CES: Cost Estimation System for Enhancing the Processing of Car Insurance Claims

Ahmed Shawky Elbhrawy *[a], Mohamed A. Belal [b], Mohamed Sameh Hassanein [c]

[a] *Business Information System Department, Faculty of Commerce and Business Administration, Helwan University, Cairo, Egypt*

[b] *Professor, Computer Science Department, Faculty of Computers and Artificial Intelligence, Helwan University, Cairo, Egypt*

[c] *Integrated Thebes Institutes for Computing & Management Science, Cairo, Egypt*

*Corresponding Author: Ahmed Shawky Elbhrawy [***Ahmed.Shawky21@commerce.helwan.edu.eg***]

ARTICLE DATA

Article history:
Received 06 Dec 2023
Revised 23 Jan 2024
Accepted 27 Jan 2024
Available online

Keywords:
Artificial Intelligence
Deep learning
Car Insurance
Cost estimation
CRISP-DM

ABSTRACT

Damage assessment is crucial in determining insurance reimbursements in the car insurance industry. However, manual inspection is time-consuming and financially costly. Artificial Intelligence (AI) offers a promising automatic damage assessment solution; we propose a Cost Estimation System (CES) for car damage volume level recognition and cost estimation. CES extracts damage estimates from mobile imagery data and combines them with structured customer data to generate accurate cost estimates for insurance purposes. This paper adopts the CRISP-DM (Cross Industry Standard Process for Data Mining) methodology to develop a robust and systematic model. Leveraging AI technology such as the (You Only Look Once) YOLO model and Transformers in image classification while expediting the claims process and mitigating fraud risk. Evaluating CES performance indicates the ability to accurately identify and locate damaged regions in car images, with an average precision of 78.50%, an average recall of 70.24%, and a mean Average Precision (mAP) of 0.66. Resulting in satisfactory performance from the curated dataset of 2508 car photos, which is classified by car body parts, and their inspected damage parts for enhancing cost estimation, productivity, accuracy, and time savings.

1. Introduction

The insurance sector is a crucial component of the Egyptian economy, contributing significantly to financial stability and risk mitigation for individuals and businesses. The diverse range of insurance offerings, including car insurance, has driven robust growth in this sector [1]. Auto insurance companies seek to enhance customer satisfaction through efficient claims processing, reducing service time and speeding damage repair quotes [2]. The traditional approach involves manual inspections by insurance adjusters, which leads to protracted periods for claim submission and its associated payment. This impediment, coupled with the issue of claims leakage, necessitates an automation approach.

Automation in companies involves reorganizing internal structures by assigning repetitive tasks, considered challenging and with minimal value, to computer systems rather than human resources. The aim is to enhance productivity by cutting costs and simplifying processes, enabling teams to dedicate more time to higher-value tasks. Over time, automation has progressed significantly, benefiting from the support of emerging technologies such as artificial intelligence [5] Digital transformation offers a promising solution by automating the claims process from inception, encompassing damaged vehicle component detection, severity assessment, and repair cost estimation based on images and structural data. Many research endeavors to optimize the auto insurance claims process, focusing on advancing customer satisfaction by expediting claim processing, minimizing service duration, and accelerating repair quotations for damaged parts. The primary objective of this paper is to facilitate the insurance claims workflow by implementing automated car damage detection, thereby facilitating insurance companies in rendering more informed pricing decisions and streamlining the claims processing timeline. Additionally, the study seeks to showcase the efficacy of artificial intelligence, specifically leveraging deep learning techniques and computer vision, in effectively addressing the challenges mentioned above.

This paper presents a complete solution for estimating vehicle damage severity using mobile phone imagery, websites, deep learning techniques, and computer vision. The proposed approach leverages deep learning models called transformers to detect damaged information from car image data, outperforming existing methods that rely solely on structured data. A computer vision model called YOLOv8 help in building a model for automating the process of detecting various car accident damage, classifying the damage, and determining the damage's severity. The two combined models' extracted damage features are also integrated with customer-provided structured data to generate reliable claim submissions and their associated payments for insurance purposes. The CES system identifies a damaged car through several panels that efficiently handle multiple images per claim, extracting and combining damage features from all images. Each panel employs semantic segmentation to analyze individual car components and evaluate the location and severity of damaged parts. The extracted and combined damage features are processed in a fusion module to generate an accurate estimate of the vehicle's damage cost along with structured data from the customer.

The main goals of this work are to demonstrate the potential of artificial intelligence by incorporating deep learning techniques and computer vision (a) to identify and locate automatically accurately damaged parts in cars through images in order (b) to enhance the insurance claims process through supporting insurance companies in making better pricing decisions and claims processing faster. To achieve these goals, (c) a CES is created to optimize the submitted claims with the minimum associated payment.

This paper is divided into five sections: Section 1 introduces the topic, Section 2 reviews related literature, Section 3 discusses applied methodology and describes the phases of the proposed solution in detail, Section 4 discusses solution implementation results, and Section 5 concludes the paper and discusses limitations and future research.

2. Related Work

Artificial intelligence addresses many problems, including detecting damage to the car's body after an accident and assessing the damage. This section will review some research in which researchers have devised methodologies or algorithms to deal with this problem. They aim to reduce the time needed for accident identification and repair costs, enhance customer satisfaction, and minimize the expenses associated with processing insurance claims.

Xin Zheng et al. [4]. They introduce BT-YOLO, a new automatic crack detection algorithm. BT-YOLO combines strengths from Transformer and YOLOv5 for better crack feature extraction, especially for small cracks. It also uses techniques such as DWConv and SimAM to reduce its number of parameters and improve processing speed. Tests using DWConv and SimAM tests show BT-YOLO is 4.5% more accurate and has a 24.9% parameter reduction.

Qianqian Zhu et al. [5]. created a computer vision-based intelligent vehicle damage assessment system with a primary objective to assess and pinpoint vehicle damage. Their system uses a method that recognizes the image's appearance through pixel-level object segmentation to evaluate and identify vehicle damage precisely. Their implemented object segmentation method is an extension of Faster R-CNN and Mask R-CNN that achieved an impressive accuracy of 87.3% for damage determination.

Singh et al. [6]. introduced a comprehensive system emphasizing automating the car damage assessment process. This system takes images of damaged cars as input, furnishing detailed insights into the impacted components and estimating the damage extent. The experimentation phase employed well-known instances segmentation models like Mask R-CNN and PANet; incorporating a VGG16 network based on transfer learning with these models. The results were noteworthy, demonstrating commendable mean Average Precision (mAP) scores of 0.38 for parts localization and 0.40 for damage localization.

Li et al. [7]. proposed a workflow for automated car damage detection and classification, utilizing the power of pre-trained deep learning models. They leverage Mask R-CNN to first locate and isolate vehicles within images, then employ various pre-trained CNN architectures, specifically focusing on VGG16, to identify and categorize bumper damage. Through extensive experiments utilizing transfer learning and a curated dataset of 864 vehicle images, They achieved impressive accuracy, with VGG16 demonstrating a top performance of 87.5%.

Gandhi [8]. Built an application that uses pre-trained models like VGG16 to check car pictures for damage. First, it uses ResNet50 to see if there's any damage at all. If there is, WPOD-Net finds the license plate, and YOLO shows exactly where the damage is on the car. Finally, DenseNet figures out how bad the damage is. He demonstrates his experiment that transfer learning, compared to fine-tuning, yielded better results. Regarding the proposed models, there are still overfitting issues in his application.

Gustian et al. [9]. introduce a methodology employing machine learning techniques to identify car body damage. Their approach involves gathering image datasets depicting cars in intact and damaged conditions, then preprocessing and segmentation to isolate damaged parts. Subsequently, a deep learning algorithm, either YOLO or Faster Region-based CNN, is employed to construct a detection model. Their model undergoes training and tuning using the collected data, followed by evaluation with test data to measure accuracy and precision in detection results. Their experimental findings demonstrate that the proposed method achieves notable accuracy and efficiency in detecting various car body defects, including scratches and cracks, with an average precision exceeding 70%.

Mallikarjuna and Arun Kumar [10]. They suggest an automatic system to assess the vehicle's damage and determine its seriousness. The model is trained using base weights from the COCO dataset and pre-trained CNN models to analyze the deep learning-based YOLOv5 target detection approach for identifying the damage within a car. The images are processed across 35–90 epochs; after processing, a color splash method is used in the final image to highlight the damaged area.

Thar, M, et al, [11] suggest an algorithm capable of assessing the extent of vehicle damage using CNN and YOLOv5 models. This algorithm comprises four models: a model for detecting vehicle body parts, a model for inspecting damage, a model for classifying damage, and a model for analyzing damage volume levels. It's important to note that the algorithm proposed in this study has not undergone implementation or testing to assess its performance.

Dwivedi, Mahavir, et al [12]. To automate the processing of auto insurance claims, they created and put into place a pipeline for categorizing and detecting auto damage. They employed models pre-trained on a large and varied dataset to prevent overfitting and learn more generic characteristics because their sample was rather tiny. They used preprocessing approaches and CNN models pretrained on the ImageNet dataset; they increased the system's performance and obtained an accuracy of 96.39%. Additionally, they employed YOLO to identify the damaged area, reaching a maximum map score of 77.78% on the held-out test set.

Zhu et al [13]. Proposed a deep learning approach for the accurate recognition of car appearance damage. Their study highlights the effectiveness and robustness of the Mask R-CNN model, particularly when based on KL-loss, in detecting vehicle damage. The researchers introduce an evaluation approach that utilizes component position instead of the traditional IOU calculation accuracy method, resulting in significantly more precise findings. Scrape, deformation, cracking, and rupture are the four primary forms of damage that may be identified using the model.

Wang, Niannian et al [14]. Proposed a Road-TransTrack tracking-counting network, which is a transformer optimization-based tracking model. They gathered road damage photos and divided them into potholes and cracks. This created a road damage dataset using a classification network built on YOLOv5. Then, the transformer and self-attention mechanism are added to the suggested tracking model to improve detection. Lastly, the model's efficacy and real road footage were used to evaluate the suggested tracking network with an accuracy of 98.59% and 91.60%, respectively.

3. Research Methodology and Steps

Previous studies have highlighted existing deep learning-based vehicle damage recognition techniques do not consider the damage volume level needed for auto insurance claims. Moreover, the various types of vehicle damage and the severity of the damage must be determined for practical applicants. These aspects [6]. To address this issue, the paper has built a prototype called CES (Cost estimation system) that determines the affected body parts, the degree of damage, and the total cost of the claim. The CRISP-DM methodology is adopted to develop the proposed prototype for detecting car damage. This methodology is used to ensure that the prototype is developed correctly and that it meets the needs of the business [15] FIGURE 1 discusses applied methodology in six phases, starting with business understanding, data understanding, data preparation, and Modeling, which will be explained in this section in detail. The remaining two phases, evaluation and deployment, will be explained in Section 5: Results and Discussion.

3.1 Business Understanding

This phase focuses on understanding business needs, which is then used to create an accurate detection prototype using deep learning techniques and computer vision to support insurance companies in making better pricing decisions and claims processing faster. This phase consists of the following two sub-phases, determining business objectives and deep learning goals as labeled phase A in FIGURE 1:

3.1.1 Determine business objective:

Classify historical information for deep learning (DL), such as insured data, vehicle data, damage reports, photographic evidence, repair estimates, and claims data, for a deeper understanding of different business information sources. DL approach provides a more detailed analysis of vehicle damage and its severity level. Providing substantial benefits throughout, automation instead of traditionally manual processes, and streamlining the claims process for insurance companies and their customers.

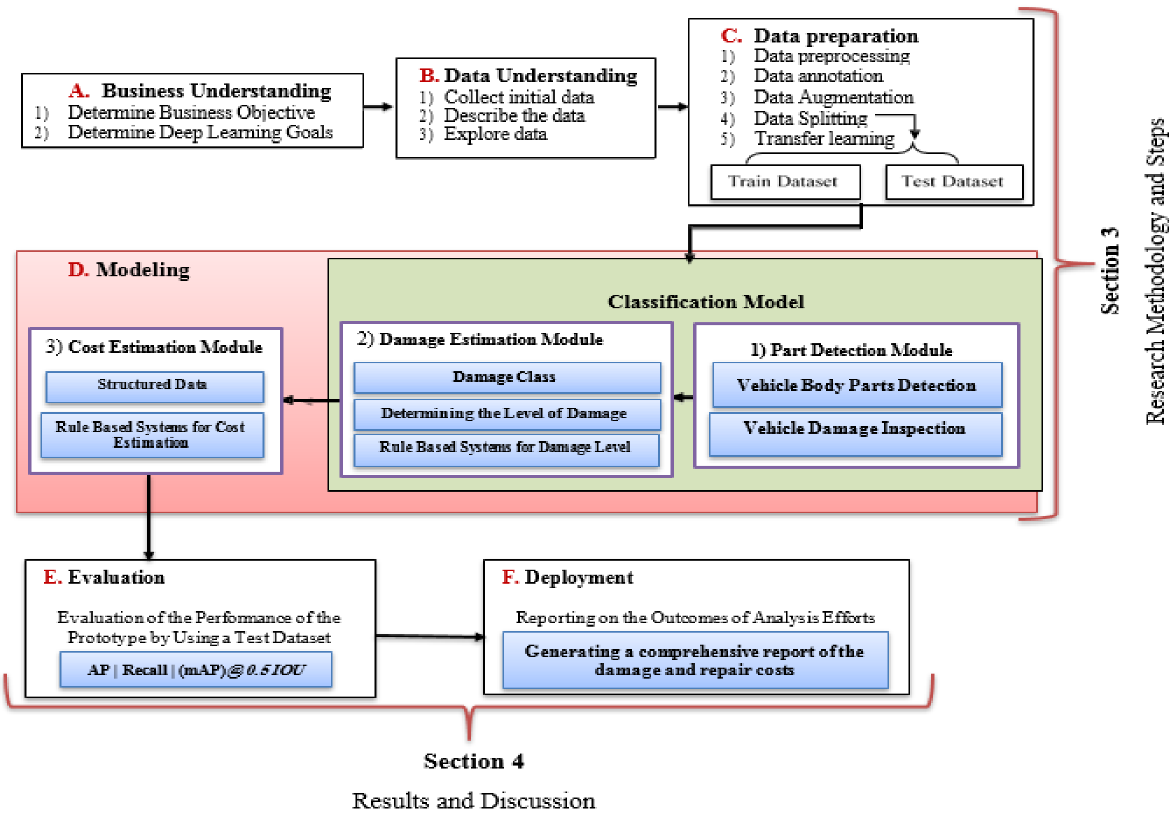

FIGURE 1. Cost Estimation System, Research Methodology and Phases

3.1.2 Determine Deep Learning Goals:

Computer vision and deep learning techniques in auto insurance can provide valuable information, such as detecting car damage from photos, to expedite the claim process [16]. A system illustrating the capabilities of computer vision and deep learning techniques for analyzing car damage estimation and assessing the severity of damage location by detecting an image of a car and its employed semantic segmentation of individual car parts. Multiple images per claim are handled by extracting and combining damage features from all images. These damage features and structured data from the customer, are processed within our system to generate an accurate estimate of the vehicle's damage cost to support insurance companies in making better pricing decisions and claims processing faster.

3.2 Data Understanding

This is a crucial initial phase of the car damage assessment process, focusing on gathering, analyzing, and comprehending the data that will be used for subsequent steps. It involves thoroughly examining the data sources, identifying and exploring the relationships between variables, and assessing the quality and consistency of the data [15]. This phase is composed of three main tasks. Collect initial data, describe the data, and explore the data, as labeled phase B in FIGURE 1.

3.2.1 Collect initial data:

Maintaining a balanced dataset was paramount to preventing bias in the deep learning algorithm, ensuring an equal representation of damaged vehicles across categories. Damage information was subsequently integrated with structured data provided by the customer through the Copart website (https://www.copart.com). Copart is an online marketplace showcasing damaged vehicles for sale, providing a rich data source for annotation. We constructed dataset categorization for vehicle damage and its severity level, so we manually annotated images retrieved from the website. We meticulously selected and categorized photos into eight commonly used categories representing distinct vehicle body parts and five types for damage assessment, as illustrated in TABLE 1. Despite its comprehensiveness, the classification task presented challenges due to the inherent similarities between different damage classes and the relatively small size of damage depicted in the images.

3.2.2 Describe the data:

Many types of damage can happen to a car's body, such as scratches, crashes, dents, shattering, cracks, etc. The damage types to be detected and the corresponding images must be carefully selected. The dataset includes eight types of vehicle body parts and five types of damage for inspection. The dataset comprises 2508 photos of damaged car areas, as shown in TABLE 1.

TABLE 1. Vehicle Body Parts, The Level of Damage, and Description of the Dataset

Vehicle Body Parts	Damage Inspection	Description of the Dataset		Level of Damage	
		Train Dataset size	Test Dataset size	Damage Level	Damage Classification
Bumper	Crash	140	28	High	3
	Dent	185	37	Medium	2
	Scratch	125	25	Low	1
Hood	Crash	155	31	High	3
	Dent	70	14	Medium	2
	Scratch	90	18	Low	1
Doors	Crash	115	23	High	3
	Dent	85	17	Medium	2
	Scratch	75	15	Low	1
Tailgate	Crash	175	35	High	3
	Dent	145	29	Medium	2
	Scratch	75	15	Low	1
Trunk	Crash	70	14	High	3
	Dent	85	17	Medium	2
	Scratch	55	11	Low	1
Mirrors	Shatter	75	15	High	2
	Cracks	50	10	Medium	1
Windshield	Shatter	85	17	High	2
	Cracks	95	19	Medium	1
Lights	Shatter	60	12	High	2
	Cracks	80	16	Medium	1

3.2.3 Explore data:

After detecting vehicle damage and its severity level between vehicle body parts and the damage inspection through preliminary examination and expert opinions, the damage level was classified into one of three possible categories of damage based on a preliminary examination and expert opinions made by the insurance company. These levels of damage are classified into low classifications of 1, medium classifications of 2, and high classifications of 3.

3.3 Data preparation

Data preparation is an indispensable step in evaluating car damage severity using image analysis. It comprises a sequence of meticulously executed sub-phases that transform raw image data into a format compatible with deep learning techniques and computer vision. This transformative process consists of five sub-phases: data preprocessing, data annotation, data augmentation, data splitting, and transfer learning. These sub-phases play crucial roles in preparing the data for subsequent analysis and Modeling, as labeled in phase C, FIGURE 1:

3.3.1 Data preprocessing:

It is a crucial step involving accurately preparing the raw image data to ensure its compatibility with deep learning models and enhance the accuracy of damage assessment; it encompasses a range of techniques, including resizing, cropping, removing outlier images and other artifacts, and normalizing images; so that they are all the same size and format for effective car damage assessment using deep learning techniques and computer vision [17].

3.3.2 Data annotation:

Is where we used an annotation tool to accurately detect and classify car damage using deep learning techniques and computer vision. It also involves meticulously tagging each image with relevant labels to differentiate between different types of damage and their severity levels so that algorithms can operate with the necessary information [18]. The labels can specify the exact type of damage, including dents, scratches, shattered glass, cracks, etc. Damage classifications are manually assigned using a bounding box for each damage category to ensure an accurate definition of damage on a specific region in an image. The annotation tool RoboFlow (https://app.roboflow.com) was employed to streamline this process.

3.3.3 Data Augmentation:

It is a method that may be used to create new, modified data from the original set to purposefully enhance the size of a training set [17]. To prevent overfitting, the training dataset appears too tiny, or you need to increase the efficiency of your model, using data augmentation is a great option. It may change the original data in several ways. Including core filtration, chromaticity adjustments, and form changes, we applied brightness between 25 and 30 percent, exposure between 20 and 30 percent, and horizontal flip to address some conflicts, such as overfitting, that may occur when one training dataset is on a smaller dataset by erasing randomly and increasing the image fusion dataset size.

3.3.4 Data Splitting:

It divides datasets into training and testing sets to evaluate the performance of our model accurately [19] , the dataset was randomly divided into 80% for training and 20% for testing. This is a common practice, ensuring the model is not overfitting the training data. This is overfitting when a model learns the training set too well and cannot generalize to new data.

3.3.5 Transfer learning:

It is a strategy used to tackle intricate issues quickly and with better performance throughout working with tiny, labeled datasets; the features needed for the specified job are extracted using a pre-trained model [17] , such as image classification as a starting point for the car damage severity assessment task. This can accelerate the training process and improve the model's performance. When the data preparation phases are completed, the dataset is ready to move on to the next phase, Modeling.

3.4 Modeling

The CES in this paper is a hybrid approach that uses image-based and structured data-based techniques. The model in the Modeling phases consists of three main modules: Part Detection Module, Damage Estimation Module, and Cost Estimation Module. The CES part detection model is implemented using a Deep learning model called transformers to detect car body damage in real efficiently. Damage Estimation Module is implemented using YOLOv8, which helped build a model for automating the process of detecting various car accident damage, classifying the damage, and determining the severity of the damage. The result of the Damage Estimation Module is used for the cost estimation module responsible for accruing claim amounts [20]. This phase of Modeling contains all these module steps, as labeled phase D in One that will be explained in this section:

3.4.1 Part Detection Module

An image training cloud-based web service called Colab, provided by Google, was used to speed up the image processing handling. Colab gives free, albeit limited, access to graphical processing units (GPUs), which can train YOLOv8 to detect various car accident damage and classify the damage's severity within a few hours [21]. A total of 2508 images for various car accident damage were processed; divided into 2090 images for training, and 418 images for model testing to accurately check our model's performance. The part detection module is separated into two parts:

a) Vehicle body parts:

It is a part of the Part Detection Module applying transformers in real time to detect efficiently the different parts of the vehicle that can be damaged in an accident, and we utilized the ability of the transformer to recognize the damage to the vehicle body parts images. However, it is primarily used to recognize texts [22]. The Vehicle body parts predefined in TABLE 1, such as the hood, the bumper, the windshield, and the doors, are Vehicle body parts we only processed.

b) Damage inspection:

The damage inspection uses a deep learning model called YOLOv8 to extract features from images of damaged cars, such as scratches, crashes, dents, and shattering predefined in TABLE 1 column damage inspection. In other words, identify the severity of the damage after the extraction performed by the transformer that identifies the different body parts of the car that have been damaged. YOLO is a highly accurate deep-learning model that can detect objects in images and videos [20] [23] . It divides the image into a grid of cells and predicts the object class and location within each cell. OpenCV is a library of programming functions utilized in YOLO to use bounding box regression to locate objects within each box precisely. Equation (1) is used for The probability of an object class, and its location in each grid cell is calculated [10].

$$\text{Probability score} = P(\text{class}) \times \text{IOU (ground truth, predicted box)} \qquad (1)$$

In this equation:

P(class) represents the probability that the object in the cell belongs to the specified class.

IOU refers to the intersection over the union of the ground truth box and the predicted box.

The grid cell with the highest probability score will likely contain an object. To address the issue of duplicate object detection, YOLO incorporates a technique called Non-Maximum Suppression (NMS). NMS is a post-processing step that eliminates overlapping bounding boxes, retaining only the most confident detections. The process involves sorting the bounding boxes based on their confidence scores. The boxes with the highest confidence scores are retained, while the rest are discarded. The remaining boxes are then checked for overlap; if two boxes overlap, the one with the lower. The confidence score is discarded. This process continues until only one bounding box remains for each object. YOLO also utilizes complex CNNs and real-time spatial analysis techniques to achieve high accuracy in object detection in images and videos. YOLO was used to train models using annotated datasets to identify and differentiate between damage to the car body and normal conditions during this stage

FIGURE 2 shows the system's experience in detecting damage to the car body. We must first determine the damaged side of the car body, then upload the image, then determine the number of working hours, the hourly wage, and the cost of changing the damaged part if it is not repairable, and then calculate the total cost. The link to the system (https://caraccident.streamlit.app/).

3.4.2 Damage Estimation Module:

The damage estimation module uses the features extracted by the part detection module to estimate the severity of the damage. Rule-based systems are commonly used for classifying damage; these systems rely on a set of predefined rules to determine the severity or category of the damage based on certain criteria [20] [24]. Applying these rules makes it possible to automatically classify and assess different types of damage consistently and efficiently. The use of rule-based systems in damage classification provides a reliable and standardized approach to assessing the severity or category of damage. The function will return the corresponding level of damage ("Low," "Medium," "High") for a certain vehicle part based on the predefined rules, as shown in TABLE 1 in damage level and vehicle body part. The damage estimation module is separated into two parts:

a) Damage class:

The damage class is the category of damage that has been identified. For example, the damage class might be "dent Hood" or "shatter mirrors", " Crash Doors," and so on for the other vehicle body parts. Damage classifications are manually assigned using a bounding box for each damage category to ensure an accurate definition of damage on a specific region in an image in phase data annotation to prepare the image to other phase damage inspection.

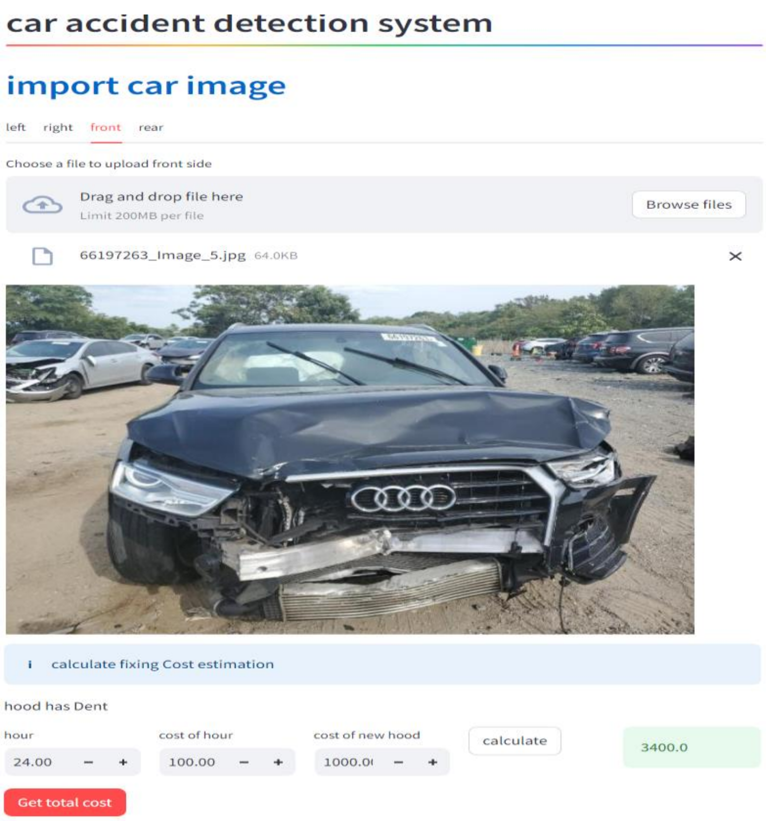

FIGURE 2. Cost Estimation System

b) Determining the level of damage:

The number of possible cases in a car with five types of accidents (crash, dent, scratch, shatter, and cracks) and eight parts of the car (bumper, hood, doors, tailgate, trunk, mirrors, windscreen, and lights). By considering the combinations of accidents and affected car parts, it is determined that there can be 40 cases. As shown in **Error! Reference source not found.**, only five cases are presented for example, and the level of damage is determined according to the opinions of experts as follows:

- Crash: major damage that requires significant repairs, such as replacement of the damaged part.

- Dent: moderate damage that can be repaired but may require cosmetic work.
- Scratch: minor damage that can be repaired easily.

The function of determining the level of damage is to analyze the vehicle body parts, damage inspection, and damage classification to determine the damage. Determining the level of damage is defined as levels 1, 2, and 3.

TABLE 2.Vehicle body parts and determining the level of damage.

Vehicle No.	Vehicle body parts	The damage inspection	The damage level	Determining The Level of Damage
A	Bumper	Crash	High	3
	Hood	Dent	Medium	
B	Bumper	Crash	High	3
	Hood	Crash	High	
C	Bumper	Dent	Medium	2
	Hood	Scratch	Low	
D	Bumper	Scratch	Low	2
	Hood	Scratch	Low	
E	Bumper	Scratch	Low	1

Here are the explanations for five cases in TABLE 2 out of 40 possible cases that could occur:

A. If the damage inspection of " crash " and "dent" is detected on the vehicle body parts, the level of damage to be level 3.
B. If two " crash " damage inspections are detected on the vehicle body parts, the level of damage to be level 3.
C. If the damage inspections of "dent" and "scratch" are detected on the vehicle body part, determining the level of damage be level 2.
D. If two "scratch" damage inspections are detected on the vehicle body part, the level of damage to be level 2.
E. If only a "scratch" is detected on the vehicle body part, the level of damage to be level 1.

As described above, the results of determining the level of damage vary according to the damage inspection and level, which is used as an input for the Cost Estimation Module to determine the amount of the claim.

3.4.3 Cost Estimation Module:

This module combines the damage estimates from the damage estimation module with the structured data provided by the customer, such as vehicle type, vehicle make, vehicle model, and year of manufacture, to generate a final and accurate cost estimate for insurance purposes. to implement that we use rule-based systems to improve the accuracy of cost estimation, rule-based systems use a set of predefined rules and conditions to make decisions or calculate costs based on given inputs.

The cost of parts required for the repair is calculated based on the prices of individual components. Cost of Parts (Cp) calculated according to this equation:

$$Cp = \Sigma \text{ (price of each part)} \qquad (2)$$

The labor cost is typically calculated based on the hourly rate (R) of the labor and the estimated time required to complete the repair (T). The cost of labor (Cl) is calculated according to this equation:

$$Cl = R \times T \qquad (3)$$

The overhead cost includes expenses associated with operating the repair shop. It can be estimated as a percentage (O) of the labor cost, the cost of overhead (Co) calculated according to this equation:

$$Co = O \times Cl \qquad (4)$$

The profit cost represents the amount the repair shop expects to make on the repair. It is typically calculated as a percentage (P) of the total cost of parts and labor, and the cost of profit (Cpr) is calculated according to this equation:

$$Cpr = P \times (Cp + Cl + Co) \qquad (5)$$

The total cost of repair is the sum of the cost of parts, labor cost, overhead cost, and cost of profit; Total Cost of Repair (Ct) is calculated according to this equation:

$$Ct = Cp + Cl + Co + Cpr \qquad (6)$$

These equations are used to estimate the cost of repairing car parts, which is calculated in (2). However, the actual cost of the repair may vary depending on several factors, such as the availability of parts, the mechanic's skill, and the repair's complexity. In addition to these equations, insurance companies and repair shops may also use other factors to estimate the cost of repairing car parts. These factors may include the car's age, the car's mileage, and the car's make and model. These factors can be determined by providing structured data about the insured and the car. In this work, the paper uses those (3,4,5,6) equations to calculate a total cost estimation that has proven its efficiency in practice.

Using these calculations, we can effectively incorporate various factors, such as damage value, make, model, and vehicle type, into the cost estimation process. This can help us better understand and predict actual outcomes by comparing predicted values with observed results for average losses per claim and total claim amounts.

4. Results and Discussion

In this section, we explain in detail the phases of evaluation and deployment to complete the CRISP-DM methodology for the CES system, as labeled phases E and F in FIGURE 1; the car damage detection model was evaluated using four metrics: average precision (AP), recall, mean average precision (mAP)@0.5 IOU, and training time (mins).AP measures the precision and recall of a model at different thresholds. It is calculated by taking the average of the precision values at each recall level. AP is a more comprehensive metric than precision or recall alone. Recall is the percentage of damaged areas correctly detected by the model.mAP@0.5 IOU is the average precision of a model on damaged areas with an intersection over union (IOU) of at least 0.5. Training time is the time it takes to train the model on the training dataset. These metrics are important for evaluating the performance of a car damage detection model [25].

4.1 Evaluation

The training process of deep learning models involves optimizing a set of parameters to achieve the desired performance. Four crucial parameters played a pivotal role in this process, as shown in Table 3 in the column parameters: number of epochs, batch size, learning rate, and momentum. The number of epochs determined the frequency with which the entire training dataset was passed through the model, balancing overfitting and under-fitting. The batch size governed how the model's weights were modified after each training data batch, influencing training speed and stability. The learning rate dictated the magnitude of adjustments made to the model's parameters during training, affecting convergence and performance. Momentum facilitated smoother training by incorporating the model's weight gradient history, potentially preventing local minima and slowing down training. The optimal values for these parameters depend on the specific task and dataset characteristics, requiring experimentation to determine the most suitable configuration for a particular application [25].

TABLE 3. Experimentation Results.

Experimentation	Parameters		Average Precision (AP)	Recall	mean Average Precision (mAP)@0.5 IOU	Training Time (mins)
1	Number of iterations Batch Learning Rate Momentum	epoch:400 44 0.0013 0.74	0.6604	0.6955	0.702	15
2	Number of iterations Batch Learning Rate Momentum	epoch:600 64 0.0013 0.949	0.7850	0.7024	0.66	17

The experimentation results are shown in Table 3. The first experiment utilized specific parameter settings, including 400 iterations (epochs), a batch size of 44, a learning rate of 0.0013, and a momentum value of 0.74. The first experiment's results revealed an average precision (AP) of 0.6604, indicating the model's ability to identify and locate damaged regions in car images accurately. The recall value, which measured the proportion of correctly detected damage, was 0.6955. The mean average precision (mAP) at an intersection over a union (IOU) threshold of 0.5 was calculated as 0.702, demonstrating the model's satisfactory performance in detecting and localizing car damage. The training process took approximately 15 minutes to complete.

The parameter settings were adjusted in the second experiment to optimize the performance. The model was trained for 600 iterations (epochs) with a batch size 64. The learning rate was set at 0.0013, and the momentum value was increased to 0.949. The results of the second experiment showed significant improvements in the model's performance. The average precision (AP) reached an impressive value of 0.7850, indicating the model's enhanced accuracy in identifying and localizing car damage. The recall value was measured at 0.7024, reflecting the model's ability to detect a significant portion of the damaged areas successfully. The mean average precision (mAP) at an intersection over union (IOU) threshold of 0.5 was calculated as 0.66, signifying the model's robustness in identifying flaws in automobile photos. The training process for the refined model needed approximately 17 minutes. These outcomes highlighted the positive impact of adjusting the parameter settings, resulting in a more precise and effective model for car damage detection and severity estimation generally. On the other hand, utilizing deep learning techniques such as YOLO and a larger and more varied dataset could also enhance the effectiveness of damage identification on automobiles.

The system achieved a good percentage of average precision (AP), recall, and mean average precision (mAP). This could be increased by using a larger dataset. Additionally, we were testing the ability of the Hugging Face transformer to recognize images, which is primarily used for recognizing text.

4.2 Deployment

In the CRISP-DM (Cross-Industry Standard Process for Data Mining) methodology, the deployment phase focused on implementing and integrating the developed solution into the operational environment [26]. Roboflow, Google Colab, and OpenCV were all technologies used for deep learning and computer vision of car damage detection. Roboflow was the platform to create and train custom machine-learning models for computer vision tasks. The platform provided a variety of features, including image annotation, model training, and deployment. Roboflow was used to annotate the dataset of damaged car images and train the model on the dataset. Google Colab was a cloud-based Jupiter Notebook environment that provided access to GPUs for machine learning tasks. Google Colab is a good platform for training car damage detection models, as GPUs could significantly accelerate the training process. OpenCV was a free and open-source computer vision library used for various tasks, including image processing, object detection, and image classification. OpenCV is used to implement a car damage detection model by using its object detection algorithms to detect damaged areas in images.

Here is an example of how Roboflow, Google Colab, and OpenCV can be used together to train a car damage detection model:

- Collect a dataset of images of damaged cars. This step was utilized in the main phase B of data understanding.
- RoboFlow was used to annotate the dataset, labeling the damaged areas in each image. This step was utilized in the main phase C of data preparation in sub-phases of data annotation.
- Train a car damage detection model on the annotated dataset using RoboFlow. Deploy the trained model to Google Colab. In addition, OpenCV was used to implement the car damage detection model on Google Colab. This step was utilized in the main phase D of Modeling.
- Once the model was deployed to Google Colab, it detected damaged cars in new images. This step was utilized in the main phase F of Deployment This was done by uploading an image to Google Colab and then using OpenCV to run the car damage detection model on the image.

Roboflow, Google Colab, and OpenCV were powerful tools for developing effective car damage detection models.

5. Conclusion and Future Work

The model's accuracy in detecting damage to the car body through images was an important factor, as it achieved 78.50% correct detection. This represented a support for the theory. Using deep learning techniques and computer vision could revolutionize the insurance of claims process. By automating the detection of car damage and the analysis of claims data; insurers could speed up the claims process, reduce the risk of fraud, and make better pricing decisions. The development of such systems was still in its early stages, but the potential benefits were significant. As the technology continues to develop, we expect to see even more innovative computer vision applications and deep learning in the insurance industry.

Here are some specific benefits of how our model was used to improve the insurance claims process: Automated car damage detection: This is used to speed up the claims process and reduce the risk of fraud. Pricing decisions: This is used to predict the likelihood of a claim being filed, the severity of the damage, and the cost of repairs. This information was used to set fair premiums for insurers and policyholders. Claims processing: This is used to automate the analysis of claims data and free up employees to focus on other tasks.

5.1 Research Limitations and Future Work

Implementing the Cost Estimation System has encountered several limitations due to image quality and quantity used for deep learning systems, especially in differentiating the damage levels and severity classifications process. The image quality depends on many metrics, such as resizing, cropping, and removing outliers affecting system accuracy. The quantity of image quality had to be maintained to have a balanced dataset and prevent overfitted bias percentage. Also, the images used had to be in a specific size, resolution, and body style (hatchbacks, sedans, SUVs, and vans) to facilitate the recognition of damaged appearance in various environmental conditions (rain, snow, dark, and strong light environments). Therefore, any deviations in damage recognition and assessment may result inaccurate dataset for the training process, leading to a challenge in recognizing the damage levels and severity classifications. The challenge in damage recognition affects the accuracy rate for damaged components in the vehicle body due to the inherent similarities and multi-categories in the scope of the image with mixed or complex damage. Consequently, the quantity of the dataset with a decent image quality can compromise the efficiency of our system in estimating the cost claim.

The future work of this research will focus on the possibility of applying this framework to different types of vehicles. It is also possible to use different learning algorithms and compare their results to determine the best algorithm for deep learning classification. Additionally, due to the small number of images related to the research topic, insurance companies and repair centers must share collected images in an indexed way to benefit from them in future scientific research.

References

[1] FRA, "FRA ANNUAL REPORT 2022," Financial Regulatory Authority (FRA), Cairo,Egypt, 2022. [Online]. Available: https://fra.gov.eg/wp-content/uploads/2023/05/Report2022-2.pdf. [Accessed 1 10 2023].

[2] H. Johnson, "investopedia.com," Investopedia, 19 Apr 2023. [Online]. Available: https://www.investopedia.com/auto-insurance-methodology-5219007. [Accessed 1 10 2023].

[3] Elhajjar, S., Yacoub, L., & Yaacoub, H. (2023). Automation in business research: systematic literature review. *Information Systems and e-Business Management*, 21(3), 675-698.

[4] Zheng, X., Qian, S., Wei, S., Zhou, S., & Hou, Y. (2023). The combination of transformer and you only look once for automatic concrete pavement crack detection. *Applied Sciences*, 13(16), 9211. https://doi.org/10.3390/app13169211.

[5] Qianqian, Z., Weiming, G., Ying, S., & Zihao, Z. (2020, April). Research on intelligent vehicle damage assessment system based on computer vision. *In Journal of Physics: Conference Series* (Vol. 1518, No. 1, p. 012050). IOP Publishing.doi:10.1088/1742-6596/1518/1/012050.

[6] Singh, R., Ayyar, M. P., Pavan, T. V. S., Gosain, S., & Shah, R. R. (2019, September). Automating car insurance claims using deep learning techniques. *In 2019 IEEE Fifth International Conference on Multimedia Big Data (BigMM)* (pp. 199-207). IEEE. doi: 10.1109/BigMM.2019.00-25.

[7] Li, L., Ono, K., & Ngan, C. K. (2021, April). A deep learning and transfer learning approach for vehicle damage detection. In *The International FLAIRS Conference Proceedings* (Vol. 34). https://doi.org/10.32473/flairs.v34i1.128473.

[8] Gandhi, R. (2021). Deep Learning Based Car Damage Detection, Classification and Severity. *International Journal*, 10(5). https://doi.org/10.30534/ijatcse/2021/031052021.

[9] Gustian, Y. W., Rahman, B., Hindarto, D., & Wedha, A. B. P. B. (2023). Detects Damage Car Body using YOLO Deep Learning Algorithm. *Sinkron: jurnal dan penelitian teknik informatika*, 8(2), 1153-1165. 10.33395/sinkron.v8i2.12394.

[10] Mallikarjuna & Arun Kumar .(2022).Vehicle Damage Detection and Classification Using Image Processing. *International Journal of Advanced Research in Science, Communication and Technology 2(9)*, pp. 568-574.

[11] Thar, M. H., Zhang, M., Nam, G. H., & Lee, D. M. (2022, November). Vehicle Damage Volume Level Recognition Algorithm based on Convolution Neural Network and Yolo Model. In Proc. *2022 International Conference on Communication and Computer Research (ICCR 2022), KICS (Korea Information and Communications Society), UCSY (University of Computer Studies, Yangon), Myanmar, ETRI (Electronics & Telecommunications Research Institute)* (pp. 21-22).

[12] Dwivedi, M., Malik, H. S., Omkar, S. N., Monis, E. B., Khanna, B., Samal, S. R., ... & Rathi, A. (2021). Deep learning-based car damage classification and detection. In *Advances in Artificial Intelligence and Data Engineering: Select Proceedings of AIDE 2019* (pp. 207-221). Springer Singapore.

[13] Zhu, Q., Hu, W., Liu, Y., & Zhao, Z. (2021, April). Research on Vehicle Appearance Damage Recognition Based on Deep Learning. In *Journal of Physics: Conference Series* (Vol. 1880, No. 1, p. 012024). *IOP Publishing*. DOI 10.1088/1742-6596/1880/1/012024.

[14] Wang, N., Shang, L., & Song, X. (2023). A Transformer-Optimized Deep Learning Network for Road Damage Detection and Tracking. Sensors, 23(17), 7395.

[15] Rakshata, P., Padma, H. V., Pooja, M., Yashaswini, H. V., & Karthik, V. (2019). "Car Damage Detection and Analysis Using Deep Learning Algorithm For Automotive," *International Journal of Scientific Research & Engineering Trends,* vol. 5, no. 6, pp. 1896-1898.

[16] Martínez-Plumed, F., Contreras-Ochando, L., Ferri, C., Hernández-Orallo, J., Kull, M., Lachiche, N., ... & Flach, P. (2019). CRISP-DM twenty years later: From data mining processes to data science trajectories. *IEEE Transactions on Knowledge and Data Engineering*, 33(8), 3048-3061. DOI: 10.1109/TKDE.2019.2962680.

[17] Dorathi Jayaseeli, J. D., Jayaraj, G. K., Kanakarajan, M., & Malathi, D. (2021). Car Damage Detection and Cost Evaluation Using MASK R-CNN. In *Intelligent Computing and Innovation on Data Science: Proceedings of ICTIDS* 2021 (pp. 279-288). *Springer Singapore*. https://doi.org/10.1007/978-981-16-3153-5_31.

[18] Maharana, K., Mondal, S., & Nemade, B. (2022). A review: Data pre-processing and data augmentation techniques. *Global Transitions Proceedings*, 3(1), 91-99. https://doi.org/10.1016/j.gltp.2022.04.020.

[19] Willemink, M. J., Koszek, W. A., Hardell, C., Wu, J., Fleischmann, D., Harvey, H., ... & Lungren, M. P. (2020). Preparing medical imaging data for machine learning. *Radiology*, 295(1), 4-15. https://doi.org/10.1148/radiol.2020192224.

[20] Wang, X., Li, W., & Wu, Z. (2023). CarDD: A New Dataset for Vision-Based Car Damage Detection. *IEEE Transactions on Intelligent Transportation Systems*. DOI: 10.1109/TITS.2023.3258480.

[21] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779-788).

[22] Google, "Welcome to colaboratory," 5 October 2023. [Online]. Available: https://colab.research.google.com/?utm_source=scs-index#. [Accessed 19 October 2023].

[23] Bazi, Y., Bashmal, L., Rahhal, M. M. A., Dayil, R. A., & Ajlan, N. A. (2021). Vision transformers for remote sensing image classification. *Remote Sensing*, 13(3), 516. https://doi.org/10.3390/rs13030516.

[24] Slimani, H., El Mhamdi, J., & Jilbab, A. (2023). Artificial Intelligence-based Detection of Fava Bean Rust Disease in Agricultural Settings: An Innovative Approach. *International Journal of Advanced Computer Science and Applications*, 14(6). DOI: 10.14569/IJACSA.2023.0140614.

[25] Guo, J., Wang, Q., & Li, Y. (2021). Evaluation-oriented façade defects detection using rule-based deep learning method. *Automation in Construction*, 131, 103910. https://doi.org/10.1016/j.autcon.2021.103910.

[26] Sirisha, U., Praveen, S. P., Srinivasu, P. N., Barsocchi, P., & Bhoi, A. K. (2023). Statistical analysis of design aspects of various YOLO-based deep learning models for object detection. *International Journal of Computational Intelligence Systems*, 16(1), 126. https://doi.org/10.1007/s44196-023-00302-w.