COMP9444 Project - Team New Bee

Ivan Luk z5463348 Zhidi Wei z5392805 Junhua Liu z5438356
Qirui Ye z5337136 Yi Jiang z5432865

I. Introduction

1. Background

The evolution of autonomous vehicle technology has introduced key challenges and opportunities in computer vision and machine learning, particularly in pedestrian detection. Critical for the safety and functionality of autonomous navigation systems, this project focuses on real-time pedestrian detection, prioritizing both speed and accuracy for rapid decision-making in various environments. Additionally, the project addresses the demand for cost-effective, efficient detection solutions suitable for portable devices, aligning with the goal of broadening the accessibility and adaptability of autonomous technologies.

2. Purpose of this project

This project has two main objectives: first, to conduct a thorough comparison of different pedestrian detection models to assess their accuracy and computational efficiency. This analysis is vital for identifying models fit for real-world, real-time applications. Second, the project aims to enhance the most promising model for pedestrian detection, focusing on improving its accuracy while optimizing for lower resource consumption, thereby enabling its use in devices with limited computing capabilities.

II. Literature Review

Pedestrian detection is critical for autonomous vehicle safety. This review summarizes key developments in this domain.

1. Early Methods

Initial pedestrian detection methods employed machine learning techniques like Support Vector Machines (SVM) with Histogram of Oriented Gradients (HOG), notably advanced by Dalal and Triggs (2005).

2. Deep Learning Approaches

The shift to deep learning, particularly Convolutional Neural Networks (CNNs), marked a significant improvement. R-CNN by Girshick et al. (2014) was a foundational development, applying deep learning to object detection.

3. Real-Time Detection

YOLO (You Only Look Once) by Redmon et al. (2016) and SSD (Single Shot MultiBox Detector) by Liu et al. (2016) introduced real-time detection capabilities, crucial for autonomous vehicles.

4. Advancements in YOLO

YOLOv4, introduced by Bochkovskiy et al. (2020), optimized the speed-accuracy trade-off, further enhancing real-time application efficiency. Developed by Ultralytics, YOLOv5 is an unofficial but popular iteration in the YOLO series. It is recognized for its ease of use and deployment, making it a favorable choice for many real-time object detection applications. YOLOv5 offers various improvements, such as enhanced performance and efficiency, over its predecessors. Suggested by Rath (2023), YOLOv8 is recognized for being faster than YOLOv5, which makes it more suitable for real-time object detection applications.

5. Segmentation Techniques

Mask R-CNN by He et al. (2017) offered fine-grained segmentation, improving detection in complex scenarios.

6. Dataset Limitations

A key challenge in pedestrian detection is the quality and nature of datasets used for training. The COCO dataset, despite its extensive use, presented limitations in pedestrian-specific scenarios. Recent works have focused on customizing existing datasets like COCO to better suit pedestrian detection needs, as seen in the works of Lin et al. (2014) and others.

III. Methods

1. Data Sources and Pre-processing

We selected the MS COCO 2017 dataset for its diversity and comprehensiveness, making it ideal for object detection tasks. Despite its versatility, the dataset's lack of a specific 'pedestrian' label presented a challenge, as the 'person' category was too broad, often including individuals in vehicles. To address this, we developed a two-pronged approach: using bounding box overlaps to filter out individuals inside vehicles and employing segmentation masks for improved accuracy in complex scenarios. This approach allowed us to redefine the 'person' category more accurately as 'pedestrians' and filter out non-pedestrians effectively.

2. Models

Our project examined four models: Faster R-CNN, SSD, YOLOv5, and YOLOv8, each offering unique strengths in pedestrian detection.

- Faster R-CNN is known for its accuracy and efficiency. It integrates a Region Proposal Network with Fast R-CNN into a single network, enhancing speed and accuracy.
- SSD balances speed and accuracy and is efficient in handling objects of various sizes, using multiple feature maps for detection, and NMS method is employed to suppress redundant detection boxes. It eliminates proposal generation and subsequent pixel and encapsulates all computation in a single network.
- YOLOv5 stands out for its exceptional speed and real-time detection capabilities. It's scalable and adaptable to various hardware requirements.
- YOLOv8 represents the latest in this series, promising even faster performance and greater accuracy, making it suitable for real-time applications.

3. Enhancement to YOLOv8

We chose to enhance YOLOv8 due to its speed and efficiency. By integrating GhostNet as the backbone network, we aimed to reduce the model's complexity while maintaining performance. This approach aligns with our goal of creating a model that's not only accurate but also agile and less resource intensive.

IV. Experimental Setup

1. Model Parameters:

Our project evaluated four models: Faster R-CNN, SSD, YOLOv5, and the enhanced YOLOv8. For each model, we maintained certain consistent parameters for a fair comparison, while also tuning specific parameters according to each model's characteristics:

- Learning Rate: Initially set to 0.001 for all models, adjusted based on validation performance.
- Optimizer: SGD with momentum (0.9) and weight decay (0.0005).
- Batch Size: Varied depending on the model and GPU capabilities, typically between 16 and 64.
- Epochs: Models were trained for up to 150 epochs, with early stopping based on validation loss to prevent overfitting.
- Input Size: All images resized to a standard input size of 640x640 pixels.

2. Evaluation Metrics:

The primary metrics used for model evaluation were:

- Mean Average Precision (mAP): A standard metric for object detection, measuring precision across different recall levels.
- Inference Speed: Measured in frames per second (FPS), indicating the model's capability for real-time detection.
- Intersection Over Union (IoU): Used to measure the accuracy of the predicted bounding boxes against ground truth.

3. Data Split:

The MS COCO 2017 dataset was split into training, validation, and testing sets as follows:

- Training Set: Comprised of 118,287 images, used for model training and parameter tuning.
- Validation Set: Consisted of 5,000 images, used for periodic evaluation during training and for hyperparameter adjustments.

4. Hardware and Software:

The experiments were conducted on NVIDIA T4

Tensor Core GPU, NVIDIA GeForce RTX 2060 GPU and NVIDIA GeForce RTX 3090 GPU. We utilized PyTorch for the implementation of Faster R-CNN, SSD, and YOLO models. The enhanced YOLOv8 model was developed based on the Ultralytics YOLOv8 framework.

This experimental setup was designed to ensure a comprehensive and fair evaluation of the models, focusing on both accuracy and efficiency in real-world pedestrian detection scenarios.

V. Results

Our experimental analysis yielded insightful findings regarding the performance of different models in pedestrian detection tasks. The key results are as follows:

1) Mean Average Precision (mAP): YOLOv5 showcased the highest mAP, followed closely by YOLOv8. The enhanced YOLOv8 model, despite the reduction in parameters, demonstrated a marginally lower mAP compared to the original YOLOv8. This slight reduction is attributed to the model's simplification in our enhancement efforts.
2) IoU Accuracy: All models exhibited strong IoU scores, with YOLOv5 leading. The precision of bounding box predictions was notably high in urban scenarios with clear pedestrian visibility.
3) Comparative Analysis: Compared to the state-of-the-art (SOTA) methods, our enhanced YOLOv8 model exhibited competitive performance, especially in speed, making it a strong candidate for real-time pedestrian detection systems.

VI. Conclusions

The key strengths of our proposed solution include high accuracy in pedestrian detection, remarkable speed suitable for real-time processing, and the adaptability of the enhanced YOLOv8 model to less powerful devices.

Key Strengths:

- Optimized Speed and Accuracy: The model successfully balanced high detection accuracy with increased processing speed.
- Reduced Model Complexity: The simplification of the YOLOv8 architecture resulted in a more efficient model, suitable for devices with varying computational power.

Weaknesses and Limitations:

- Small Object Detection: All models, especially Faster R-CNN, struggled with the detection of small or distant pedestrians.
- Complex Overlapping Scenarios: In highly congested pedestrian scenes, there was a noticeable decline in detection accuracy.
- Dataset Limitations: The lack of a specific 'pedestrian' label in the COCO dataset posed challenges in accurate labeling and training.

Recommendations for Future Work:

1) Enhanced Training on Diverse Datasets: Incorporating a wider range of pedestrian-focused datasets could improve the model's robustness.
2) Algorithmic Improvements for Small Object Detection: Further research is needed to enhance the model's capability in accurately detecting small or distant pedestrians.
3) Real-World Testing and Validation: Extensive testing in real-world scenarios, including varying weather and lighting conditions, could validate the model's practical applicability.

In summary, our enhanced YOLOv8 model stands out for its efficiency and speed, proving to be a viable option for real-time pedestrian detection in autonomous vehicles. Future enhancements and broader testing will be crucial in optimizing its performance and applicability in diverse operational environments.

VII. References

[1] Bochkovskiy, A., et al. (2020). YOLOv4: Optimal Speed and Accuracy of Object Detection. arXiv preprint arXiv:2004.10934.

[2] Chandan, G., Jain, A., & Jain, H. (2018). Real-Time Object Detection and Tracking Using Deep Learning and OpenCV. 2018 International Conference on Inventive Research in Computing Applications (ICIRCA), 1305–1308.

[3] Dalal, N., & Triggs, B. (2005). Histograms of Oriented Gradients for Human Detection. International Conference on Computer Vision & Pattern Recognition.

[4] Girshick, R., et al. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. CVPR.

[5] He, K., et al. (2017). Mask R-CNN. IEEE Transactions on Pattern Analysis and Machine Intelligence.

[6] Lin, T.-Y., et al. (2014). Microsoft COCO: Common Objects in Context. European Conference on Computer Vision (ECCV).

[7] Liu, W., et al. (2016). SSD: Single Shot MultiBox Detector. European Conference on Computer Vision (ECCV).

[8] Reddy, E. R. V., & Thale, S. (2021). Speed and Accuracy in Pedestrian Detection for Autonomous Cars Using YOLOv5. In 2021 IEEE Transportation Electrification Conference (ITEC-India), DOI: 10.1109/ITEC-India53713.2021.9932534.

[9] Redmon, J., et al. (2016). You Only Look Once: Unified, Real-Time Object Detection. CVPR.