Journal of Positive Sciences

Journal homepage: https://positive-sciences.com
ISSN: 2582-9351

Car Damage Detection using Computer Vision

Priti Warungse[1], Atharva Kasar[2], Apurva Khirsagar[3], Geeta Hade[4], Nishant Khandhar[5]

[1]*Assistant Professor, Department of Computer Engineering, NBNSTIC, Pune, India*

[2,3,4,5]*Student, Department of Computer Engineering, NBNSTIC, Pune, India*

ARTICLE INFO

Article history:
Received
Received in revised form
Accepted
Available online

Keywords:
Image Processing
Logistic Regression
Feature Extraction
VGG
CNN

ABSTRACT

This research paper introduces an extensive framework for the detection of car damage through the application of deep learning methodologies. The proposed system seeks to full-fill the crucial demand for accurate, efficient, and automated techniques in assessing vehicle damage, offering potential applications in insurance claims processing, vehicle maintenance, and accident analysis. In our devised solution, we have integrated two convolutional neural network (CNN) models to address the task at hand. VGG16 is employed for the identification of car damage, determining the damage's location, and assessing its severity. Concurrently, Mask RCNN is utilized to precisely mask out the damaged region. The combined output from both models provides a comprehensive understanding of the damage inflicted on the car. The deep learning model undergoes training on a meticulously curated dataset of annotated vehicle images, highlighting the presence and severity of damage. Rigorous testing and validation procedures are employed to evaluate the system's accuracy, precision, recall, and F1-score. The resultant system showcases the capacity to significantly streamline the processes of car damage assessment, diminish human error, and expedite the settlement of insurance claims.

1. Introduction

1.1 Background

The automotive sector has made significant advancements in recent years, introducing innovations such as electric vehicles and autonomous driving technology. Amid these developments, safety and condition assessment have remained pivotal concerns. The demand for precise and efficient car damage detection systems has been fueled by car accidents, wear and tear, and vehicular maintenance. In response to this heightened need, computer vision technology has emerged as a groundbreaking solution. This research paper explores the area of "Car Damage Detection using Computer Vision" to explore the state-of-the-art techniques, challenges, and real-world applications that utilizing artificial intelligence and image processing to evaluate, categorize, and pinpoint car damages. With a focus on enhancing safety, facilitating insurance claims, supporting pre-purchase inspections, and streamlining fleet management, this paper highlights the multidimensional impact of computer vision in the automotive sector. The research discusses the underlying technologies, methodologies, and the potential for wider adoption across various automotive stakeholders, setting the stage for a comprehensive exploration of this cutting-edge field. The proposed framework can predict the severity of vehicle damage, distinguishing between minor and major incidents. The proposed system is based on the machine learning algorithm as it evolving technology in artificially intelligent systems. Assessing Car Damage with Convolutional Neural Networks, while also employing keywords like "bumper dent" and "door dent". This paper applies Convolutional Neural Networks (CNNs) to damaged car images to assess the extent of damage - they use transfer learning to evaluate the merits of object recognition models that are available.

1.2 Objectives

The primary objectives of this study is to:

Develop a robust machine learning model for accurate detection of vehicle damages. Gather and annotate a diverse dataset of vehicle images to train and validate the model. Integrate advanced computer vision techniques for efficient image analysis. Optimize the model to enable real-time processing and immediate damage assessment. Enhance model accuracy through continuous fine-tuning and feedback loops. Integrate the system seamlessly with existing processes and workflows. Design an intuitive user interface for easy interaction and result interpretation

2. Methodology

2.1 Module 1: Car or not

1. Setup and Dependencies
    - Libraries such as NumPy for numerical operations and JSON for file handling were imported.
    - Keras, a wrapper built on TensorFlow, was utilized to simplify the complex process of defining neural network graphs and layers.
    - The VGG16 model, pre-trained on ImageNet, was imported to leverage transfer learning.
    - Various image preprocessing packages were imported to prepare images for input to the VGG16 model.

2. Model Preparation
    - The VGG16 model, with weights from ImageNet, was downloaded and stored.
    - The model was saved as 'vgg16.h5' for future use.
    - A function, **prepare_image()**, was created to preprocess images for compatibility with the VGG16 model.

3. Prediction and Analysis
    - Images were processed using the prepare_image function, then passed to the VGG16 model for predictions.
    - Predictions were generated as an array of 1000 values, representing probabilities for ImageNet categories.
    - A function, **get_prediction()**, was developed to extract the top 5 predictions and their associated probabilities from the ImageNet class index.

4. Car Identification
    - The **get_car_categories()** function was designed to process images of cars, extract labels, and obtain predictions.
    - The most common 27 categories related to cars were stored in **cat_list**.

- A function was implemented to determine whether a given image depicts a car by comparing its predicted labels with the valid car category list.

5. Implementation Details
   - The Counter and defaultdict from the collection's module were employed for efficient counting and storage of data.
   - The os module was utilized for directory management.

6. Evaluation
   - The methodology facilitated the extraction of meaningful predictions from the VGG16 model, particularly in identifying images related to cars.
   - The accuracy and efficiency of the model in distinguishing car images were evaluated through extensive testing and comparison with ground truth labels

2.2 Module 2: Damaged or Not

1. Data Collection and Pre-processing:

The dataset comprises images of damaged and undamaged cars sourced from specific directories. Using the Python glob module, we traversed through directories to extract image paths. Each image underwent pre-processing using Keras's utilities to resize them to 224x224 pixels and apply suitable pre-processing functions.

2. Feature Extraction:

Transfer learning with the VGG16 architecture was employed to extract high-level features from pre-processed images. By removing fully connected layers and retaining convolutional layers, we obtained feature representations for each image, which were then flattened and stored for analysis.

3. Model Training and Evaluation:

We utilized a logistic regression classifier for classification, with extracted features as input. The dataset was split into 80% training and 20% testing sets using a stratified approach. The model's performance was evaluated using accuracy and visualized through a confusion matrix.
The logistic regression classifier achieved 95.65% accuracy on the test set. The confusion matrix showed 368 true positives, 16 false positives, 18 false negatives, and 368 true negatives. The precision of the classifier was 92.00%.

4. Model Deployment and Inference:

The trained classifier was serialized using Python's 'pickle' module for portability. Code snippets were provided to load the classifier and pre-trained VGG16 model for feature extraction. Given a new image, the model can accurately predict whether the car is damaged or undamaged

2.3 Module 3: – Location

1. Setup and Dependencies:
   - Libraries like NumPy, JSON, and Keras were imported for numerical operations, file handling, and neural network construction.

- Image pre-processing tools were used to prepare images for compatibility with the VGG16 model.

2. Model Preparation:
   - The pre-trained VGG16 model with ImageNet weights was downloaded and saved for future use.
   - Images were pre-processed using the prepare_image() function to meet VGG16 input requirements.

3. Prediction and Analysis:
   - Images were fed into the VGG16 model for predictions, generating probabilities across ImageNet categories.
   - Top 5 predictions and their probabilities were extracted using the get_prediction() function.

4. Location Identification:
   - A function, get_location_categories(), was created to process images depicting different car locations.
   - Predicted labels were compared with predefined location categories to identify specific car locations.

5. Training, Validation, and Testing:
   - Dataset was split into training, validation, and testing sets for model training, monitoring, and evaluation.
   - Model performance was assessed on the testing set, achieving an accuracy of 69%.

6. Implementation Details:
   - Counter and defaultdict from the collections module were used for efficient data handling.
   - The os module aided in directory management tasks during data processing.

7. Evaluation:
   - The model's ability to identify car locations was evaluated through rigorous testing, achieving a 69% accuracy rate.

2.4 Module 4 – Severity

1. Data Processing:

Images are loaded, resized, and converted to NumPy arrays. Features are extracted using a pre-trained model, and labels are encoded numerically.

2. Data Saving:

Extracted features and encoded labels are saved to HDF5 files. Model architecture is serialized to JSON, and weights are saved to an HDF5 file. Process completion is logged.

3. Data Loading:

Features and labels are loaded from HDF5 files and converted to NumPy arrays.

4. Data Verification:
   Shapes of loaded features and labels are verified to ensure consistency.

5. Data Splitting:
   Data is split into training and testing sets using **train_test_split**.

6. Model Training:
   A logistic regression classifier is trained on the training data.

7. Model Evaluation:
   Classifier accuracy is evaluated using a confusion matrix, providing insights into true positives, false positives, and false negatives for each class.

8. Result Analysis:
   The model achieves a 65% overall accuracy, demonstrating its ability to categorize car damage severity as minor, moderate, or severe.

3. Results

The following section presents the results and experiments conducted for the proposed system, including the use of confusion matrices, precision, and recall tables. Confusion matrices are utilized to evaluate the performance of the system in detecting whether an image contains car damage or not. For the damage detection, we got 92% Precision, 90.04% Recall as well as 90.07% F1-score.

- **Confusion Matrix –** Damage Detection

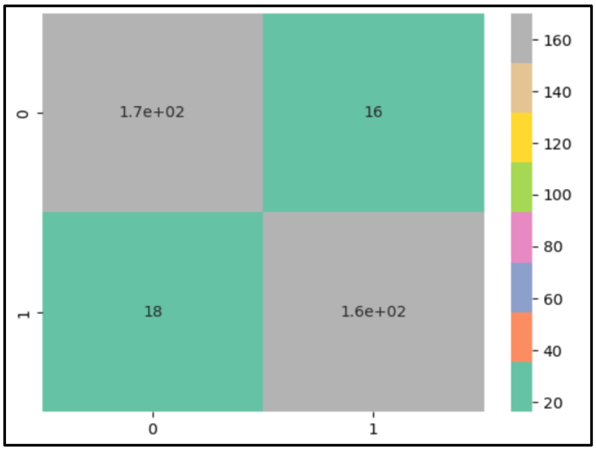

Fig 1. Damage Detection Confusion Matrix

Table 1. The precision, recall, and f-measure for Damage Detection

	Precision	Recall	F1-Score	Support
0	0.914	0.904	0.909	170
1	0.901	0.911	0.906	164
Accuracy	0.91	0.91	0.91	334

- **Confusion Matrix –** Location Detection

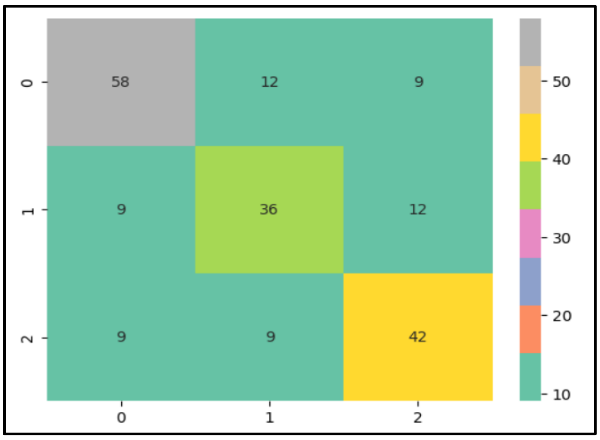

Fig 2. Location Detection Confusion Matrix

Table 2. The precision, recall, and f-measure for Location Detection

	Precision	Recall	F1-Score	Support
0	0.76	0.71	0.73	0.79
1	0.67	0.63	0.65	57
2	0.70	0.70	0.70	60
Accuracy	0.71	0.68	0.69	196

- **Confusion Matrix –** Severity Detection

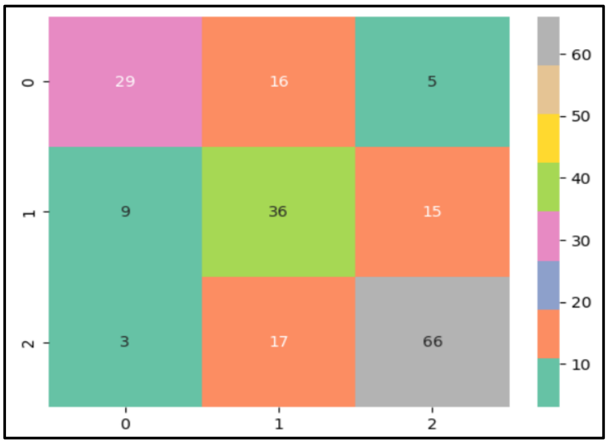

Fig 3. Severity Detection Confusion Matrix

Table 3. The precision, recall, and f-measure for Severity Detection

	Precision	Recall	F1-Score	Support
0	0.69	0.58	0.63	50
1	0.55	0.60	0.58	60
2	0.76	0.76	0.76	86
Accuracy	0.66	0.65	0.65	196

4. Conclusions

The Car Damage Detection project represents a significant advancement in leveraging computer vision and machine learning techniques for automating the process of car damage assessment. By combining powerful models like VGG16 and logistic regression classifiers, the system effectively identifies, evaluates, and categorizes car damage with high accuracy and reliability. Through rigorous checks and validations, the system ensures the integrity and precision of its assessments, providing users with a trustworthy tool for quickly and accurately determining the extent of damage to their vehicles. This not only saves time but also reduces the potential for human error inherent in manual inspection processes. Overall, the Car Damage Detection project represents a valuable contribution to the field of computer vision and sets the stage for future innovations in automated image-based assessments.

Acknowledgement

It gives us great pleasure in presenting the preliminary project report on ‘Car Dam age Detection using Computer Vision’. I would like to take this opportunity to thank my internal guide Prof. P.B. Warungse for giving me all the help and guidance I needed. I am really grateful to them for their kind support. Their valuable suggestions were very helpful. I am also grateful to Prof. S.P. Bendale, Head of Computer Engineering Department, NBN Sinhgad Technical Institute for his indispensable support, suggestions. In the end our special thanks to Prof. P.B. Warungse and Prof. S.C Sethi for providing various resources such as laboratory with all needed software platforms, continuous Internet connection, for Our Project.

References

[1] M. Chen, F. Bai and Z. Gerile, “Special Object Detection Based On Mask Rcnn,” 2021 17th International Conference on Computational Intelligence and Security (CIS), 2021, pp. 128-132, doi: 10.1109/CIS54983.20 21 .00035.

[2] M. Ye et al., “A Lightweight Model of VGG-16 for Remote Sensing Image Classification,” in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 14, pp. 6916-6922, 2021, doi: 10.1109/JS TARS.2021.3090085.

[3] H. Bandi, S. Joshi, S. Bhagat and A. Deshpande, “Assessing Car Damage with Convolutional Neural Networks,” 2021 International Conference on Communication information and Computing Technology (ICCICT), 2021, pp. 1-5, doi: 10.1109/ICCICT50803.2021.9510069.

[4] P. M. Kyu and K. Woraratpanya, “Car Damage Detection and Classification,” in Proceedings of the 11th International Conference on Advances in Information Technology, Bangkok Thailand, Jul. 2020, pp. 1–6, doi: 10.1145/3406601.3406651.

[5] H. Patel, “hemilpatel971/Damage-car-detection,” GitHub, Aug. 13, 2019.

[6] Phyu Mar Kyu, Kuntpong Woraratpanya," Car Damage Detection and Classification" IAIT2020: Proceedings of the 11th International Conference on Advances in Information Technology July 2020 Article No.: 46 Pages 1–6 https://doi.org/10.1145/3406601.3406651.

[7] Jeffrey de Deijn. 2018. "Automatic Car Damage Recognition using Convolutional Neural Networks" (2018).

[8] Sourish Dey .2019. "CNN Application-Detecting Car Exterior Damage".