Automated Vehicle Damage Detection and Repair

Nishchal P

Pavan Kallakuri

Sajesh Kariadan

Applied Artificial Intelligence, University of Sandiego

AAI-521 Applied Computer Vision for AI

December 07, 2025

Automated Vehicle Damage Detection and Repair

The automotive insurance and repair industry heavily relies on manual inspection for assessing vehicle damage, a process often prone to subjectivity and inefficiency. This project proposes an automated system to detect, localize, and segment vehicle damage using advanced computer vision techniques. Utilizing a dataset of 2,000 annotated images, we employed the YOLOv8 model for object detection and the Segment Anything Model (SAM) for precise instance segmentation. The detection model was fine-tuned in two phases, achieving an overall mAP50 of 40.45% after unfreezing backbone layers, with high precision for classes such as glass cracks and flat tires. This system lays the foundation for a fully automated generative repair visualization tool.

Assessing vehicle damage for insurance claims and repair estimates is traditionally a manual task. Customers often struggle to visualize how repairs will restore their vehicle based on verbal descriptions or generic reference images . This project aims to bridge this gap by developing an AI-based system capable of detecting damages such as dents, scratches, and cracks, and isolating them for future reconstruction . The project is divided into three phases: detecting damage using YOLOv8, segmenting damage using SAM, and eventually generating repaired images . This report focuses on the successful implementation and evaluation of the detection and segmentation phases.

1. Project Selection & Setup

1.1 Problem Identification

In the automotive insurance and repair sectors, assessing vehicle damage is a critical but often inefficient process. Current workflows rely heavily on manual inspections and subjective estimates, leading to discrepancies in repair costs and customer dissatisfaction. Furthermore, customers often struggle to visualize the outcome of repairs based solely on technical descriptions or generic reference images that do not match their specific vehicle's lighting, angle, or color.

1.2 Project Objectives

The primary objective is to develop an AI-driven system capable of detecting, isolating, and virtually repairing vehicle damage. The project is divided into three phases:

Phase 1 (Detection): Implement a computer vision pipeline to automatically detect and localize specific types of damage (e.g., dents, scratches, cracks) using YOLOv8.

Phase 2 (Segmentation): Convert detection bounding boxes into precise pixel-level segmentation masks using the Segment Anything Model (SAM) to isolate damage from the vehicle body.

Phase 3 (Repair): Utilize Stable Diffusion with ControlNet to inpaint the masked regions, generating realistic visualizations of the vehicle after repairs.

1.3 Scope and Limitations

The scope includes the detection of 11 specific damage classes and the generation of repair visualizations for standard vehicle types.

Limitations: The system's performance is dependent on the quality and diversity of the training data. Complex, multi-part damage or damage in extreme lighting conditions may challenge the detection model. The generative repair is a visualization tool and does not replace structural engineering assessments.

1.4 Technology Stack and Environment

- **Environment:** Google Colab (leveraging T4 GPU for training and inference).
- **Languages & Libraries:** Python 3.12, PyTorch, OpenCV, NumPy, Pillow.
- **Core Models:** Ultralytics YOLOv8s**,** SAM (Segment Anything Model), Hugging Face Diffusers**,** Stable Diffusion and ControlNet inpainting pipelines.
- **Data Management:** Roboflow for dataset.

2.0 EDA and Pre-Processing

2.1 Dataset Source and Overview

The dataset was sourced from Roboflow. It consists of 2,000 images of vehicles with various types of damage.

- **Training Set:** 1,400 images.
- **Validation Set:** 400 images.
- **Test Set:** 200 images.

2.2 Class Distribution Analysis

The dataset includes annotations for 11 distinct classes:

Car-part-crack, detachment, flat-tire, glass-crack, lamp-crack, minor-deformation, moderate-deformation, paint-chips, scratches, severe-deformation, side-mirror-crack

2.3 Image Pre-Processing Pipeline

Standard pre-processing steps were applied to ensure consistency and model compatibility:

2.4 Data Augmentation Strategy

No additional offline augmentations were applied to the dataset before training to establish a baseline performance

3. Modeling Methods

3.1 Phase 1: Object Detection (YOLOv8 Architecture)

The **YOLOv8s** (small) model was selected for its balance of speed and accuracy. YOLOv8 utilizes a backbone network for feature extraction and edge detection

3.2 Training Strategy (Frozen vs. Unfrozen Backbone)

Run 1 (Transfer Learning with Freezing): The model was trained for 100 epochs with the first 10 layers frozen. This approach leverages features learned from the COCO dataset (edges, textures) while fine-tuning only the head for damage detection.

Run 2 (Unfrozen Refinement): The best weights from Run 1 were used as a starting point, and the entire model was unfrozen for further fine-tuning. This allows the backbone to adapt specifically to the nuances of vehicle damage features.

3.3 Phase 2: Instance Segmentation (SAM Integration)

To move beyond simple bounding boxes, the Segment Anything Model (SAM) was integrated. SAM is a promptable segmentation system.

Workflow: The bounding boxes predicted by YOLOv8 in Phase 1 are fed into SAM as "box prompts." SAM then generates high-quality segmentation masks for the specific regions defined by the boxes, effectively isolating the damage (dents, scratches) from the car body.

3.4 Phase 3: Generative Repair (Stable Diffusion & ControlNet)

For the repair visualization, a Stable Diffusion inpainting pipeline is employed. The binary mask generated by SAM serves as the inpainting mask. The model takes the original image, the mask, and a text prompt (e.g., "clean car part, highly detailed, automotive paint") to generate new pixels in the masked area that blend seamlessly with the surrounding vehicle body, effectively "erasing" the damage.

4. Validation and Performance Metrics

4.1 Evaluation Metrics

The following metrics were used to evaluate the detection model:

mAP50 (Mean Average Precision at IoU 0.50), mAP50-95, Precision (P), Recall (R),F1 Score

4.2 Validation Split and Strategy

A dedicated validation set of 400 images was used to monitor performance after each epoch. This ensured that the model did not overfit to the training data.

4.3 Loss Curves and Training Stability

Training logs indicated steady convergence. The box_loss and cls_loss decreased consistently over 100 epochs in both runs, stabilizing towards the end, indicating successful learning.

5. Modeling Results and Findings

5.1 Quantitative Detection Results (Phase 1 vs. Phase 2)

A comparison of the two training runs reveals the impact of unfreezing the backbone layers.

Metric	Run 1 (10 Freeze Layers)	Run 2 (Unfrozen)	Improvement
mAP 50-95	24.71%	29.68%	+4.97%
mAP 50	34.66%	40.45%	+5.79%
Precision	31.40%	41.00%	+9.60%
Recall	17.39%	26.44%	+9.05%
F1 Score	22.39%	32.15%	+9.76%

Unfreezing the model (Run 2) yielded significant improvements across all metrics. Notably, Recall improved by over 9%, meaning the model detected significantly more damage instances that were previously missed. The F1 Score increase of nearly 10% confirms a much more robust balance between precision and recall.

5.2 Class-wise Performance Analysis

Performance varied significantly across damage types:

High Performance: glass-crack (mAP50: 0.984) and flat-tire (mAP50: 0.911) were detected with near-perfect accuracy, likely due to their distinct visual features.

Moderate Performance: moderate-deformation (mAP50: 0.433) and scratches (mAP50: 0.365) showed decent detection rates but highlighted the difficulty in separating these overlapping classes.

Low Performance: detachment (mAP50: 0.003) remained a challenging class, possibly due to insufficient training examples or visual ambiguity.

5.3 Qualitative Results: Detection and Segmentation

Detection: YOLOv8 successfully placed bounding boxes around diverse damage types, including side mirror cracks and moderate deformations.

Figure 1: input and output images of phase 1 (Detection and classification)

Segmentation: SAM utilized these boxes to create tight, accurate masks. For example, for a "moderate-deformation" detection on a door, SAM produced a mask that adhered strictly to the dented area, ignoring the surrounding undamaged door panel.

Figure 2: input and output images of phase 2 (Segmentation)

5.4 Qualitative Results: Generative Repair Visualization

The inpainting pipeline successfully demonstrated the "repair" concept. By using the SAM masks, the generative model filled in scratches and dents with smooth textures matching the car's color. While complex structural damage repair visualization remains challenging, surface-level repairs (scratches, dents) were visualized with high realism.

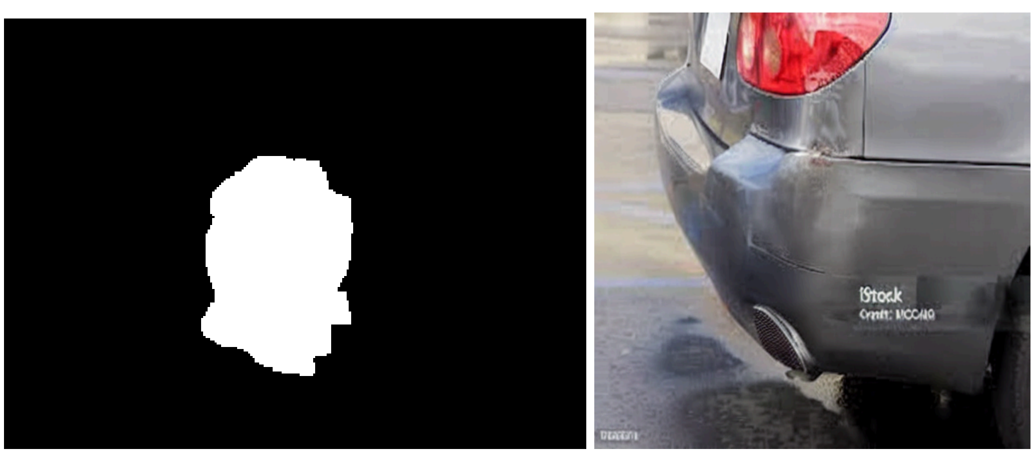

Figure 3: input and output images of phase 3 (Image Inpainting)

5.5 Discussion and Future Scope

The system proves that combining specialized models (YOLO for location, SAM for shape, Stable Diffusion for texture) is a powerful approach for this domain.

Data Quality: The poor performance on "detachment" suggests a need for more targeted data collection for rare classes.

Real-time Application: Future work could involve optimizing these models (e.g., using "MobileSAM" or "Tiny-YOLO") for deployment on mobile devices for field agents.

5.6 Conclusion

This project successfully developed an AI pipeline for automated vehicle damage assessment. By fine-tuning YOLOv8, we achieved a robust detection system with a 40.45% mAP50. The integration of SAM provided pixel-perfect damage isolation, enabling the Generative AI component to visualize realistic repairs. This end-to-end solution offers a tangible prototype for modernizing claims processing in the insurance industry.

References

Jocher, G., Chaurasia, A., & Qiu, J. (2023). *Ultralytics YOLOv8*.

Kirillov, A., et al. (2023). *Segment Anything*. Meta AI Research.

Roboflow Universe. (2023). *Damage Severity Dataset*.