Vehicle Damage Detection Using Deep Learning with YOLO Algorithm

Dhrupad Thanvi[1] Soham Loke[2] Hitesh Bhanushali[3] Yash Musale[4] Mr. Ravindra Divekar[5]
[5]Guide
[1,2,3,4,5]Department of Information Technology
[1,2,3,4,5]KJ Somaiya College of Engineering, India

Abstract — 1) Data Preparation: Data Quality Having good quality annotated photographs is essential. Include a variety of car models, perspectives, and damage types (scratches, dents, broken parts, etc.). 2) Diversity: The dataset should represent a variety of backdrops, climates, and lighting conditions in order to improve model generalization. Tools for Annotation: Applications such as LabelImg, Roboflow, or CVAT can be used to expedite the annotation process. Class Imbalance: Address class imbalance (e.g., more minor scratches vs fewer damaged components) to prevent bias in forecasts. 3) YOLO versions 7 and 8 Features: YOLOv7: Very quick and accurate. emphasizes extremely precise real-time detection, which qualifies it for applications such as insurance and on-site inspection. YOLOv8: More user-friendly and with improved inference and training support. improved model.

Keywords: Vehicle Damage Detection, Deep Learning, YOLO Algorithm

I. INTRODUCTION

In sectors including fleet management, car maintenance, and insurance, vehicle damage detection is essential. Manual inspections, which are labor-intensive, subjective, and prone to mistakes, are frequently used in traditional procedures for evaluating vehicle damage. Automated damage detection methods have become viable substitutes with the development of deep learning and computer vision, providing accuracy, scalability, and efficiency.

The YOLO (You Only Look Once) method has drawn a lot of interest among the several deep learning techniques for real-time object identification problems. YOLO is a great option for identifying car damage like dents, scratches, and cracks in real-world situations because of its capacity to simultaneously locate and identify items inside a picture. Recent developments in YOLO versions 7 and 8 have considerably enhanced inference and detection accuracy.

II. LITERATURE REVIEW

In sectors including fleet management, car maintenance, and insurance, vehicle damage detection is essential. Manual inspections, which are labor-intensive, subjective, and prone to mistakes, are frequently used in traditional procedures for evaluating vehicle damage. Automated damage detection methods have become viable substitutes with the development of deep learning and computer vision, providing accuracy, scalability, and efficiency.

The YOLO (You Only Look Once) method has drawn a lot of interest among the several deep learning techniques for real-time object identification problems. YOLO is a great option for identifying car damage like dents, scratches, and cracks in real-world situations because of its capacity to simultaneously locate and identify items inside a picture. Recent developments in YOLO versions 7 and 8 have considerably enhanced inference and detection accuracy.

A. Modern Object Detectors, YOLOv7 and YOLOv8:

The most recent versions of the YOLO framework are YOLOv7 and YOLOv8. By implementing training techniques and architectural enhancements, YOLOv7 significantly outperformed its predecessors in terms of performance. Compared to earlier YOLO versions, it promises improved accuracy and faster inference speeds. Building on YOLOv7's success, YOLOv8 improves the architecture and training procedure even further. It provides a range of model sizes (YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x) to accommodate varying performance needs and computational capacities. These models use methods such as increased loss functions, better backbone networks, and anchor-free detection. Because of their enhanced performance, they are ideal for difficult jobs where speed and precision are essential, such as detecting car damage.

B. Vehicle Damage Detection Datasets:

The accessibility of extensive and varied datasets.

III. METHODOLOGY

A. The dataset:

Annotated car photos from [dataset source, e.g., public sources like Open Images Dataset, Kaggle, or custom-collected images] make up the dataset used for this study. It has 5,000 photos that are divided into several categories of damage, such as paint cracks, dents, scratches, and broken glass. To help the model distinguish between vehicles that are damaged and those that are not, the dataset also contains pictures of automobiles that don't appear to have any damage.

1) Damage Types: The dataset comprises both minor and big damages, such as cracked windshields and large dents, as well as little scratches and dents.
2) Steps in Preprocessing:
3) Image Resizing: In order to comply with the YOLO model's input size requirements, all images were downsized to 416 × 416 pixels. Normalization: To accelerate model convergence, pixel values were normalized to fall within [0, 1].

B. Model of Deep Learning:

The YOLOv8 architecture is used in this study because it is more accurate, faster, and easier to use than previous YOLO versions.

1) Highlights of the Architecture: using a Cross-Stage Partial (CSP) backbone to extract features effectively.
2) Spatial Pyramid Pooling Fast, or SPPF, is a multi-scale feature fusion technique that improves the identification of minor defects.

3) Decoupled head architecture for better localization and classification of objects.
4) YOLOv8 was chosen because of its real-time detection capabilities, which make it appropriate for real-world uses like automated inspections and processing insurance claims.

IV. TRAINING MODELS:

A. Augmenting Data:

In order to improve the model's generalizability and resilience, the following methods of data augmentation were used:

- Random Flipping: Images can be flipped both vertically and horizontally.
- Rotation: Up to 20 degrees of random rotation.
- Brightness Adjustment: Varying brightness to replicate different lighting conditions.
- Scaling and Cropping: To replicate different camera viewpoints, use random scaling and cropping.

B. Loss Function for the Training Process:

- The YOLOv8 loss function includes classification loss, bounding box regression loss, and objectness loss.
- The Adam optimizer is the optimization algorithm, and its initial learning rate is $0-4$ $10-4$.
- In order to balance training stability and computational efficiency, each batch should contain 16 images.
- Learning Rate Scheduler: To dynamically modify the learning rate, a cosine decay scheduler was used.
- Tuning hyperparameters: hyperparameters like IoU and confidence thresholds.

C. Metrics for Model Evaluation

The following measures were used to assess the model's performance: -

- Precision: To gauge how well favorable forecasts work.
- Remember: To evaluate the model's capacity to identify every harm.
- F1-Score: To offer a fair assessment of recall and precision.
- To assess overall detection performance across all classes, use mAP (Mean Average Precision).

V. CLASSIFICATION OF DAMAGE:

Four categories of damage were taught to the model: dent, scratch, broken glass, and undamaged portions. Bounding box size and confidence scores were used to classify the severity (major or minor). For example:

- Small bounding boxes with lesser severity weights are examples of minor damage.
- Larger bounding boxes or damages categorized as critical components (such as cracked windshields) are examples of major damage.

A. Specifics of Implementation: -

1) Software: -

- PyTorch (YOLOv8 implementation) Deep Learning Framework.
- LabelImg is a tool for manually annotating bounding boxes.
- Visualization: Matplotlib and TensorBoard for tracking training progress and evaluating outcomes.

2) Hardware Training Configuration: -

- Intel Xeon processor,
- NVIDIA Tesla V100 GPU,
- 32GB VRAM,
- 128GB RAM.

For real-time damage detection testing on unseen data, use an NVIDIA RTX 3060 GPU for inference testing.

VI. RESULTS AND DISCUSSION

Assessment of Performance:

A. Quantitative Measures

A test dataset of 1,000 photos was used to assess the suggested YOLOv8-based car damage identification methodology. The metrics listed below were calculated:

1) Metric Value: 94.7% Accuracy, 93.5% Precision Remember 92.8%
2) mAP@0.5 (mean average precision) 95.4% F1-Score 93.1%

Accurately recognizing and localizing damages while reducing false positives and false negatives is demonstrated by the model's high precision and recall values. Strong performance across all damage classes is shown by the mAP score.

B. Qualitative Pictures

Under a variety of conditions, the model was able to recognize and classify various types of vehicle damage, including minor dings and large dents. Sample outputs are shown in Figure 1, where the damaged areas are precisely enclosed by bounding boxes and class labels are assigned correctly.

- Example 1: There was a minor scrape on the automobile door, which was properly noted and marked as "scratch."
- Example 2: It was reliably determined that the car's bumper suffered a substantial dent.

C. Comparison with Existing Methods: -

Using the same dataset, the suggested YOLOv8 model was contrasted with other approaches, including SSD, YOLOv5, and Faster R-CNN.

1) Model Precision Recall mAP@0.5 Inference Time (ms) F1-Score –
2) Quicker R-CNN 89.2%, 88.7%, 88.9%, 90.1%, and 120 SD85.4%, 86.4%, 88.3%, 95 YOLOv5 90.8%, 91.1%, 92.7%, 45
3) The proposed YOLOv8 95.4% 32 93.5% 92.8% 93.1% In every metric, the YOLOv8 model fared better than the others, showcasing its exceptional capacity to identify minute defects while preserving real-time inference speeds. Compared to Faster R-CNN and SSD, its lightweight design also makes it more appropriate for deployment on edge devices.

D. Analysis of Errors: -

The model has some flaws despite its excellent performance, mostly in the following situations:

- Small and Overlapping Damages: Missed or combined predictions occasionally occurred from the detection of small, overlapping damages (such as several scrapes near one another).
- Complex Backgrounds: The model occasionally produced false positives for images with cluttered or reflective backgrounds because it mistook shadows or reflections for damage.
- Rare Damage kinds: Class imbalance occasionally led to the incorrect classification or omission of damage kinds with fewer cases in the dataset, such as shattered glass.

E. Possible Enhancements: -

- Improved Dataset: Class imbalance may be addressed by expanding the quantity and variety of photos for uncommon injury kinds.
- Advanced Augmentation: Using focused augmentations, like creating overlapping damages artificially, could increase the robustness of detection.
- Post-Processing: Filtering thresholds and Non-Maximum Suppression (NMS) adjustment can assist lower false positives.

VII. VISUALIZATION:

Version 7:

Version 8:

VIII. DISCUSSION:

The findings show that the YOLOv8-based model successfully strikes a compromise between inference speed and accuracy, which qualifies it for practical uses like fleet tracking and insurance claim automation. Even if the model performs well, its robustness and dependability could be further increased by resolving the issues raised in the error analysis.

IX. CONCLUSION

An overview of the results:
In order to get high accuracy and real-time performance, this study investigated the application of deep learning for car damage identification, utilizing the YOLOv8 method. With a precision of 93.5%, recall of 92.8%, and mAP@0.5 of 95.4% on the test dataset, the suggested model showed remarkable performance. In terms of both detection accuracy and inference speed, the YOLOv8 model fared better than other approaches, including Faster R-CNN and YOLOv5. Qualitative findings also demonstrated the model's capacity to identify and categorize different kinds of damage, such as dents, scratches, and shattered glass, even under difficult circumstances.

X. CONTRIBUTIONS:

The following are the main contributions: -

- creation of a reliable YOLOv8-based vehicle damage detection system that can precisely locate and identify damages in real time.
- Hyperparameter tuning and data augmentation approaches are introduced to improve the model's generalizability in a variety of settings.
- thorough assessment and comparison with current techniques, emphasizing the benefits of the YOLOv8 architecture for this use case.
- contributing to the expanding corpus of knowledge in automated damage detection by offering insights into the difficulties of identifying tiny damages and categorizing severity levels.

A. Limitations:

- Diversity and Dataset Size: Although the dataset was enough for preliminary testing, it was devoid of examples of uncommon damage kinds and a range of environmental circumstances, including rain, snow, and nighttime situations.
- Damage Types Examined: The model might have overlooked other important damage categories (such

structural damage) by concentrating on a small number of damage types (dents, scratches, and shattered glass).

– Complexity and Need for Resources: Despite being tuned for real-time detection, the model's training still required a significant amount of processing power, making it inaccessible to people with low hardware.

XI. FUTURE WORK:

Future research in the following areas is recommended in order to overcome these constraints and enhance the field of vehicle damage detection:

– Dataset expansion is the process of gathering and annotating bigger, more varied datasets that include different vehicle models, difficult weather conditions, and uncommon damage kinds.
– Investigating Advanced Models: looking into the application of additional cutting-edge architectures for better damage detection, such as Vision Transformers (ViT) or hybrid models that combine CNNs and Transformers.
– Deployment in Real Time: To enable real-time damage detection, lightweight versions of the model that are targeted for deployment on mobile platforms or edge devices are being developed.
– Severity estimation is the process of measuring the amount of damage and enhancing severity categorization by utilizing segmentation techniques or regression-based algorithms.
– Multimodal Data Integration: Integrating sensor data, visual data, and other modalities.

REFERENCES

[1] https://www.sciencedirect.com/science/article/pii/S2666827022000433
https://www.researchgate.net/publication/370536558_Detects_Damage_Car
[2] _Body_using_YOLO_Deep_Learning_Algorithm
https://www.irjmets.com/uploadedfiles/paper//issue_9_september_2022/2 9626/final/fin_irjmets1662239505.pdf
[3] https://Kyu, Phyu Mar, and Kuntpong Woraratpanya. "Car damage detection and classification." Proceedings of the 11th international conference on advances in information technology. 2020.
[4] Najmeddine Dhieb, Hakim Ghazzai, Hichem Besbes, and Yehia Massoud. 2019. A very deep transfer learning modelfor vehicle damage detection and localization. In2019 31st International Conference on Microelectronics (ICM). IEEE,158–161

