Team Members:
Mohammed Munawwar - 120241642 (Driver)
Tathya Bhatt - (Navigator)

Midterm Proposal - Group 13 - Human Detection and Avoidance Module

Introduction

The objective of this project is to equip a mobile robot with the ability to navigate safely in dynamic environments. The module will detect and track human obstacles, providing real-time location data in the robot's reference frame. This functionality will allow the robot to dynamically adjust its path, preventing collisions and enhancing its safe operation in environments like public spaces or workplaces.

The Human Detection and Avoidance module will utilise a monocular video camera to detect human obstacles in real time and determine their positions relative to the robot. While the system will not account for occluded objects, it will focus on identifying and tracking visible humans. It will integrate with the robot's motion control system, supplying human location data to enable adaptive path planning and ensure collision-free navigation.

Tools Used

- ***Operating System:*** *Ubuntu 22.04*
- ***Programming Language:*** *C++17*, chosen for its performance and widespread use in robotics.
- ***Build System:*** *CMake* for easy project configuration and management.
- ***Libraries:***
  - ***OpenCV*** for image processing and human detection using deep learning-based object detection models.
  - ***Google Test*** for unit testing and TDD.
- ***Git*** is used for version control.

Development Methodology

1. ***Preprocessing:*** We will use OpenCV to preprocess the camera input (like image resizing, turning it into grey scale and getting the input image ready for processing it further).
2. ***Human Detection:*** In this step we will use a pre-trained deep learning model like YOLO to detect humans in the camera frame. In this step we plan to extract bounding boxes for detected humans. This approach allows for accurate localization of humans with a relatively high frame rate.
3. ***Motion Detection/Tracking:*** After processing the input image and detecting humans in the dynamic environment we will use a tracking algorithm like SORT(Simple Online and Realtime Tracking) or optical flow to predict human positions and detect motion of any other object in the dynamic environment.
4. ***Coordinate Transformation:*** Once we detect moving objects/human and track their motion we will convert the detected bounding box positions into the robot's reference frame.

5. ***Avoidance Strategy:*** The final step would be to integrate this with the robot's motion planning system to enable dynamic path adjustments based on the detected human positions.

Risks

- We might have limited accuracy of human detection in low-light or cluttered environments(with high population density of humans).
- There will be difficulty integrating with the existing motion planning system.
- The storage and processing resources required for real-time processing requirements may not be met due to limited size.

References

1. https://docs.opencv.org/3.4/d4/dee/tutorial_optical_flow.html
2. https://github.com/roboflow/notebooks/blob/main/notebooks/train-yolov8-object-detection-on-custom-dataset.ipynb