riportiamo di seguito possibili modifiche da apportare:

- Aggiungere un algoritmo per effettuare SLAM, in modo da poter pianificare il moto con algoritmi come A* [21] o RRT [22]
- Utilizzare le informazioni ottenute dalla camera (e.g. flusso ottico [23]) anche per migliorare le stime sulla posizione.
- Utilizzare una depth-cam per stimare con più precisione la distanza delle persone
- Tracciamento delle persone in real-time
- Utilizzare più robot distribuiti nell'ambiente per aggiornare in modo collaborativo [24] le stime sulla distribuzione delle persone, eventualmente anche con una mappatura semantica degli ambienti [25]



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Grazie per l'attenzione!



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Bibliografia |

- [1] Petrônio CL Silva, Paulo VC Batista, Hélder S Lima, Marcos A Alves, Frederico G Guimarães, and Rodrigo CP Silva.
Covid-abs: An agent-based model of covid-19 epidemic to simulate health and economic effects of social distancing interventions.
Chaos, Solitons & Fractals, 139:110088, 2020.
- [2] G. Bradski.
The OpenCV Library.
Dr. Dobb's Journal of Software Tools, 2000.
- [3] A Rosebrock.
Imutils.
- [4] J. D. Hunter.
Matplotlib: A 2d graphics environment.
Computing in Science & Engineering, 9(3):90–95, 2007.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Bibliografia II

- [5] Charles R. Harris, K. Jarrod Millman, St'efan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fern'andez del R'io, Mark Wiebe, Pearu Peterson, Pierre G'erard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant.
Array programming with numpy.
Nature, 585(7825):357–362, September 2020.
- [6] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al.
Scikit-learn: Machine learning in python.
Journal of machine learning research, 12(Oct):2825–2830, 2011.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Bibliografia III

- [7] Jordi Pages, Luca Marchionni, and Francesco Ferro.
Tiago: the modular robot that adapts to different research needs.
In *International workshop on robot modularity, IROS*, 2016.
- [8] Tiago datasheet.
- [9] Webots.
Webots user guide: Pal robotics' tiago iron.
- [10] Webots.
Lidar specifications.
- [11] J. Borenstein and L. Feng.
Gyrodometry: a new method for combining data from gyros and odometry in mobile robots.
In *Proceedings of IEEE International Conference on Robotics and Automation*, volume 1, pages 423–428 vol.1, 1996.

Bibliografia IV

- [12] Kurt Seifert and Oscar Camacho.
Implementing positioning algorithms using accelerometers.
Freescale Semiconductor, pages 1–13, 2007.
- [13] S. Maeyama, N. Ishikawa, and S. Yuta.
Rule based filtering and fusion of odometry and gyroscope for a fail safe dead reckoning system of a mobile robot.
In *1996 IEEE/SICE/RSJ International Conference on Multisensor Fusion and Integration for Intelligent Systems (Cat. No.96TH8242)*, pages 541–548, 1996.
- [14] Debabrata Ghosh and Naima Kaabouch.
A survey on image mosaicing techniques.
Journal of Visual Communication and Image Representation, 34:1–11, 2016.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Bibliografia V

- [15] P. Adarsh, P. Rathi, and M. Kumar.

Yolo v3-tiny: Object detection and recognition using one stage improved model.
In *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*, pages 687–694, 2020.

- [16] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al.

A density-based algorithm for discovering clusters in large spatial databases with noise.
In *Kdd*, volume 96, pages 226–231, 1996.

- [17] MK Tay, Kamel Mekhnacha, Manuel Yguel, Christophe Coue, Cédric Pradalier, Christian Laugier, Th Fraichard, and Pierre Bessiere.

The bayesian occupation filter.

In *Probabilistic Reasoning and Decision Making in Sensory-Motor Systems*, pages 77–98. Springer, 2008.

Bibliografia VI

- [18] Nobuyuki Otsu.
A threshold selection method from gray-level histograms.
IEEE transactions on systems, man, and cybernetics, 9(1):62–66, 1979.
- [19] Satoshi Suzuki et al.
Topological structural analysis of digitized binary images by border following.
Computer vision, graphics, and image processing, 30(1):32–46, 1985.
- [20] I. Kamon, E. Rivlin, and E. Rimon.
A new range-sensor based globally convergent navigation algorithm for mobile robots.
In *Proceedings of IEEE International Conference on Robotics and Automation*, volume 1, pages 429–435 vol.1, 1996.

Bibliografia VII

- [21] Akshay Guruji, Himansh Agarwal, and Deep Parsadia.
Time-efficient a* algorithm for robot path planning.
Procedia Technology, 23:144–149, 12 2016.
- [22] S. Karaman, M. R. Walter, A. Perez, E. Frazzoli, and S. Teller.
Anytime motion planning using the rrt*.
In *2011 IEEE International Conference on Robotics and Automation*, pages 1478–1483, 2011.
- [23] Heng Deng, Usman Arif, Kun Yang, Zhiyu Xi, Quan Quan, and Kai-Yuan Cai.
Global optical flow-based estimation of velocity for multicopters using monocular vision in gps-denied environments.
Optik, 219:164923, Oct 2020.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Bibliografia VIII

- [24] Don Miner.
Swarm robotics algorithms: A survey.
Report, MAPLE lab, University of Maryland, 2007.
- [25] Ioannis Kostavelis and Antonios Gasteratos.
Semantic mapping for mobile robotics tasks: A survey.
Robotics and Autonomous Systems, 66:86–103, 2015.



UNIVERSITÀ
DEGLI STUDI
DI PALERMO