CAR DAMAGE DETECTION USING TRANSFER LEARNING

Megha Manoj and Shifa Parveen Allaudeen
BITS Pilani Dubai Campus, United Arab Emirates
f20200016@ dubai.bits-pilani.ac.in; f20200040@dubai.bits-pilani.ac.in

Abstract: Vehicles have become the main mode of transport in the progressing world which has led to a rise in the number of traffic and other vehicle related accidents taking place. When it comes to recovering the vehicle from the damage of such accidents, insurance companies play a strong role in financially supporting the vehicle owners. The field of image-based inspection is expanding and holds significant potential for automation. If vehicle damages could be automatically classified, it would greatly accelerate and streamline the insurance claims process.This paper presents an image classification model using transfer learning combined with an artificial neural network model using ResNet-50 to create a convolutional neural network. An imbalanced dataset is adopted and converted into an uniform one using data augmentation. This project aims to detect damaged vehicles through convolutional neural networks and evaluate the model.

Keywords: ResNet-50, Google Colab, CNN, Data Augmentation,image datasets, Artificial neural network, AUC-ROC curve, accuracy, Loss function, binary cross entropy, ADAM optimizer

---

1.Introduction

Assessing car damage is a crucial task in the automotive industry, whether it is for insurance purposes, resale value, or safety concerns. Traditionally, this assessment has been done by experts in the field, relying on their knowledge and experience to accurately identify and classify damages. However, with the advancements in machine learning, there is a growing interest in automating this process using computer vision techniques. In recent years, Transfer Learning combined with Artificial Neural Networks has emerged as a powerful tool for image recognition and classification tasks, and it has shown promising results in detecting car damage accurately. Artificial Neural Networks (ANNs) are a subset of machine learning algorithms that mimic the structure and function of the human brain, making them well-suited for image recognition and classification tasks. In this approach, pre-trained models are leveraged, and the transfer learning technique is used to retrain them on new data with limited labeled data, achieving high accuracy.

The assessment of car damage is not only essential for the automotive industry but also for car owners who want to keep their vehicles in good condition and avoid potential safety hazards. Traditional methods of assessing car damage are time-consuming, subjective, and prone to errors, which can lead to inaccurate estimations and delayed repairs. Computer vision-based approaches can automate the assessment process, which can significantly reduce the time and cost required for assessment while improving the accuracy and consistency of results.

This paper aims to explore the application of Transfer Learning combined with Artificial Neural Networks in the assessment of car damage and its potential benefits for the automotive industry.

2.Literature Review

The R.E. van Ruitenbeek[1] discusses research conducted on automatic inspection to prevent delays caused by inspections, done before handover of vehicles to the customers, on the basis of expenses and usability. The research aims to extend the application of convolutional neural networks for damage detection by exploring 3 different methods. The authors used a self-made dataset through manual annotation and further extended the collection of images from the internet. They also created a demo environment for real deployment of the model by placing cameras within a light street.For preprocessing the data, redundant images were removed, excluding the images depicting the same type of damage from different angles. A manual process is then implemented to clear the dataset of images with no visible damage or cars in them. The data is then split for training and testing and annotated with polygons. The dataset is run through different experiments to train the model for damage detection using Adam optimizer. The results indicate that deep learning provides results with high accuracy in detection and classification of vehicle images. It concludes that the use of external data improves detection of damage in internal dataset. Although the model is not precise at recognising the background in light street, it is still capable of classifying the damage.

The research conducted by Qinghui Zhang [2] focuses on building a vehicle damage detection model to resolve problems related to traffic accident compensation caused by traffic congestion. The model consists of Mask RCNN, a deep learning model capable of performing target detection and segmentation. Mask RCNN is used in the research to detect and classify damaged parts of automobiles in traffic accidents and also reduce the number of layers involved in the neural network, strengthen the model's regularization and improve its generalization capability. The dataset used in the study consists of images annotated through LabelMe annotation tool. Mask RCNN performs feature extraction ,classification prediction and segmentation masking to output a car damage detection model. The detection accuracy of the resultant model is quite high however the model failed to detect damaged areas of the vehicle when it was hard to spot.

Umer Waqas [3] created a model using deep learning techniques to classify car damage based on its severity via the application of MobileNet and also determine whether the images are fake or real. The concept of moire effect was also taken into consideration to identify fake images.A dataset consisting of damaged as well as undamaged images were derived from the internet and images taken through phones and computer screens, for moire detection. The preprocessed dataset contained grayscale images and given labels - medium damage, huge damage and no damage. For image classification and moire detection MobileNet was used with additional layers of fine tuning and transfer learning, which was made up of convolutional networks and ReLu nonlinearity. Fake images were identified with the help of a Fake Image Analysis Module to detect editing. The resultant model showed 95% accuracy in vehicle damage detection and 99% in moire detection.

The paper takes into consideration the concept of 2 dimensional image processing and machine learning algorithms to estimate car damage. The authors have implemented Convolutional Neural Network(CNN) to classify the severity of the damage via transfer learning. H. Bandi [4] trained a model which takes 2 datasets, each consisting of 300 images, for differentiating between damaged and undamaged cars. The images used were obtained from google and converted from RGB(red-green-blue) format to gray-scale. and given a fixed size in order to uniform the data. Feature extraction was performed on the entire dataset followed by transfer learning using VGG16 architecture and Keras model to provide the required Conv2D and MaxPooling2D layers . a softmax activation layer is added at the end along with additional dense trainable layers. 2 Binary classification models built using the dataset provided an accuracy of 63% in detecting damaged cars using CNN and 87.9% using the VGG16 model. Its research concludes that use of higher quality images can improve the performance of the model.

Sruthy C M [5] address the concern of time wastage in handling insurance claims due to inspection and testing by building model capable of car damage detection and classification using Convolutional Neural Networks and transfer learning models from the Keras Library, namely - InceptionV3, Xception, VGG16, VGG19, ResNet50, and MobileNet.The model created was successful in predicting the images with an accuracy of 97.28%.

Atharva and Tejas [6] did research on detecting damages on a car using deep learning algorithms which is an alternative for the manual inspection done by the insurance company during the testing of the vehicle .They have applied two CNN models. The first model is VGG16 which was used to find the damaged location on the car and how severe the damage is. The second model is Mask R CNN which was used to figure out the exact damaged location. Both these models were successful in finding out the damage caused to the car. Deep learning and Machine learning are used as it can process the data, perceive frauds and reduce the risk of claim process in insurance industries. Four main algorithms were used, VGG-19, VGG-16, Inception V3 and Resnet50.VGG 16 and 19 models gave accuracy of approximately 82-85%, Resnet gave accuracy of 57% and InceptionV3 gave 72% accuracy.

They also used a pre-trained model from the ImageNet dataset to detect whether the object in the image is a car or not. These models were able to discover the exact location of the damage and the severity of the damage.VGG models were useful for giving accurate predictions and Mask RCNN model masked out the damaged locations from the images.

In this research work Daniel, Endang and Yosi [7] aims to create a model which will make it easier for the car insurance company to accelerate the process of insurance claims using deep learning systems. The dataset used consists of three parts, segmentation of car parts and damage and the classification dataset of the car damage. The author used Mask R-CNN for the segmentation.MobileNetV2 and EfficientNet was used for the classification task. A simple CNN model was also used for the classification model as an addition so that a masked segmentation model can be received as an additional input. The result was that the Mask R-CNN can segregate the damage .When looking at classification models ,the models which used additional simple CNN models provided a better output compared to the models which did not. When a simple CNN model is used as an addition there is a 9% increase in the F1 score. The study also showed that when CNN model was used along with MobileNetV2 gave an F1 score of 91%.This study done by the authors can produce a decent and trustworthy model that can detect and classify the damage done to the vehicle.

Kalpesh Patil, Mandar , Anand and Shirish [8] did a research on detection and classification of car damage by experimenting with various techniques such as training a CNN model and later on they used a pre-trained model which used auto encoder along with fine-tuning and transfer/ensemble learning. This study was done as to reduce the issue of illegal insurance claims and leakage of the claims faced by the car insurance industry.CNN based model is used for the categorization of car damage types.The dataset used by the author contained some specific types of damage done to a car such as door/bumper dent, shattered glass, broken headlight/tail light, smashed/scratches done to the vehicle. The images for the dataset were collected from the net and the authors manually marked them. Ensemble learning combined with transfer learning gives out best outcome. Additionally, a technique was developed to locate the type of damage. This study shows that performance of transfer learning method is better and this model can categorize only some specific car damage features.

This research paper focuses on detecting the damage vehicle parts when the car went through a low speed crash (velocity difference of not more than 16 km/h) .Milan Koch [9] used time series data for machine learning in this study. Feature selection algorithm is used to detect the most significant parts. The dataset used in this study is collected from the sensors installed in production vehicles and 100 cases are considered. To estimate the potential of such a system, crash tests are conducted, and the sensor data from these crashes such as the acceleration of the crash is recorded. Post each test; the effected parts of the vehicles are identified. The author came up with a technique that uses time-series data for machine learning in an efficient, optimized and automated manner. This technique is good for small datasets. Tsfresh software package was used which is helpful in computing a diverse set of features and the boruta algorithm assists in identifying the most significant and common features and EGO software optimizes the hyper-parameters of the machine learning algorithm(random forest) in a proficient manner. The final F1 prediction score can reach up to 94%, which demonstrates the potential of predicting damaged vehicle parts solely based on the onboard data.

J.D Dorathi Jayaseeli [10] did research on detecting damage done to a car using image-based processing methods. The idea of automatically detecting and quantifying the severity of vehicle damage externally would be advantageous for car insurance companies, rental companies, agencies and services. The author suggests the utilization of convolutional neural networks to create a Mask RCNN model which is capable of detecting the damaged region on a vehicle. The dataset used consists of images of damaged vehicles with a class named scratch. The images are arranged according to the area of damage. Mask R-CNN COCO dataset was used to train the model. The images used undergo 21 epochs of processing. Post processing the final outcome is visualized using a color splash method in which the damaged area is highlighted. This model is useful to reduce the cost of handling insurance claims. Using this model fraudulent vehicle insurance claims can be minimized. This model produced a total loss value of 0.3888.

3. Dataset Description

There is a lack of availability of datasets for vehicle damage detection, due to which the amount of annotated datasets found is scarce.

The dataset 'Fast, Furious and Insured' chosen for this project was taken from Kaggle. It consists of train and test folders,each containing images of damaged and non damaged cars in jpg format and a .csv file.

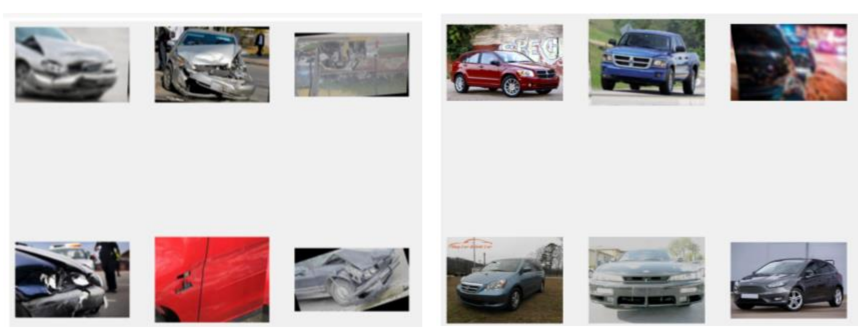


Fig 1: Images in train dataset

The csv files store records for each image along with the following insurance related features-

- Image_path - the name of the image file in .jpg format
- Insurance_company - the name of the insurance company

- Cost_of_vehicle - the cost of a vehicle purchased.
- Min_coverage- the minimum amount of money the insurance firm can provide during a certain period.
- Expiry_date- Expiry date of an insurance contract after which the vehicle owner can no longer avail insurance unless renewed.
- Max_coverage- It refers to the maximum amount of money the insurance firm can provide during a certain period.
- Condition - A binary feature classifying whether the vehicle in the image is damaged or not. Here, if the value of the feature is 1, the car is damaged and vice versa.
- Amount- the amount of monet the insurance company offered for the vehicle

Since the purpose of this project is to assess car damage, only the relevant features, Image_path, and Condition, are extracted from the dataset. The images present in the dataset consist of cars in different backgrounds, perspectives and lighting. To enhance the diversity of images the model is capable of correctly predicting, the quality of the images varies from blurred to defined.

4. Methodology

The methodology adopted for this project is damage detection of vehicles through Transfer Learning combined with an Artificial Neural Network.

Based on the literature review done, the authors have found a suitable image dataset of damaged and undamaged cars for image classification. With reference to the models mentioned in the reviewed research papers, the concept of transfer learning and artificial neural networks have been opted for constructing the classification model.

4.1. Dataset preprocessing

To equalize the number of images present in the training dataset, an ImageDataGenerator is imported from Keras for preprocessing. when the model generated is fit by creating batches of images that are derived artificially at runtime from the same dataset. In this manner, train, test and validation sets are generated for classification.

4.2. Image augmentation

ImageDataGenerator applies various augmentation techniques on the entire image data inputted and presents the model with assorted images when it is run by working on the pixel. It helps create consistency in the dimensions of the images by rescaling them. It also creates variations in the angles of the images through parameters for rotating and flipping the images as well as randomly zooming into some of the images. To ensure that no pixels of each image data are ignored during execution, the boundaries of each image are filled. A certain quantity of the data can be reserved for validation purposes too. The advantage of using this image augmentation technique is that at each epoch, the input images are different and it takes less memory space due to its live augmentation while the model is run. The results of image augmentation are as follows-

- The train dataset consists of 2015 images belonging to 2 classes.
- The test dataset consists of 600 images belonging to 1 class.
- Validation dataset consists of 671 images belonging to 2 classes, from a 25% validation split performed.

Refer to this link to view the dataset

https://drive.google.com/file/d/1QZIj1n06VK4zMeFWdrwynrHub6fq_h35/view?usp=share_lin

4.3. Transfer Learning

Transfer Learning refers to the application of pre-trained models as the first layer of a new learning model in order to increase efficiency of the new model.

In this project, transfer learning has been implemented using a deep learning model from the Keras library. .An artificial neural network is built by adding layers to a pre-trained ResNet-50 model for image classification. A ReNet-50 is a convolutional neural network that comprises 50 layers where one layer is a max pool layer,48 layers are convolutional, and an average pool layer.

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 keras_layer_1 (KerasLayer)  (None, 2048)              23561152

 dropout (Dropout)           (None, 2048)              0

 dense (Dense)               (None, 8)                 16392

 dense_1 (Dense)             (None, 1)                 9

=================================================================
Total params: 23,577,553
Trainable params: 16,401
Non-trainable params: 23,561,152
```

Fig 2: CNN model

A sequential model is created using the pre-trained ResNet-50, one Dropout, and 2 Dense layers, as shown in the figure. It is used as a feature extractor for the image dataset in the ResNet-50 layer sequential layer created.The model has not learned 23,561,152 parameters as the ResNet-50 has already been trained on these hidden layers.

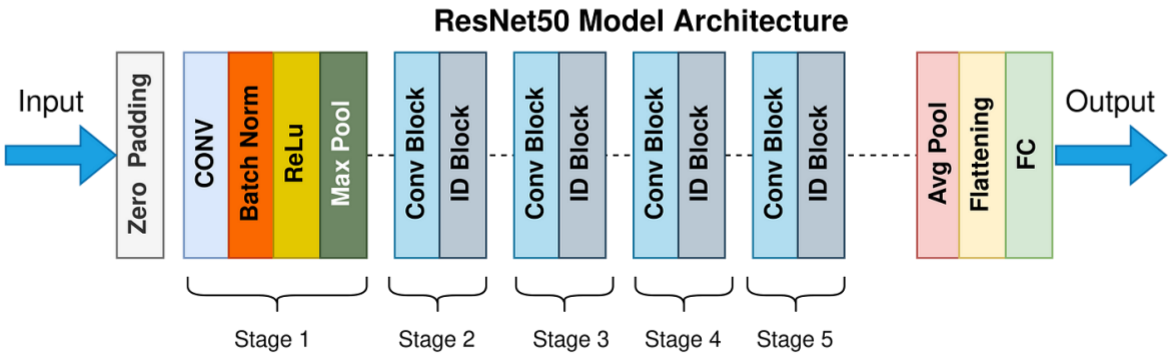


Fig 3: Architecture of ResNet50

The model is compiled with with following parameters:

- optimizer- The optimizer applied on the model is the Adam optimization algorithm to reduce the computational time taken by the model.
- loss- Binary cross-entropy is implemented as the loss function of the model to compute the cross entropy between the predicted labels and true labels.
- metrics- Area Under the ROC curve, Precision, Recall have been chosen as a metric to analyze the performance of the model.

Additionally, to prevent the model from overfitting, an early stopping rule has been inserted as a callback to stop fitting the model when the validation loss of the model starts giving similar values.

4.4. Evaluation

Finally, the validation set is evaluated on it to obtain the metric results and the performance of the model shall be compared at epoch level along with interpreting the overall results.

5. Implementation

In this section, the implementation details will be discussed. This section consists of information about the experimental setup and data description.

5.1. Experimental Setup

The core system components include a Windows 11 Home as operating system, with 1TB of memory. The platform used for implementing the project was Google Colab executed on Google Chrome. To speed up the execution of the code, the Colab environment is set to GPU. The dataset is accessed from the author's google drive and unzipped into the Colab content directory.

Table 1: Overview of experimental setup

System Components	Specification
Operating System	Windows 11 Home
Memory	1TB
Platform	Google Colab
GPU	NVIDIA-SMI 525.82.15
Driver version	525.82.15
CUDA version	12.0

5.2. Experimental Data

The dataset consists of 2 csv files- train.csv, which contains information related to the images in the train folder, and test.csv, containing information about the images in the test folder. The train folder is made up of 600 damaged and non-damaged car images and test.csv does not contain the feature 'Condition' as these images are used for testing purposes. The training dataset consists of a total of 1399 images out of which 1300 images are damaged and 99 are undamaged cars, which poses the problem of the dataset being imbalanced. The imbalance present is sorted by segmenting the dataset of training images, based on whether they are damaged or not, into different directories created. These image directories are then used in data augmentation, to perform oversampling.

6. Discussion and Results

This section will discuss the results of the training and evaluation of the transfer learning model using the following terminologies-

- Positive- An image is correctly classified.
- Negative- An image is wrongly classified.
- True positive(TP)- An image is correctly classified as positive.
- False positive(FP)-An image is incorrectly classified as positive
- True negative(TN)- An image is rightly classified as negative.
- False negative(FN)- An image is wrongly classified as negative.

The formulas listed below were used to calculate precision and recall:

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP+FN}$$

The model was fitted for a total of 80 epochs out of which 12 epochs were completed before the early stopping rule came into action. Epoch 12 gave the best results in terms of loss of 16.2%, maximum AUC score of 98.21%, precision of 93.32%, and recall of 91.62% in comparison to the previous epochs. However, the performance of epoch 12 on the validation set shows fluctuating scores, the validation loss of the epoch is 54.09% indicating that about half of the target and predicted labels match. On the contrary, the validation AUC is as high as 90.81% depicting that the majority of the images were correctly classified. The validation precision, which indicates the relevance of the results, is highest at epoch 5 and gives similar values within the range of 73% to 80% for each epoch. This means that about 20 to 30% of the classification were false positives. The validation recall scores rise in a fluctuating manner with each epoch and stop at 94.15%, hence very few of the images were false negatives.

Table 2: Model Fitting

Epoch no.	Loss	AUC	Precision	Recall	Validation loss	Validation AUC	Validation precision	Validation recall
1	0.5689	0.7792	0.7107	0.6451	0.5457	0.8285	0.7303	0.8000
2	0.3857	0.9067	0.8095	0.8195	0.4716	0.8675	0.7885	0.8031
3	0.3131	0.9409	0.8521	0.8626	0.5025	0.8687	0.7848	0.8031
4	0.2704	0.9566	0.8802	0.8821	0.4758	0.8883	0.7917	0.8769
5	0.2681	0.9570	0.8910	0.8718	0.4748	0.8883	0.8052	0.8523
6	0.2438	0.9652	0.8854	0.8872	0.4842	0.8903	0.7861	0.8523
7	0.2165	0.9724	0.9167	0.8913	0.4748	0.8984	0.7861	0.8954
8	0.1986	0.9778	0.9216	0.9159	0.5010	0.8956	0.7914	0.9108
9	0.1886	0.9795	0.9283	0.9169	0.4838	0.8983	0.7889	0.9200
10	0.2075	0.9740	0.9180	0.9067	0.5409	0.8995	0.7650	0.9415
11	0.1743	0.9821	0.9332	0.9169	0.5388	0.9014	0.7775	0.9569
12	0.1620	0.9847	0.9379	0.9292	0.5266	0.9081	0.7775	0.9415

Table 3: Evaluation Of Validation Set

Loss	Accuracy	Precision	Recall	F1-Score

56.03183507919312%	90.12138247489929%	77.24999785423279%	95.07692456245422%	85.24137860104243%

Table 3 displays the overall scores for each of the applied evaluation metrics when the validation set is evaluated. The loss score reveals that 56.03% of the predictions were True positives and the model gave an accuracy of 90.12% indicating that it performs well. The precision score (77.25%) shows more than half the predictions were rightly classified whereas the recall indicates 95.07% of the actual damaged cars were classified as damaged. the F1-score was calculated using precision and recall using the following formula:

$$\boldsymbol{F1 - score} = \frac{\boldsymbol{2 * Precision * Recall}}{\boldsymbol{Precision + Recall}}$$

The evaluation gave an F1-score of 85.25% revealing that both precision and recall values of the model are good indicators of the model's performance. Hence the overall performance of the model is good.

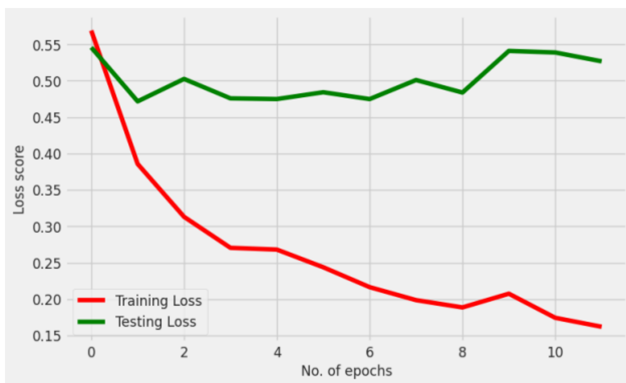
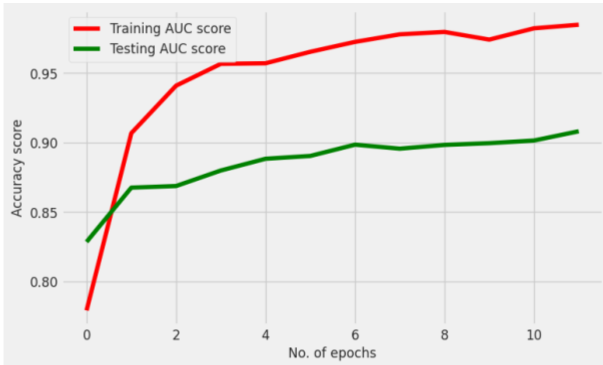


Fig 4: Loss Curve

The Loss curve in figure. 4 suggests that the testing loss is higher than the training loss. Hence the model is not generalized enough and the data used is underrepresented when compared to the testing data.

The Area Under the Curve graph (figure 5) is a summarized graphical representation of the evaluation metric Receiver Operator Characteristic (ROC) curve that is used to evaluate the working of binary classifiers. It makes use of False positive rate(FPR) and True positive rate(TPR) to plot a probabilistic curve. Here, depicts that the model gave best results training data in comparison to the testing dataset.



$$FPR = \frac{FP}{TN+FP}$$

$$TPR = \frac{TP}{TP+FN}$$

Fig5: Area Under the Curve

6.1. Comparison with previous works

To compare the performance of our model with those mentioned in the reviews papers, we have used evaluation metrics such as accuracy, precision and recall. If the values of the evaluation metrics are higher for one model, it can be concluded that this particular model performs better than the other.

When compared with other models based on transfer learning with data augmentation, the precision and recall of the constructed model show the best performance [8]. The authors built CNN models which make use of pre-trained models such as VGG-19, Alexnet, Inception were used for feature extraction and the extracted features were used as data for linear Supervised Vector Machine(SVM) and a SoftMax classifier followed by creating an ensemble learner which makes use of the 3 pre-trained models which gave the best performance. The precision of the ensemble learner was 85.88% with image augmentation while the recall value was 78.91%. When all the pre-trained models were used as estimators for the ensemble model, the model gave precision and recall values slightly less than that using the best 3 pre-trained models. A CNN model was also built with 10 layers where using random initialization and an augmented image dataset were used for training and prediction which resulted in 64.03% precision and a recall score of 52.5%.

The model gave a better performance in terms of accuracy when compared to the Convolutional Neural Network(CNN) models built by H. Bandi et. al.[4]. The first model built by the authors is a simplistic CNN model which makes use of randomized images for car damage detection and the latter is a transfer learning model built using pre-trained VGG16 from the Keras library resulting in an accuracy of 87% when trained and tested on a set of 300 images. Therefore, it can be concluded that the newly built model performs better than that proposed in the research paper due to the larger dataset. Furthermore, the use of ResNet-50 decreases the computational time and is deeper than the VGG16 model.

The author introduced a Residual Network (ResNet) that employs the residual module to facilitate model convergence and expedite neural network training. This network is integrated with the Mask RCNN object detection model. The integration of this Residual Network with the Mask RCNN model leads to improved accuracy of object detection and segmentation within the model.The paper aims to enhance the model's network structure by reducing the number of layers in the Residual Network and strengthening the model's regularization and generalization capabilities. The parameters of the anchor box and the loss function are then adjusted to improve the accuracy of car damage detection and segmentation.While the detection accuracy is high, the mask instance segmentation is not always completely accurate and may not be able to segment areas where the damage is not obvious.Therefore the model in our paper performs better compared to the model suggested in this paper.

To implement this paper, four algorithms were considered, namely VGG16, VGG19, Resnet50, and Inception V3. Among all the models trained, VGG 16 achieved the highest accuracy of approximately 85%, followed by InceptionV3 with an accuracy of 72%, VGG 19 with an accuracy of 82%, and Resnet50 with an accuracy of 57%.The project is implemented in two parts. The first part involves training three VGG models using transfer learning.In the second part of the implementation, Mask R-CNN was used to localize the car damage.The best performance was achieved over the entire car test set, with a false positive rate of 0.63% and a false negative rate of 0.97%.During the training process, they initially encountered the problem of overfitting.The CNN model had an average accuracy of 86.99%. However, the increased processing time caused by analyzing sections that do not correspond to vehicles was observed as a drawback.The final classification accuracy achieved was about 87%.Therefore comparing the accuracy of the model our model performed better.

6.2. Challenges

The dataset used for implementation was highly imbalanced which made the model more exposed to having a bias towards damaged cars. The lack of a label attribute ‘Condition’ in the test dataset to compare any predictions made through a classification model. The data augmentation technique produces batches of data that are directly passed through the built classification model, which prevents any possible evaluation of the predicted data because of differences in the size of the original data and the augmented data thereby limits the use of evaluation metrics that use predicted values and true values of the test dataset. The derived accuracy of the model is not ample to evaluate the performance of the model if there is any imbalance present within the classes. While the dataset is diverse in terms of the damages present, a more intrinsic model can be built capable of detecting which part of the vehicle is damaged. The computational time of the model is severely large without the assistance of the GPU, therefore execution of such large programs may be taxing on a CPU.

Although the testing and validation data were shuffled to provide better model performance, the training dataset was still under-represented when compared to the validation data, giving large loss scores.

7. Conclusion

The aim of this paper was the assessment of car damage with Transfer Learning combined with an Artificial Neural Network has shown promising results.The study has demonstrated that deep learning techniques can effectively detect and classify vehicle damages with high accuracy.The assessment of car damage is a crucial aspect of the automotive industry, and recent advances in machine learning have presented an opportunity for more accurate and efficient evaluation. Transfer learning combined with artificial neural networks has shown promising results in accurately detecting and classifying car damages from images. By leveraging pre-trained models and transfer learning, this approach significantly reduces the time and resources required for training and achieves high accuracy with limited labeled data. The potential applications of this technology are vast, from streamlining the insurance claims process to improving vehicle safety through early detection of damages. As research in this field continues to progress, we can expect more efficient and accurate solutions to emerge, revolutionizing the way we assess car damage in the future.

8. Author Contributions

The paper was a collaborative effort by two authors who made equal contributions.The literature review was performed by both authors.Megha was responsible for drafting the abstract,Implementation,dataset description and challenges sections, while Shifa wrote the introduction, conclusion, and methodology sections.Additionally, both authors collaborated on implementing the algorithms to generate the results.

9. References

[1]R. E. van Ruitenbeek and S. Bhulai, “Convolutional Neural Networks for vehicle damage detection,” Machine Learning with Applications, vol. 9. Elsevier BV, p. 100332, Sep. 2022. doi: 10.1016/j.mlwa.2022.100332.

[2]Q. Zhang, X. Chang, and S. B. Bian, “Vehicle-Damage-Detection Segmentation Algorithm Based on Improved Mask RCNN,” IEEE Access, vol. 8. Institute of Electrical and Electronics Engineers (IEEE), pp. 6997–7004, 2020. doi: 10.1109/access.2020.2964055.

[3]U. Waqas, N. Akram, S. Kim, D. Lee, and J. Jeon, “Vehicle Damage Classification and Fraudulent Image Detection Including Moiré Effect Using Deep Learning,” 2020 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE). IEEE, Aug. 30, 2020. doi: 10.1109/ccece47787.2020.9255806.

[4]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). IEEE, Jun. 25, 2021. doi: 10.1109/iccict50803.2021.9510069.

[5]C M, S. Kunjumon, and N. R, “Car Damage Identification and Categorization Using Various Transfer Learning Models,” 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI). IEEE, Jun. 03, 2021. doi: 10.1109/icoei51242.2021.9452846.

[6] A. Shirode, T. Rathod, P. Wanjari, and A. Halbe, “Car Damage Detection and Assessment Using CNN,” 2022 IEEE Delhi Section Conference (DELCON). IEEE, Feb. 11, 2022. doi: 10.1109/delcon54057.2022.9752971.

[7]D. Widjojo, E. Setyati, and Y. Kristian, “Integrated Deep Learning System for Car Damage Detection and Classification Using Deep Transfer Learning,” 2022 IEEE 8th Information Technology International Seminar (ITIS). IEEE, Oct. 19, 2022. doi: 10.1109/itis57155.2022.10010292.

[8]K. Patil, M. Kulkarni, A. Sriraman, and S. Karande, “Deep Learning Based Car Damage Classification,” 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA). IEEE, Dec. 2017. doi: 10.1109/icmla.2017.0-179.

[9]M. Koch, H. Wang, and T. Back, “Machine Learning for Predicting the Damaged Parts of a Low Speed Vehicle Crash,” 2018 Thirteenth International Conference on Digital Information Management (ICDIM). IEEE, Sep. 2018. doi: 10.1109/icdim.2018.8846974.

[10]J. D. Dorathi Jayaseeli, G. K. Jayaraj, M. Kanakarajan, and D. Malathi, “Car Damage Detection and Cost Evaluation Using MASK R-CNN,” Intelligent Computing and Innovation on Data Science. Springer Nature Singapore, pp. 279–288, 2021. doi: 10.1007/978-981-16-3153-5_31.

[11]https://www.shaip.com/blog/training-data-to-train-vehicle-damage-detection-model/#:~:text=Vehicle%20damage%20detection%20uses%20machine,vision%20and%20imaging%20processing%20tools.

[12]https://www.kaggle.com/datasets/infernape/fast-furious-and-insured

[13] https://youtu.be/JcU72smpLJk]

[14]https://www.tensorflow.org/api_docs/python/tf/keras/preprocessing/image/ImageDataGenerator

[15]https://neptune.ai/blog/keras-metrics

[16]https://www.geeksforgeeks.org/matplotlib-pyplot-imshow-in-python/

[17]https://towardsdatascience.com/using-convolutional-neural-network-for-image-classification-5997bfd0ede4#:~:text=The%20Convolutional%20Neural%20Network%20

[18]https://builtin.com/data-science/transfer-learning

[19]https://www.analyticsvidhya.com/blog/2022/09/image-classification-in-stl-10-dataset-using-resnet-50-deep-learning-model/

[20]https://link.springer.com/article/10.1007/s12652-022-04105-3

[21]https://medium.com/analytics-vidhya/car-damage-classification-using-deep-learning-d29fa1e9a520

[22]https://www.analyticsvidhya.com/blog/2023/04/deep-learning-for-image-segmentation-with-tensorflow/#Visualizing_Dataset_Samples