Mobile Robot Navigation in Dynamic Environments with Human Detection and Categorization

Ishara Dilshan
Department of ENTC,
University of Moratuwa,
Colombo, Sri Lanka
Email: rajarathnagkmid.20@uom.lk

***Abstract*—Your abstract text goes here.**

I. Introduction

Your introduction section goes here.

II. Methodology

A. Human Detection and Categorization Module

We developed a custom dataset containing images of human agents in various environments to train a YOLOv10 small model [1] for detecting and categorizing human agents. This dataset included images derived from the study [2], which explored pedestrian categorization to enhance autonomous vehicle navigation. The dataset primarily focused on four human agent classes, categorized based on their physical appearances: child, normal adult, elder (non-disabled), and disabled.

To ensure the dataset was diverse and robust, we applied various data augmentation techniques such as rotation, contrast adjustment, brightness variation, noise injection, and flipping. These augmentations expanded the dataset coverage of potential light conditions and addressed challenges associated with the robot's actual camera input. The final dataset included 4,796 images, distributed across three subsets: 4,003 images in the training set, 397 images in the test set, and 196 images in the validation set.

To maintain the integrity of performance evaluations, we ensured that the test and validation sets were not augmented and did not overlap with the training set. Using this well-structured dataset, we trained the YOLOv10 small model, achieving notable performance metrics that demonstrated the model's ability to detect and categorize human agents effectively. During training, the model performed optimally at epoch 61, which corresponded to the point of optimal validation performance. Using early stopping, we terminated training at this point and saved the best-performing model weights.

The trained model detects human agents and their respective classes Figure 1, Figure 2 in each video frame, producing bounding box coordinates and class labels for each agent. To integrate this information into the robot's dynamic environment map, we transform the 2D coordinates from the video frames into a 3D reference frame. Then, we cross-reference these transformed coordinates with human agents identified through LiDAR data, enabling unique identification of each agent and their assigned class within the environment.

The acquired class data provides insights into each human agent's movement capabilities and related factors. We use this information for human path planning, which feeds into the robot path planning process. Additionally, the class data defines the human "footprint" in the 2D environmental map. Instead of representing human agents as dots, we use circles to represent their footprints. These footprints account not only for the physical area occupied by each human agent but also for the uncertainty in their movements, which varies depending on their class and other factors. For instance, children tend to exhibit more unpredictable movements than adults, resulting in larger footprints to reflect this higher uncertainty. By incorporating these class-dependent footprints into the 2D map, the robot can better predict and adapt to dynamic human behavior. This enables the robot to plan its path more effectively, navigating crowded environments while maintaining appropriate distances and ensuring socially acceptable interactions.

B. Power Management System and Self-Charging

We designed the robot power management system to monitor its battery levels during operation. A voltage sensor integrated in to the system alerts the robot when the battery level drops below a predefined threshold. Upon detecting low battery levels, the robot navigates autonomously to a charging dock station, using the navigation stack. The robot aligns with the docking station through an IR emitter-based guidance system to establish a secure connection for charging.

The dock station operates on a wall power supply (230V AC) and outputs a maximum of 14V, 30A DC. When in charging mode, the robot is powered directly by the dock station supply while simultaneously charging its battery. A battery management system (BMS) ensures overcharge and discharge protection during this process. A relay-based switching system facilitates transitions between charging and discharging states.

The power management system designed as a dual source system, ensuring the robot is always powered either by the battery or the dock station. The robot uses a 4-cell LiFePO4 battery with a capacity of 10,000mAh, providing up to 45 minutes of operation on a full charge. We selected LiFePO4 over lithium ion and lead acid batteries due to its superior

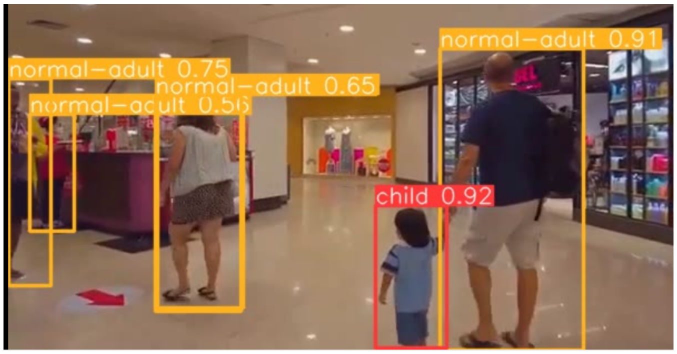

Fig. 1. Real-time detection of human agents using the trained YOLOv10-small model.

Fig. 2. Categorization of human agents in dynamic scenarios using YOLOv10-small model.

discharge rate, lower bulk, and compatibility with the robot's voltage and performance requirements. To meet the charging conditions of battery management system (14V to 18V), we employed a 600W step-up converter to boost the dock station's output voltage during charging. Additionally, a 600W step down converter regulates the battery output voltage during discharging, ensuring a steady 14V supply to meet the robot power requirements. This regulation is crucial since the battery voltage can fluctuate between 16.5V when fully charged and lower levels during discharge.

During charging, the BMS draws a maximum of 10A from the dock station while also powering the robot. Conversely, in discharging mode, the robot powered exclusively by the battery. The dual source circuit guarantees continuous operation, regardless of the robot state. A schematic diagram of the system is shown Figure 3.

III. Evaluation

A. Human Detection and Categorization Module

We validated the YOLOv10 small model using the test data set after completing its training on the training set. Validation demonstrated reasonable precision and recall metrics for each class, as presented in Table I.

The metrics in Table I show fair and reasonable precision, recall, and mean average precision (mAP) values for each class. During training, we analyzed the training losses and validation losses, including box regression loss, classification loss, and distribution focal loss (DFL), against training epochs to identify the optimal set of weights. Figure 4 illustrates these loss plots along with precision and recall values across epochs.

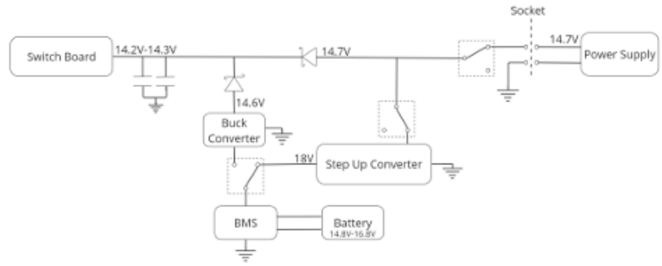

Fig. 3. Schematic diagram of the dual-source power management system.

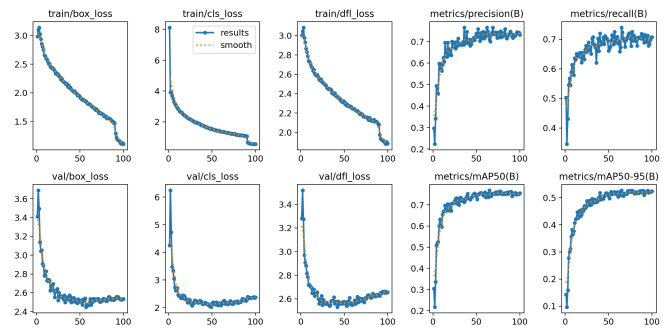

Fig. 4. Training losses, precision, and recall values across epochs.

We tested the trained model on video sources to evaluate its ability to detect and classify human agents. Figures 1 and 2 demonstrate the model performance in real-time environments. The results show that the model accurately detects human agents and classifies them into the predefined categories, even in dynamic scenarios.

To further evaluate performance, we compared the YOLOv10 small model against other models trained on the same dataset. Table II presents the observations, highlighting differences in average precision (AP) and latency across models.

Based on these results, we determined that the YOLOv10 Mega model, although achieving the highest precision, was unsuitable for real time applications due to its high latency. Conversely, the YOLOv10 small model offered an optimal balance between precision and latency, making it the most practical choice for accurately detecting and categorizing human agents in dynamic environments.

IV. Conclusion

Your conclusion text goes here.

Based on these results, we determined that the YOLOv10-Mega model, although achieving the highest precision, was unsuitable for real-time applications due to its high latency. Conversely, the YOLOv10-small model offered an optimal

TABLE I
CLASS METRICS

Class	Images	Instances	Precision	Recall	mAP@0.5
All	397	1296	0.71	0.74	0.76
Child	127	333	0.75	0.83	0.86
Normal-adult	207	631	0.66	0.62	0.66
Elder	102	163	0.65	0.64	0.67
Disabled	148	169	0.76	0.85	0.87

TABLE II
MODEL COMPARISON

Model	Parameters (M)	AP	Latency (ms)
YOLOv10 Nano	2.3	0.41	1.28
YOLOv10 Small	7.2	0.76	2.8
YOLOv10 Mega	15.4	0.78	4.9
YOLOv8 Small	11.3	0.64	7.4
RTMDet	8.3	0.42	6.6

balance between precision and latency, making it the most practical choice for accurately detecting and categorizing human agents in dynamic environments.

REFERENCES

[1] A. Wang, H. Chen, L. Liu *et al.*, "Yolov10: Real-time end-to-end object detection," *arXiv preprint arXiv:2405.14458*, 2024.
[2] D. Sharma, T. Hade, and Q. Tian, "Comparison of deep object detectors on a new vulnerable pedestrian dataset," *arXiv preprint arXiv:2212.06218*, 2022.