

Rilevamento e Localizzazione di Persone tramite Kinect, GPS e YOLO, con Data Fusion tramite Filtro di Kalman

1. Introduzione

Il progetto si è concentrato sul migliorare e integrare il progetto precedente, aggiungendo ai sensori già presenti (quello RGB e di profondità del Kinect) il segnale gps per ottenere la posizione globale del sistema. I dati gps e di profondità vengono poi uniti ed elaborati attraverso un filtro di Kalman. L'obiettivo principale è quello di sviluppare un sistema di rilevamento dei volti e di misurazione della distanza in tempo reale che applica data fusion per determinare la possibile posizione globale dei volti individuati.

2. Librerie, algoritmi e framework utilizzati

- **YOLO11**: L'ultimo modello di deep learning di Ultralytics per il rilevamento rapido di oggetti, addestrato su un dataset personalizzato
- **MEGAFACE**: Dataset contenente le immagini di volti/persone per il training
- **PyTorch**: Framework per il training di YOLO
- **Freenect**: Libreria per interfacciarsi con il Kinect
- **OpenCV**: Utilizzato per il pre-processing delle immagini e per l'output
- **NumPy**: Utilizzato per la gestione di matrici e operazioni numeriche
- **Folium**: Libreria per la creazione di mappe interattive
- **Cdist**: Libreria per calcoli di distanze spaziali

3. Struttura del sistema

Il sistema è composto da diverse fasi e supporta sia l'elaborazione real-time che la post-elaborazione. Questa differenza dal primo progetto (che era invece interamente in real-time) è stata resa necessaria purtroppo dalla pesantezza dell'analisi con YOLO e l'utilizzo del filtro di Kalman, che portava, su macchine non performanti, l'analisi completa di un singolo frame a più di 1 secondo.

L'elaborazione **real-time** è composta dalle seguenti fasi:

- **Calibrazione**: Il processo di calibrazione delle telecamere del sensore Kinect è basato sull'acquisizione di immagini di una scacchiera posta in diverse posizioni davanti al dispositivo. La scacchiera viene utilizzata dal Kinect come riferimento fisso in quanto i punti d'angolo sono disposti in un pattern regolare. Durante la calibrazione, il sistema rileva gli angoli della scacchiera nelle immagini RGB e di depth, associando ogni punto rilevato alle coordinate corrispondenti nello spazio. Una volta raccolte le immagini, un algoritmo calcola le matrici intrinseche delle telecamere e i loro coefficienti di distorsione, necessari per correggere l'immagine e migliorare la precisione

della percezione spaziale. I parametri ottenuti vengono poi salvati e possono essere usati successivamente per migliorare l'allineamento tra immagini e dati di profondità.

- **Acquisizione delle Immagini RGB, della Mappa di Profondità e del segnale GPS:** In questa fase i sensori del Kinect catturano i dati RGB e di profondità, mentre il GPS i dati di posizione globali
- **Rilevamento degli Oggetti con YOLO:** Una volta acquisita l'immagine, questa viene preparata per essere processata dal modello YOLO. Nel codice viene caricato il modello "last.pt" e le classi degli oggetti "obj.names".
- **Stima della distanza e della posizione:** I dati della mappa di profondità vengono passati insieme alla latitudine, alla longitudine, e al centro x-y della bounding box dell'oggetto rilevato come vettore dello stato iniziale. Insieme a questo vettore viene passato al filtro anche la Matrice di transizione dello stato, la Matrice di osservazione, la Covarianza del rumore di processo, la Covarianza del rumore di misurazione, la Covarianza dell'errore iniziale, e il Vettore di controllo. Ad ogni frame vengono abbinate le rilevazioni agli oggetti esistenti utilizzando la distanza euclidea del centro delle bounding box. In questo modo il programma sa quali oggetti sono ancora in scena, se ce ne sono di nuovi e, di conseguenza, di quali fare le predizioni.
- **Visualizzazione e Feedback:** Gli oggetti rilevati vengono visualizzati sull'immagine RGB con un bounding box, il loro ID e la distanza stimata dal filtro. In aggiunta ad ogni frame viene generato e riscritto un file html contenente una mappa centrata nel segnale gps, con dei cerchi che indicano la possibile posizione degli oggetti. Si utilizzano dei cerchi perché, non avendo un sensore per sapere la direzione in cui è rivolto il sistema, gli oggetti si possono trovare su una circonferenza che ha per raggio la distanza, centrata nella posizione del Kinect.

La **post-elaborazione** invece verte sulle seguenti fasi:

- **Acquisizione delle immagini RGB, della Mappa di Profondità e del segnale GPS:** In questa fase il programma carica il video RGB, il video di profondità e il segnale GPS (contenuto in un file).

La **Stima della distanza e della Posizione**, e la **Visualizzazione e Feedback** sono le medesime dell'elaborazione real-time

4. Conclusioni

Questo progetto integra con successo il progetto precedente, aggiungendogli dati da un sensore GPS e un meccanismo di data fusion tradizionale.