Draft ricerca e sviluppa soluzioni innovative per l’intelligenza artificiale al fine di migliorare l’autonomia della tecnologia dei droni.Per affrontare i vari aspetti del progetto il Team è diviso in squad:

Robotic Systems Engineering

Deep Learning and Computer Vision

Obstacle Avoidance and Motion Planning

Simultaneous Localization and Mapping

Business & Outreach

Io faccio parte di “Obstacle Avoidance and Motion Planning” che si occupa principalmente della pianificazione del percorso più efficiente del drone per raggiungere punti di una mappa e della previsione del comportamento del drone utilizzando dati in tempo reale.

OAMP è formata da 13 componenti coordinati da uno squad-leader.

Le riunioni tra i membri della squadra sono avvenuti in videochiamata tramite “Meet”.

Le prime videochiamate sono servite allo squad-leader per spiegare gli scopi del team, l’organizzazione, i lavori che sono stati effettuati in passato, quali bisognava effettuare in futuro, le varie applicazioni utilizzate per i lavori e le applicazioni per mantenersi in contatto.

Il primo compito assegnato è stata l’installazione di Ubuntu, un sistema operativo nato nel 2004 e basato su Linux, e ROS (Robot Operating System), un framework open-source progettato per lo sviluppo di applicazioni per robot.

ROS è un sistema che consente di eseguire nodi di una stessa 'rete'. Ad esempio vari programmi (in c++ o python) che vanno dalla logica di alto livello, a script che pubblicano i dati. Questi nodi possono scambiarsi informazioni tramite topic e possono interagire richiedendo informazioni l'uno all'altro tramite i service.

Il compito successivo è stato quello di comprendere e prendere dimestichezza con i codici della squad OAMP, presentati dallo squad-leader, in particolare il Path Planning, A\* e DWA.

Path planning è un nome generico che include tutti i programmi che partendo dalla posizione del drone, un goal e la conoscenza del mondo circostante generano un percorso.

L'*A\** è un algoritmo di pathfinding.

Il pathfinding è la rappresentazione, tramite un'applicazione per computer, del percorso più breve tra due punti. Questo campo di ricerca si basa sull'algoritmo di Dijkstra per trovare il percorso più breve su un grafico.

L'*A\** è utilizzata per generare un percorso in una rappresentazione discreta di una mappa, quinditrova il percorso più corto possibile tra due punti.

Inoltre è stato aggiunto all’A\* un sistema che consente di modificare il comportamento di quest’ultimo per far passare il drone lontano dagli ostacoli, e/o più in alto rispetto a dove passerebbe normalmente.

Il DWA invece è un algoritmo di motion planning che non viene più utilizzato dal Team.

Il motion planning al contrario del path planning è un algoritmo di basso livello.

Nel caso del Team gli algoritmi di motion planning sono quelli che danno i comandi di velocità al drone, mentre quelli di path planning creano la lista di waypoint intermedi da cui far passare il drone. Esistono anche altri casi dove il limite tra motion planning e path planning potrebbe essere meno evidente o addirittura non esistere.

Successivamente lo squad-leader ha dato come obiettivo a 4 membri della squadra di rendere più facilmente leggibile epiù efficiente il codice DWA già esistente.

Ci siamo organizzati in due videochiamate, nella prima il codice è stato diviso in quattro parti, assegnandone una ad ognuno di noi. Autonomamente ognuno ha indentificato e corretto inefficienze ed errori e riordinato le varie funzioni del codice.

Durante la seconda videochiamata ci siamo confrontati e abbiamo messo insieme in maniera opportuna ogni parte, in modo tale da ottenere il nuovo codice del DWA chiaro e pulito.

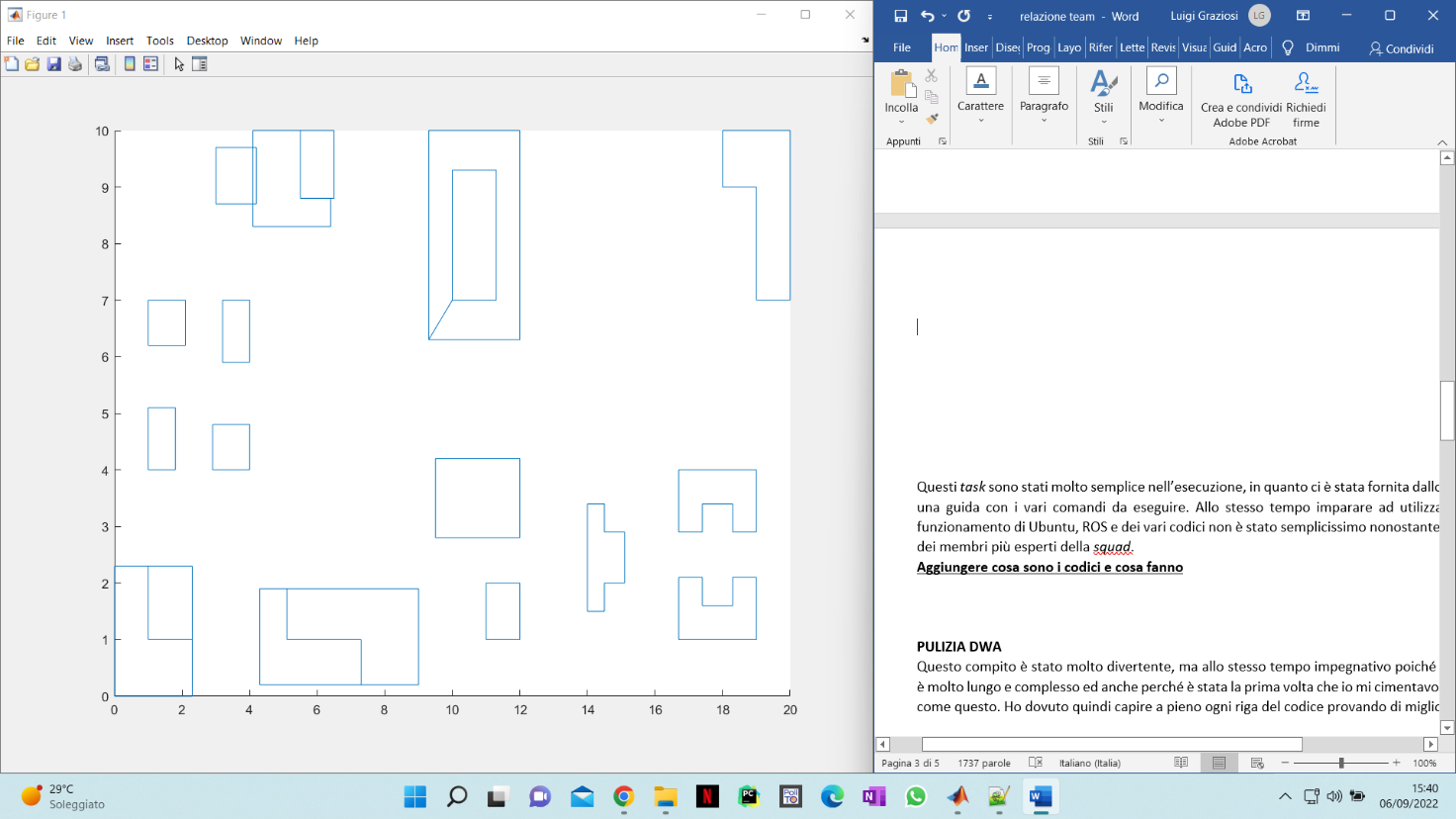
Terminata la pulizia del codice DWA ho elaborato uno script in Python.

Data una lista di waypoint**,** che arrivano dall’ A\*, il nuovo codice suddivide la lista in "punti importanti", per punti importanti si intendono o tutti i punti distanziati di almeno 1.5m, o che cambiano direzione, cioè che hanno le coordinate x e y entrambe diverse dal punto precedente, o in caso di percorso rettilineo i punti di inizio e fine.

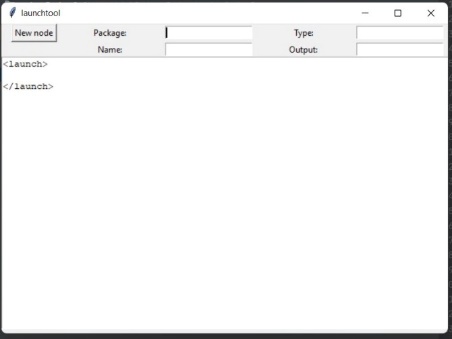
Successivamente lo squad-leader dopo una breve spiegazione di GitLab, ci ha chiesto di imparare ad usare lo stesso seguendo alcune guide online.

Inoltre per verificare le conoscenze acquisite e per prendere dimestichezza abbiamo clonato l’archivio del Team sul nostro computer, con sistema operativo Ubuntu, e caricato online alcuni file di prova.

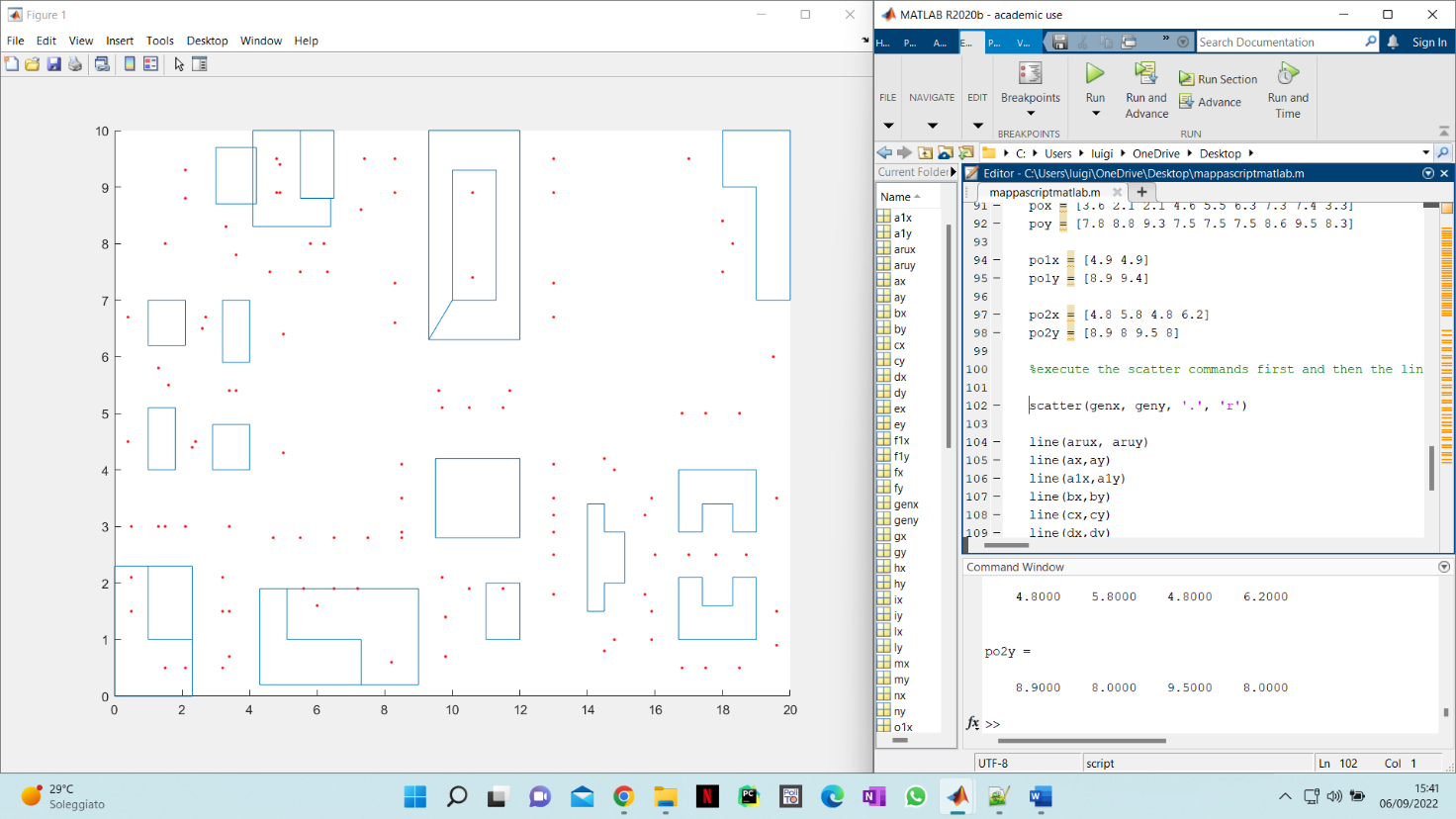
Dopo aver imparato le basi di GitLab mi è stato chiesto di scrivere una funzione in Python chiamata ‘Automatic field generator’. Questa doveva aiutare gli altri membri del team a disegnare a computer le mappe virtuali su cui bisognava fare le prove con il drone.

Si riesce quindi ad ottenere una lista di punti tridimensionali in cui si trovano gli ostacoli, ovvero punti che non possono essere utilizzati dal drone come waypoint.

Le mappe sono formate da ostacoli di forma riconducibile a parallelepipedi. Da un file human-readable, cioè un file con qualsiasi codifica di dati o informazioni che può essere letta dall'uomo, contenente le coordinate dell’angolo in basso a sinistra e in alto a destra di ogni parallelepipedo si riesce facilmente a realizzare l’elenco dei punti appartenenti all’ostacolo.

Il compito successivo è stato quello di sviluppare una versione iniziale di un tool per completare automaticamente i file ".launch" di ROS. I cosiddetti “launch file” sono dei file XML contenenti la lista dei nodi e sono utili per avviare facilmente quest’ultimi con i rispettivi parametri. Per la riuscita del task mi sono servito della libreria python “tkinter”, la quale ci permette di creare delle interfacce grafiche per programmi.

Disegno della mappa del Leonardo Drone Contest 2022 sviluppata con l’utilizzo del software Matlab

Uno dei punti importanti del Leonardo Drone Contest 2022 è la scansione verticale degli ostacoli per ricercare gli aruco.

Gli aruco sono dei quadrati composti da un ampio bordo nero e da una matrice binaria interna di colore bianco, simili a codici QR. Questi si trovano nel campo gara e devono essere raggiunti dal drone per far guadagnare punti al team.

Utilizzando la mappa ufficiale della sfida di quest’anno, fornitaci da Leonardo, ho stilato una lista di punti importanti, ovvero tutti quei punti in cui il drone deve posizionarsi in modo da riuscire a scansionare i lati degli ostacoli su cui sono presenti gli aruco.

Campo gara e punti per scansione verticale realizzati con matlab.

Dopo aver completato la ricerca dei punti, con l’aiuto del software Matlab ho disegnato la mappa ufficiale in 2D per verificare approssimativamente la veridicità delle informazione ottenute.

**Results and recommendations**

**Questa sezione mostra e discute in modo approfondito i risultati dell'attività. Devono essere evidenziati i principali risultati, così come i punti di forza. Devono essere individuate le criticità e proposte eventuali azioni correttive. Sono inoltre incoraggiate raccomandazioni per lavori futuri.**

I Risultati del mio lavoro si possono suddividere in due parti.

La prima è stata di supporto agli altri membri del team infatti, aver pulito il codice e averlo reso più efficiente ha migliorato sicuramente le prestazioni del drone, in aggiunta essendo più chiaro e più pulito si può prestare meglio a modifiche future ed alla correzione di eventuali errori.

Anche la scrittura della funzione ‘Automatic field generator’ riduce i tempi di lavoro e per di più si limitano, se non addirittura eliminano, gli errori nella generazione della mappa. Infatti non si perde tempo a scrivere tutti i punti appartenenti agli ostacoli, ma bisogna fornire alla procedura solo i 2 punti prestabiliti.

Un altro elemento che ha aiutato molto è il tool per completare automaticamente i file ".launch”. Quest’ultimo facilita notevolmente l’inserimento degli input rendendo il lavoro più pratico e veloce, aiutando a scrivere in maniera dinamica il codice. Per mancanza di tempo però, il programma da la possibilità di inserire solo le informazioni principali, ma è possibile perfezionarlo aggiungendo all’interfaccia grafica più voci che inseriscono i vari valori.

La seconda parte invece è finalizzata all’applicazione pratica del drone.

Trovare i punti in cui il drone deve posizionarsi per riuscire a scansionare le facce degli ostacoli è un obbiettivo fondamentale ai fini della gara, quindi la scrittura della mappa con matlab è stata molto utile per verificare la correttezza delle informazioni ottenute e rendere anche più semplice l’eventuale correzione da parte degli altri membri del team.

In generale l’esperienza nel Team è stata molto proficua e affascinante sia dal punto di vista didattico che personale. Interagire con ragazzi più grandi di me e con più esperienza mi ha fatto capire e imparare moltissimo. I task che mi sono stati assegna durante la permanenza nel team non sono stati facili da eseguire, ma allo stesso tempo mi hanno insegnato ad utilizzare nuovi software, hanno ampliato la mia conoscenza dei linguaggi ed ho assunto maggiore consapevolezza degli ambienti informatici.

Per le esperienze future mi permetto di suggerire di svolgere le riunioni e i vari lavori in presenza in modo da avere più iterazioni tra i membri del team e di fornire più risorse didattiche, soprattutto ai nuovi reclutati.

**Motivations**

**Questa sezione descrive il contesto e gli obiettivi dell'attività.**

****Il Leonardo Drone Contest è una sfida ideata da Leonardo in collaborazione con sei atenei italiani per sviluppare l’Intelligenza Artificiale applicata alla guida autonoma dei sistemi. Le università partecipanti sono: Politecnico di Torino , Politecnico di Milano e Università di Bologna, Sant’Anna di Pisa, Roma Tor Vergata e Federico II di Napoli.

Il drone si sposta nel campo gara, contro il tempo, per raccogliere informazioni necessarie al team per compiere le missione in maniera corretta. In un secondo momento il drone si muove tra gli ostacoli per portare a termine gli obbiettivi.

Nel 2021 il team del Politecnico di Torino ha raggiunto la terza posizione. Abbiamo svolto un ottimo lavoro durante tutto l’anno, però bisognava perfezionare il progetto per provare a vincere la gara del 2022.

Gli algoritmi che venivano utilizzati era troppo pesanti e poco chiari. Infatti le principali modifiche che sono

state apportate riguardano maggiormente i codici più importanti. Ci siamo adoperati per rendere più efficiente i nostri algoritmi di path plannig e motion planning, e/o sostituirli con nuovi codici secondo noi migliori.

Uno degli obbiettivi della gara: atterraggio del drone su un aruco.

In più, noi membri meno esperti del team, abbiamo applicato le nostre conoscenze per creare strumenti e scrivere programmi che rendono più facile e più efficiente il lavoro dei più qualificati.

La mia ricerca si è focalizzata per lo più su queste criticità al fine di ottenere una performance migliore nella prossima gara**.**

****

Campo gara del Leonardo Drone Contest