HDAL

(Human Detection and Localization)

Proposed By:
Vikram Setty (119696897)
Vinay Lanka (12041665)

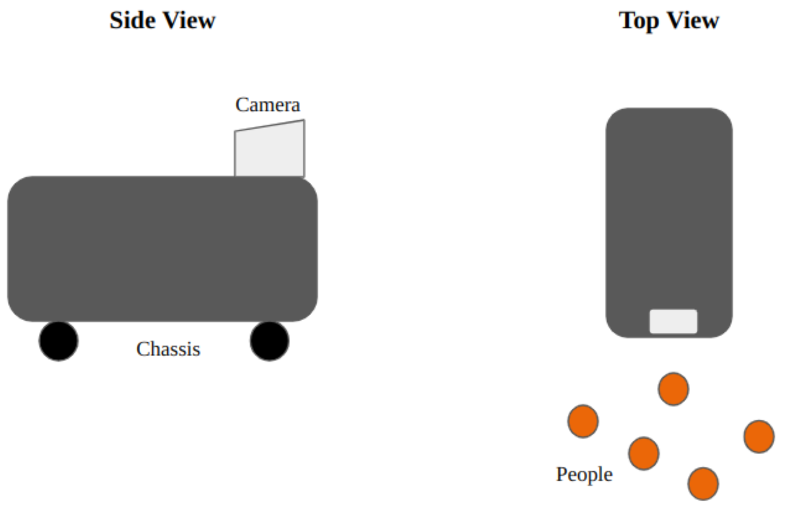
The Product

We propose building a perception system for Acme Robotics' delivery robot group. This system would be able to detect and localize humans moving in front of the robot as it moves on sidewalks delivering packages to the front doors of houses. The software would help the robot understand its surroundings and provide information to its path planners to choose an appropriate trajectory to move along.

Delivery robots, a subset of goods-to-person (G2P) robots are becoming an integral part of freight and cargo movement. Lying at the last leg of the supply chain, the delivery of packages to consumers is one of the most challenging tasks, and building perception stacks for robots in this domain is an integral step.

A robust perception stack is necessary for a robot to have a clear understanding of its surroundings. Camera footage, lidar data, and dextrous load information are the most widely used information sources for a robot for this purpose. With tremendous progress in computer vision in recent years, using pictorial information from a real-time camera serves Acme Robotics' needs perfectly.

The software would take real-time video feed from the robot's camera and simultaneously update the positions of the human obstacles in front of it relative to the camera mounted on it.



Software Design

The system would be built using two main interlinked components, the human detector, and the human tracker.

The human detector would use a pre-trained YOLO model[1] to identify all the humans in each chosen video frame along with the corresponding bounding boxes (position information in terms of picture coordinates).

The human tracker would then use principles of computer vision to identify the position of each identified human with respect to the robot camera's frame of reference. It is assumed that the camera would be placed at a known fixed height for this technique to work.[2]

Algorithms/Techniques to be Used

The human detector would use a pre-trained open-source YOLO convolutional neural network (CNN) model (in a JSON/H5 format) whereas the human tracker (the localization part) would make use of geometric computer vision concepts.

The YOLO model is most preferred for human detection as it is considered a benchmark and the tradeoff of slightly higher computation for high performance is worth it. For human tracking, however, classical computer vision methods are quite stable, and going out of the way to use heavier systems increases both complexity and computation.

Tools to be Used

The main tools that would be used to develop the software include the following:

- C++ (main programming language)
- ONNX (for using the pre-trained YOLO model): Apache 2 License
- OpenCV (for using geometric computer vision concepts): Apache 2 License
- CMake (for the build system)

Potential Bottlenecks

It is worth noting that maintaining a reasonable synchronization between the camera feed and human detector so that human presences and positions are updated without excessive time lag. Building an efficient system using good programming practices would be key to maintaining this.

Another aspect to keep in mind would be making sure the perception system is stable and does not fluctuate with small disturbances. For example, the fixed height of the robot's camera from the ground may be slightly inaccurate or can wobble and the system should still output a reasonably accurate position of the tracked human targets. Using verified computer vision algorithms and enabling high code coverage and testing would be key to ensuring this.

Final Deliverable

A software that on building and execution would take real-time camera footage and instantaneously mark and update human presence and position.
For the purpose of demonstration, the application would make use of the user's computer webcam for the video feed.
The final source code would have all unit tests (written in Google Test) passing with a code coverage greater than 90%. It would also be well documented (using Doxygen) and follow Google C++ style patterns.

Development Process

The software development cycle would make use of pair programming using the Agile Iterative Process (AIP). While Vinay Lanka would work on the human detection component of the perception system, Vikram Setty would work on the human tracking system. Further Vinay would set the unit tests while Vikram would run the tests to verify working. This order is specifically chosen to cater to the fact that the tracking system delivers the final output of the perception system. Both Vinay and Vikram would work equally on their respective documentation.

References

1. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 779–788). https://doi.org/10.1109/CVPR.2016.91
2. S. Kecman and D. Osmankovic, "Perspective Projection Approach to a Faster UAV Navigation and Obstacle Avoidance Learning," 2023 XXIX International Conference on Information, Communication and Automation Technologies (ICAT), Sarajevo, Bosnia and Herzegovina, 2023, pp. 1-6, doi: 10.1109/ICAT57854.2023.10171205.