Vision-based Deep Learning Framework for Assessing Car Damage and Identifying Damaged Components

Siwarat Laoprom, Thanaphit Sutthipanyo, Kanok Vongsaroj,
Wittawin Susutti, Warin Wattanapornprom, and (…)

Abstract—**In the realm of vehicle damage assessment, ensuring precision is vital for averting losses and minimizing costs. However, conventional assessment methods, reliant on manual inspections by trained experts, often prove time-consuming, subjective, and influenced by various factors. In this work, we harness the capabilities of Convolutional Neural Networks (CNNs) within the domain of computer vision to construct a comprehensive car damage detection and assessment framework. This framework comprises 3 interconnected components designed to detect the damage locations, identify damaged components, and specify damage types and severity levels. The 3 integral components are Car-Part Segmentation, Damage Type Detection, and Deformation Severity Classification. Our dataset encompasses image of damaged cars from diverse sources, and we have diligently reassessed annotations and labels to harmonize the data annotations originating from various sources with distinct class definitions. In our experiments, we evaluated one-stage and two-stage instance segmentation models for Car Part Segmentation and Damage Type Detection. YOLOv8, a one-stage model, demonstrated real-time efficiency and high accuracy, making it our preferred choice. For Deformation Severity Classification, we explored different models, including ResNet50, VGG16, and CAFORMER. Among these, ResNet50 offered the best balance of precision and recall, making it our selection for its accuracy and lightweight nature, aligning with our application objectives.**

Keywords—**Car Damage Detection, Car Damage Severity Assessment, Car Parts Segmentation, Object Detection, Instance Segmentation, Image Classification, Deep Learning**

I. INTRODUCTION

In an ever-changing transportation landscape, vehicles play an important role in facilitating personal and commercial activities. However, as convenient and useful as they are, car accidents are still an unfortunate reality that can result in significant loss of lives and money. One of the most important measures to prevent these incidents is how an accurate assessment of the severity of the vehicle damage. Determining the extent of damage after a collision is important not only for insurance claims and maintenance decisions but also for the safety of passengers and other road users.

Traditional methods of assessing vehicle damage severity are often based on manual inspection by trained experts, which can be time-consuming, dependent on human discretion and many factors. Advances in computer vision , machine learning (ML) and artificial intelligence have an increased chance of automating this process and making it more accurate by applying to locate damaged components to the extent.

The field of the Deep Learning (DL) area is achieving remarkable results in many computers vision tasks. The application of DL techniques, notably Convolutional Neural Networks (CNNs), has revolutionized object detection and instance segmentation tasks. various state-of-the-art frameworks are proposed for example, R-CNN [1] combines CNNs with region proposal methods. It significantly improves object detection accuracy by allowing the model to focus on relevant regions and leverage the power of deep learning for classification and localization tasks. However, R-CNN's multi-stage design can be computationally expensive, leading to subsequent developments like Fast R-CNN [2] and Faster R-CNN [3] that optimize this process for efficiency. And another example, stand out framework like YOLO (You Only Look Once) [4] that aims to achieve high accuracy in object detection with real-time speed.

In the present time, with the remarkable capabilities of CNNs as mentioned, they have garnered attention and have been effectively employed for the purposes of detecting and assessing car damage [5] [6] [7]. However, within this domain, there is a lack of standardized criteria or clear benchmarks, making direct comparison of works in this domain challenging. Additionally, some studies do not categorize or classify damage types adequately. This work aims to introduce a framework for car damage detection and assessment, including the identification of damaged components. Our approach consists of 3 main components, Car Part Segmentation, Damage Type Detection, and Damage Severity Classification. Currently, our method supports specific severity classification only for damage of deformation, and thus, the final component can also be referred to as 'Deformation Severity Classification'.

II. RELATED WORK

Studies within the domain of Car Damage Detection often focus on identifying the location of damage occurring on the vehicle itself. Detecting damaged components presents an intricate challenge due to these reasons. Firstly, a damaged car looks distinct from its original state, rendering models trained on datasets of undamaged vehicles less effective. Secondly, damages differ significantly from everyday objects, introducing fresh hurdles to existing algorithms in the broader field of object detection [8]. Prior works employ sensor-based approaches [9] [10] [11] that require additional devices, causes the limitations in applications.

Indeed, there have been prior works proposing vision-based approaches [12] [13]. but these methods have not consistently yielded satisfactory results. However, when CNNs are applied in conjunction with various techniques for tasks of this nature, they demonstrate outstanding capabilities. For instance, P. Kalpesh et al. [5] employed CNNs along with transfer learning and ensemble learning techniques to classify car damage images. D. Najmeddine et al. [6] presented an approach that combines an instance segmentation model with transfer learning techniques to improve damage localization efficiency. Furthermore, S. Atharva et al. [7] utilized 2 CNN models—one for detecting damage positions and another for generating masks around damaged areas.

However, even within the same field of research, the definitions of damage vary significantly. In this field, there is no standardized or well-defined set of rules or criteria. For example, in some works, methods are presented for detecting only the locations of damage without specifying the type and severity or only focusing on 1 type of damage: P. Li et. al. [14] introduce car damage detection dataset which is collected from various search engine using “dent” and “scratch” as searching keyword and proposed the anti-fraud system utilizing YOLO model for detecting car damage without identifying type and severity of damage. Q. Zhang et. al. [15] presented improved MRCNN for detecting car damage focusing on only scratches. In other cases, in addition to detecting the locations, the severity is assessed but without specifying the type of damage; A. Dasari et. al. [16] presented car damage detection system which categorizes damage severity into minor, moderate and major damage but does not specify which type the detected damages are. P. M. Kyu and K. Woraratpanya [17] proposed the framework to detect the location of damaged part where it locates in front, rare or side and to assess damage severity where the damage level is defined as follows [18]. Some works classify the types of damage and even identify which components are damaged; H. S. Malik et. al. [19] proposed the pipeline combining the classification and detection models which identifies damaged part by predicted class of damage, but none provide a comprehensive and detailed definition of damage types and severity. In this study, we propose a more detailed and comprehensive approach that also allows for precise identification of the damaged components.

When it comes to identifying damaged car components, one intriguing approach is Car Part Segmentation. Current research in this area is somewhat limited in addressing our specific interests. Some studies have utilized image processing techniques in conjunction with 3D CAD models to segment vehicle parts [13] [20]. However, these approaches have certain limitations, as they cannot precisely determine the identity of the segmented parts. W. Lu et al. [21] introduced graphical models that integrate segmentation appearance consistency (SAC) coupling terms between neighboring landmarks to assess which part of the car each pixel in the image represents. Their method can differentiate between windows, lights, the car body, and license plates but is not more granular. R. Singh et al. [22] presented a process for identifying damaged components by finding the overlapped area between damage and detected car parts. In their work, they classified car components into approximately 32 detailed classes, while the damage classification was less detailed, including categories like scratch, minor dent, major dent, crack, and missing. They applied a two-stage model along with ensemble techniques. However, the results of these approaches are not highly accurate.

[1] https://github.com/dsmlr/Car-Parts-Segmentation

Pasupa K. et al. [23] compared the accuracy of several deep learning models and algorithms for car part segmentation using a self-collected dataset named DSMLS Car Part dataset that they shared as a public resource[1]. This dataset comprises 500 images of SUVs, with annotations for 18 car part classes. Unfortunately, this dataset lacks completeness in terms of labeling; some images may lack annotations for certain parts, as depicted in Figure 1. Furthermore, Yusuf S. A. et al. [24] utilized the DSMLS dataset in conjunction with a locally collected dataset, sourced from local car repair workshops, to evaluate the capabilities of SipMask and YOLACT models for real-time instance segmentation of car parts. However, a recent development is the 'Humans in the Loop' dataset, which offers a reasonably detailed and comprehensive annotation of both damaged and undamaged car images, totaling over a thousand images [25]. Fortunately, this dataset has been dedicated to the public domain under a CC0 1.0 license, providing a valuable resource for future research.

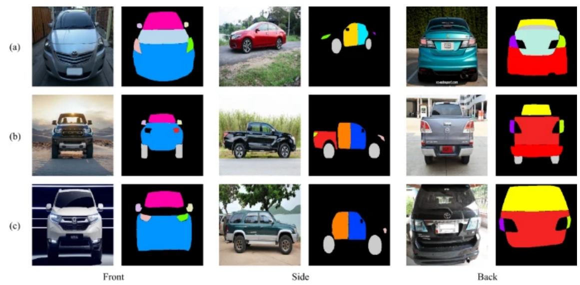

Figure 1. Samples of pair images and instance mask from the DSMLR Car Part data set: (a) sedan, (b) pickup and (c) sports utility vehicle (SUV) (ref. Figure 8. in [23])

III. METHODOLOGY

In this section, we will elucidate the methodology employed in this study. We initiate by discussing the entirety of the dataset used, followed by an exploration of each component within our framework. Finally, we state about evaluation of the framework.

A. Dataset

Given that each component within the framework is compassed by a distinct model, it becomes necessary to employ separate datasets for each component. These datasets include the Car-Part Instance Segmentation Dataset, Damage-type Detection Dataset, and Deformation Severity Dataset.

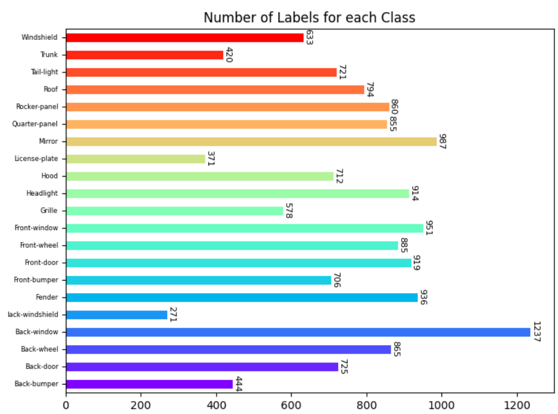

Figure 2. Bar chart depicts the number of instances for each class of Car-Part Instance Segmentation Dataset

Figure 3. Sample images of Car-Part Instance Segmentation Dataset

Car-Part Instance Segmentation Dataset we employ the open-source dataset for the car part and car damage analysis provided by the Humans in the Loop [25] that consists of consists of a total of 1,812 images but we chose only 998 images fully annotated with polygons either for car parts (the bounding boxes are automatically generated by Roboflow). The dataset is split into 3 sets which are the 70% training set, the 20% validation set, and the 10% testing set. The dataset includes a total of 15,784 annotated instances of 21 car part classes. Figure 2 show amount of instance for each class of Car-Part Instance Segmentation Dataset and Figure 3 show some sample images from the dataset.

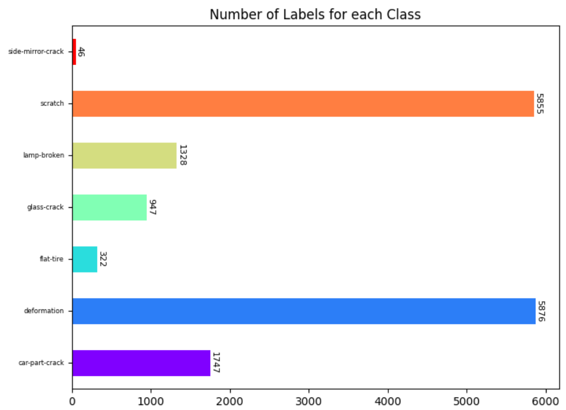

Figure 4. Bar chart depicts the number of instances for each class of Damage-type Detection Dataset

Figure 5. Sample images of Damage-type Detection Dataset

Damage-type Dataset The dataset for car damage detection was collected from various sources (e.g., the CarDD dataset [8], private dataset from insurance company, and public datasets on Roboflow and Kaggle) that consist of a total of 7,700 images of the damaged car (split dataset with 70% for the training, 20% for the validation set, and remaining for the testing set). We reassessed annotation masks with 7 damage-type classes to syncretize the annotations of various sources of data that have different classes. We defined criteria for annotating masks, the following list is the annotation criteria for each damage-type class:

- **Crack**, cryptic chips, cracks, fractures, and breaks on car parts or massive scratches that make a hole in the surface of the car.
- **Deformation**, cryptic dents, and damages that deform the shape of some parts or the whole body of the car, also including on the surface of the car.
- **Flat Tire**, the tire is not inflated where you can observe it obviously.
- **Glass Crack**, all glass cracks other than front and back lamps also side mirror.
- **Lamp Broken**, the glass of the lamp or the lamp is cracked or broken.
- **Scratch**, cryptic scuffs and scratches or multiple scratches which appear in an adjacent area.
- **Side Mirror Crack**, the glass of the side mirror or the side mirror is cracked or broken.

Figure 4 show amount of instance for each class of Damage-type Dataset and Figure 5 show some sample images from the dataset.

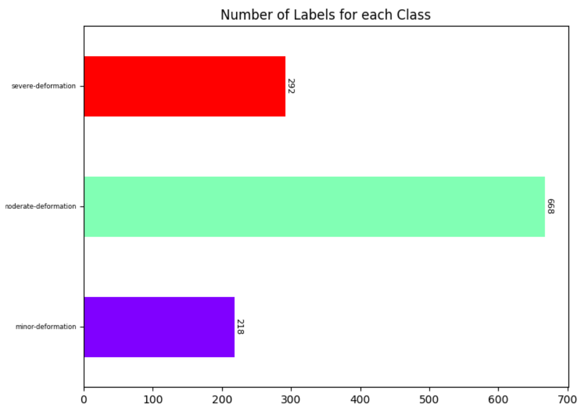

Figure 6. Bar chart depicts the number of instances for each class of Deformation Severity Dataset

Figure 7. Sample images of Deformation Severity Dataset

Deformation Severity Dataset the dataset for deformation severity classification was collected from various sources like Damage-type Dataset but we filter only deformation damage. The dataset contains total of 845 images of car within deformation damage. Like Damage-type Detection Dataset, we reassessed annotation masks of deformation damage by the following criteria which is adopted from [16] and [18]:

- **Minor Deformation**, small or narrow dents, that do not deform the shape of the whole part of the car, on the surface of some car parts.
- **Moderate Deformation**, large or wide dents and damages that deform the shape of some parts but not the whole body of the car.
- **Severe Deformation**, large or wide dents and damages that deform the shape of some parts but not the whole body of the car.

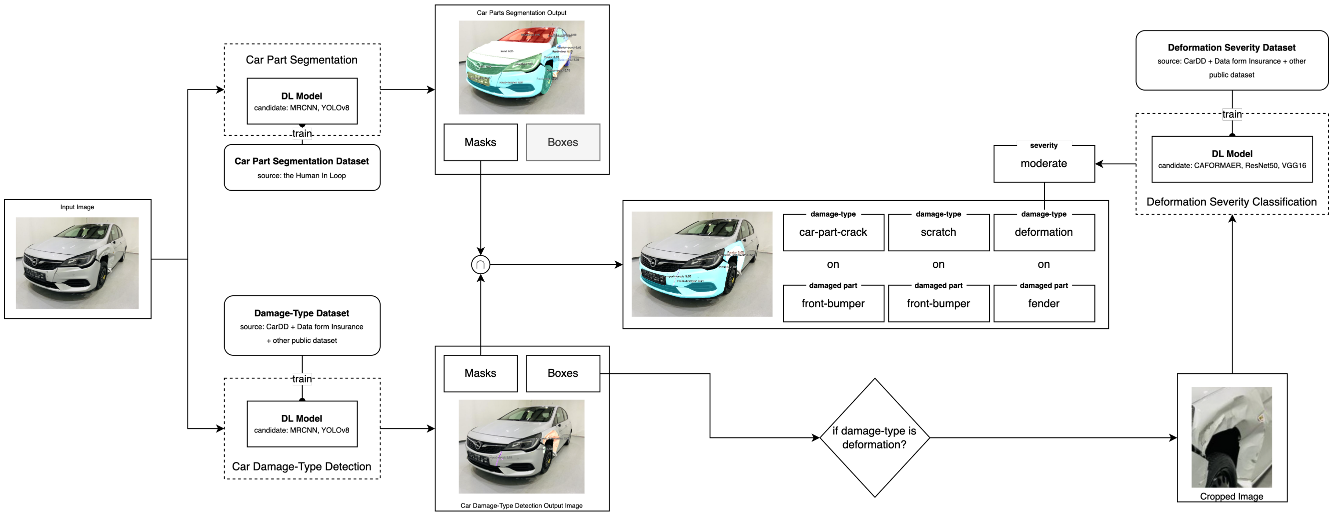

Figure 8. Schematic diagram show overview of the framework

Figure 6 show amount of instance for each class of Deformation Severity Dataset and Figure 7 show some sample images from the dataset.

B. *Car Damage Detection and Assessment Framework*

We design our method underlying the idea that by separating the damage type and damage degree detection components into two distinct stages, we improved the performance of our system in the damage type and damage degree detection parts, drawing on our prior strategy. To locate and identify damage areas, we first utilize an instance segmentation model. Then, we apply damage severity classification models to classify the identified areas.

Therefore, the framework consists of 3 main components: Car Part Segmentation, Damage Type Detection and Deformation Severity Classification. Starting with Car Part Segmentation, we take the entire view of car image as input to segment its constituent parts. Parallel with Damage Type Detection, we utilize the same input image as in the segmentation to locate the incurred damage and classify its type. The overlapping instances of Car Part Segmentation and Damage Type Detection resultant masks will identify which parts of the car are damaged. Furthermore, in instances of deformation as damage, we crop the image from that specific region to facilitate Deformation Severity Classification, which assesses the degree of deformation-related damage. The Figure 8. depict the schematic diagram of overview of the framework.

In our experimentation phase, we rigorously evaluated both one-stage and two-stage instance segmentation models, which are the popular methods for object detection and image segmentation tasks, to determine the most appropriate technique for our tasks of car part segmentation and damage type detection. We conducted a comprehensive analysis, considering elements including accuracy, pace, and ease of implementation. Therefore, we selected MRCNN [26] and YOLOv8 [27], which are one-stage and two-stage instance segmentation model respectively, for car-parts segmentation and damage type detection tasks.

In the deformation severity classification task, we explored a diverse range of model architectures. In particular, we evaluated the efficacy of two prominent approaches: transfer learning on traditional Convolutional Neural Network (CNN) based models, and transfer learning on Vision Transformer (ViT) based models. This experimentation delves into our findings and considerations of these two distinct architectural choices. Hence we evaluate 1 ViT based model and 2 CNNs classification models which respectively are CAFORMER [28], ResNet50 [29] and VGG16 [30] as the candidate models for deformation severity classification task.

C. *Evaluation*

For performance evaluation in this study, we categorize the evaluation into 3 distinct subsections. These subsections encompass measuring results tailored to each component within the 3 parts of framework. To quantify the effectiveness of the Car Part Segmentation and Damage Type Detection components, we employ the mean Average Precision (mAP) and mAP-50 metrics for bounding box evaluation in the context of detection tasks. For segmentation tasks, we utilize the annotation mask metric. It's worth noting that mAP represents the average precision across multiple levels of IoU (Intersection over Union) threshold in range of 0.5 to 0.95 stepping by 0.05, offering a comprehensive performance assessment for object detection and instance segmentation. Additionally, mAP-50 represents the mean Average Precision at a fixed IoU level of 0.5, providing insights into the model's performance at a specific recall threshold.

$$mAP = \frac{1}{|N|}\sum_{C \in N} AP_C \; ; \; N \text{ is set of classe} \qquad (1)$$

As for the Deformation Severity Classification component, we employ common classification metrics including F1 score, Precision, Recall, and Accuracy.

IV. Result and Discussion

In this section, we will elaborate on our experimental outcomes, categorized into Car Part Segmentation Experiments, Damage Type Detection Experiment, Deformation Severity Classification, Model Representation, and Data Issues encountered in this study, along with the corresponding mitigation strategies. To avoid confusion between results pertinent to the detection task and

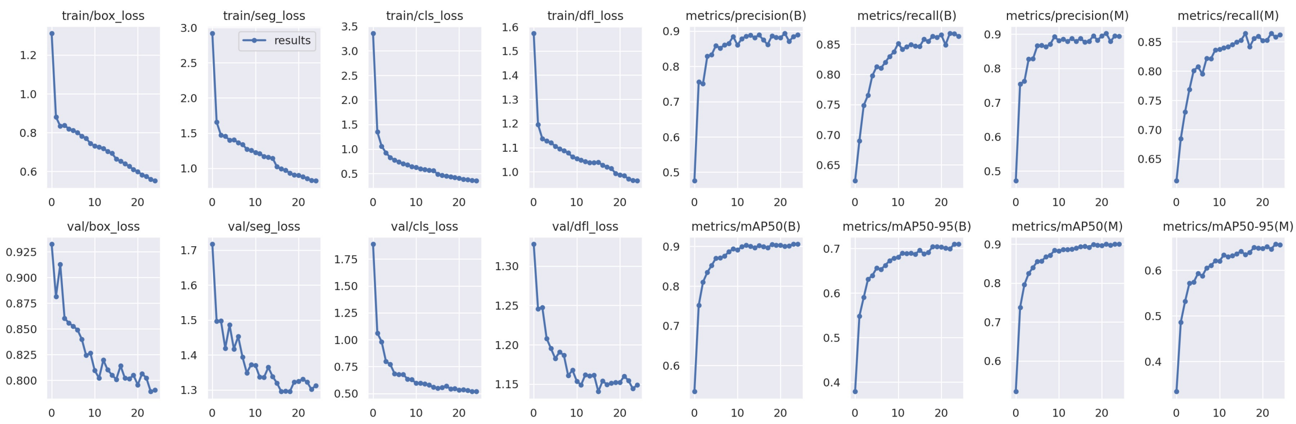

Figure 9. Graphs show the trajectory of training and validation while training YOLOv8x for car part segmentation

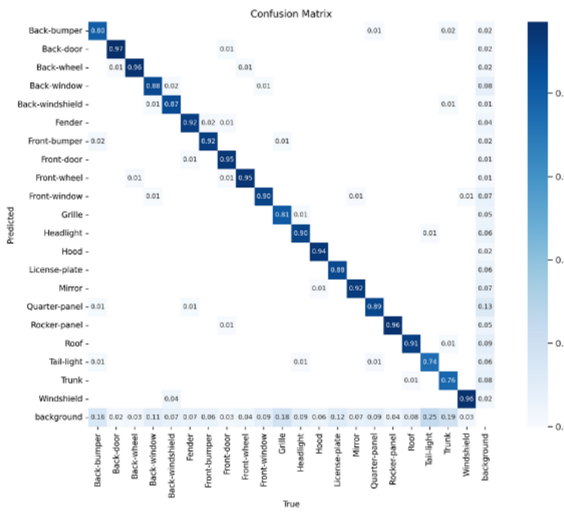

Figure 10. Confusion Matrix of YOLOv8x for car part segmentation respecting testing set

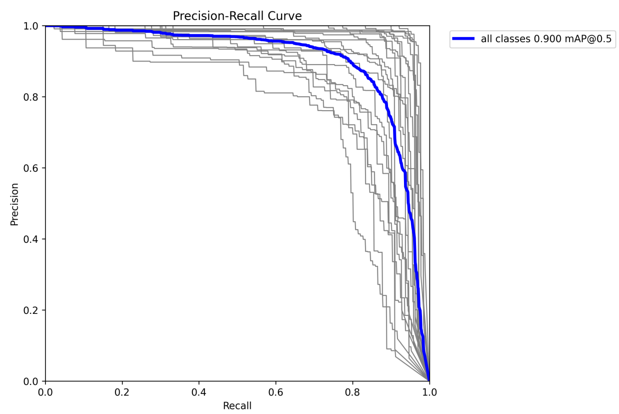

Figure 11. Precision-Recall Curve of YOLOv8x for car part segmentation respecting testing set

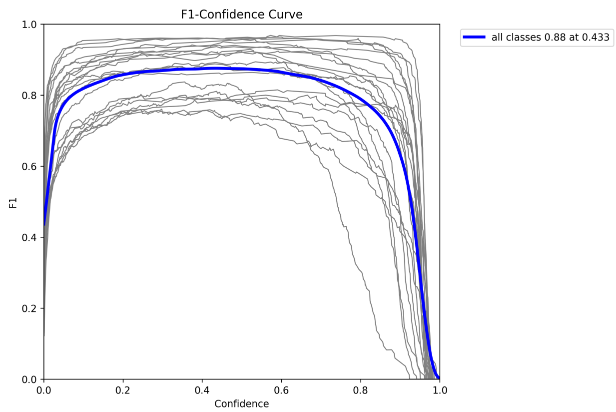

Figure 12. F1-Confidence Curve of YOLOv8x for car part segmentation respecting testing

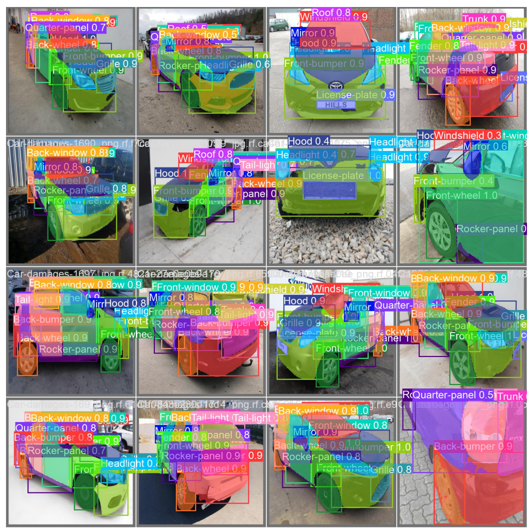

Figure 13. Sample output of car part detection and segmentation by YOLOv8x within confidence threshold as 0.25

segmentation tasks, we specify that when discussing the results of the detection task, we are referring to outcomes related to bounding box evaluation. Conversely, for the segmentation task, the results pertain to the annotation mask evaluation.

A. *Car Part Segmentation Experiments*

For car part detection and segmentation, we conducted experiments to compare the capabilities of MRCNN and YOLOv8, both trained using the Car-Part Instance Segmentation Dataset. We evaluate them by mAP and mAP-50, as presented in Table I. The results indicated that the YOLOv8x model outperformed others in terms of detection and segmentation accuracy. Figure 9. shows the trajectory of training and validation while training YOLOv8x for car part segmentation and Figures 10., 11., and 12., depicting the confusion matrix, precision-recall curve, and F1-confidence curve respecting car part segmentation testing set of YOLOv8x respectively, provide clear evidence of its high overall and instance specific performance (sample of the output are shown in Figure 13.).

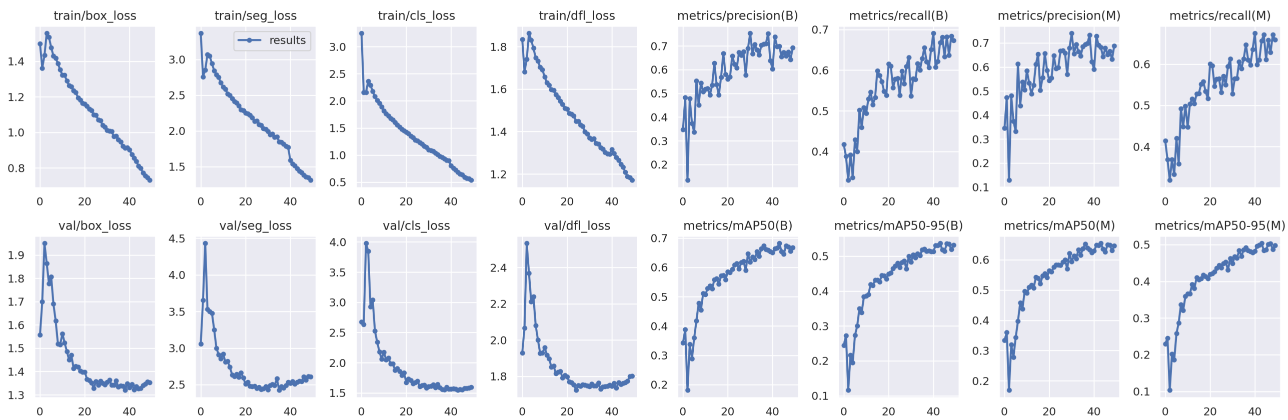

Figure 14. Graphs show the trajectory of training and validation while training YOLOv8l for damage type detection

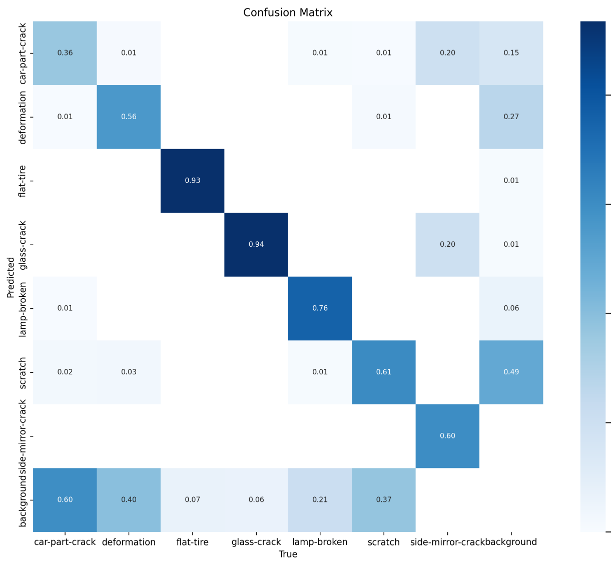

Figure 15. Confusion Matrix of YOLOv8l for damage type detection respecting testing set

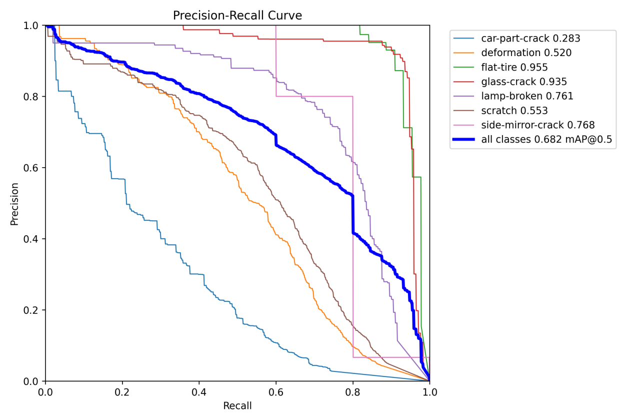

Figure 16. Precision-Recall Curve of YOLOv8l for damage type detection respecting testing set

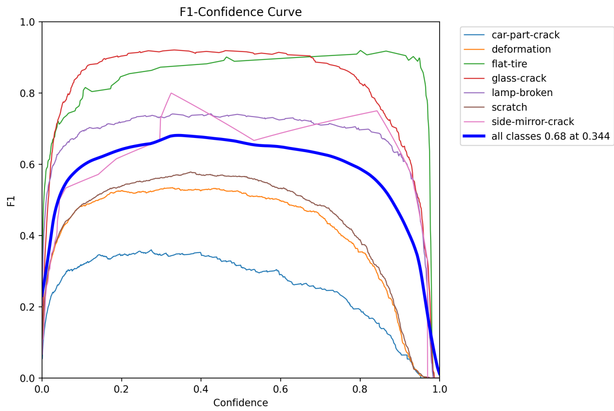

Figure 17. F1-Confidence Curve of YOLOv8l for damage type detection respecting testing set

Figure 18. Sample output of damage type detection and segmentation by YOLOv8l within confidence threshold as 0.25

Furthermore, upon comparing our results with other state-of-the-art methods, as presented in Table II, it becomes evident that our model achieves higher mAP values in both detection and segmentation tasks.

B. Damage Type Detection Experiments

Given the outcomes of car part segmentation from the previous section, we observed that YOLOv8 outperforms MRCNN in terms of efficiency. Therefore, we prioritize YOLOv8 as our primary choice. To ensure YOLOv8's capability to yield satisfactory results in preliminary damage type detection, we proceeded to train YOLOv8 using the smaller CarDD dataset, as opposed to our comprehensive Damage Type Detection Dataset. This enables us to later compare these results with outcomes from other studies employing MRCNN, as shown in Table III.

TABLE I. EXPERIMENTAL RESULT TO COMPARE MODELS CAPABILITY IN CAR PART SEGMENTATION

Model	mAP		mAP-50	
	Box	Mask	Box	Mask
MRCNN-ResNet101-FPN	0.51	0.50	0.77	0.75
YOLOv8l	0.69	0.63	0.89	0.89
YOLOv8x (best)	**0.70**	**0.65**	**0.90**	**0.90**

TABLE II. COMPARING CAR PART DETECTION AND SEGMENTATION RESULT WITH STATE-OF-ART METHODS

Model [Ref.]	mAP		mAP-50	
	Box	Mask	Box	Mask
Parts MRCNN [22]	0.35	-	-	-
Parts PANet [22]	0.32	-	-	-
Parts MRCNN+PANet [22]	0.38	-	-	-
Mask R-CNN+ResNet-50 [23]	0.56	0.59	0.81	0.82
SipMask++ [24]	0.56	0.51	0.84	0.78
SipMask [24]	0.59	0.53	0.86	0.80
YOLACT [24]	0.45	0.49	0.80	0.75
Our best model (YOLOv8x)	**0.70**	**0.65**	**0.90**	**0.90**

TABLE III. COMPARING PRELIMINARY DAMAGE TYPE DETECTION RESULT WITH STATE-OF-ART METHODS

Model [Ref.]	mAP [Box]	mAP-50 [Box]
Damage MRCNN [22]	0.4	-
Our model (YOLOv8l)	**0.61**	**0.74**

TABLE IV. EXPERIMENTAL RESULT OF DAMAGE TYPE DETECTION

Model	mAP [Box]	mAP-50 [Box]
YOLOv8l	0.5	0.68

From the results in Table III, it is evident that YOLOv8 is likely to possess better capabilities in damage type detection. Consequently, we employed YOLOv8 as the primary model for the damage type detection within our framework. The outcomes presented in Table IV are the results of experiments where we trained the model using our dataset. Figure 14. shows the trajectory of training and validation while training YOLOv8l for damage type detection and Figure 15., 16., and 17. represent the confusion matrix, precision-recall curve, and F1-confidence curve respecting damage type detection testing set, respectively also sample of the output are shown in Figure 18.

During the evaluation of one-stage object detection model, YOLOv8, we observed their impressive efficiency in processing images in real-time, making them particularly well-suited for our automotive applications. These models demonstrated remarkable performance in accurately segmenting both car parts and damage types within the car images. On the other hand, the two-stage image segmentation model, MRCNN, shows lower accuracy in localizing objects and segmenting complex regions. and their increased computational demands and slightly slower inference times were notable drawbacks for our use cases,

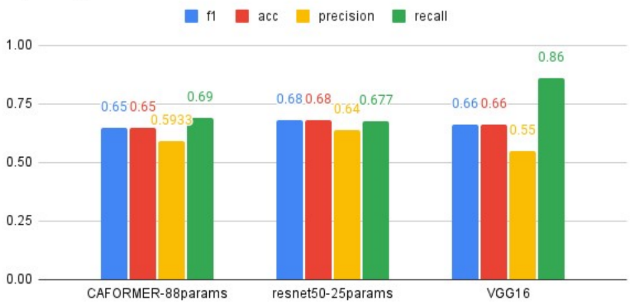

Figure 19. Deformation severity classification results of 3 CNNs

where speed was a critical factor when it came to application services.

After a thorough assessment, we made the strategic decision to adopt one-stage object detection, YOLOv8, for both Car Part Segmentation and Damage Type Detection in our method. The effectiveness and speed offered by one-stage models proved to be paramount for our applications, enabling rapid and precise analysis of vehicle images. This choice not only ensured efficient real-time processing but also maintained high levels of accuracy, which aligns perfectly with our objectives in the automotive domain.

C. Deformation Severity Classification Experiments

We conducted a comparative experiment involving three CNN models: CAFORMER [28], ResNet50 [29] and VGG16 [30], for the task of deformation severity classification. The objective was to determine the model that achieves the highest classification accuracy. We evaluated the results using metrics such as F1 score, precision, recall, and accuracy, as presented in Figure 19.

We conducted experiments on CNN-based models, ResNet50, VGG16, and the ViT-based model, CAFORMER. They transferred weights to the COCO dataset. Our experiment shows that VGG16 achieves the highest recall at 0.86 but is slightly lower in precision at 0.55 because of the limited size of our dataset and the large number of parameters leading to overfitting. compared to Resnet50, which performs better in both precision at 0.64 and recall at 0.68 with the fewest parameters and also outperforms the results from CAFOMER.

We have therefore decided to move forward with the ResNet50 model for our Deformation Severity Classification task in consideration of our application objective of obtaining accurate findings while being lightweight.

D. Result Representation

In this subsection, we will present examples of the outputs generated by the framework. These outputs are categorized into two types: image outputs and summary tabular outputs. For image outputs, they consist of car part segmentation images, damage detection images that have undergone deformation damage severity assessment, and images displaying only instances that intersect between car part instances and damage instances, as illustrated in Figure 20. As for tabular outputs, the tables have a number of rows equal to the detected car parts. There are 4 columns: 'car_part,' representing the identified car part in the image; 'damage_type,' listing the damage types observed on that particular car part; 'damage_severity,' listing the severity levels of damage within the same 'damage_type' index

(a) (b) (c)

Figure 20. Example of (a) car part segmentation image output, (b) damage detection image output, (c) damaged car part image output

(Unless it's a deformation-type damage, in which case it will be labeled as 'None'); and 'damage_instance,' which is a list of damage instance IDs, as exemplified in Table VI.

TABLE V. EXAMPLE OF TABULAR OUTPUT

car_part	damage_type	damage_severity	damage_instance
Back-door	[]	[]	[]
Back-wheel	[]	[]	[]
Back-window	[]	[]	[]
Fender	[car-part-crack]	[None]	[0]
Front-bumper	[car-part-crack, deformation]	[None, Moderate]	[0, 1]
Front-door	[]	[]	[]
Front-wheel	[]	[]	[]
Front-window	[]	[]	[]
Grille	[]	[]	[]
Headlight	[car-part-crack, deformation]	[None, Moderate]	[0, 2]
Hood	[deformation]	[Moderate]	[2]
Mirror	[]	[]	[]
Quarter-panel	[]	[]	[]
Rocker-panel	[]	[]	[]
Roof	[]	[]	[]
Windshield	[]	[]	[]

F. Data Issue

Throughout our research, we encountered several data challenges that required careful handling. Firstly, the presence of duplicate images within the dataset created testing issues, particularly when these duplicates had inconsistent labels since manual labeling. Addressing this problem necessitated a comprehensive dataset preprocessing step to eliminate repeated images. Secondly, disparities in data granularity and class categorization from various sources of dataset mandated extensive relabeling efforts. This misalignment stemmed from a lack of clarity and domain knowledge in damage type definitions, leading to multiple rounds of corrections and relabeling. Furthermore, an overabundance of finely detailed classes in the dataset resulted in the creation of numerous unnecessary classes, many of which had limited data. To streamline our dataset, we consequently removed redundant classes. Additionally, classifying images based on damage size proved challenging, especially when dealing with images captured from various angles. To mitigate this issue, we adopted a strategy of selecting images with similar characteristics and avoiding zoomed-in-damage images. Finally, the inclusion of small or ambiguously defined wound classes adversely affected model accuracy. To address this concern, we made the decision to exclude classes representing small-sized or unclear damages from our dataset, enhancing the overall quality and focus of our data.

V. CONCLUSION

We have presented a framework for detecting car damage and identifying damaged car parts, along with specifying the severity of deformation damage. Additionally, we have collected data from various sources and relabeled this data to align with our defined criteria, creating a dataset for this work. The framework we propose comprises 3 main components: Car Part Segmentation, which divides the car into its constituent parts; Damage Type Detection, which identifies and categorizes damage types by assessing the area of overlap between car parts and damage areas; and finally, Deformation Severity Classification, which evaluates the level of deformation damage.

Through experimentation, we conducted a thorough evaluation of one-stage and two-stage instance segmentation models, widely used in object detection and image segmentation tasks, to identify the most suitable technique for our Car Part Segmentation and Damage Type Detection tasks. Notably, the one-stage object detection model YOLOv8 demonstrated exceptional real-time image processing efficiency, making it ideal for our automotive applications. It exhibited high accuracy in segmenting both car parts and damage types within car images. In contrast, the two-stage image segmentation model Mask-RCNN showed lower accuracy in localizing objects and complex region segmentation, with increased computational demands and slower inference times. Consequently, we opted for YOLOv8 for both Car Part Segmentation and Damage Type Detection due to its efficiency and speed, maintaining accuracy in line with our automotive objectives.

Regarding Deformation Severity Classification, we explored various model architectures, particularly focusing on transfer learning using traditional CNNs like ResNet50 and VGG16, as well as ViT-based models like CAFORMER. Our experiments showed VGG16 achieving the highest recall at 0.86 but lower precision at 0.55 due to dataset size limitations and overfitting. In contrast, ResNet50 outperformed with better precision at 0.64 and recall at 0.68, boasting fewer parameters and superior results compared to CAFORMER. Consequently, we selected the ResNet50 model for our Deformation Severity Classification task, prioritizing accurate findings while maintaining a lightweight architecture in alignment with our application goals.

ACKNOWLEDGMENT

(Thank everyone)

REFERENCE

[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*, 2014.

[2] G. Ross, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015.

[3] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in *Advances in Neural Information Processing Systems 28*, 2015.

[4] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, 2016.

[5] K. Patil, M. Kulkarni, A. Sriraman and S. Karande, "Deep Learning Based Car Damage Classification," in *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, Cancun, Mexico, 2017. doi: 10.1109/ICMLA.2017.0-179.

[6] N. Dhieb, H. Ghazzai, H. Besbes and Y. Massoud, "A Very Deep Transfer Learning Model for Vehicle Damage Detection and Localization," in *2019 31st International Conference on Microelectronics (ICM)*, Cairo, Egypt, 2019.

[7] A. Shirode, T. Rathod, P. Wanjari and A. Halbe, "Car Damage Detection and Assessment Using CNN," in *2022 IEEE Delhi Section Conference (DELCON)*, New Delhi, India, 2022. doi: 10.1109/DELCON54057.2022.9752971.

[8] 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. doi: 10.1109/TITS.2023.3258480.

[9] S. Karbacher, J. Babst, G. Häusler and X. Laboureux, "Visualization and Detection of Small Defects on Car-Bodies," *Modeling and Visualization,* vol. 99, 1999.

[10] S. Gontscharov, H. Baumgärtel, A. Kneifel and K.-L. Krieger, "Algorithm Development for Minor Damage Identification in Vehicle Bodies Using Adaptive Sensor Data Processing," *Procedia Technology,* vol. 15, pp. 586-594, 2014.

[11] H. Baumgärtel, A. Kneifel, S. Gontscharov and K.-L. Krieger, "Investigations and Comparison of Noise Signals to Useful Signals for the Detection of Dents in Vehicle Bodies by Sound Emission Analysis," *Procedia Technology,* vol. 15, pp. 716-725, 2014.

[12] T. Lilienblum, P. Albrecht, R. Calow and B. Michaelis, "Dent detection in car bodies," in *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*, Barcelona, Spain, 2000. doi: 10.1109/ICPR.2000.903032 .

[13] S. Jayawardena, "Image based automatic vehicle damage detection," 2013. [Online]. Available: http://hdl.handle.net/1885/11072. [Accessed 2023].

[14] P. Li, B. Shen and W. Dong, "An Anti-fraud System for Car Insurance Claim Based on Visual Evidence," *arXiv:1804.11207 [cs.CV],* 2018.

[15] Q. Zhang, X. Chang and S. B. Bian, "Vehicle-Damage-Detection Segmentation Algorithm Based on Improved Mask RCNN," *IEEE Access,* vol. 8, pp. 6997 - 7004, 2020. doi: 10.1109/ACCESS.2020.2964055.

[16] A. R. Dasari, S. Shambharkar, K. Jyothsna and V. M. Kumar, "Automatic Vehicle Damage Detection Classification framework using Fast and Mask Deep learning," in *Conference: 2022 Second International Conference on Computer Science, Engineering and Applications (ICCSEA)*, 2022. doi:10.1109/ICCSEA54677.2022.9936208.

[17] P. M. Kyu and K. Woraratpanya, "Car Damage Detection and Classification," in *The 11th International Conference on Advances in Information Technology (IAIT2020)*, 2020, Bangkok, Thailand.

[18] W. R. Harshani and K. Vidanage, "Image processing based severity and cost prediction of damages in the vehicle body: A computational intelligence approach," in *2017 National Information Technology Conference (NITC)*, Colombo, Sri Lanka, 2017.

[19] H. S. Malik, M. Dwivedi, S. N. Omakar, S. R. Samal, A. Rathi, E. B. Monis, B. Khanna and A. Tiwari, "Deep Learning Based Car Damage Classification and Detection," EasyChair, 22 March 2020. [Online]. Available: https://easychair.org/publications/preprint/RlQ2.

[20] Q. Liu, A. Kortylewski, Z. Zhang, Z. Li, M. Guo, Q. Liu, X. Yuan, J. Mu, W. Qiu and A. Yuille, "Learning Part Segmentation through Unsupervised Domain Adaptation from Synthetic Vehicles," in *IEEE/CVF conference on computer vision and pattern recognition*, 2022.

[21] W. Lu, X. Lian and A. Yuille, "Parsing Semantic Parts of Cars Using Graphical Models and Segment Appearance Consistency," *arXiv:1406.2375 [cs.CV],* 2014.

[22] R. Singh, M. P. Ayyar, T. V. S. Pavan, S. Gosain and R. R. Shah, "Automating Car Insurance Claims Using Deep Learning Techniques," in *2019 IEEE Fifth International Conference on Multimedia Big Data (BigMM)*, Singapore, 2019. doi:10.1109/BigMM.2019.00-25.

[23] K. Pasupa, P. Kittiworapanya, N. Hongngern and K. Woraratpanya, "Evaluation of deep learning

algorithms for semantic segmentation of car parts," *Complex & Intelligent Systems,* vol. 8, p. 3613–3625, 2022.

[24] S. A. Yusuf, A. A. Aldawsari and R. Souissi, "Real-Time Instance Segmentation Models for Identification of Vehicle Parts," *Complexity,* vol. 2023, 2023.

[25] "Car Parts and Car Damages Dataset," Humans in the Loop, 2023. [Online]. Available: https://humansintheloop.org/resources/datasets/car-parts-and-car-damages-dataset/. [Accessed July 2023].

[26] P. Burlina, "MRCNN: A stateful Fast R-CNN," in *2016 23rd International Conference on Pattern Recognition (ICPR)*, Cancun, Mexico, 2016. doi:10.1109/ICPR.2016.7900179.

[27] D. Reis, J. Kupec, J. Hong and A. Daoudi, "Real-Time Flying Object Detection with YOLOv8," *arXiv:2305.09972 [cs.CV],* 2023.

[28] W. Yu, C. Si, M. L. Pan Zhou, Y. Zhou, J. Feng, S. Yan and X. Wang, "MetaFormer Baselines for Vision," *arXiv:2210.13452 [cs.CV],* 2022.

[29] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," *arXiv:1512.03385 [cs.CV],* 2015.

[30] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *arXiv:1409.1556 [cs.CV],* 2015.