Car Damage Detection and Assessment Using CNN

Atharva Shirode
Department of Information Technology
Sardar Patel Institute of Technology
Mumbai, India
adshirode8@gmail.com

Tejas Rathod
Department of Information Technology
Sardar Patel Institute of Technology
Mumbai, India
tejasrathod227@gmail.com

Parth Wanjari
Department of Information Technology
Sardar Patel Institute of Technology
Mumbai, India
wanjariparth@gmail.com

Aparna Halbe
Department of Information Technology
Sardar Patel Institute of Technology
Mumbai, India
aparna_halbe@spit.ac.in

***Abstract*— In today's digital world, most businesses are adopting technology in every possible way. Many time it occurs that when the car is damaged insurance claims are done. If the car is insured, a person from the insurance industry visits and takes survey of the customers car and prepares the report. The manual verification is a tedious process. But with the major advancement in field of deep learning algorithms, it can be used in the insurance industry to solve these problems. In the proposed solution we have implemented 2 CNN models. VGG16 is used to detect the damage on the car, location of the damage and its severity. Mask RCNN is used to mask out the exact damaged region. Both the models give a fair idea about the damage caused to the car which can help insurance company to proceed further with the insurance claims without wasting time and resources on manual verification.**

Keywords—Car Damage, Detection, Classification, VGG, Mask RCNN, Severity, Location, Masking

I. INTRODUCTION

In today's world, the use of private vehicles has increased drastically which has also lead to increased rate of accidents. Most of the vehicles being insured, they go for insurance claims. There are lots of insurance companies present in the market and majority of them reside on manual inspection of the car which is damaged. In this process they waste a significant amount of time as well as resources. Claim leakage becomes a major threat to the company in such situations. One of the feasible solutions of this issue lies in the use of proper technology which can greatly reduce the efforts, time and resources in this process. Machine learning and deep learning can help problems such as analysing and processing data, detecting frauds, reducing risks and automating claim process in insurance industries. So, insurance firms are looking for faster damage assessment and agreement of claims.

A very well-known technique which has worked effectively in case of small labelled data is transfer learning. A network which is trained on a source task is used as a feature extractor for target task. There are many CNN models trained on ImageNet which are available publicly such as VGG-16, VGG-19, Inception, Resnet. Transferable feature representation learned by CNN minimizes the effect of over-fitting in case of small, labelled set.

For the implementation of this paper 4 algorithms were considered, VGG16, VGG19, Resnet50, Inception V3. Of all models trained VGG 16 gave accuracy of approximately 85%, VGG 19 gave accuracy of 82%, Resnet50 gave accuracy of 57% and InceptionV3 gave accuracy of 72%. VGG16 being effective in terms of its object detection(car) capability and classification (severity and location) because of its simple linear architecture and hence compatibility with the required use case. Hence VGG16 model was best suitable for implementation.

The most complex challenge is reducing model training time. Performing image classification tasks with a traditional CNN model and identifying the optimal weights for the network over several forward and backward iterations might take a long time. Using GPUs, this process could take days or even weeks to finish. Fortunately, using pre-trained CNN models that have been previously trained on big benchmark datasets like the ImageNet dataset, the model training effort can be decreased. Through transfer learning, weights can be freely extracted and their designs can be used for other particular tasks.

The implementation of this project is done two parts, first one consists of VGG models [10] wherein 3 models are trained using transfer learning. Pre trained model based on ImageNet dataset detects the object from the image whether it is car or not. Out first model identifies the car as damaged or not. If it finds that the car is damaged, then it proceeds to the second model to detect the location of the damaged portion and the third model find the severity of the damage. The models are trained using self-defined labels by classifying the dataset manually.

In the second part of implementation, Mask R-CNN [11] has been used for localizing the damage of the car. The images were annotated using annotation tool and generate a json file for the labels and coordinates for annotated images. This json file was then used to train R-CNN model [12] on the labelled dataset. The labels used were Scratch, Dents, Shatter and Dislocation. The model identifies this local damage on the car image and masks it. This gives a clear idea of the damage location and its type.

II. LITERATURE SURVEY

The authors in [1] have used mask RCNN architecture for detecting the damage and masking it. Region of interest of every image is stored in a json file in the form of polygon coordinates and used while training the model. Coco weights are used to train the model and those weights are later modified and migrated for fine tuning. These algorithms mask the damaged portion on the car image but do not help to find the severity of damage.

2022 IEEE Delhi Section Conference (DELCON) | 978-1-6654-5883-2/22/$31.00 ©2022 IEEE | DOI: 10.1109/DELCON54057.2022.9752971

978-1-6654-5883-2/22/$31.00 ©2022 IEEE

According to [2], they have created 3 VGG models for Detecting the damage (Damaged or not), Severity of damage (Minor, Moderate, Severe) and Damage location (Front, Rear, Side). They used pre-trained VGG models and applied transfer learning on those models to defeat training times and overfitting problems for smaller datasets. But this method does not show the exact location of damage on the car.

The authors in [3] collected the photographs and prepared the data set for the broken cars. They split the data set into the bumper dent, door dent, broken glass, broken headlamp, broken lamp, scratch, smash, none. They zoomed in on the dataset four times applying a twenty-degree rotation, 0.2 cut, 0.2 variable zooms, and horizontal flip instead of instructing CNN without pre-trained weights. Furthermore, they opted for pre-trained models (Alex net, V3 origin, VGG19, Resnet50, mobile networks). Furthermore, they exclusively formed FC layers and each layer for each pre-trained model. In this case, they used YOLOV3 to locate the damage and highlight it using the bounding boxes.

The authors in [4] collected the images and sorted the dataset into 8 classes (bumper dent, door dent, broken glass, broken headlamp, broken tail light, scratch, crushing, no damage). Since the dataset containing the images is smaller, they synthetically enlarged the dataset 5 times using the random rotation increase between 20 and 20 with horizontal flipping. They formed a convolutional neural network and CNN-AE (autoencoder) from scratch. The accuracy did not reach 75%. So, they went to train pre-trained models. They used pre-trained models like Inception, Alexnet, VGG19, VGG16, and Reset and SVM. From the output of the pre-trained and SoftMax models with Delta optimization, cross-entropy loss. By experimenting with using the SoftMax classifier, it performs better than linear SVM and is faster to train.

In [5] dataset was created by capturing images of cars on various streets and extracting additional features from them. Cropping of the region of interest was performed using strictly demarcated bounding boxes to easily extract the positions of damaged parts. In addition to train and test images, a set of localization tests is provided to compare the localization accuracy of their algorithms. They claim that their dataset is compatible to develop comprehensive automatic detection algorithms and coin-based automatic detectors. To detect an object, the image is divided into a superimposed grid of sub-images, on which a test analysis is performed. The best performance is obtained over the entire car test set, with the false positive and false negative rates being 0.63% and 0.97% respectively.

The authors in paper [6] utilized Alexnet architecture for training CNN and the image is divided into segments for searching for the presence of scratch. There are only 2 categories, sections with and without scratches, for this reason it is necessary to change the number of classes in the fully connected output to 2 and retrain. Based upon the architecture they modified their dataset into multiple sections (with and without scratches) and data augmentation was done in order to generalize the network. They faced the problem of overfitting initially while training. The average accuracy (of both with and without scratches) of the CNN model was 86.99%. The drawback of increased processing time due to analysis of sections not corresponding to vehicles was seen. Use of R-CNN to extract the region of interest and further detect scratches was proposed to overcome the above limitation.

The author in [7] proposes the use of CNN for damage classification of the car. The concept of transfer learning is used. The author used various available CNN architectures for deep learning like VGG 16, VGG 19 and Resnet. VGG 16 gave the highest accuracy. The images were divided into various classes according to implementation requirements. The final classification accuracy achieved was about 87%.

The authors in [8] highlighted the problem of cumbersomeness and claims leakage in insurance. This process can be automated by using machine learning. The dataset used here for training was preprocessed according to the need. The entire dataset was divided into various classes like scratch, dent, glass shattered, broken lamp. A CNN architecture was used for model training and predicting the class and type of damage.

Authors in [9] proposed a 2-step approach based on object detection that improves mean average precision by 30%. One method is to use R-CNN and the other one is to use a well labeled dataset for training to detect the regions. In the region proposal they extract 4096 dimensional features using the caffe implementation. The author used a bit of fine tuning to improve accuracy. It was observed that an improvement of 8% was obtained by fine tuning the model. The HOG is replaced with DPM HSC. The final model was tested on Pascal VOC.

III. PROPOSED METHODOLOGY

The implementation is divided into 2 major parts. First part is based on VGG 16 architecture which consists of 3 models, first model checks whether uploaded car image is damaged or not and rest 2 models checks the location of the damage and severity respectively. If the first VGG model confirms the damage of car then it is carried forward to next part of implementation which is based on Mask RCNN architecture for masking the exact damage portion.

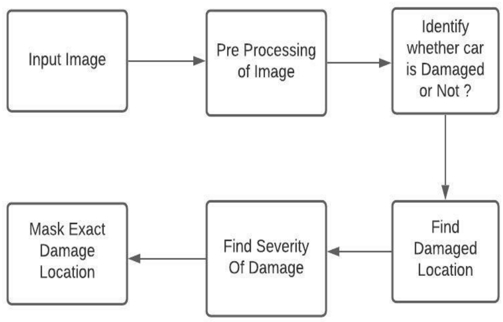

Fig. 1. Implementation flow diagram

A. VGG 16

The VGG [2] network's input is a two-dimensional image (224, 224, 3). As per Fig2 The first two layers have the same padding and 64 channels of 3*3 filter size. Then, after a stride (2, 2) max pool layer, two layers of convolution layers of 256 filter size and filter size (3, 3). This is followed by a stride (2, 2) max pooling layer, which is the same as the previous layer. Following that, there are two convolution layers with filter sizes of 3 and 3 and a 256 filter. Following that, there are two sets of three convolution layers, as well as a max pool layer. Each has 512 filters of the same size (3, 3) and padding. This image is then fed into a two-layer convolution stack. The filters utilised in these convolution and max pooling layers were 3*3 instead of 11*11 in AlexNet and 7*7 in ZF-Net. It can be found in some of the layers.

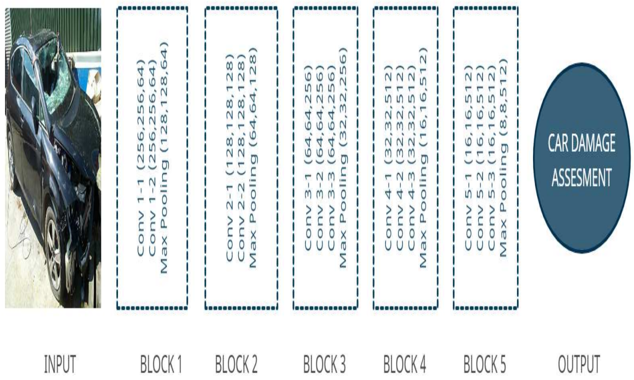

Fig. 2. VGG-16 Architecture

In this paper three VGG models [4] are created for assessing the initial damage condition. For pre-processing of the image dataset, the height and width of the image is set to 256*256. PreprocessInput() function is used to make the data compatible with ImageDataGenerator() function from Keras. It is used for augmentation of dataset for accurate training of model. After pre-processing of the images, the categorical model is trained by adding a flattening layer followed by 2 dense layers. SGD optimizer is used as it converges faster than batch training on large dataset. All models are trained for 50 epochs and then finetuning over 10 epochs.

1) Damaged or Not (Model 1)

This model identifies whether the car is damaged or not. It is trained using 198 images of damaged cars and 262 images of whole (non-damaged) cars along with the labels Damaged and Whole. The activation functions used for the last 2 dense layers are relu and softmax respectively and the learning rate is set as 0.00001 and binary_crossentropy is used for loss as the output is binary.

2) Location of damage (Model 2)

This model tells on which side of the car the damage is caused and is trained on a total of 979 images divided under 3 labels: Front (419 images), Rear (272 images), Side (288 images). The activation functions used for the last 2 dense layers are relu and sigmoid respectively and the learning rate is set as 0.0001 and categorical_crossentropy is used for loss.

Fig. 3. Confusion matrix of VGG Model-2

3) Severity of damage (Model 3)

This model classifies the damage into its level of severity and is trained on a total of 979 images divided under 3 labels: Minor (315 images), Moderate (278 images), Severe (386 images). The activation functions used for the last 2 dense layers are relu and softmax respectively and the learning rate is set as 0.0001 and categorical_crossentropy.

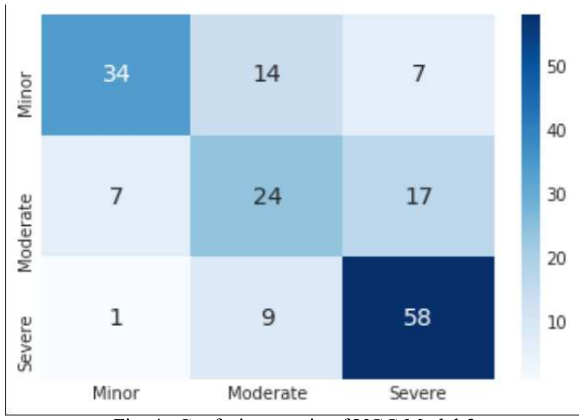

Fig. 4. Confusion matrix of VGG Model-3

B. Mask R-CNN

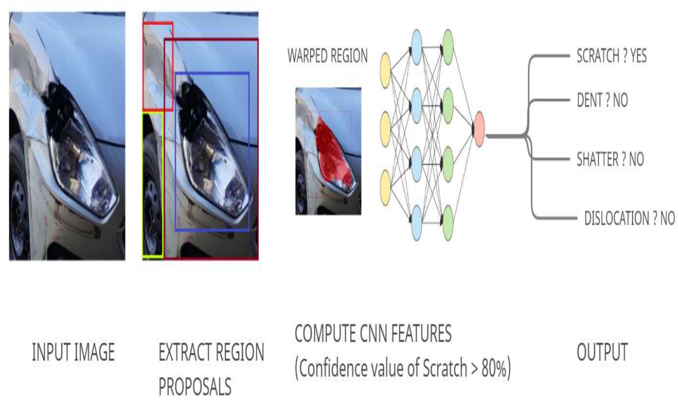

Fig. 5. Mask R-CNN Architecture

As shown in to Fig 5. the RCNN mask combines a faster RCNN for object detection (class + bounding box) with a fully convolutional network (FCN) for pixel-level boundary detection.

For each candidate element, Faster RCNN [11] has two outputs: a class label and a bounding box offset; to that a third branch is added that emits the object mask, which is a binary mask that shows the pixels where the object is in the bounding box. The extra mask output, however, differs from class and box outputs, making it necessary to extract a much finer spatial arrangement of an element. Therefore, Fully Convolutional Network (FCN) is a common semantic segmentation approach and Mask RCNN uses it. To decompress an image to 1/32 of its original size, this model uses multiple convolution blocks and maximum pool levels. This level of detail is then used to construct a class prediction. Finally, it resizes the image to its original size using sampling and deconvolution levels.

Mask RCCN being a supervised learning algorithm for object detection requires every ROI to be labelled [1]. Using the VGG Image annotation tool v-1.0.6, the coordinates of bounding polygons along with the labels were stored in json format for respective images in the dataset. They were further used while training the model. 4 labels are used to train this model and the dataset was divided accordingly for it. The labels were Scratch, Dent, Shatter and Dislocation.

In the second part of implementation, mask R-CNN [12] has been used for instance segmentation. In the first part, only the type and location of damage was shown but, in this part, the damaged part is highlighted using mask R-CNN. The mask R-CNN works in 2 stages. In stage 1 there are two networks and a RPN (Region Proposal network). In the second stage the network predicts bounding boxes and object classes. Five classes are made namely background (BG), scratch, dent, shatter, dislocation. When an image is fed to the model it will try to match any one of the labels. Threshold probability for every label is set as 80 percent and if the identified label crosses the threshold set, then it is considered as damaged under that particular type and masked.

The implementation is extended using Matterport's RCNN repository. It is based on feature pyramid network (FPN) and Resnet101 backbone. It helps in creating segmentation mask according to how the model is trained. COCO weights are also provided which can be used according to the requirement or the use case. To get the optimum accuracy and masking of damaged portion several combinations of learning rate, pool size, steps per epoch etc were tried and tested. After several trials the combination of learning rate as 0.001, mask pool size as 14, pool size as 7 and steps per epoch as 80 gave the best suited result. The model was trained for 15 epochs with the mentioned parameters and gave appropriate results.

IV. Result and Observation

Table I shows how the accuracy of model is varying. Whenever the input images are given to the model, it first detects it whether the given image is car or not and then identifies the location of damage as well the severity of damage as shown in the Fig 6 and Fig 7 below.

TABLE I. VGG Model Classification Report

Model	precision	Recall	f1score	support
Identify car	0.91	0.90	0.90	460
Location	0.76	0.75	0.75	171
Severity	0.68	0.68	0.67	171

Example 1:

Fig. 6. Implementation- Output1

As it can be seen from Fig 6 the headlight of the car is broken hence it can be considered as Moderate level damage and it is on the front side of the car, the implemented VGG model also predicts the same results. Further the Mask RCNN model masks the damage portion of headlight precisely.

Example 2:

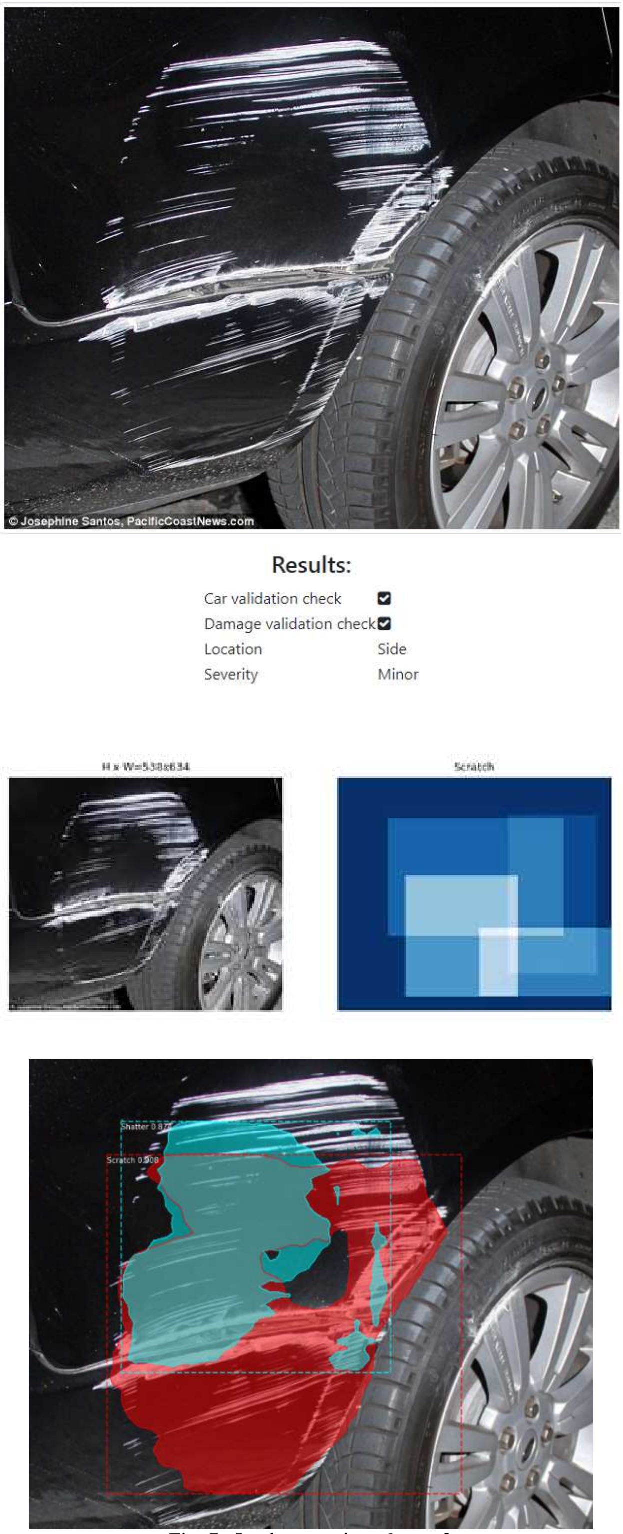

Fig. 7. Implementation- Output2

From Fig 7 it can be noted that the car has a minor scratch one the side portion and both VGG and Mask RCNN models are able to detect it correctly.

The above results vary because of the noise present in the images. As all the images are taken at different light intensity and different background. It becomes tough for the model to detect the exact location of the damage. A zoomed in image with the damage location will show more accuracy than the zoomed-out image with low pixels quality.

V. Conclusion

The implemented models are able to detect the damage, severity level and its exact location. Fine tuning of VGG models helped in increasing its accuracy and giving accurate predictions for edge cases. Mask RCNN model is also precisely masking out the damaged regions from the images considering various factors such as different backgrounds, light intensity and multiple angles. In future work data expansion can be carried out to train the model effectively and get better results. The approximate cost of repair can also be calculated by use of proper resources and data.

.

VI. References

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

[2] Kyu, Phyu & Woraratpanya, Kuntpong. (2020). Car Damage Detection and Classification. 1-6. 10.1145/3406601.3406651.

[3] Dwivedi, Mahavir & Hashmat, Malik & , Shadab & S N, Omkar & Bosco, Edgar & , Monis & Khanna, Bharat & Samal, Satya & Ayush, & , Tiwari & Rathi, Aditya. (2019). Deep Learning Based Car Damage Classification and Detection. 10.13140/RG.2.2.18702.51525.

[4] 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), 2017, pp. 50-54, doi: 10.1109/ICMLA.2017.0-179.

[5] P. Rostami, A. Taimori, S. Sattarzadeh, H. O. Shahreza and F. Marvasti, "An Image Dataset of Vehicles Front Views and Parts for Vehicle Detection, Localization and Alignment Applications," 2020 10th International Symposium onTelecommunications (IST), 2020, pp. 25-30, doi: 10.1109/IST50524.2020.9345889.

[6] Suescún, César & Useche Murillo, Paula & Moreno, Robinson. (2018). Scratch Detection in Cars Using a Convolutional Neural Network by Means of Transfer Learning. International Journal of Applied Engineering Research. 13. 12976-12982.

[7] 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.

[8] Aniket Gupta, Jitesh Chogale, Shashank Shrivastav, Prof. Rupali Nikhare"Automatic Car Insurance using Image Analysis",International Research Journal of Engineering and Technology (IRJET) Volume: 07 Issue: 04 | Apr 2020 e-ISSN: 2395-0056

[9] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 580-587, doi: 10.1109/CVPR.2014.81.

[10] 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/JSTARS.2021.3090085.

[11] K. He, G. Gkioxari, P. Dollár and R. Girshick, "Mask R-CNN," 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 2980-2988, doi: 10.1109/ICCV.2017.322.

[12] 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/CIS54983.2021.00035.