**ABSTRACT**

Accurate Vehicle damage detection play’s most important part in the field of vehicular insurance and also in the rental car services industry. The precise discovery and appraisal of the damage occurring in a vehicle is imperative to estimating the insurance claim amount for rebursal to the client. The method also plays a pivotal role in the rental car services as it allows the rental agency to detect any non-pre-existing damage occurring to a vehicle after getting returned back to the company by the client toward the finish of the rental period. This damage detection method helps the company estimate the severity of the damage also, set forward the support and servicing charges accurately to the client.

The project puts forward an automatic approach towards detecting damage in a vehicle and also examining the extremeness of the damage to make the cycle bother free and fast. The methodology utilizes convolutional neural organizations (CNN) and utilizations picture preparing strategies to analyze the harm. A wide range of calculations have been concentrated to get more noteworthy exactness from the model in identifying the harm and furthermore to empower it to arrange the damage in terms of its severity.

1. **INTRODUCTION**

In the fast pacing world today which is rapidly evolving in terms of technology and innovation transport is very imperative to the growth.However, it is noted that more than 35000 vehicle accidents take place per annum. The large number of accidents then calls for an efficient insurance company, but unfortunately it is noted that the insurance policy holders have to wait for a considerably large amount of time in order to receive any rebursal related to their insured vehicle on the occurrence of an accident.

At present, just after a vehicle meets an accident, the insurance policy holder claims tor a compensation from the insurance agency and waits for their arrival at the place of accident. Then an operative from the company arrives at the spot and performs a manual analysis of the damage caused to the vehicle and following a customary methodology computes the examination. However, a lot of times the estimated price doesn’t match up the maintenance cost of the vehicle which in turn, results in a loss for the policy holder and leads to the disappointment of the client from the insurance firm.

Also, it is an often occurrence that the protection staff pass up a great opportunity on some hidden damages of the vehicle which is then not covered in the insurance appraisal. The process of approximation of dent is not very transparent and is in no way interactive with the client, which often leads to confusions and is not very effective either. Time taken to complete such claims and release the appraisal amount is another point for making the process dull and ineffective.

The approach followed in this project makes the whole process automatic an in turn less time consuming. It enables the user to click a picture of the damaged portion of the vehicle. The image captured by the driver or the insurance policy holder is then sent to the company which feeds it to the software in discussion. The approach followed here uses image processing techniques.

It then uses the image to detect the damage and categorize it into one of the accompanying classes of damages:

1. Minor damage:

This includes all the small dents and scratches and scrapes to the not so critical parts of the car.

For ex. Broken headlamps, minor scratches etc.



Figure 1.1– minor dent

1. Intermediate damage:

This includes all the large bumps or dents in a vehicular body, and other moderately critical damages like deformation of doors or deployment of airbags etc.

Figure1.2–deformation of body

1. Critical damage:

This encompasses all the heavy duty damage occurring to a car in severe road accidents like breaking if axle, overall deformation of vehicular structure, twisting of frames, breaking of doors etc.



Figure 1.3 – severe deformation and dent

**1.1 About Dataset**

Collection of an adequate dataset was one of the major challenges for us in the project. For accomplishing high exactness in the detection model it is imperative for us to make it learn the models with large volumes of data. In our approach we separately created two different datasets for classification purposes. The first was images with damaged cars while the other had undamaged car pictures.

**Damaged cars dataset:**

Initially since there exists no such dataset with pre-collected pictures of damaged cars we had to get creative and make our own dataset. For this purpose, we searched google for images and found out relevant images. For the diversification of our dataset it was imperative for us to select images with the damages in all different ranges i.e. from images of minor damages like scratches and dents to severe damages like broken headlamps and bumps. We also categorized these images in terms of type of damage, location of damage and the severity of damage.

**Undamaged cars dataset:**

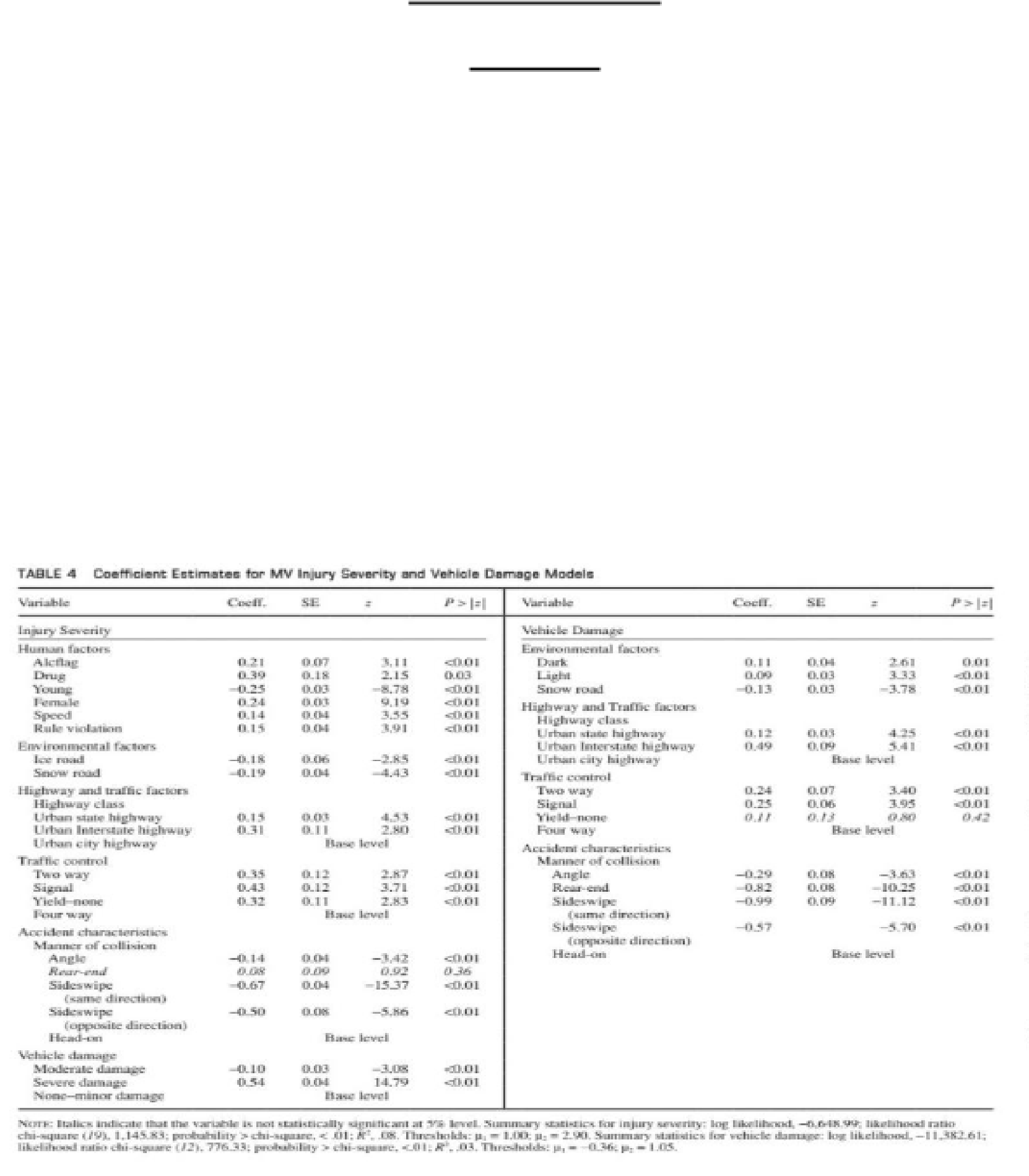
There are various datasets available which contain images of cars in different orientations but for the project in hand we used the “cars” dataset Krause et al. It consists of 16185 images of 196 different car models and was created for getting fine-grained classification. The dataset contains of small scale vehicles like those of passenger cars and not the bigger vehicles like trucks, buses or vans. Main challenges has been was to select related dataset and images for which we had to do a lot of work.

**1.2 Purpose of the project**

As stated in the introduction part of the project, the project aims at developing an computerized framework for distinguishing and analyzing the damage in a car due to multifarious reasons. It progresses towards providing a hassle free, cost effective and cheaper innovation towards the analysis of vehicular damage. The project finds it’s application in multiple fields, insurance field vehicle rental administrations are to name a few.

In the insurance field it makes hassle free for the related companies to detect the damage all the more precisely as analyzed to manual methods. It makes the process less time consuming and hassle free se initially the insurance agent had to pay a manual visit to the aggrieved party and perform a complete inspection and it is highly probable to miss judge the damage assessment as there would be no set criteria for the process. But with the help of this project, the insurance agency gets the picture of the harmed vehicle from the customer it takes care of the picture to the framework and the framework examinations that image and gives an accurate interpretation of the assessment thus leading to a quicker insurance claim process which makes it a win-win situation for the client as well as the insurance company.

Presently in the vehicle rental assistance organization, the rental office needs to precisely decipher the harms if any has been brought about by the tenant in the time of his vehicle rental residency. This project helps in making it easier for the company or agency to perform a quick but accurate check and thus ensure the well-being of their vehicle and put forward a claim if there is any damage. And the client can also rest assured that he will not be responsible for any harm that did not incurred by him during the rental tenure.



**Literature Review**

**Paper – 1**

**Analysis of crash severity based on vehicle damage and occupant injuries**

By- Xiao Qin, Kai Wang and Chase.E.Cutler

They considered the severity of damage to the vehicle and the damages incurred by the occupants of the vehicles to come to the conclusion of the severity of the crash responsible. The project had the injury severity and vehicle damage were separately modelled to explore the 18 factors affecting the overall crash by occupant and vehicle. Four sets of models are specified in this paper: SV injury severity model, MV injury severity model, SV vehicle damage model, and MV vehicle damage model.

In the SV crash results, it was detected that, the utility function was more or less dependent on the magnitude value. The positively larger values could lead to possibly increased injuries while highly negative values could reduce the injuries incurred. In the MV crash results similar findings were noticed but the severity was dependent more on the human error and vehicle 19 damage was more severely related to serious injuries.

**Paper – 2**

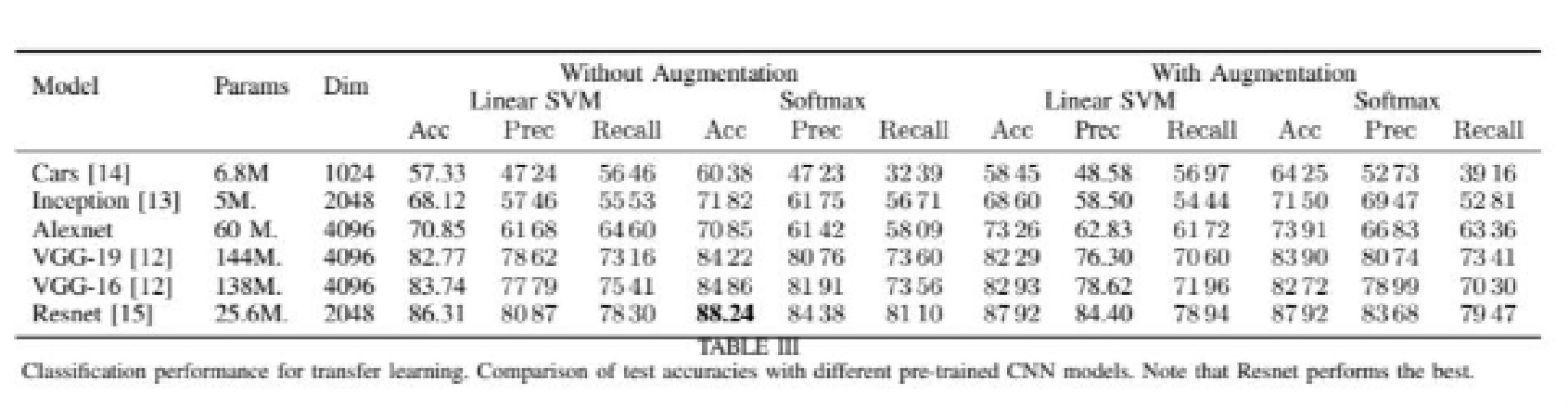
**Deep Learning Based Vehicle Damage Classification**

By –Kalpesh Patil, Mandar Kulkarni, Anand Sriram, and Shirish Karande

They deployed Connected Neural Networks based methods to classify the vandalism of the vehicle in the terms of crash or destruction associated with it. They consider the common types of damages like bumper dent, broken headlights, scratches, rear light damage, side disfiguration, broken axle etc. Since there was no availability of a preprepared dataset of vehicular damage images they had to create their own dataset by employing manual google search and using the basic augmentation techniques.

They made their CNN model learn on imageNet and used the characteristics of Transfer Learning to create a well trained classifier on top of already existing pretrained classifiers.

They found out that Transfer learning when paise with ensemble learning works the best and is efficiently able to complete the task with a very high accuracy rate.

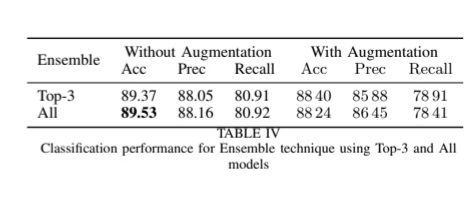


**Paper-3**

**Automatic Car Damage Recognition Using CNN**

By –Jeffrey de Dejin

They aimed at providing an aromatic way to recognize the damaged car images by employing CNN techniques. The highlight of this paper is that it looks into account the recognition of the image as an actual var before the assessment of damage incurred by the car due to some crash.

Along the project their main issue was finding the relativity of some unique hyper- parameters and compounding theoretically proven techniques to adapt them, in order to just progressively reach to a result which is highly accurate and satisfactory.

**2. METHODOLOGY**

The method we have implemented for Car Damage Assessment using hybrid classifiers can be divided into 4 steps.

1. Input the damage image
2. Feature extraction
3. Classification
4. Vehicle damage detection results

DATA COLLECTION

INPUT THE DAMAGE IMAGE

FEATURE EXTRACTION

CLASSIFICATION

VEHICLE DAMAGE DETECTION RESULTS

DEPLOYMENT OF THE MODEL

Figure 2.1 work flow

* 1. **FEATURE EXTRACTION**

The information i.e the data we create or collect is a collection of large fresh data, in which many features are either same or irrelevant in that case, we use feature extraction.

Feature extraction is the process of dividing initial raw data into small manageable groups. These consist of many variables, which requires lot of computing resources to process them. Therefore, feature extraction helps to reduce data set in a way such that the information set after processing, is still resulting the dataset with same accuracy and originality.

**2.1.1. EXTRACTION IN IMAGE CLASSIFICATION**

Major application of feature extraction is in the area of image classification. It is used to discover and segregate various desired features of image like edge, corners and many more. Feature extraction take the input image and represent the intersection parts of an image as a compact feature vector.Therefore, we have applied three feature extraction methods or techniques for image classification.

**2.2. DATA AUGMENTATION**

Data augmentation is a technique which exponentially increases the size and diversity of the dataset. It comes in very handy at the times when there is a shortage of data to train the model. The DA techniques the limited dataset and with some minor readjustments transform it into a dataset which is larger in size by manifolds. It has been in use for a long time and has proved beneficial in various large as well as small scale projects. Especially in our case since there exists no predefined dataset of damaged cars, so we had to get creative and collect our own dataset and collecting a dataset large enough to train our model to near perfection is a tedious job. Thus we used data augmentation tools and techniques to overcome our problem and create a large enough dataset for our project requirements.

**Data augmentation techniques :-**

There are various technique to data augmentation some of which we used are mentioned below

**2.2.1 Flip**

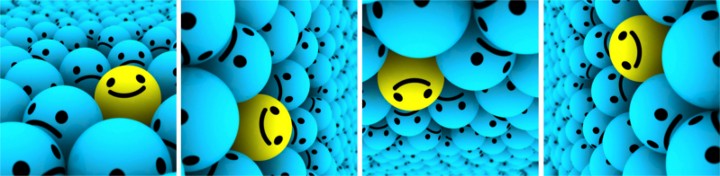
It flips the images vertically and horizontally. A lot of frameworks do not have the functionaliteit toprovide for vertical flips. But, a vertical flip is the same as rotating an image by 180 deg followed by a horizontal flip. Below examples will make it more clear.



Figure2.5 – flipped images example

**2.2.2. Rotation**

The most important aspect of this operation is that of the image might not be preserved after rotation. If the image fed is a square, it’s rotation along right angles preserved the image size. If it’s a rectangle, then its rotation by 180 degrees would have the size preserved.

Figure 2.6 – rotation of images

**2.2.3. Scale**

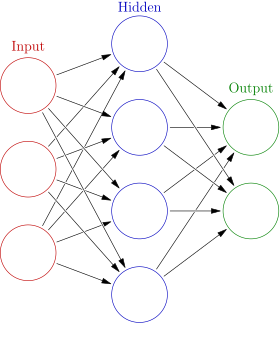
The scaling can be inward or outward. While scaling in ward, the final image size will be smaller than the real image size. A lot of frameworks crop out a section of the new image, of equal size.

Figure 2.7 – scaled images example

**3.Classification**

Deep learning-

Deep learning is a subset of machine learning and most effective technique when it comes to classify or predict information. This technique is also effective to recognize pattern and classify images. Deep learning involves use of neural network.



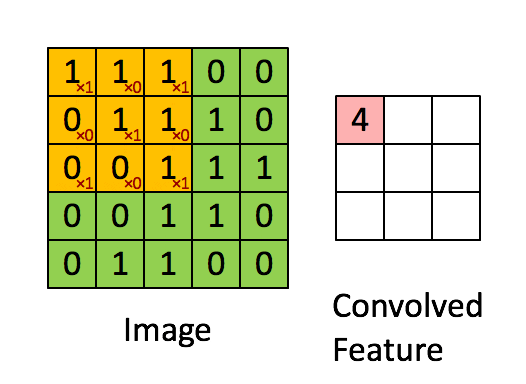
Above image shows the structure of basic neural network consisting of input layer, output layer, and hidden layer.

1. Convolutional neural network-

It is the common technique used foranalysing, classifying and recognizing pattern in an image. CNN is used everywhere from self-driving car to Instagram’s photo tagging feature. They are effective as well as quite impressive when it comes to speed even after updating weights.

CNN has several layers which do feature mapping in sequence. Those are –

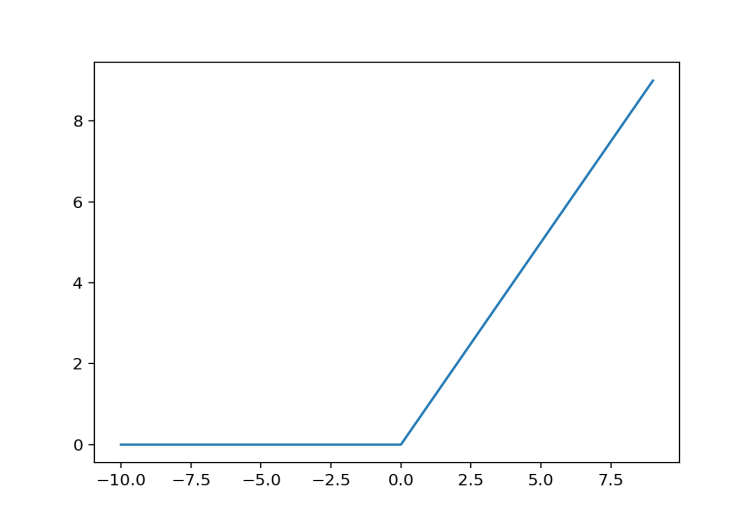
1. Convolutional layer- This is the 1st layer which uses a filter to do feature mapping



1. ReLU Layer-

This layer applies the function to all the values of the input. Basically what this layer does that, it outputs the same input if it is positive else itgives zero.

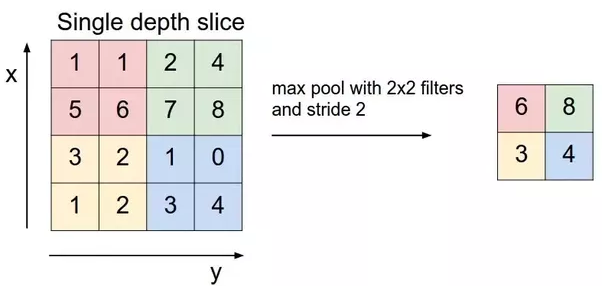
This layer allows us to use stochastic gradient descent to train models. It helps to update weights to gain perfect accuracy. If we talk about images, then out weights are filters which are updated time to time.



In the above figure,ReLu activation for negative and positive inputs is showed

1. Pooling-

While the convolution layer separates significant shrouded highlights, the quantity of highlights can at present be entirely enormous. We can utilize pooling to lessen the size of the information in respect of stature and width measurements. This permits the model to perform less calculations and eventually train quicker. It likewise forestalls overfitting, by separating just the most striking highlights and disregarding expected contortions or unprecedented highlights found in a couple of models.



1. Fully-connected layer

**1-Fully-connected layer**

A total associated layer of size 1024 (for instance the amount of neurons in the layer) to the yield data of the second pooling layer is being applied. The amount of units is somewhat self-confident. Enough to be stunning , however not really as to be excessively asset serious. The reason for taking completely associated layer is to total the information highlights before we convert them to classes. This permits the model to improve forecasts than if we had quite recently changed over the pooling yield straightforwardly to classes.



Convolutional neural network model

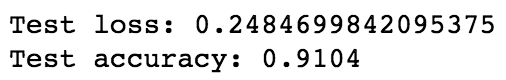
CNN has several models such as-

1. Alex net
2. Vggnet
3. Resnet

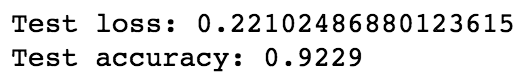
In this report we will be talking about some of the previous models which were used to classify images on some different data set of the images.

1. CNN with one layer-

CNN model was created containing only 1 layer. that single layer involved convolve operation with ReLU activation function and kernel size 3\*3. It also had one max pooling layer of size 2\*2. The results were

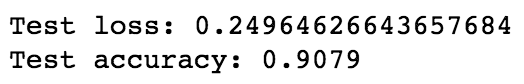


After performing data augmentation

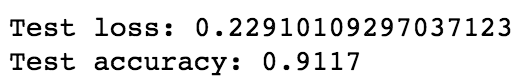


1. CNN with three layer-

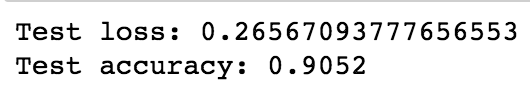
In this, CNN model was created comprising of only 3 layer



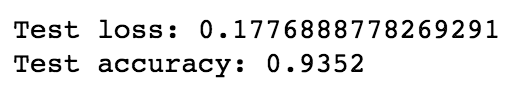
After performing data augmentation



1. CNN model with 4 layers-



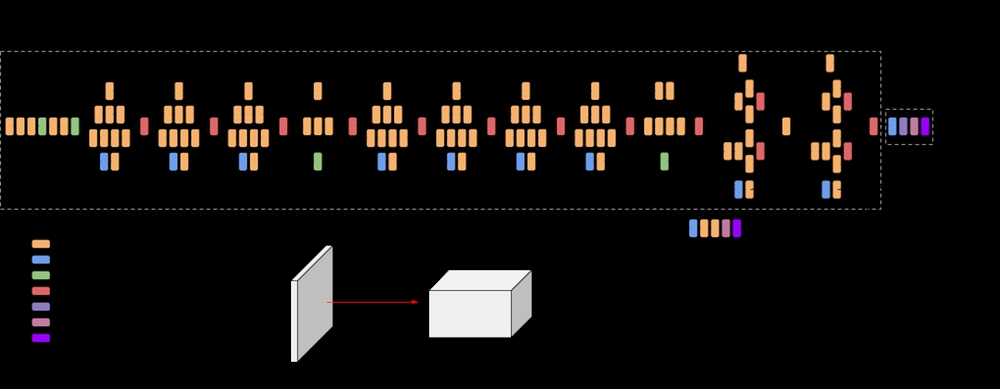
After performing data augmentation



**3.1.Inception**

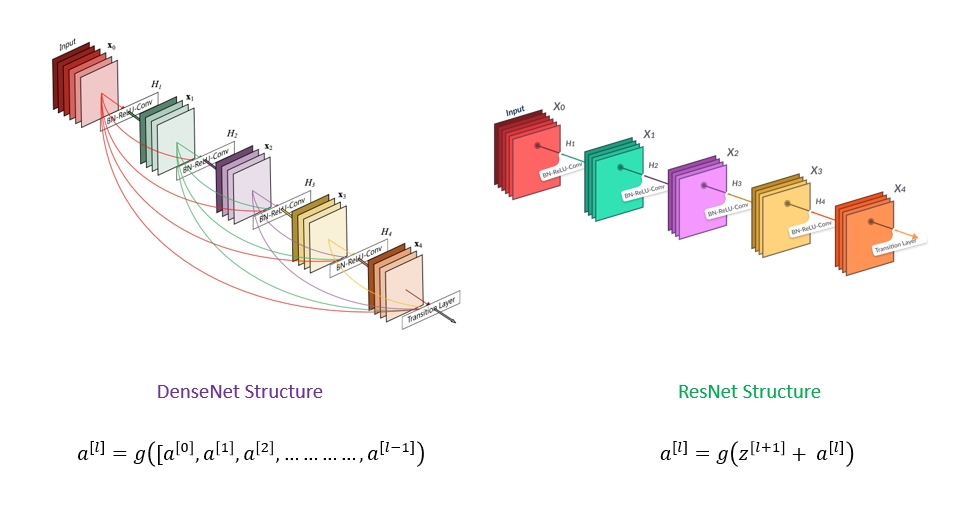
Inception v3 is a widely- utilized picture acknowledgment model that has been shown to attain greater than 78.1% accuracy on the ImageNet dataset. The model is the culmination of many ideas developed by multiple researchers over the years. It is based on the original paper:”Rethinking the Inception architecture for computer vision”by Szegedy, et. al.

The model itself is made up of symmetric and asymmetric building blocks, including convolutions, average pooling, max pooling, concats, dropouts, and fully connected layers. Batchnorm is used extensively throughout the model and applied to activation inputs. Loss is computed via Softmax.

**** A high-level diagram of the model is shown above:

**3.2.DenseNet**

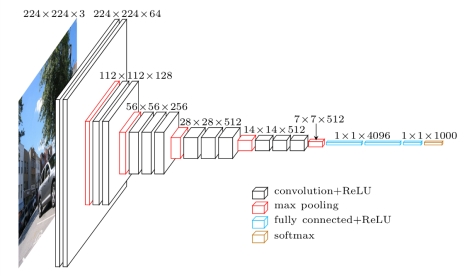
In neural networks, DenseNet is one of the new discoveries for visual object recognition. DenseNet is quite similar to ResNet with some fundamental differences. ResNet utilizes an added substance strategy (+) which joins the past layer (personality) with the new or next layer, while DenseNet connects (.) the yield of the past layer with the future layer developed specifically to improve the declined accuracy caused by the vanishing gradient in high-level neural networks. In simpler terms,because of huge distance between put[ut and input layers, the information vanishes before reaching its destination.

**Densenet Architecture :**DenseNet falls in the category of classic networks.This image shows a 5-layer dense block with a growth rate of k = 4 and the standard ResNet structure

An output of the previous layer acts as an input of the second layer by using composite function operation. This operation consists of thenon-linear activation layer, pooling layer, batch normalization, andconvolution layer. These connections mean that the network has L(L+1)/2 direct connections. L is the number of layers in the architecture.

The DenseNet has different versions, like DenseNet-121, DenseNet-160, DenseNet-201, etc. These numbers implies the number of layers in the neural network.

**3.3.VGG-16**

VGG16 is c convolution neural network used to win the ILSVR(ImageNet) completion in 2014. Till date, VGG16 is one among the best vision model architecture. Most and best unique thing about it is that, VGG16 focuses on getting convolution layer of 3x3 filter with a stride of 1and with same padding and always used maxpool layer of 2x2 filter having stride 2.

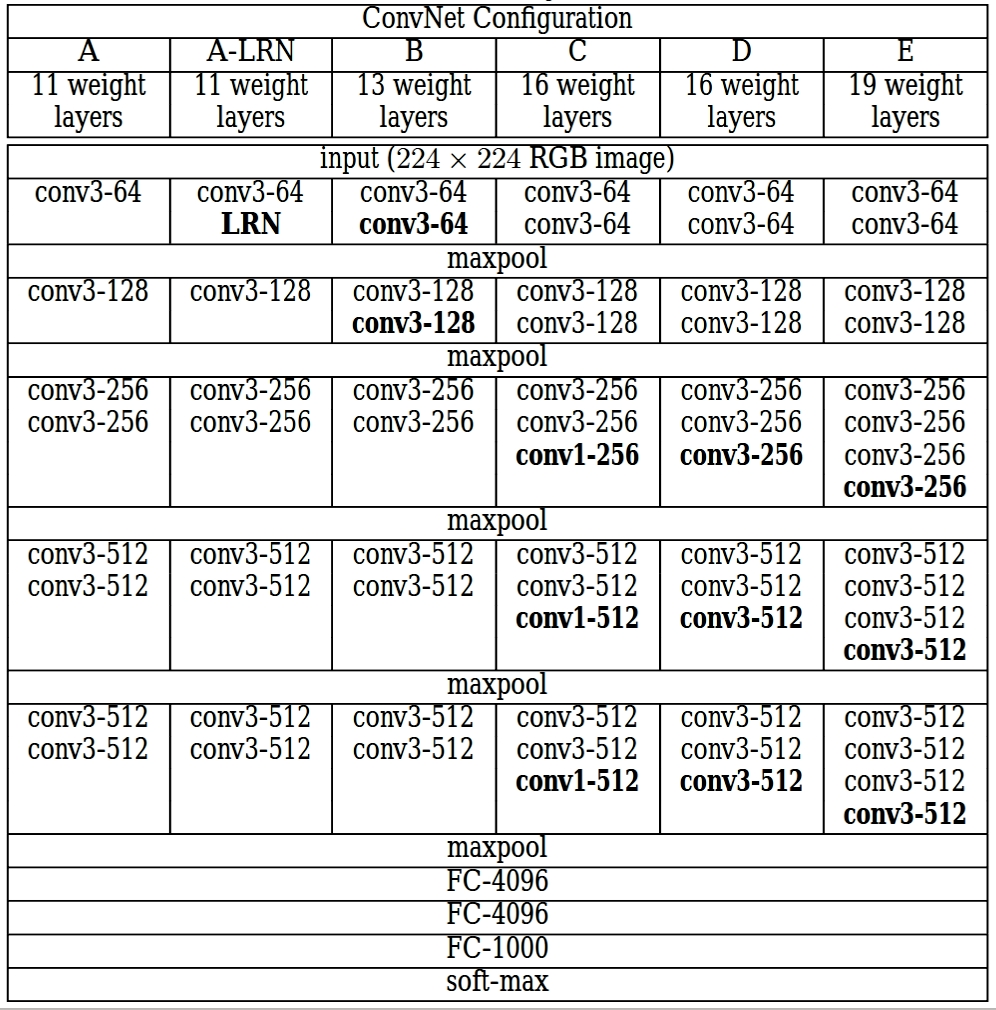
Throughout the whole architecture, this arrangement of max pool and convolutional layer is used very frequently. At the end in the architecture, it contains 2 fully connected layers after which softmax layer is used for getting output. In VGG16, 16 means, there are 16 layers having weights. With about 138 million (approx.) parameters, this is one of the pretty large network.

**3.4.VGG-19**

VGG19 is a kind of VGG model with only one difference, it contains 19 layers instead of 16 layers. VGG19 consist of 16 convolution, 3 fully connected, 5 MaxPool and 1 SoftMax layer.There are other variants of VGG like VGG11, VGG16 and others. VGG19 has **19.6 billion FLOPs**.

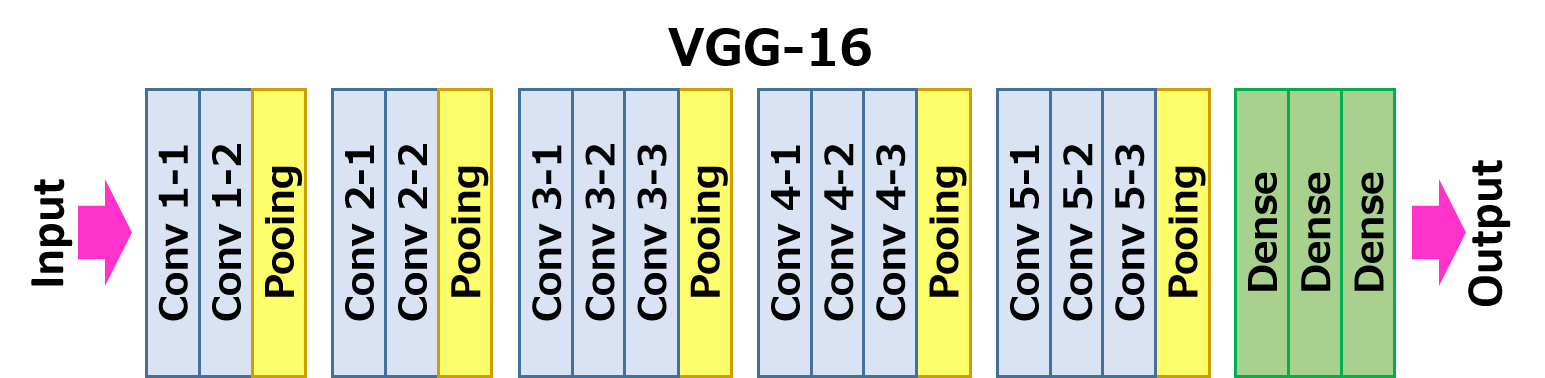
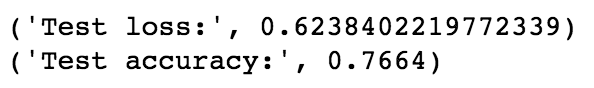
# **Architecture**

* A fixed size image of 224\*224 (in RGB format) is used as input in this network i.e. shape of matrix is (224,224,3).
* In preprocessing, from each pixel, we subtract the mean RGB value and compute it all over the whole training set.
* For covering the whole notion of image, kernels of size 3\*3having the stride of size 1 pixel is used.
* For maintaining the spatial resolution of the image, spatial padding technique is used.
* On windows of size 2\*2 having stride 2, max pooling is applied.
* For improving the model classification better and to improve the computational time, Rectified Linear layer (ReLu) is used for getting non-linearity.
* Two fully-connected layer of size 4096 is implemented and after that a layer having 1000 channels for 1000-way ILSVRC classification implements and then finally softmax function is implemented on final layer.

In the given table, the column C and E are for VGG16 and VGG19. And rest columns are the other variants of VGG model (varies on the basis of number of layers).

Transfer learning is very effective and fast way to predict the categories and data or image. This basically involves the concept of using pre-trained model. A pre-trained model is a model which is trained on large dataset or large scale image classification. Such models are much accurate than normal and basic layer models.

VGG 16 gave a test loss and test accuracy of



Such low accuracy could be possible because vgg16 is pre-trained model and it was trained over huge datasets. And above models consisting of 1,3,4 layers models were trained over small data, which require less layer.

**Result**

Model – (Damage Location).This is graph between accuracy and epoch on the left side which tells that our model has performed well on training data but not that good on validation data which clearly shows an accuracy of 68% and 92% on training data. This is a case of overfitting epoch 6.

If we talk about learning rate on training data is good if we check right side graph but learning rate is very high and unstable on our validation data.

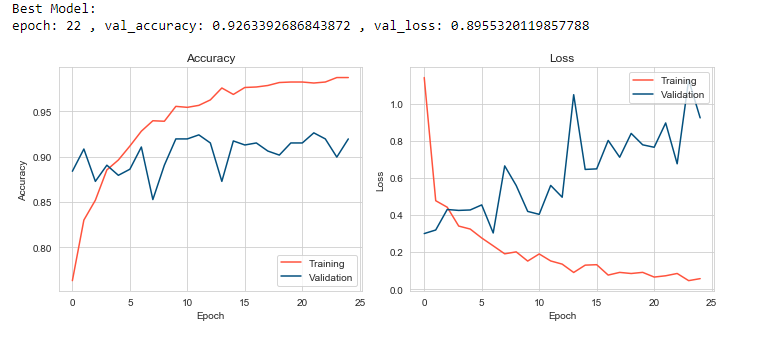


Figure- Graphs between accuracy vs epoch and loss vs epoch

Model(Damage or not) – In the above image you see our model vgg16 here has performed better over the datasetand there is less overfitting then the other model. Learning rate on training data is great but the there is a high learning rate on our test data which can eventually affect our results

Confusion matrix of model Damage or not –

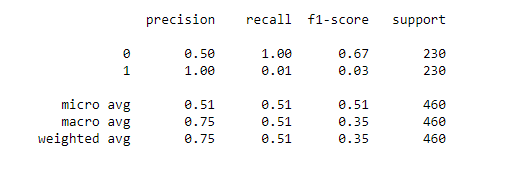


Figure- Confusion Matrix

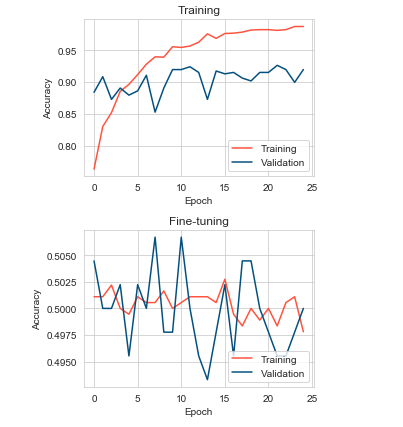
Generalization of model before and after finetune of damage or not- 

Figure- Graphs comparision

Now measures we can take to avoid overfitting which is the most general problem.

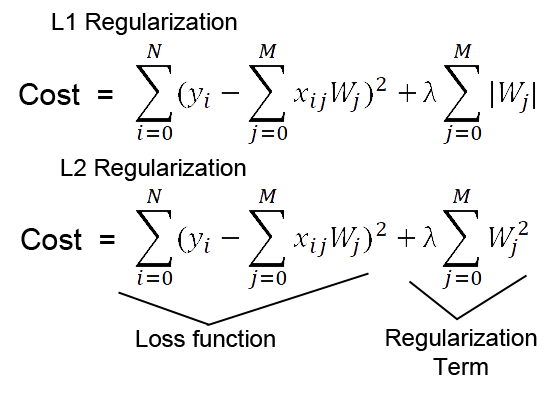
1. Regularization-

This should be the 1st method which should be involved to avoid overfitting. What this does is that it basically punishes our model as being complex. It balances out too high values of weight matrix.

There are 2 types of versions –

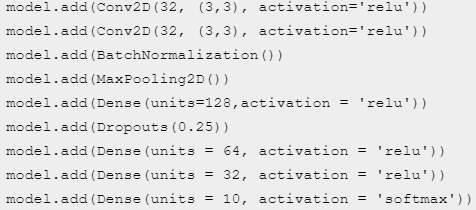
1. L1 – Least Absolute Deviation
2. L2 – Least Square Errors

The most commonly used is L1 as it reduces the weights of less usable features to zero.

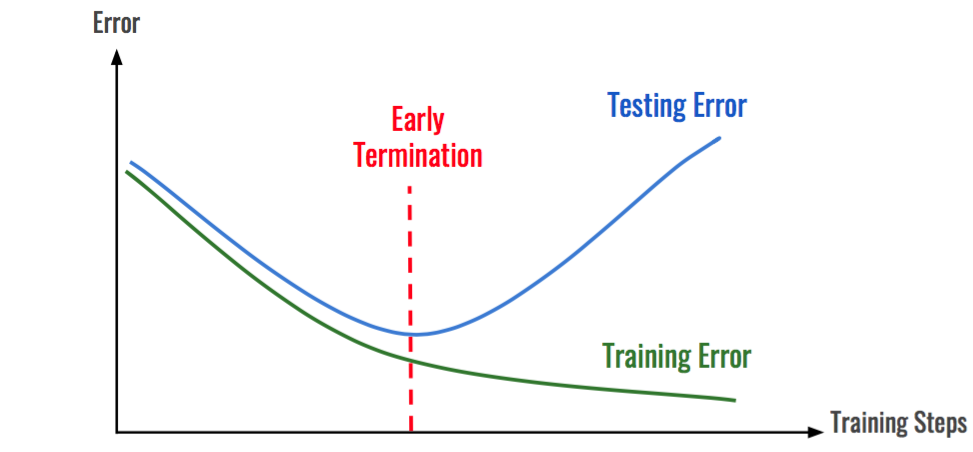


1. Dropout- it is basically a strategy of regularization just like L1 and L2 . Function Drop out layer is to deactivate some of the neurons.By doing so we can avoid overfiting.

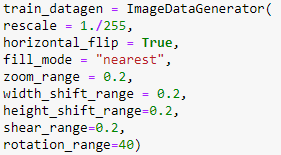
A simple code snippet for this looks like this-



1. Early stopping- It’s the process stopping our model from getting over trained . it provides the guidance to when to stop as over training can overfit our data.



1. Data Augmentation- Data Augmentation is a method which fundamentally builds the size and variety of the dataset. It comes in helpful at the occasions when there is a deficiency of information to prepare the model. The information expansion methods the restricted dataset and for certain minor corrections change it into a dataset which is bigger in size by manifolds. It has been in need for a long while and has demonstrated advantageous in different enormous just as limited scale projects. Particularly for our situation since there exists no predefined dataset of harmed vehicles, so we needed to get imaginative and gather our own dataset and gathering a dataset sufficiently enormous to prepare our model to approach flawlessness is a dreary work



Key points to note is that

1. Regularization helps the model to generalize well but can hurt our model performance
2. Dropout layer should be only used while feed forward and backpropagation and only during training data not while testing.

Now, main point while discussing our results we also used densenet in our model which is predicting the severity of the damage. We used densenet because this model is efficiently designed for low level feature extraction such as edges and pattern recognition, so to have the feature extraction of such minute damage we needed this model. Features of this model have been discussed above already and its architecture.

Accuracies of our model before fine-tune-

Figure- Comparison of Accuracies of our models

**Conclusion**

In this project, we performed various highlight extraction procedures in picture division to detect our damage part in the car and assess its damage intensity and area. We used edge detection techniques such as canny edge detector, harris corner detection and sobel operator. This was completely done in the first 50% of the portion of the project. After performing feature extraction we did data augmentation to increase the data, eventually increasing the accuracy of the model.

After that in the other half of the project we studied various algorithms that we used to categorize our images. We looked into some state of the art classifiers like Densenet, Inception and the VGG models. We also compared the accuracies of different layer model in CNN and tried to understand how it can help classify our damages parts and intensity of damage. We also tried to understand how we can frame our data to categories our damaged part and intensity of the damage. The potential which this project has is huge, in future work we plan to train create a model which can predict the price, to determine the price of repairs based on market value rates which shall be applicable pertaining to the intensity of the damage caused to the car.

**References**

* https://www.google.com/url?sa=t&source=web&rct=j&url=https://www.researchgate.net/publication/263619076\_Image\_Based\_Automatic\_Vehicle\_Damage\_Detection&ved=2ahUKEwjvspqVx6\_wAhUf83MBHUZ7CMAQFjABegQIBBAC&usg=AOvVaw0uQWW27deN5qrUnpiXBBA7
* https://www.google.com/url?sa=t&source=web&rct=j&url=https://openresearch-repository.anu.edu.au/handle/1885/11072&ved=2ahUKEwjvspqVx6\_wAhUf83MBHUZ7CMAQFjACegQIGxAC&usg=AOvVaw1wvkHgV6aacmc-P9YCL3fd
* https://www.google.com/url?sa=t&source=web&rct=j&url=https://medium.com/analytics-vidhya/vggnet-architecture-explained-e5c7318aa5b6&ved=2ahUKEwiviL28x6\_wAhWS63MBHSuqBxAQFjAJegQIFRAC&usg=AOvVaw2wiPKDt7cRd2QpM0GuHKkW
* https://www.google.com/url?sa=t&source=web&rct=j&url=https://www.kaggle.com/blurredmachine/vggnet-16-architecture-a-complete-guide&ved=2ahUKEwiviL28x6\_wAhWS63MBHSuqBxAQFjAeegQIExAC&usg=AOvVaw0HsL5nEa0KR1l664hzjjEU
* https://www.google.com/url?sa=t&source=web&rct=j&url=https://towardsdatascience.com/canny-edge-detection-step-by-step-in-python-computer-vision-b49c3a2d8123&ved=2ahUKEwjFva7kx6\_wAhVOIbcAHTotDTMQFjAYegQIIxAC&usg=AOvVaw1JbQI-UYduuWgVSBESrEja
* https://doi.org/10.1145/3406601.3406651
* https://www.google.com/url?sa=t&source=web&rct=j&url=https://www.ee.iitb.ac.in/student/~kalpesh.patil/material/car\_damage.pdf&ved=2ahUKEwilsbz1x6\_wAhU1gtgFHUhSDLkQFjAEegQICRAC&usg=AOvVaw2P053cO-cvo4ESeQYc2ALn
* https://www.google.com/url?sa=t&source=web&rct=j&url=https://ieeexplore.ieee.org/iel7/6287639/8948470/08950115.pdf&ved=2ahUKEwilsbz1x6\_wAhU1gtgFHUhSDLkQFjAOegQIBhAC&usg=AOvVaw3aF5rStj7oGFdJYtC6uQ1m
* https://www.tech-quantum.com/implementing-drop-out-regularization-in-neural-networks/#:~:text=%20The%20following%20steps%20will%20be%20used%20to,%28D%20%E2%80%93%20dropout%29%20that%20has%20the...%20More%20
* https://analyticsindiamag.com/everything-you-should-know-about-dropouts-and-batchnormalization-in-cnn/
* https://medium.com/analytics-vidhya/the-perfect-fit-for-a-dnn-596954c9ea39#:~:text=Measures%20to%20prevent%20overfitting%201%20Decrease%20the%20network,neural%20networks%20from%20overfitting.%20...%205%20Early%20Stopping
  + [https://w](http://www.tensorflow.org/tutorials/images/data_augmentation)ww.[tensorflow.org/tutorials/images/data\_augmentation](http://www.tensorflow.org/tutorials/images/data_augmentation)
  + [https://medium.com/@RaghavPrabhu/cnn-architectures-lenet-alexnet-vgg-googlenet-and-resnet-](https://medium.com/%40RaghavPrabhu/cnn-architectures-lenet-alexnet-vgg-googlenet-and-resnet-)7c81c017b848#:~:text=VGG%2D16%20is%20a%20simpler,2%20with%20stride%20of%202.&text=The% 20winner%20of%20ILSVRC%202014,also%20known%20as%20Inception%20Module.
  + https://towardsdatascience.com/the-4-convolutional-neural-network-models-that-can-classify-your- fashion-images-9fe7f3e5399d
  + https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural- networks/
  + https://towardsdatascience.com/image-recognition-with-machine-learning-on-python-convolutional- neural-network-363073020588
  + https://towardsdatascience.com/wtf-is-image-classification- 8e78a8235acb#:~:text=The%20convolutional%20neural%20network%20(CNN,the%20scenes%20in%20i mage%20classification.
  + https://beta.vu.nl/nl/Images/stageverslag-deijn\_tcm235-882561.pdf [https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/](http://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/)[9]https://[www.analyticsvidhya.com/blog/2020/03/beginners-guide-random-forest-](http://www.analyticsvidhya.com/blog/2020/03/beginners-guide-random-forest-)
* https://homepages.inf.ed.ac.uk/rbf/HIPR2/sobel.htm
* https://docs.opencv.org/master/da/d22/tutorial\_py\_canny.html
* [https://www.mygreatlearning.com/blog/feature-extraction-in-image-processing/](http://www.mygreatlearning.com/blog/feature-extraction-in-image-processing/)
* [https://www.analyticsvidhya.com/blog/2019/04/introduction-image-segmentation-techniques-python/](http://www.analyticsvidhya.com/blog/2019/04/introduction-image-segmentation-techniques-python/)
* https://opencv-python tutroals.readthedocs.io/en/latest/py\_tutorials/py\_feature2d/py\_features\_harris/py\_features\_harris.html