2nd 3D Perception Project Report

Authors: Francesco De Patre, Davide Faroldi Lo Presti

Detection and Localization of People via Kinect, GPS and YOLO, with Data Fusion via Kalman filter

1. Introduction

The project focused on improving and integrating the previous project, adding to the existing sensors (the RGB and depth sensors of the Kinect) the GPS signal to obtain the global position of the system.
The GPS and depth data are then merged and processed through a Kalman filter.
The main goal is to develop a real-time face detection and distance measurement system that applies data fusion to determine the possible global position of faces.
identified.

2. Libraries, algorithms and frameworks used

- **MEGAFACE:** Dataset containing images of faces/people for training.
- **PyTorch:** Framework for training YOLO**.**
- **Freenect:** Library for interfacing with Kinect.
- **OpenCV:** Used for image pre-processing and output.
- **NumPy:** Used for managing matrices and numerical operations.
- **Folium:** Library for creating interactive maps
- **Cdist:** Library for calculating spatial distances
- **Yolo11:** 'Ultralytics' latest deep learning model for fast object detection, trained on a custom dataset.

3. System structure

The system consists of several stages and supports both real-time processing and post-processing.
This difference from the first project (which was entirely in real-time) was unfortunately made necessary by the heaviness of the analysis with YOLO and the use of the Kalman filter, which
brought, on non-performing machines, the complete analysis of a single frame to more than 1 second.

Real-time processing consists of the following phases:

- **Calibration:** The Kinect sensor camera calibration process is based on the acquisition of images of a chessboard placed in different positions in front of the device. The chessboard is used by the Kinect as a fixed reference because the corner points are arranged in a regular pattern. During calibration, the system detects the corners of the chessboard in the RGB and depth images, associating each detected point with the corresponding coordinates in space. Once the images are collected, an algorithm

calculates the intrinsic matrices of the cameras and their distortion coefficients, necessary to correct the image and improve the precision of spatial perception. The parameters obtained are then saved and can be used later to improve the alignment between images and depth data.

- **RGB Image, Depth Map and GPS Signal Acquisition:** At this stage, the Kinect sensors capture RGB and depth data, while the GPS captures global position data.
- **Object Detection with YOLO**: Once the image is acquired, it is prepared to be processed by the YOLO model. The “last.pt” model and the “obj.names” object classes are loaded into the code.
- **Distance and location estimation:** The depth map data is passed along with the latitude, longitude, and the x-y center of the bounding box of the detected object as the initial state vector. Along with this vector, the State Transition Matrix, the Observation Matrix, the Process Noise Covariance, the Measurement Noise Covariance, the Initial Error Covariance, and the Control Vector are also passed to the filter. At each frame, the detections are matched to the existing objects using the Euclidean distance of the bounding box centers. This way, the program knows which objects are still in the scene, if there are any new ones, and consequently, which ones to make predictions about.
- **Viewing and Feedback:** The detected objects are displayed on the RGB image with a bounding box, their ID and the distance estimated by the filter. In addition to each frame, an html file is generated and rewritten containing a map centered in the GPS signal, with circles indicating the possible position of the objects. Circles are used because, not having a sensor to know the direction in which the system is facing, the objects can be found on a circumference that has the distance as a radius, centered in the position of the Kinect.

Post-processing instead focuses on the following phases:

- Acquisition of RGB images, Depth Map and GPS signal: In this phase the program loads the RGB video (which is associated with a file containing the position of the bounding boxes previously processed with YOLO), the depth video and the GPS signal (contained in a file).

Distance and Position Estimation, and Visualization and Feedback are the same as real-time processing.

4. Conclusions

This project successfully integrates the previous project by adding data from a GPS sensor and a traditional data fusion mechanism.