CAR EXTERIOR DAMAGE DETECTION USING MASK R-CNN

Ms.C. Pabitha[1], Pradhip Kumar.R[2], Sanjeev Kumar Patel[3], Shankar.K.R[4]

[1] Assistant Professor, Department of Computer Science and Engineering, SRM Valliammai Engineering College, Tamil Nadu, India
[2] UG Student, Department of Computer Science and Engineering, SRM Valliammai Engineering College, Tamil Nadu, India
[3] UG Student, Department of Computer Science and Engineering, SRM Valliammai Engineering College, Tamil Nadu, India
[4] UG Student, Department of Computer Science and Engineering, SRM Valliammai Engineering College, Tamil Nadu, India

ABSTRACT

Recently, the production of image-based vehicle insurance is an important area with considerable scope for automation reach. In this paper we consider the issue of classifying car damage, where some of the categories may be fine-granular. To this reason, we are exploring deep learning-based techniques. Initially we try to train a CNN directly with a collection of training data. However, it's not working well due to a small collection of labeled data. Hence, we investigate the domain-specific pre-training effect accompanied by fine-tuning with a large number of annotated training-data. As Faster R-CNN and SVM have not identified damaged cars with high accuracy, and the Cascade R-CNN takes an immense amount of time to train and check the data that we are working on to fit. Hence, we are training data into a R-CNN Mask that produces adequate results compared to traditional Neural Networks. Though there is a lot of unknowns such as partial images, Hence the classifier was built to detect amorphous damages. The model is layered over 3 classifications of detecting the car and examining whether the damage dealt is high or low. Finally, the classifier is projected with the flask environment to make the working experience easier, since it runs on a localhost the compile time does not exceed 5 seconds irrespective of the quality of the image. Experimental results indicate that Mask R-CNN works better than convolutional R-CNN as transfer learning works better than domain specific fine-tuning such as Cascade and Faster R-CNN. We achieve 89.5 per cent accuracy with Mask R-CNN through transfer combination.

Keyword: - *Car Damage Classification, CNN, Transfer Learning, Convolutional Auto-encoders*

---

1. INTRODUCTION

Nowadays, a lot of money is being wasted in the car insurance business due to leakage claims. Claims leakage / Underwriting leakage is characterized as the discrepancy between the actual payment of claims made and the sum that should have been paid if all of the industry's leading practices were applied. Visual examination and testing have been used to may these results. However, they impose delays in the processing of claims. Efforts have been made by a few start-ups to reduce the processing time of statements. In this paper, we use Convolutional Neural Network (CNN) based methods for classifying car damage types.

We experimented with a variety of techniques, such as direct CNN training, pre-training CNN using auto-encoder followed by fine-tuning, using transfer learning from broad CNNs trained on ImageNet, and creating an ensemble classifier on top of the collection of pre-trained classifiers. We find that transfer learning, coupled with group learning, works best. We also set up a method to locate a particular type of damage. Experimental results confirm the effectiveness of our proposed solution.

2. PROPOSED METHOD

Convolution neural networks (ConvNets), we concentrate on the type of machine learning that we will use, which is classification. The classifier is a program that performs the so-called score function. This implies that, provided the data case, it calculates the score for all possible classes of C. The class with the highest score is then predicted to be the true class for the data case. In the special case where C= 2 (e.g. damaged or undamaged) is referred to as this differential classification. An algorithm that returns a classifier based on a set of labeled training data is called a learner. In order to obtain some structure in choosing a learner from the countless amounts of possibilities, characterizes learning as a combination of three components: representation, evaluation and optimization.

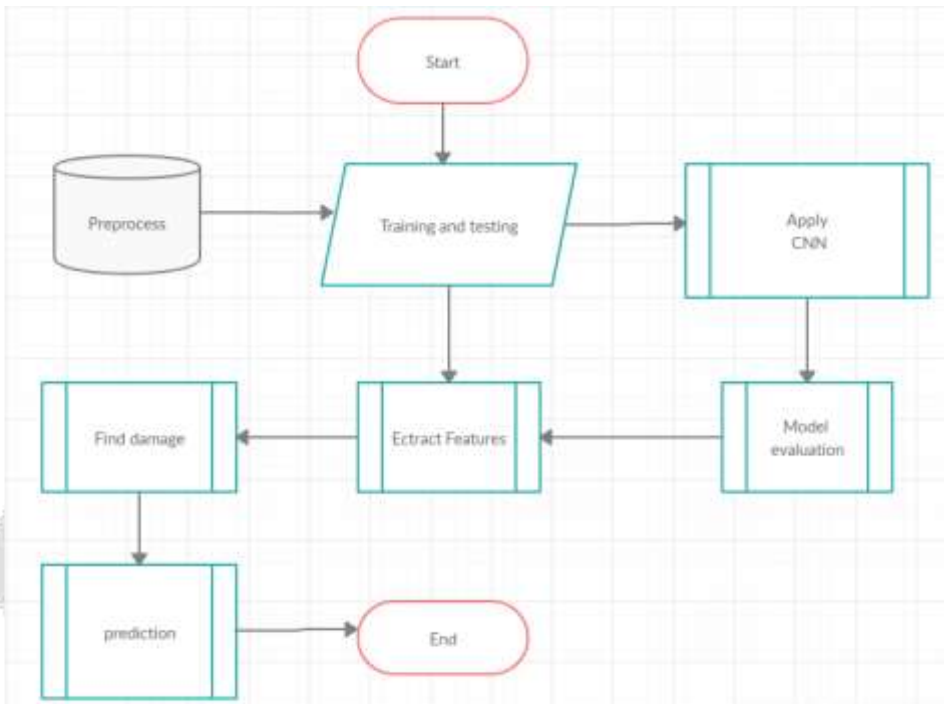

Fig -1: A pictorial representation of the proposed methodology

The representation of the learner, commonly referred to as the model, is its most characteristic aspect. This defines the learning space principle, i.e. the set of classifiers that can be learned. The representation of the ConvNet is determined by the network architecture. In other words, the ConvNet is represented by a set of nodes2 ordered in one or more layers which are connected in a feed-forward manner (so without any cycles).

A three-layer neural network of three inputs, two hidden layers of four nodes each, and two outputs. Remember that all connections are driven (so we can call them arcs) in a feed-forward fashion (from left to right) and there are no connections between nodes in the same layer.

2.1 ML Algorithms:

Like most of the real-world computer vision issues here, we'll also use transfer learning from sufficient pre-trained CNN to save a huge amount of time in retraining the entire weight matrix. One of the most powerful algorithms designed to predict vehicle exterior damage detection is Cascade R-CNN. In brief, like every object detection mission, here, too, we have three subtasks:

Extracting Regions of Interest (ROI): Image is passed to a ConvNet that returns a region of interest based on methods such as selective search (RCNN) or RPN (Region Proposal N / W for Faster RCNN) and then a RoI pooling layer on the extracted ROI to ensure that all regions are of the same size.

Classification task: Regions are transferred to a fully connected network that classifies them into different image classes. In our case, it's going to be scratch ('damage ') or background (car body without damage).

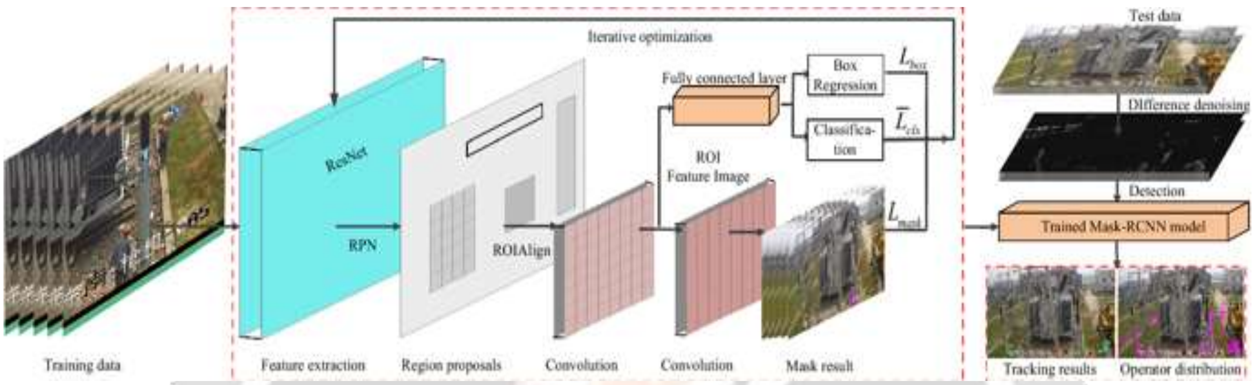

Fig -2: pictorial representation of Mask R-CNN model

Regression task: At last, a bounding box (BB) regression is used to predict bounding boxes for each identified region to tighten bounding boxes (get exact BB definition of relative coordinates)

In our case, however, only arriving at square/rectangular shaped BBs is not sufficient as the car scratches/damages are amorphous (without a clearly defined shape or shape). We need to identify the exact pixels in the bounding box that correspond to the class(damage). The exact location of the scratch pixel will only help to identify the location and quantify the damage accurately. So, we need to add another step-semantic segmentation (pixel-wise shading of the class of interest) to the entire pipeline for which we will use the CNN (Mask R-CNN)-based Masked Region architecture.

2.2Mask R-CNN:

Mask R-CNN is an instance segmentation model that helps us to define pixel-wise delineation for the object class of our interest. So Mask R-CNN has two specific tasks: 1)BB dependent object detection (also called localization task) and 2)Semantic segmentation, which allows the segmentation of individual pixel objects within a scene, regardless of the shape. Put together these two tasks. The R-CNN Mask does get the instance segmentation for a given image.

For object detection tasks, it uses a similar architecture to Faster R-CNN. The only difference in Mask R-CNN is ROI step-instead of using ROI pooling, it uses ROI align to allow the pixel to maintain ROIs and prevent the loss of information. For Semantic segmentation activities, it uses fully convolutional n/w(FCN). FCN generates masks (in our case binary masks) around BB objects by generating a pixel-wise classification of each area (distinct object of interest). Thus, overall Mask R-CNN minimizes the cumulative loss of the following losses at each point of the Proceeding Segmentation.

Symbol	Explanation
µ	True class label, u 0, 1…., *K*; by convention, the catch-all background class has u = 0.
p	Discrete probability distribution (per ROI) over K+1 classes: p = ($p_{0,}$……,p_k),computed by a SoftMax over the K+1 outputs of a fully connected layer
v	True bounding box v = (v_x, v_y, v_w, v_h).
t^u	Predicted bounding box correction, t^u = (t_x^u, t_y^u, t_w^u, t_h^u)

Chart -1: Symbol Explanation

rpn_class_loss : RPN anchor classifier loss is measured for each ROI and then summed up for all ROIs for a single image, and network rpn_class_loss must sum up to rpn_class_loss for all images (train / validation). But this is nothing but a loss of cross-entropy.

$$L_{rpn\_cls} = -\sum \log(p_u)$$

rpn_bbox_loss: Network RPN BB regression loss is aggregated as rpn_class_loss The bounding box loss values reflect the distance between the true box parameters, i.e. the (x, y) coordinates of the location of the box, its width and its height, and the predicted values. It is, by its definition, a lack of regression and penalizes greater absolute differences (in an approximately exponential fashion for smaller differences, and linearly for larger differences).

$$L_{rpn\_bb\_reg}(t^u,v) = \sum L_i^{smooth}(t_i^u - v_i)$$

While the first losses are produced during the BB object detection phase, the last three losses are generated during the Semantic Segmentation process. Therefore, during training the network minimizes to a total loss of all components (for each train and validation).

$$L_{overall}=L_{rpn\_cls}+L_{rpn\_bb\_reg}+L_{mrcnn\_cnn}+L_{mrcnn\_bb\_reg}+L_{mrcnn\_mask}$$

3.DATASET DISTRIBUTION

In order to improve the accuracy of the classifier, a combination of 1000 damaged and 1000 perfectly-looking car pictures were gathered and annotated in an amorphous way. To order to test the data set and the classifier, an additional 500 harm car images and 500 perfectly-looking car images have been fed. These images are annotated using third-party tools called labellmg.

Fig -3: Dataset and Annotations

5. MODEL VALIDATION AND TRAINING

In this project, we use the tensorflow backend to train the model. The dataset after it has been cleaned will be translated to a.csv file (separate comma values) and loaded into the training model. The csv file is then fed into the passive training model. Data manipulation is done using python language which, in effect, runs all the libraries to compile this R-CNN Mask. With each iteration, the model learns how to predict the damage caused by the vehicle.

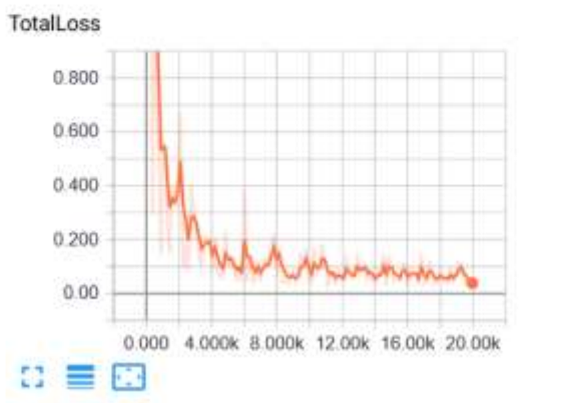

Fig -4: Total Loss

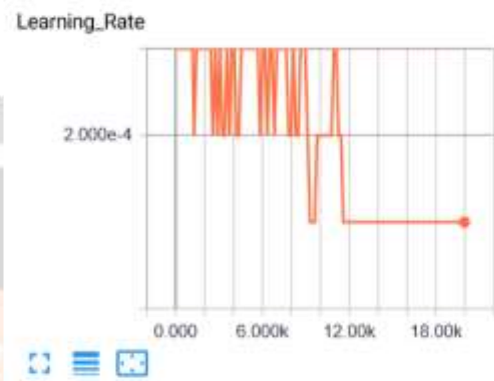

Fig -5: Learning Rate

6.MODEL ANALYSIS AND OUTPUT

Model Prediction: After adequate and beneficial loss monitoring ideally monotonically decaying both training and validation loss, we can test the model object on randomly selected validation images to see the accuracy of the prediction (car damage masking).

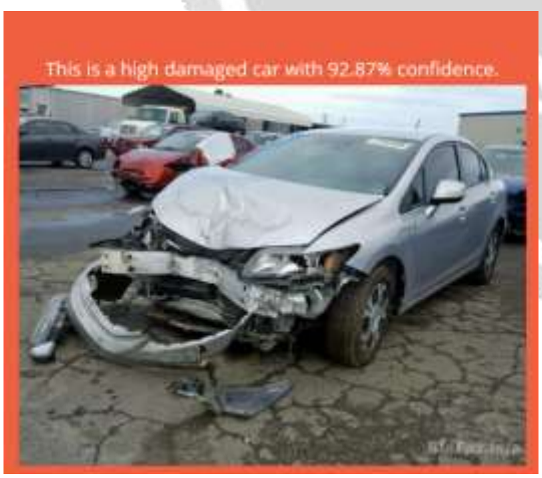

Fig -6: High Damage

Fig -7: Low Damage

The model is attached to the front end of the python flask, which serves as the user interface for uploading the desired image, so that the image is then assessed using the car external damage detector classifier and the result is displayed on the basis of how much damage the car has dealt.

Fig -8: No Damage

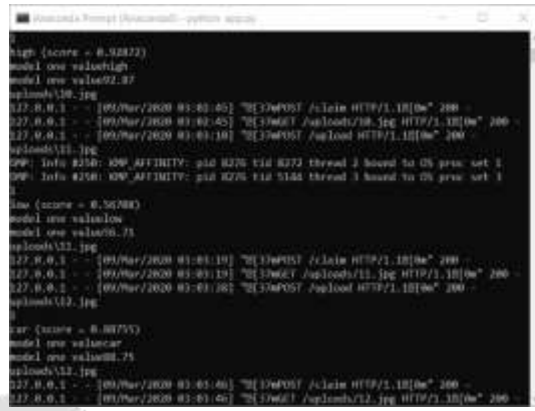

Fig -9: CMD output

7. CONCLUSION

In this paper, we proposed a deep-learning solution for the classification of car damage. We experimented with deep learning techniques such as R-CNN Mask Machine Learning Techniques, Convolution Auto Pre-Training Encoder followed by supervised fine tuning and transfer learning. We have found that transfer learning has worked the best. We also note that only common features of the car cannot be sufficient for the detection of damage. It thus underlines the supremacy of the feature representation learned from the broad collection of training courses. The suggested approach offers an effective and high degree of accuracy.

In the future, the vehicle shall be based on the external damage detected; and the estimated cost of repair of the vehicle shall be determined on the basis of the external damage identified and the internal damage assumed on the basis of access to the parts inventory, which shall include repair and labor costs for each component in the first and second sets of parts.

6. REFERENCES

[1]. Image processing-based severity and cost prediction of damages in the vehicle body: A computational intelligence approach, W.A. Rukshala Harshani; Kaneeka Vidanage, 2017 National Information Technology Conference (NITC)

[2]. Research on the Application of Deep Learning Target Detection of Engineering Vehicles in the Patrol and Inspection for Military Optical Cable Lines by UAV, Mingjiang Zhang; Hongwei Li; Guijin Xia; Weihu Zhao; Shuai Ren; Chengyuan Wang, 2018 11th International Symposium on Computational Intelligence and Design (ISCID)

[3]. Auto Detection of Joints and Axle Failure in Heavy Load Vehicles Using Artificial Intelligence, B. Sivasankari; V. Akash Kumar; M. Elavenil; A. R. Gomathi, 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS)

[4]. Vehicle-Damage-Detection Segmentation Algorithm Based on Improved Mask RCNN Qinghui Zhang; Xianing Chang; Shanfeng Bian IEEE Access

[5]. Small-Scale Rail Flaw Detection Car Design and Damage Position Estimation, Jinling Yao; Gang Yu, 2017 International Conference on Sensing, Diagnostics, Prognostics, and Control (SDPC)