“Automated Vehicle Damage Analysis and Cost Prediction with Mask R-CNN for Insurance and Repair Optimization”

AUTOMATING CAR INSURANCE CLAIMS

Arpit Nandkishor Chauhan
SVKM Mithibai College
Mumbai, India
arpitchauhan2610@gmail.com

Amol Joglekar
SVKM Mithibai College
Mumbai, India
amol.jogalekar@mithibai.ac.in

***Abstract*—One part of the automotive industry that could be poised for a major disruption is the automatic damage detection on vehicles, and this is most likely to occur through image data for insurance companies, car rental companies, and repair shops. In this work, we use one of the most powerful neural network architectures in its class: Mask R-CNN.**

We trained this model on a COCO-Formatted dataset containing images,where each image contains annotations of damaged areas. Using a standard 2D image capture, the system detects and assesses damage to a vehicle and helps drivers speed up the insurance claims process without the need for a physical inspection.

This study uses cutting-edge computer vision techniques to identify damage and categorize its severity. To determine the degree of damage, the algorithm calculates the Intersection over Union (IoU) score between the ground truth and anticipated bounding boxes. Based on the extent of the damage found, these scores are subsequently utilized in a linear regression model to calculate repair costs.

This study’s main goal is to transform the auto insurance claim procedure by making it quicker, more economical, and more successful. It also enhances consumer satisfaction, aids in the detection of fraud, and shows how AI-powered damage detection might change the direction of the vehicle insurance sector.

I. Problem Statement

Both policy holders and insurers may find the traditional method of filing auto insurance claims to be extremely time-consuming and unpleasant. Before repair costs can be calculated, they need extensive documentation, drawn-out inspection procedures, and manual assessments. In addition to increasing operational expenses and delaying claim settlement, these inefficiencies generate difficulty for car owners who must wait for repairs to be completed.

II. Introduction

Filing car insurance claims has always been a headache-endless paperwork, lengthy inspections, and long wait times. These problems not only frustrate policy holders, but also pose problems for insurance companies, causing unnecessary stress and delays for everyone. We therefore developed a system that would automate and streamline the whole process, making it faster, easier, and more efficient for both the drivers and the insurers In this regard, the system relies on the most advanced technology in image segmentation and machine learning. Through CNNs, the system will be able to examine vehicle photographs, identify certain components, and precisely identify areas of damage. Automation of the process accelerates the assessment of damage, often obviating the need for an in-person inspection for minor to moderate damage. The end result is a faster and more painless claims experience for drivers and insurers alike.

Our system will use machine learning to analyze the past claims made and estimate repair costs based on the damage incurred. This automation hastens the settlements of claims in a more accurate and satisfying way for customers.

Drivers get to enjoy faster repairs, simpler claims, and quicker payout, which eliminates stress. Insurance companies save money, detect fraud better, and improve customer experience. Even repair shops benefit through streamlined workflows and faster service.

This study demonstrates how technology can revolutionize car insurance claims and make it faster, easier and less stressful for both parties. This is a step toward a smarter, more efficient future for insurance.

III. Literature Review

The adoption of the COCO dataset and the customization of models in Detectron2 have greatly improved instance segmentation over the past few years. However, as real-world applications become more demanding in terms of accuracy, adaptability, and efficiency, segmentation models are continually being refined to improve performance and generalization across diverse datasets.

The evolution has been driven by key studies that paved the way for current object detection and segmentation approaches. These improvements have looked at critical problems, including enhancing the performance of the model, dealing with challenging real-world scenarios, and improving computational efficiency.

In this chapter, major contributions in the field are discussed, along with the development of instance segmentation techniques and datasets that have informed this study. Furthermore, this chapter describes how these advancements have been incorporated into the proposed methodology.

A. *Initial Foundations in Instance Segmentation*

The foundation of the modern object detection was established by R-CNN (CNNs with regions) Girshick et al. (2014) [1], that enhanced the detection accuracy by combining CNNs with region proposals. This approach solved the bottlenecks of the previous object detection methods by incorporating a high-capacity CNN for object localization and segmentation. In addition to that, R-CNN also revealed the potential of supervised pre-training on related tasks and then fine-tuning the network for the target task. Thus, the success of R-CNN was achieved by applying CNNs to bottom-up region proposals and gained about 30% improvement in mAP on the PASCAL VOC 2012 dataset as well as setting a new standard for object detection.

B. *How the COCO Dataset Became a Benchmark*

These have included the PASCAL VOC and ImageNet dataset, which helped improve classification and detection tasks in the early object detection datasets. However, they did not capture the complexity of real world scenes. The Microsoft COCO dataset [2] filled that gap by adding per-instance level detailed segmentation, more contextual relationships between objects and images, and dense annotations of the images. COCO has more than 328,000 images and over 2.5 million instance-level annotations, making it the go-to dataset for computer vision model evaluation. Mask R-CNN and Detectron2, powerful instance segmentation models, were pushed by exhaustively labelled data and diverse real world images.

C. *Advancements with Detectron and Detectron2*

Detectron and its upgraded version, Detectron2, present highly improved object recognition and instance segmentation through deep learning. These frameworks have been developed by Facebook AI Research (FAIR) to streamline the use of powerful models such as Faster R-CNN and Mask R-CNN, making them more user-friendly and adaptable for researchers and developers.

Detectron2 [3] was built based on the emerging need for a high-performance, fast, and flexible framework. The updated version built on PyTorch included many significant enhancements such as training pipelines, sup- port for distributed training, and interaction with the most recent models such as Cascade R-CNN and DensePose. With such enhancements, Detectron2 is a popular tool both for indus- trial and academic use in building extremely accurate and efficient models for segmentation

D. *Image-Based Vehicle Detection Using Various Features*

Real-Time Vehicle Detection is one the most difficult task in recent times, especially under different different conditions, such as different lighting, reflections and background clutter. Researchers into developing more innovation because there is growing need to have accurate detection and analysis of vehicles automated in computer vision techniques. A recent study [4] for auto detection applies image-based detection using thresholding, bounding boxes, and heatmaps as features.These are two methods, one of them is Haar Cascade Classifier, another one is Lens-based Vehicle Classifier (LVC).

The first couple of images of a car taken by a DSLR camera are saved for future analysis as a part of detection procedure. An area of interest is isolated with the help of processing these photographs with thresholding, merging of boxes, bounding boxes, and heat maps, and then finally, a model of bounding box prediction assesses the selected area. It has reduced the false negatives through applying an 85% probability threshold, enhancing the accuracy of the images. The image processing is carried out using a regressor model combining pixel data for a maximum of 85% accuracy in detection.

The method promotes the improvement of real-time vehicle detection while it increases the safer control of traffic and effective road monitoring.

E. *Assessing the repair expenses for vehicles damaged in road incidents poses a significant challenge, particularly for insurance firms and accident evaluators*

Estimating the repair costs for vehicles damaged in road accidents is a challenging and a complex task, especially for insurance companies and accident analysis. Traditional methods are often insufficient in giving an accurate estimation because they lack accurate data that correlates the physical deformation with the repair cost. Recent research [5] attempts to address this challenge by suggesting a methodology based on deformation measurements and crash reconstruction variables such as energy absorption and velocity changes ($DeltaV$)

This approach uses a detailed data collection from real world accidents to establish how the amount of damage results in repair cost. Using a tool such as the NASS CDS database for crash data, and the Audaplus © estimation system in evaluating the repair costs, the paper develops a simplified process for matching damage measurements to repair costs. The methodology not only improves the accuracy in the estimation of repair costs but also provides excellent insights for improving pricing and accident analysis in insurance.

F. *A Refined Mask R-CNN Method for Vehicle Damage Detection and Segmentation*

Detection of vehicle damage has become an essential process in accident claims assessment, optimized repair, and processing of insurance claims. Traditional object detection models fail to precisely segment such objects, particularly for complex real-world damage scenarios. To solve these problems, recent research by [6] introduced an improved Mask R-CNN approach tailored to vehicle damage detection.

It improved segmentation precision by tuning up the backbone network of the Residual Network and adding a feature pyramid to be able to use better feature extraction, and refining RPN in a way to adapt anchor boxes for all sizes of an object while making fine tuning adjustments to weights on loss function, further helping it to provide accurate detection precision. Through various optimizations and improvements, Mask R-CNN demonstrates notably increased precision and superior segmentation accuracy when compared to baseline models

G. Summary

If put together, these development aspects illustrate the evolution from initial object detection methods with R-CNN [1] and standard benchmarking through COCO [2] to even the most optimized deep learning frameworks like Detectron2 [3]. Such advances have enabled more applications through image-based vehicle detection [4], the estimation of repair cost through deformation analysis [5], and even enhanced segmentation algorithms based on improved models of Mask R-CNN [6].

Research has gradually optimised object detection techniques, dataset implementation, and damage analysis, allowing for the development of AI-based systems that can be accurate, fast, and scalable. These technological advancements pave the way for our proposed framework, which utilizes deep learning, feature extraction, and segmentation techniques to enhance automation and efficiency in vehicle insurance claims processing.

IV. Methodology

To reach our goal, we rely on the latest and most advanced state-of-art deep learning framework for object detection and segmentation, that is, Detectron2 **Detectron2** [3], to create an automatic vehicle damage detection system. It's from the Facebook AI Research group, a modular and fast implementation, which permits you to use a variety of different object detection models (including Mask R-CNN).

To achieve accurate location and categorization of the vehicle Damages, we use Mask R-CNN [1] in our approach. Mask R-CNN is an extension of Faster R-CNN. This addition includes another branch which generates the segmentation masks for objects that are detected, and hence increases the performance hugely for detecting things like vehicle damage

Our model is trained on the Car Damage Detection (CarDD) dataset [7],which includes an extensive number of annotated damage vehi-cle images. The total large-scale number of images are high-resolution labelled with systematic categorization into different categories such as dents, scratches, cracks, shattered glass, broken lamps, and flat tires.

The augmentation techniques in our pipeline pre- processing are used to increase the generalization of the model. Hyperparameters such as learning rate, batch size and iterations to further fine-tune the model are defined following the changes on the Detectron2 default training parameters; To evaluate model performance, we utilize Intersection over Union (IoU) [8] and Mean Average Precision (mAP), two widely recognized metrics in object detection and segmentation. These measures ensure proper localization and segmen- tation of vehicle damage, thus making it possible to provide a realistic assessment of the model. A description of the methodology used is given below. This in-cludes details of dataset preparation, model architecture, training configurations, and evaluation.

A. Dataset Used

The Car Damage Detection (CarDD) dataset [7], is employed for the research, which offers a vast library of images labelled to include different kinds of damage. The dataset is divided as follows:

- Training set: 2,819 images
- Validation set: 813 images
- Test set: 377 images

Each image is categorized into one of the following damage types:

1) Dent
2) Scratch
3) Crack
4) Glass Shatter
5) Lamp Broken
6) Tire Flat

B. Model Architecture

Mask R-CNN [9] is an enhancement over Faster R-CNN wherein the model embeds an auxiliary branch for pixel-wise segmentation and thus well suits the case of instance-level segmentation.

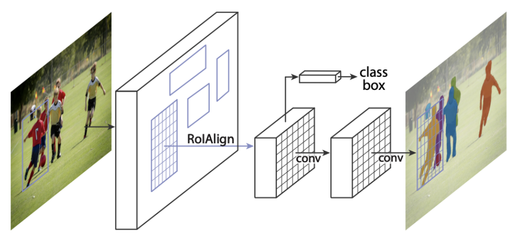

Fig. 1. Mask R-CNN Architecture, as proposed by He et al. [9].

This model enables vehicle damage detection by localizing and segmenting damages in images. The core components of the architecture include:

- **Backbone:** ResNet-50 [3] is used as the feature extractor, processing input images and generating hierarchical feature maps.
- **Feature Pyramid Network (FPN):** Enhances multi-scale object detection, providing feature maps at different levels.
- **Region Proposal Network (RPN):** Proposes candidate regions for possible vehicle damages.
- **ROIAlign:** Ensures that feature extraction is aligned for accurate bounding box and mask predictions.
- **Segmentation Head:** Generates pixel-wise segmentation masks for each detected damage category.

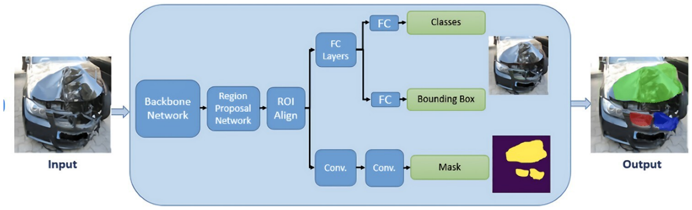

Fig. 2. Mask R-CNN model architecture for car damage detection [?].

C. Detailed Network Structure

The architecture used in our model, based on Detectron2 [3], is structured as follows:

TABLE I
MASK R-CNN MODEL SUMMARY

Layer (Type)	Output Shape	Param #
Input Layer	(None, 64, 122, 1)	0
Conv2D	(None, 64, 122, 32)	320
BatchNorm	(None, 64, 122, 32)	128
ReLU	(None, 64, 122, 32)	0
MaxPooling2D	(None, 32, 61, 32)	0
FPN Layer	Multi-scale Feature Maps	-
ROIAlign	Pooled feature maps	-
RPN	Bounding box proposals	-
Mask Head	Segmentation mask output	-
Total Params	**39,989**	**(156.21 KB)**

D. Custom Modifications

To improve performance and avoid overfitting, the following modifications were made:

- Added **Dropout Layers** at key points in the Mask R-CNN head.
- Adjusted **learning rate scheduling** for better convergence.
- Fine-tuned the model using the **CarDD dataset** [7], ensuring it learns vehicle-specific damages effectively.

E. Intersection over Union (IoU) in Model Training and Evaluation

To improve the correctness of our proposed vehicle damage detection model, we apply the IoU as a critical performance metric during both the training and testing processes. IoU is used to measure the overlap between the predicted bounding boxes and the ground truth annotations and ismathematically defined as

$$IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}} \tag{1}$$

Figure 3 shows the mathematical formulation of IoU that shows the intersection and union of two bounding boxes [10]. This helps in understanding how the metric quantifies object detection performance.

1. IoU in Model Training: In the training process, the Detectron2 library uses the IoU value to improve the object detection results. The RoI heads employ IoU-based thresholding to generate valid object proposals. The model employs the following default IoU threshold values:

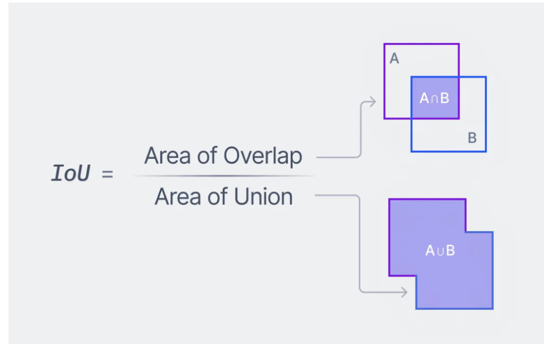

Fig. 3. Intersection over Union (IoU) formula and visualization of overlapping bounding boxes [10].

- IoU threshold for positive samples:

```
cfg.MODEL.ROI_HEADS.IOU_THRESHOLDS = [0.5, 0.7
```

2. IoU in Evaluation: For model evaluation, we employ the COCO Evaluator [10], which is based on the IoU as a key performance metric. The function:

```
evaluator = COCOEvaluator("Valid", cfg)
val_loader = build_detection_test_loader(cfg, "Valid")
inference_on_dataset(predictor.model, val_loader, evaluator)
```

assesses the precision of the predicted bounding boxes by calculating the IoU scores. The IoU scores are used to measure the spatial correspondence between the predicted damage regions and the ground truth annotations and a higher value indicates a better match.

Figure 4 demonstrates IoU in action for tracking multiple objects in real-world applications [11].

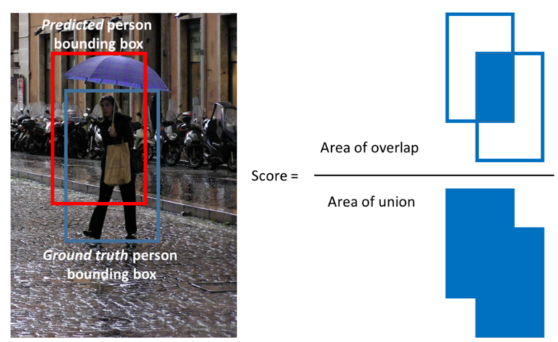

Fig. 4. Application of IoU in object tracking, comparing predicted and ground truth bounding boxes [11].

3. IoU in Prediction Phase: When applying the model to new images, an IoU-based confidence threshold is used to filter out low-quality detections

```
cfg.MODEL.ROI_HEADS.SCORE_THRESH_TEST = 0.70
```

This guarantees that only damage regions with high IoU confidence scores are considered real detections. We improve the robustness of our vehicle damage detection system by including IoU at multiple stages in our pipeline, namely training, validation, and inference, and thereby enhance localization accuracy and reduce repair cost estimation.

F. *IoU for Repair Cost Estimation*

Intersection over Union (IoU) quantifies the overlap between predicted damage regions and ground truth. A higher IoU indicates more severe damage and impacts cost estimation.

$$\text{Estimated Repair Cost} = \alpha \times IoU + \beta \quad (2)$$

where α and β are regression coefficients from a linear regression model [5].

1.Repair Cost Assignment: Custom cost functions are defined for different vehicle parts:

$$C_{\text{repair}}(IoU) = \begin{cases} \text{High}, & IoU \geq 0.7 \\ \text{Moderate}, & 0.5 \leq IoU < 0.7 \\ \text{Low}, & 0.3 \leq IoU < 0.5 \\ \text{Minimal}, & \text{otherwise} \end{cases} \quad (3)$$

2.Machine Learning for Cost Prediction: A regression model predicts repair costs using bounding box features:

$$Y = \beta_0 + \beta_1 X_1 + ... + \beta_n X_n + \varepsilon \quad (4)$$

where Y is the repair cost, X_i are bounding box features, and ε is error.

3.Final Cost Calculation: Total repair cost is computed as:

$$\text{Total Repair Cost} = \sum_{i=1}^{n} C_{\text{repair},i} \quad (5)$$

where n is the number of detected damages.

G. *Creation of Repair Cost Dataset*

The dataset for the cost of repair was generated using a multi-step process. Here is the car damage detection model initially run on a set of training images to predict bounding boxes. These predicted bounding boxes were integrated with ground truth annotations to form a full dataset of detected and annotated auto damage cases.

The following phase was based on the computed IoU scores of the overlap regions between the ground truth and forecasted bounding boxes. IoU was a crucial indicator for assessing how accurate the model's forecast was.

The allocation of the repair costs followed a methodical methodology. Data from several websites was also used along with previously extracted pose sequence and IoU scores to label the corresponding repair costs in the dataset. This methodology ensured that repair costs were allocated in proportion to the model's predictions and the ground truth, making precise the repair costs estimation.

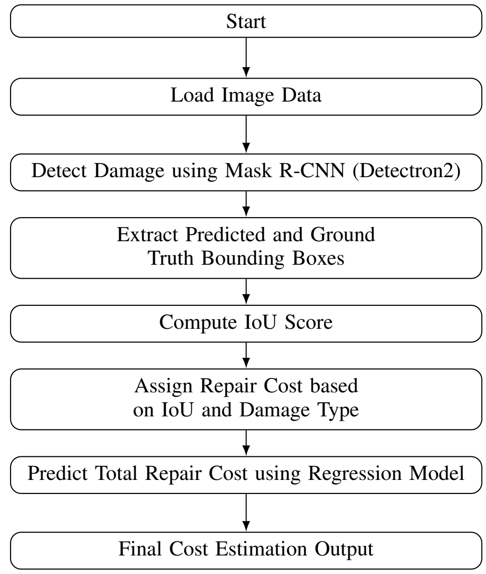

Fig. 5. Pipeline for damage detection and repair cost estimation.

V. Results

A. *Object Detection and Segmentation Results*

Figure 6 illustrates the performance of our model in detecting and segmenting car scratches. The evaluation metrics provide a quantitative measure of its effectiveness:

- The model achieves an **Average Precision (AP) of 41.04**, indicating overall segmentation accuracy.
- At an **IoU threshold of 50% (AP50)**, the model attains **52.118**, while at a stricter threshold of **75% (AP75)**, it reaches **42.84**.
- Performance varies based on object size: **Small objects (APs) score 0.528**, **medium-sized objects (APm) achieve 2.398**, and **large objects (APl) perform significantly better at 43.829**.

These results suggest that the model excels in detecting larger damages but struggles with smaller ones, which could be improved with further refinement.

B. *Damage-Specific Detection Performance*

To better understand category-wise performance, the following table summarizes the **detection accuracy across different damage types**:

Damage Type	AP Score
Dent	15.853
Scratch	17.030
Crack	3.591
Glass Shatter	85.744
Lamp Broken	47.809
Tire Flat	76.214

TABLE II
Per-category damage detection performance based on AP scores.

From these results, it is evident that the model performs exceptionally well in detecting **glass shatter (AP: 85.744),**

tire flats (AP: 76.214), and broken lamps (AP: 47.809). However, **cracks and dents show relatively lower accuracy**, indicating a need for further dataset augmentation and model optimization.

C. Bounding Box Detection Performance

Apart from segmentation, the model's ability to accurately localize damaged regions using bounding boxes is also evaluated. The overall **Bounding Box Average Precision (BBox AP) is 40.646**, highlighting its effectiveness in pinpointing damage locations. While this is a promising result, improving localization accuracy for smaller damages remains a key area for enhancement.

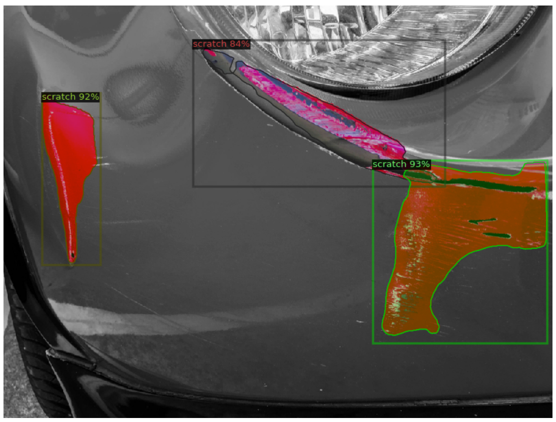

Fig. 6. Example of detected and segmented damages on a car's surface. The model successfully identifies a flat tire with 97% confidence.

D. Repair Cost Estimation

The damage was detected on **tire** with a confidence threshold of **97%** using bounding box and segmentation techniques. The estimated repair cost is **Rs 1617**.

Fig. 7. Estimated repair costs for various types of vehicle damage.

E. Conclusion and Future Work

This paper uses Detectron2 and Mask R-CNN to demonstrate an automated solution for detecting car damage and estimating repair costs. The system correctly detects and classifies damaged car parts by using a deep learning-based segmentation model. Additionally, damage severity is measured using the Intersection over Union (IoU) score, which is then used to predict repair costs in a linear regression model.

The suggested method improves the insurance claims procedure by offering a quick, precise, and automated evaluation of vehicle damage. This decreases fraudulent claims, speeds up claim settlements, and lessens the need for manual inspections. High accuracy in damage detection and cost estimate has been shown when machine learning is integrated with structured information.

Future work could concentrate on enhancing the system by:

- Adding more varied real-world damage cases to the dataset.
- For more accurate cost forecasts, deep learning-based regression models should be used rather than linear regression.
- Using real-time mobile applications to estimate costs and assess damage instantly.
- To expedite automated claim processing, an end-to-end API is being developed for insurance firms.

These enhancements might make the system even more resilient and flexible, which would make it a useful instrument for evaluating car damage in the insurance and automotive sectors.

These results demonstrate the model's effectiveness in real-world damage assessment scenarios.

References

[1] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 580–587.

[2] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European Conference on Computer Vision*. Springer, 2014, pp. 740–755.

[3] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, "Detectron2," 2019, accessed: January 28, 2025.

[4] S. Goswami, M. S. Uddin, M. R. Oishi, S. I. Shehab, and N. Shakib, "Image-based vehicle detection using various features," *International Journal of Innovative Science and Research Technology*, vol. 7, no. 1, pp. 709–713, 2022.

[5] C. Sevillano and F. J. P. Ayuso, "Methodology of repair cost estimation in vehicles based on deformation measurements in real world accidents," in *Symposium SEGVAUTO-TRIES-CM: Technologies for a Safe, Accessible and Sustainable Mobility*, Madrid, Spain, 2016.

[6] Q. Zhang, X. Chang, and S. Bian, "Vehicle-damage-detection segmentation algorithm based on improved mask r-cnn," *IEEE Access*, 2020.

[7] X. Wang, W. Li, and Z. Wu, "Cardd: A new dataset for vision-based car damage detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 7, pp. 7202–7214, 2023.

[8] "Intersection over union for object detection," accessed: January 2025.

[9] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2961–2969.

[10] "Intersection over union (iou) guide - v7 labs," accessed: January 2025.

[11] W.-J. Ahn, K.-S. Ko, M. T. Lim, D.-S. Pae, and T.-K. Kang, "Multiple object tracking using re-identification model with attention module," *Applied Sciences*, vol. 13, p. 4298, 2023.