Analysis of damaged cars using multiple convolutional neural networks

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering degree in Computer Science and Engineering

by

V.SANTHOSH

(39110900)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade "A" by NAAC | 12B Status by UGC | Approved by AICTE
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI - 600119

APRIL - 2023

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade "A" by NAAC| 12B Status by UGC | Approved by AICTE
(Established under Section 3 of UGC Act, 1956)
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI– 600119
www.sathyabamauniversity.ac.in

DEPARTEMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **V.SANTHOSH (Reg. No. 39110900),**who carried out the Project Phase-2 entitled **"ASSESSING CAR DAMAGE"** under my supervision from January 2023 to April 2023.

Internal Guide
Mrs.Dr. Bangaru kamatchi M.E., Ph.D.,

Head of the Department
Dr. L. LAKSHMANAN, M.E., Ph.D.,

SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY
DEPARTMENT OF CSE

Submitted for Viva-voce Examination held on 19.4.2023

Internal Examiner **External Examiner**

DECLARATION

I, **V.SANTHOSH (39110900),** hereby declare that the Project Phase-2 Report entitled **“ASSESSING CAR DAMAGE”** done by me under the guidance of **Mrs.Dr. BANGARU KAMATCHI M.E., Ph.D.,** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in **Computer Science and Engineering**.

DATE: 19.4.2023
PLACE: Chennai

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to the **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala M.E., Ph. D**, **Dean**, School of Computing, and **Dr. L. Lakshmanan M.E., Ph.D., Head,** Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Mrs.Dr. BANGARU KAMATCHI M.E., Ph.D.,** for her valuable guidance, suggestions, and constant encouragement that paved the way for the successful completion of my phase-2 project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

Visual image classification is a research area that involves both computer vision and machine learning. The task of visually classifying an object consists in assigning an object to a category, or set of categories the object belongs to. Traditionally, visual classification tasks are performed using a two layered system, made up of a first layer featuring an out-of-the-shelf feature extractor and detector, and a second classifier layer. In most recent years, convolutional neural networks have been shown to outperform such previously used systems. Cars have a paramount role in today's world, and being able to automatically classify damages in cars is of great interest specially to the car insurance industry. Car insurance companies deal with car inspections on a daily basis. Such inspections are a manual, lengthy and sometimes faulty processes. Processes that bring costs and inconveniences to costumers and insurance companies alike. Even though the total replacement of such manual inspection processes might still be far away, developing systems to aid, accelerate or enhance the process might bepossible with today's technology.

Nowadays, the proliferation of automobile industries is directly related to the number of claims in insurance companies. Those companies are facing many simultaneous claims and solving claims leakage. In Advanced Artificial Intelligence (AI), machine learning and deep learning algorithms can help to solve these kinds of problems for insurance industries.

Manual estimation of damages in fields like construction, vehicular accidents has beenthe mainstay of the insurance business.

However, such methods are replete with biases and inaccurate estimations.

This project deals with estimating car damage, primarily with auto insurers as our keypotential customers.

For this purpose, three distinct Transfer Learning approaches are used which detect thepresence of damage, location, and severity of the damage.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 GENERAL BACKGROUND

Automatically detecting damages in images containing cars is a special case of an image classification task because, at it's most basic level, detecting damages in images consists of assigning an image to a particular category or set of categories. There is a lot of research done in image classification but, there aren't many works, that I am aware of, in car visual damage detection. Nevertheless, being able to automatically detect damages in cars is a research topic that has many possible real-world applications. Car insurance companies and car rentals have to deal with car damages on a daily basis. It often happens that cars have to be inspected for damages, often in circumstances that are inconvenient for customers, and costly to the companies themselves. It is therefore important to be able to automate car damage detection, making it both more convenient and cheaper. This report starts by giving a broad overview of the current existing solutions for the problem of automatically identifying car damages in images. Solutions and strategies to solve problems that are only similar to the ones addressed here, will often be mentioned due to the lack of more specific research on the field of automated car damage detection. Some of these solutions, mainlythe more promising ones, are later applied to the problem of automatically identifying damages in car images. A comparison of results with the ones obtained by previous works on the same problem will be given whenever possible. The methodology used throughout this project consists of adapting solutions that are known to work in a variety of image classification problems, to the particular problem of visually identifying damages in cars and then evaluating their performance relatively to other existing solutions. Several classification systems will be tested, favouring the ones that can be used in practice. The dataset used to validate and evaluate the performance of the validate and evaluate performance of the developed solutions will be composed of images previously gathered from search engines, labelled for training and testing purposes.Do you know that the insurance industry is one of the first industry which invested in innovation, the latest technology and AI? In today's world, when the rate of car accidents are increasing depending on expanding car industries, car insurance companies waste millions of dollars annually, due to claims leakage. The sense of AI technology based on machine and deep learning can solve problems such as analyzing and processing data, detecting frauds, lessening risks and automating claim process in insurance industries . So, insurance organizations are looking for faster damage assessment and agreement of claims

The manual inspection of vehicles for external damages is a process known to most and particularly prevalent in the vehicle rental industry, where the accuracy of damage detection is of high importance. Not only are such visual inspections time consuming for lessee as well as a rental company, the high level of complaints often due to unexpected supplementary charges for subjectively assessed vehicles damages is leading to negative customer experiences as well as an additional cost for all parties involved for further investigation and formal responses before an acceptable resolution can be achieved. These shortcomings call for an objective solution. With the significant progress made in object detection and instance segmentation, this field of computer vision provides a promising solution. With the possibility to replace manual inspections with visual automated ones, the dependency on the expertise of the assessor is removed. Furthermore, artificial damage recognition reduces the time and effort required for an inspection while also providing a reproducible and objective assessment for all parties involved. While fully supervised models, such as Mask R-CNN have proven to achieve high accuracy rates for the detection of objects the required effort to annotate a large number of images to achieve these results requires high levels of effort and time.

Currently, AI is advancing at a great pace and deep learning is one of the contributors to that. It is good to understand the basics of deep learning as they are changing the world we live. Deep learning is a sub-field of machine learning dealing with algorithms inspired by the structure and function of the brain called artificial neural networks.In neural networks. In other words, It mirrors the functioning of our brains Deep learning algorithms are similar to how nervous system structured where each neuron connected each other and passing information. One of the differences between machine learning and deep learning model is on the feature extraction area. Feature extraction is done by human in machine learning whereas deep learning model figures out by itself Today, in the car insurance industry, a lot of money is wasted due to claims leakage. Money lost through claims management inefficiencies that ultimately result from failures in existing processes (manual and automated). In other words, it's the difference between what you did spend and what you should have spent on a claim. The cause can be procedural, such as from inefficient claim processing or improper/errant payments, or from human error, such as poor decision-making, customer service, or even fraud. Claim Leakage is often discovered through an audit of closed claim files.

1.2 OBJECTIVE

Cars have a central role in today's world. Many people use cars every single day. There are businesses that depend on cars, and some of them base theiractivity in dealing with car damages and accidents. Non automated visual inspection of cars is a common task in some businesses. This is mainly the case of insurance companies. Cars are inspected both when a new coverage is bought andwhen an issue is reported to the insurance company. In both situations, inspections are only done by sampling. This happens because the cost of having an expertdrive down to an often damaged vehicle, in order to inspect it is too high. It also causes delays for customer and company alike. This is especially important in cases where the insurance is not bought in one of the insurance companies dealerships, e.g. car insurance bought through the Internet.

If the car for which the insurance was bought had to be inspected by an insurance company, the whole purpose of selling insurance through the Internet would be defeated; The customer would no longer have the convenience of quickly buying aninsurance, since he would have to wait for the inspection to be performed before the effectiveness of the insurance was granted. On the other hand, the insurance company would no longer have the benefit of reduced costs in selling its products through these channels, because an expert would have to drive down to the insured car location, possibly a remote location, in order to perform the inspection. Both these inconveniences would be mitigated if the client uploaded some photos of the car to be automatically inspected by a system that could assert if the car has damages or not. Insurance companies often mitigate the problem by reducing the number of inspections they make. This is achieved either by not making them at all, or by making them by sampling, in an attempt to reduce fraud while not spendingtoo much money inspecting it.

Although an automated inspection might not be as effective as an inspection performed by an expert, it makes it possible to inspect all the vehicles, instead of just a few. On the other end, performing such inspections alongside with expert inspections might contribute to detect systematic biases, making it easier to discover potential fraudulent inspection processes or mistakes. The possibility of classifying and distinguishing between different kinds of damages opens up the door for other more complex car damage related problems to be solved in an automated way. Other regression and classification problems that might include repair cost estimation or damaged parts estimation, problems of the uttermost importance for car insurance and car repair companies. These processes are also susceptible of fraud, and are also not automated nowadays.

Even tough cars are very important to many people, they have been somehow neglected by the computer vision research community. Cars are very challenging to work with in computer vision. They present great variation in shape and form by slight variations of viewpoint. This, combined with the fact that cars usually have highly reflective metal bodies that make them very susceptible to inter object reflection, makes the problem of automatically of detecting damages in car images a very challenging one. This is especially the case if the damages are relatively small, when they are easily mistaken by reflections. There is some research done in the field of automatic damage detection in vehicles. The approach taken often involves the usage of CAD models to estimate image perspective and assert differences between what's in the image and what the differences between what's in the image and what the image is expected to have . On the other hand, there is the research done in the broader field of image classification. This research often takes a different strategy in solving image classification problems. Image classification technology has endured tremendous changes in the last decade. We now have image classification systems that can allegedly outperform human beings in some tasks, a feat that could not be achieved by decades long research with traditional approaches. While the non traditional approach followed made perfect sense when the state of the art image classification systems yielded results that were not good enough for this tasks, it now makes complete sense to apply the more performant and up to date state ofthe art, technology to tackle the same problems, hopefully with better results. Automated car damage detection using images is, therefore, an interesting research topic both due to the lack of work done in the field and due to the hope that new state-of-the-art methods might bring serious performance improvements. Also, due to the importance cars have in many places and the fact that cars present interesting unsolved challenges that state-of-the-art image classification technologies might now be able to overcome, this make this an even more interesting research topic.

In this document, we'll see the development of a prototype of a system, capable of identifying, and possibly locating damages in car images and evaluate its performance by comparing it to existing systems. A dataset will be gathered for the purpose of training the system and evaluating its performance. This dataset will be built by gathering images from search engines and other publicly available datasets.

1.3 DEEP LEARNING

Deep-learning networks are distinguished from the more commonplace single-hidden layer neural networks by their depth; that is, the number of node layers through which data must pass in a multistep process of pattern recognition. Earlier versions of neural networks such as the first perceptron were shallow, composed of one input and one output layer, and at most one hidden layer in between. More than three layers (including input and output) qualify as "deep"learning. So deep is not just a buzzword learning. So deep is not just a buzzword to make algorithms seem like they read Sartre and listen to bands you haven't heard of yet. It is a strictly defined term that means more than one hidden layer. In deep-learning networks, each layer of nodes trains on a distinct set of features based on the previous layer's output. The further you advance into the neural net, the more complex the features your nodes can recognize since they aggregate and recombine features from the previous layer. This is known as feature hierarchy, and it is a hierarchy of increasing complexity and abstraction. It makes deep-learning networks capable of handling very large, high-dimensional data sets with billions of parameters that pass through nonlinear functions.

Above all, these neural nets are capable of discovering latent structures within unlabeled, unstructured data, which is the vast majority of data in the world. Another word for unstructured data is raw media; i.e., pictures, texts, video, and audio recordings. Therefore, one of the problems deep learning solves best is in processing and clustering the world's raw, unlabeled media, discerning similarities and anomalies in data that no human has organized in a relational database orever put a name to. For example, deep learning can take a million images, and cluster them according to their similarities: cats in one corner, ice breakers in another, and in a third all the photos of your grandmother. This is the basis of so-called smart photo albums. Deep-learning networks perform automatic feature extraction without human intervention, unlike most traditional machine-learning algorithms.

Given that feature, extraction is a task that can take teams of data scientists years to accomplish, deep learning is a way to circumvent the chokepoint of limited experts. It augments the powers of small data science teams, which by their nature do not scale. When training on unlabeled data, each node layer in a deep network learns features automatically by repeatedly trying to reconstruct the input from which it draws its samples, attempting to minimize the difference between the network's guesses and the probability distribution of the input data itself. Restricted Boltzmann machines, for example, create so-called reconstructions in this manner. In the process, these neural networks learn to recognize correlations betweencertain relevant features and optimal certain relevant features and optimal results – they draw connections between feature signals and what those features represent, whether it be a full reconstruction, or with labeled data. A deep-learning network trained on labeled data can then be applied to unstructured data, giving it access to much more input than machine-learning nets.

1.4 NEURAL NETWORKS

A neural network is a series of algorithms that endeavors to recognize underlying relationships in a set of data through a process that mimics the way the human brain operates. In this sense, neural networks refer to systems of neurons, either organic or artificial in nature. Neural networks can adapt to changing input; so the network generates the best possible result without needing to redesign the output criteria. The concept of neural networks, which has its roots in artificial intelligence, is swiftly gaining popularity in the development of trading systems. A neural network works similarly to the human brain's neural network. A "neuron" in a neural network is a mathematical function that collects and classifies information according to a specific architecture. The network bears a strong resemblance to statistical methods such as curve fitting and regression analysis. A neural network contains layers of interconnected nodes. Each node is a perceptron and is similar to multiple linear regression. The perceptron feeds the signal produced by a multiple linear regression into an activation function that may be nonlinear. 14 In a multi-layered perceptron (MLP), perceptron are arranged in interconnected layers. The input layer collects input patterns. The output layer has classifications or output signals to which input patterns may map. Hidden layers fine-tune the input weightings until the neural network's margin of error is minimal. It is hypothesized that hidden layers extrapolate salient features in the input data that have predictive power regarding the outputs. This describes feature extraction, which accomplishes a utility similar to statistical techniques such as principal component analysis.

1.5 CNN

Convolutional Neural Network is an algorithm of Deep Learning. That is used for Image Recognition and in Natural Language Processing.Convolutional NeuralNetwork (CNN)

Network (CNN) takes an image to identify its features and predict it. It captures the spatial and temporal dependencies of the image. Each CNN layer learns filters of increasing complexity. The first layers learn basic feature detection filters like edges, corners. The middle layer learns filters that detect parts of objects. The last layers have higher representations they learn to recognize full objects in different shapes and positions. Suppose, when you see some image of a Dog, your brain focuses on certain features of the dog to identify. These features may be the dog's ears, eyes, or it may be anything else. Based on these features your brain givesyou a signal that this is a dog. Similarly, Convolutional Neural Network processes the image and identifies it based on certain features. Convolutional Neural Network is gaining so much popularity over artificial neural networks.

Steps in Convolutional Neural Network In a Convolutional Neural Network, there are basically the following steps -

1. Convolution Operation.
2. ReLU Layer.
3. Pooling.
4. Flattening.
5. Full Connection.

Generally, a convolutional neural network architecture generally has several components:

- A convolution layer - you can think of this layer as "what relevant featuresare we picking up in an image?" In a convolutional neural network, we have multiple convolutional layers that extract low to high-level features depending on what specific layer we are focusing on. To give an (over-simplified) intuition, earlier convolutional layers pick up lower-level features (i.e. like lines and edges) while later convolutional layers pick up higher-level features based on inputs from lower-level features (i.e. shapes, structures) - analogous to how vision works in the human brain.

- A pooling layer - convolutional neural networks are typically used for image classification. However, images are high-dimensional data - so we would prefer to reduce the dimensionality to minimize the possibility of overfitting. Pooling essentially reduces the spatial dimensions of the image based on certain mathematical operations such as average or max-pooling (there’s a nice graphic here). We generally incorporate pooling since it

(1) generally, acts as a noise suppressant

(2) makes it invariant to translation movement for image classification and

(3) helps capture essential structural features of the represented images without being bogged down by the fine details.

- Fully-connected layer - You can think of a series of convolution and pooling operations as dimensionality reduction steps prior to passing this information over to the fully connected (Dense) layer. Essentially, what the fully connected layer does is that it takes the "compressed" representationof the image and it tries to fit a basic NN (multi-layer perceptron) when doing classification.

S.NO	CNN Layers
1.	Convolutional layer
2.	Pooling Layer
3.	Fully connected Layer

Table 1.1 CNN layers used

The change in paradigm that the usage of NNs encompasses is a very important one. Prior to the usage of NNs in image classification, the practitioner had to use explicitly coded algorithms for detecting features. This was a job that often involved a lot of work. Machine Learning was not used throughout the whole system, only the last layer of the system, the classifier, had the capacity to learn and adapt itself to the specific problem. NNs, and especially CNNs, dramatically change that. CNN's extract features from images and learns how to do it. The practitioner doesn't need to craft complicated hard-coded algorithms to extract those features. Since the introduction of the first CNNs in 2012, all the winning teams for the classification task have used different types of CNNs. CNN's systems now clearly outperform other image classification systems such as the ones described. State-of-the-art CNNs have even crossed the boundary of what is considered to be the error rate for human beings in image classification tasks. Ever since it's introduction in 2012, CNNs have been making steady progress in decreasing the error rate in both tasks. This is due to the introduction of new strategies that further improve the convergence and training speed of CNNs. CNNs are a particular kind of NNs that give their name to the fact that they make extensive use of Convolutional Layers.

Convolutional Layers consist of a learnable filter of fixed size (kernel) to be applied to images. The name comes from the fact that at each forward pass, the learned filter is convolved with the image resulting on a 2-dimentional map. Another type of layer widely used in CNNs are the pooling layers. These layers perform a type of down-sampling on the input signal. There are various types of Pooling layers, max-pooling being the most common. ReLUs or Rectified Linear Units, are also widely used in CNNs. ReLUs are units that apply a function similar to Equation 2.1. There are often alternatives for these types of units, Equation 2.2 and 2.3 show twoof them.

ReLU : $f(x) = \max(0, x)$ (2.1)

Tanh : $f(x) = \tanh(x)$ (2.2)

Sigm : $f(x) = (1+e^{-x})^{-1}$ (2.3)

Fully Connected Layers are used too, as in more standard NNs. These layers consist of an array of neurons, each of which is connected to every single output of the previous layer. Loss Layers are typically used at the end of the CNN to figure out the penalty for a given output and to provide feedback used by the CNN to learn. In addition to these more standard set of layers, modern CNNs make use of a very extensive set of techniques and strategies that greatly enhance their performance. Dropout is one of such techniques. It consists on temporarily deactivating some neurons within the net, preventing it from over-fitting. This temporary deactivation of neurons is typically applied only to 8 Fully-connected layers. CNNs nowadays also make extensive use of GPU processing power that allows for a faster training, an idea pioneered in 2012 that is now a standard. Evidence shows that one way of improving a CNN performance is by stacking more layers and making the network effectively deeper The problem that often arises is that as more layers are stacked, the performance saturates and eventually starts to decrease at some point. To address this issue, the idea of deep residual networks has been proposed. Residual Networks try, with minor changes to the architecture, and with virtually no impact on performance, to approximate residual functions instead of standard functions.

If we have a CNN with input x and we want the output to approximate a function O(x), we can instead, approximate $F(x) = O(x)-x$ and then compute O(x) from F(x) since $O(x) = F(x) + x$ as shown in Figure 2.2. Evidence shows that CNNs train better and converge faster when the approximated function is in fact F(x), a residual function. This approach opens door for usage of even deeper and more powerful models. Some pre-processing techniques are also used to improve the data fed to the CNN. These techniques include normalization and zero-centering of the data. These techniques are mainly used in order to overcome the vanishing and exploding gradient problem, that often affects NN systems' training. Both problems arise when models are trained using gradient-based methods. Gradients are often computed using the chain rule. In case of near-zero gradients, this lead to several very small numbers being multiplied together in order to calculate the changes in the last layers. This causes the last layers to suffer virtually the last layers. This causes the last layers to suffer virtually no change at all in its outputs as the training progresses. The opposite happens when activation functions have high derivative values, leading 9 Related Work to the uncontrolled increase of the gradient of the last layers. As NN systems tend to require a lot of examples, better results are obtained when data augmentation techniques are used. CNN frameworks often incorporate mechanisms to augment data. Cropping is a very simple technique where several crops of a single image are fed to the NN, augmenting the dataset by a factor equal to the number of crops. Rotations [KSH12] of the image are also used, augmenting the dataset by a factor equal to the number of rotations done. The techniques summarized here, and others, together with a very complete set of open-source tools, foster the usage of CNNs, allowing for the fast creation of systems that perform very well in noisy datasets, clearly outperforming older image classification systems.

CHAPTER 2

LITERATURE REVIEW

2.1. INFERENCES FROM LITERATURE SURVEY

Deep learning is an efficient method used for classification. Kalpesh Patil, et. al. in [1] have used the concept of deep learning in order to classify car damage. The model used is trained on CNN directly. The preprocessing includes the steps of domain-specific pre-training followed by fine-tuning. The paper has conducted a combined and separate study of Transfer Learning and Ensemble Learning. The research has a setback of unavailability of a proper dataset which has resulted in creation of dataset by annotating images. The use of Convolution Autoencoder based pre-training followed by supervised fine-tuning and transfer learning is a novelty factor of this research.

Deep learning methodology can also be used for detecting presence or absence of damage and conducting further analysis. The researchers in [2] have applied this in the field of automotives. In this paper, CNN is used for object recognition. The task of classification has been performed on Damaged Vehicle dataset. Mask RCNN is used for segmenting, decomposing and sub-dividing the various instances of Machine Learning. The scope of research is limited to a particular dataset. Extensive research on new data can be performed for testing the quality of the model. Yet the fact that it is an automated system that can classify the damaged vehicle and predict how the damage has occurred remains a unique factor of this research.

The concept of faster R-CNN can be helpful for real-time object detection with Region Proposal Networks. This concept is implemented in [3]. RPN (Region Proposal Network) is trained end-to-end to generate high-quality region proposals, which are used by Fast R-CNN for detection. RPN and Fast R-CNN are merged into a single network by sharing their convolutional features using neural networks with attention mechanisms. The RPN component is essentially used for the unified network to focus on a particular object. The research does not include exploitation and preprocessing on the data. This process could have been used to improve results. The research has built a unified, deep learning

Computer Vision Technology can be used for assessment of damage to an object. Xianglei Zhu, et. al. in [4] have developed an unified intelligent framework based on this concept. This paper uses RetinaNet algorithm to identify damaged parts. The accuracy with this algorithm is improved. Mask R-CNN is adopted for the identification of vehicle parts, the damaged parts are determined by the method of sampling, and the time complexity is greatly reduced. The accuracy achieved in this research can be improved in order to get better results. A combination of characteristics of vehicle damage data and suitable data can further strengthen this system. The research has successfully reduced time complexity in damage detection and the use of RetinaNet gives good accuracy in damage detection.

The use of Improved Mask RCNN can be used for vehicle damage detection. In [5], this approach is followed using Segmentation algorithm. A deep learning approach is used to detect vehicle-damage for compensation problem in traffic accidents. The algorithm has achieved good detection results in different scenarios. Regardless of the strength of the light, the damaged area of multiple cars, or a scene with an overly high exposure, the fitting effect is better and the robustness is strong. The limitation of this research lies in the mask instance segmentation. In many cases the obvious damage is not considered and segmented leading to inaccurate results. This research contributes to detection of damage of vehicles in a more efficient method through improved Mask algorithm.

Convolutional Neural Network (CNN) is a widely used algorithm for the purpose of classification problems. This method is used by Jeffrey de Deijn in [6]. The research was able to detect car damage with fairly accurate results. The type, location and size of damage is detected with moderate accuracy. The addition of Ensemble learning could have further improved the results from this research. The use of ConvNets to detect car damage detection and transfer learning are the novelty factors of this research.

[7] Now days claiming and settlement of vehicle insurance is done through online, where customers are allowed to upload the image of damaged vehicle taken using their mobile phones and request for claim. However there can be chance of repeated claim for the

repeated claim for same case which can be loss for insurance company. So the main objective is to develop an Anti-Fraud checking system to process the request and speed up the insurance claim process. YOLO detector is used as object detection framework. YOLO detects the damage on vehicle by learning features though regression in 4 coordinates. To extract local features, pre-trained VGG16 object recognition model is employed as a feature extractor. Global deep features and color histogram are present in Global feature.

[8] In this paper the authors have suggested an automated system that can classify the damaged vehicle and predict how the damage has occurred. Convolution Neural Network (CNN) can be used for understanding, detecting and analyzing various classes of damage in the minor and major parts of car. The damages canbe of any types like bumper dent, door dent, glass shatter, head lamp broken, tail lamp broken, scratch and smash. CNN is used for object recognition task, in the proposed system it is being applied in the specific context of car damage recognition. Classification task is done on Damaged Vehicle dataset. This allows us to separate different objects and give bounding boxes, classes and masks. Once after locating bounding boxes, it can be colored and individually extract the features.

[9] Due to the development of deep learning, in recent years, the field of computer vision grows rapidly. A large amount of computer vision technologies have been applied in actual problems. At present, the industry of vehicle damage assessment requires a lot of manpower, and new automatic intelligent damage assessment technology can greatly reduce industrial costs. In this paper, a framework of intelligent vehicle damage assessment algorithm based on object detection technology and image classification technology is proposed. This algorithm can automatically identify the damage position, type and degree according to photos provided by users, so as to offer appropriate maintenance price and reach the accuracy that can meet actual application requirements.

[10] Traffic congestion due to vehicular accidents seriously affects normal travel, and accurate and effective mitigating measures and methods must be studied. To resolve

resolve traffic accident compensation problems quickly, a vehicle-damage-detection segmentation algorithm based on transfer learning and an improved mask regional convolutional neural network (Mask RCNN) is proposed in this paper. The experiment first collects car damage pictures for preprocessing and uses Labelme to make data set labels, which are divided into training sets and test sets. The residual network (ResNet) is optimized, and feature extraction is performed in combination with Feature Pyramid Network (FPN). Then, the proportion and threshold of the Anchor in the region proposal network (RPN) are adjusted. The spatial information of the feature map is preserved by bilinear interpolation in ROIAlign, and different weights are introduced in the loss function for different-scale targets. Finally, the results of self-made dedicated dataset training and testing show that the improved Mask RCNN has better Average Precision (AP) value, detection accuracy and masking accuracy, and improves the efficiency of solving trafficaccident compensation problems.

[11] Objective is to investigate the vehicle frontal body damage detection using roadway surveillance camera images. It implements the deep learning technique and image classification methods to identify the vehicle status. First need to detect the vehicles within the raw images, secondly by using the cropped images it represents the deep features of the vehicle, finally by applying the classification operations to the damaged © 2019 IJSRET 1898 International Journal of Scientific Research & Engineering Trends Volume 5, Issue 6, Nov-Dec-2019, ISSN (Online): 2395-566X vehicle. Average accuracy results of damaged and nondamaged class through SVM classifiers using reported features on test images. Here there is a distinct accuracy can be established between damaged and non-damaged class, sothat it would not mislead overall performance results. It requires larger dataset for getting minimum accuracy results, many models need to encounter.

The authors in [12] create our three datasets based on 1150 car damaged images, which consist of different types of car damage when there is no openly obtainable dataset for car damage classification. To reach the classification procedure, they need to have our three datasets which are manually collected from google images using selenium, which must include respective images without cars, with undamaged cars, and with more undamaged cars, and with damaged cars, and the ImagNet dataset. While they were dealing with small datasets, the authors were required to use data augmentation to artificially expand and adapt our datasets to improve their performance and decrease their tolerance to the over-fitting issue during training. Therefore, the authors utilized random rotation, zooming, dimension shift, and flipping renovation plans to differ the generated data. We explain more about our three datasets in the dataset subsection.

2.2 OPEN PROBLEMS IN EXISTING SYSTEM

According to [13], they proposed an end to end system with a transfer learning based on CNN models on ImageNet dataset to perform different tasks of localization and detection but not calculate the level of damage part. The similarity in paper [14], they also trained CNN model with both of transfer learning and ensemble learning by comparing with the result of finetuning in the pre-trained CNN model on ImageNet dataset focusing on the accuracy of damage detection. The researchers used the basic concept of transfer learning and ensembling in pre-trained CNN model on ImageNet dataset to get the result of damage classification from car images.

Nowadays, the proliferation of automobile industries is directly related to the increasing number of car incidents. So, insurance companies are facing many simultaneous claims and solving claims leakage. The sense of Artificial Intelligence (AI) based on machine learning and deep learning algorithms can help to solve these kinds of problem for insurance industries. In this paper, we apply deep learning-based algorithms, VGG16 and VGG19, for car damage detection and assessment in real-world datasets. In a new style approach as [15], they applied the YOLO object detection model [16] to train and detect damage region as their important pipeline to improve their performances of damage detection.

Development of modern applications to overcome such problems is still challenging, especially in applying deep learning for car damage assessment. Deep learning is an efficient approach for solving complex tasks, but it needs more resources for model development, i.e., for training a model, deep learning requires a huge dataset and takes more computation time. To realize deep learning approach for car damaged assessment, this paper focuses on two challenges for creating an efficient model: (i) car damaged datasets for training and (ii) reduction of computation time. Since car damaged assessment is a specific domain, it is lack of publicly available datasets for car damaged images with labelling. Training a model with a small dataset is the most challenging.

In this case, [17,18] demonstrated significant progress on how to solve classification problems when the small dataset is not enough to train a CNN model. Using data augmentation by manually collecting and labelling data on the Web can solve this problem.

An Anti-fraud System for Car Insurance Claim Based on Visual Evidence - they proposed an approach to generate robust deep features by locating the damages accurately and used YOLO to locate the damage regions.

It [20] tries to solve vehicle body damage by using multi sensor-data fusion. They proposed an industrial solution for car damage by hail, by applying a high-resolution Multi-camera.

CHAPTER 3

METHODOLOGY

Manual estimation of damages in fields like construction, vehicular accidents has been the mainstay of the insurance business. However, such methods are replete with biases and inaccurate estimations. This project deals with estimating car damage, primarily with auto insurers as our key potential customers. For this purpose, three distinct Transfer Learning approaches are used which detect the presence of damage, location, and severity of the damage. The basis for algorithms used lies in Convolutional Neural Networks, customized to optimize accuracy. Each approach is analyzed and varying degrees of accuracy were achieved across different models deployed ranging from 68% to 87%. Accuracy as high as 87.9% was obtained during the course of experiments. This research fine-tunes a number of existing approaches and opens doors for collaboration in image recognition, particularly for the car insurance domain.

3.1 SYSTEM ARCHITECTURE

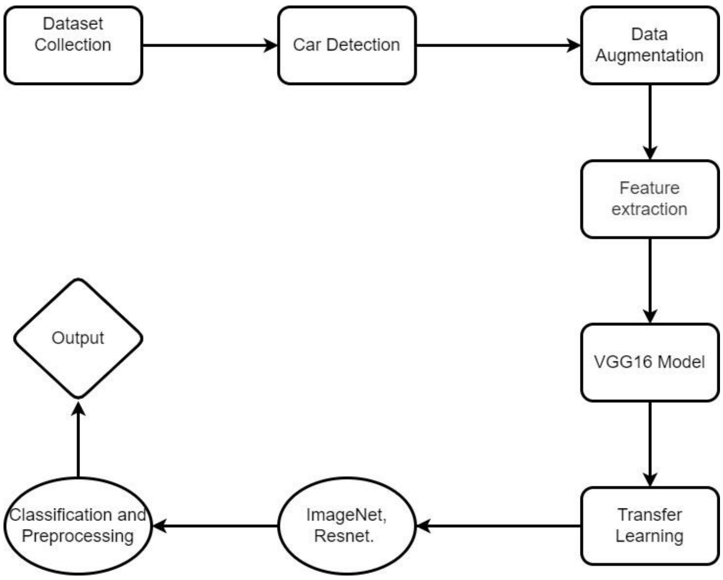


Fig 3.1 System Architecture

3.2 DATASET DESCRIPTION

Generally, the phase of data preparation is very time consuming depending on the requirement of the labelling data. But we did not need to do data cleaning since we used pretrained VGG because it just requires the original images as input. Using cross-validation to estimate our models would need too much computational time because it is very expensive to train VGG for a long time. Therefore, we decided to split the dataset randomly into separate sets for training (80%) and validation (20%).We randomly put train and validation sets, because creating and training with an ensemble of models would have taken very much time. After training multiple times with different splits, this split proved useful. Finally, the train and test were split such that they have similar images. We created our three datasets based on 1150 car damaged images, which consist of different types of car damage when there is no openly obtainable dataset for car damage classification. To reach our classification procedure, we needed to have our three datasets. So, we manually collected damaged and undamaged car images on the Web using selenium. We describe our three collected datasets in the following.

- ImageNet dataset—Having 12 sub-trees: mammal, bird, fish, reptile, amphibian, vehicle, furniture, musical instrument, geological formation, tool, flower, fruit.

- Dataset 1—Train and validation sets with undamaged and damaged cars.
- Dataset 2—Train and validation sets with damaged cars.
- Dataset 3—Train and validation sets with damaged cars.

3.3 CAR DETECTION

Convolutional Neural Network (CNN) is a widely used algorithm for the purpose of classification problems. Firstly, detection of the presence of car damaged takes place (logistic or logit classification). The approach narrows down to two separate models pipelined. The first task is to differentiate between a whole and a damaged car followed by detecting the extent of damage and classify accordingly. Each classhas at least 300 images to train upon. Data augmentation can develop the size and quality of training datasets for deep learning models. Moreover, the augmented data will correspond to a more wide-ranging set of possible data points, decreasing the distance between the training and validation set, as well as testing sets. Connor Shorten surveyed of data augmentation, a solution to the data-space problem of limited data, to improve the execution of their models and expand limited datasetsto take benefit of the abilities of big data. According to the lack of car damaged datasets for training, we use data augmentation to artificially expand and adapt our small datasets, to improve their performance and decrease their tolerance to the overfitting issue during training. Therefore, we apply it randomly rotation, zooming, dimension shift and flipping renovation plans to differ the generated data.

3.4 FEATURE EXTRACTION

Secondly, the extraction of the features of the car damages has been explained further. An extensively comparison of the performances of many deep feature approaches was done in terms of feature extraction and decided to use the VGG16 model with ImageNet weights due to its simplistic model architecture and computational efficiency. It was observed that transfer learning combined with additional layers provides the best results, that is building an ensemble classifier on the top of the set of pretrained classifiers.

3.5 TRANSFER LEARNING

To overcome the problem of overfitting on small datasets, instead of training the CNN models from scratch we can use Transfer learning which has shown a significant improvement on classification problems when the dataset available is scarce. So it is more common to train a CNN on an already available large dataset (e.g. ImageNet, which contains 1.2 million images with 1000 categories) and then transfer this knowledge to the target task with the intuition that some knowledge may be common between the source domain and target domain. Learning features from a large and diverse input data distribution has been found to be more effective than using features from a data distribution specific to just the target class. Instead of just pretraining our model on a large car dataset, it is better to learn features on a more diverse dataset. After training on Imagenet dataset, we retrain the classifier on top of the CNN on our dataset. We also fine-tune all the layers of the CNN while keeping in mind that the earlier layers learn more generic features that are commonin all classification tasks.

- Alexnet AlexNet was designed by Alex Krizhevsky , is one of the deep ConvNets designed to deal with complex scene classification task on Imagenet data.The Network had a very similar architecture to LeNet, but was deeper, bigger, and featured Convolutional Layers stacked on top of each other. It contains eight layers, the first five are convolutional layers, some of them followed by max-pooling layers, and the last three are fully connected layers. To reduce overfitting alexnet uses another technique called dropout that was introduced by G.E. Hinton in a paper in 2012.

- VGG19 The outstanding accomplishment achieved by submission took advantage of smaller receptive window size and smaller stride of the initial convolutional layer. VGG focus on another important aspect of ConvNet architecture design – its depth. They kept other parameters of the architecture constant, and steadily increase the deepness of the network by adding more convolutional layers, which is convenient due to the fact that they use a very they used very small (3 × 3) convolution filters in all layers. As an outcome, came up with significantly more accurate ConvNet architecture.

- Inception V3 The InceptionV3 or GoogleNet is an Architecture with a large set of 42 layers & was created by Google. With such a large set of layers , error rate is reduced effectively. All other architectures preceding to the Inception, performed the convolution on the channel and spatial wise domain together.Inception is a multiple step process where a pattern of convolutional layer is repeated multiple times and in each step the Architecture has set of convolutions layers through which the input is passed which ultimately leads to better learning rate. By performing the one-on-one convolution, the inception block is doing cross-channel correlations, ignoringthe spatialdimensions. This is followed by cross-spatial and cross-channel correlations via the 3x3 and 5x5 filters. All of this then go through dimension reduction to end up in 1×1 convolutions.

- MobileNets As consumer technology becomes thinner and lighter, interest in lightweight neural networks for mobile applications has gained traction. MobileNets is based on the concept of factorised convolutions, where a standard convolution is split into a depthwise convolution and a point wise convolution, to reduce the number of parameters, originally introduced in. Considering M as the number of input channels , N as the number of output channels and Dk as the spatial dimension of the kernel which is assumed to be square, the reduction in computation is.

- ResNet50 Resnet works by using microarchitecture modules to create a “Network of Networks“ architecture. In recent times, deep modules have increased the accuracy multiple times. This network introduces residual learning first introduced by He et al. in their 2015 paper. Residual learning aims to try to learn residuals instead of trying to learn features. We used ResNet50 pretrained on the ImageNet dataset for Feature Extraction. Keeping the Conv layer freezed we trained the densely connected classifier on the augmented data. In order to make the model generalize well with our dataset, we fine-tuned it. In which we dataset, we fine-tuned it. In which we unfreezed a few of the top layers of the freezed model used for feature extraction, and jointly trained the fully connected classifier and these top layers. Validation accuracy increased in all the models.

3.6 CLASSIFICATION

After successfully extracting the features for the two classes, two binary classification model for the pair of two classes were Built. The RGB (Red-Green-Blue) images are Gray-scaled. The images are are resized throughout the dataset using a predefined image size in order to change them into a desirable format. The image data and corresponding class index are appended to training data. The training data is randomly shuffled to ensure that each data point creates an independent change on the model, without being biased by the same points before them. Pickle file is generated to save the serialized format of training data to a file and load it later to directly train the different models without repeating the hassle of data preprocessing. All models are assessed using validation accuracy and loss metrics. The experimentation starts with a customised CNN model which yields an accuracy of around 63%. Next, the VGG-16 model is retrained for the dataset, which yields an accuracy of 87.9%. Forthe use case of this project, the VGG-16 model is employed for further training two different models, each outputting the location and severity of the model using the target classes respectively. Graphs in Fig. 3, Fig. 4, and Fig. 5 represent the performance of models for car damage detection, car damage location identification, and damage severity analysis respectively.

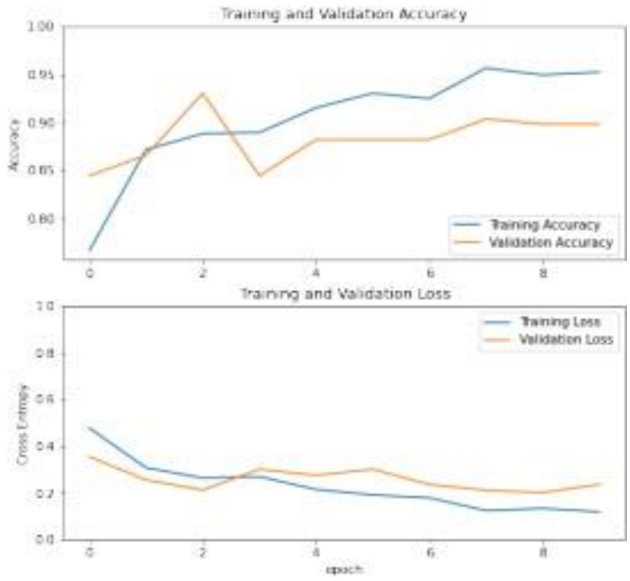


Fig 3.2 Training and Validation accuracy

CHAPTER 4

DATA COLLECTION AND PREPROCESSING

4.1 DATA COLLECTION AND PREPROCESSING

Data Processing is the task of converting data from a given form to a much more usable and desired form i.e. making it more meaningful and informative. Using Machine Learning algorithms, mathematical modeling, and statistical knowledge, this entire process can be automated. The output of this complete process can be in any desired form like graphs, videos, charts, tables, images, and many more, depending on the task we are performing and the requirements of the machine.

The performance and reliability of the models depend upon the dataset used for training. The dataset must contain realistic car damage images. The dataset for car images were collected from a website like Kaggle, a few of the images from Google and other images are collected from Indian traffic management database..

Table 1. Dataset Description

Classes	**Train Size**	**Aug. Train Size**	**Test Size**
Bumper dent	150	750	30
Scratch	112	560	22
Door dent	146	730	25
Glass shatter	104	520	25
Head-lamp broken	107	535	20
Tail-lamp broken	39	195	11
Smashed	256	1280	30
No damage	947	4735	225

Table 4.1 Dataset Description

The dataset images are collected from online sources making noisy and low-quality images. The final dataset had about 7000 images. The dataset used for the classification model is preprocessed before it is fed to the model to improve accuracy and increase robustness. The model accepts input in image and video format. The video is divided into frames and the captured frames are stored as images. After all the images are collected, the preprocessing process starts with removal of unwanted car images that are not needed by the system. This step is necessary as the model would work only on a certain set of images in which cars can be indentified, classified and so can be the severity of damage. It is followed by removal of noise, unwanted elements from th dataset images. There is also a chance that there are images in the database which are out of focus or we can say the blurred images. It is necessary to remove them as such images provide no actual information to they system and it is necessary to remove them. The next step includes annotating all the images in the dataset and labelling the bounding boxes with the labels damaged or not damaged. The images are also labelled according to the class of the damage level as well as damage position.

After the collection of images of car damage, we performed the pre-processing of the collected image dataset. During pre-processing of the dataset, the following various operations were performed on the images.

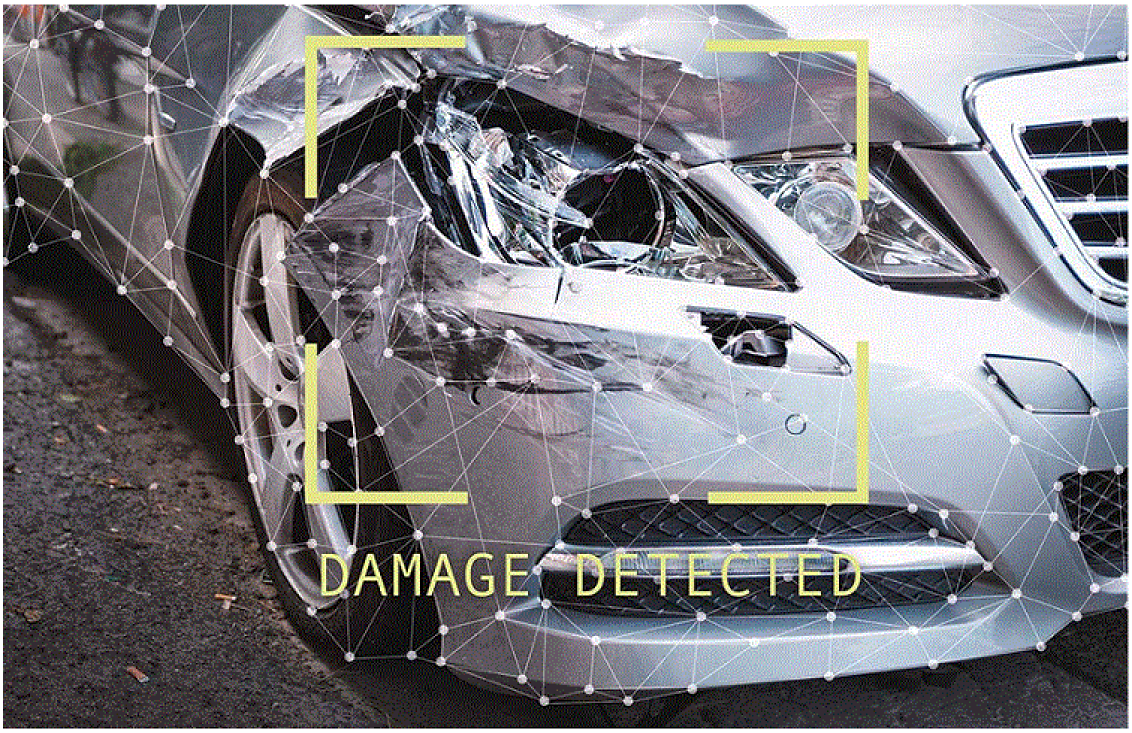


Fig.4.1:Detected Damage

An image can be defined by a two-dimensional array specifically arranged in rows and columns. Digital Image is composed of a finite number of elements, each of which elements have a particular value at a particular location.These elements are called as pixels.

4.2 Types of Image

Binary Image

Black and White Image

8-bit color format

16-bit color format

Resizing of images

First,We will resize all images to same size.

It will help to reduce computization power.

The cropped images are resized to 224X224 Pixel, which is the input size required for our pretrained network.

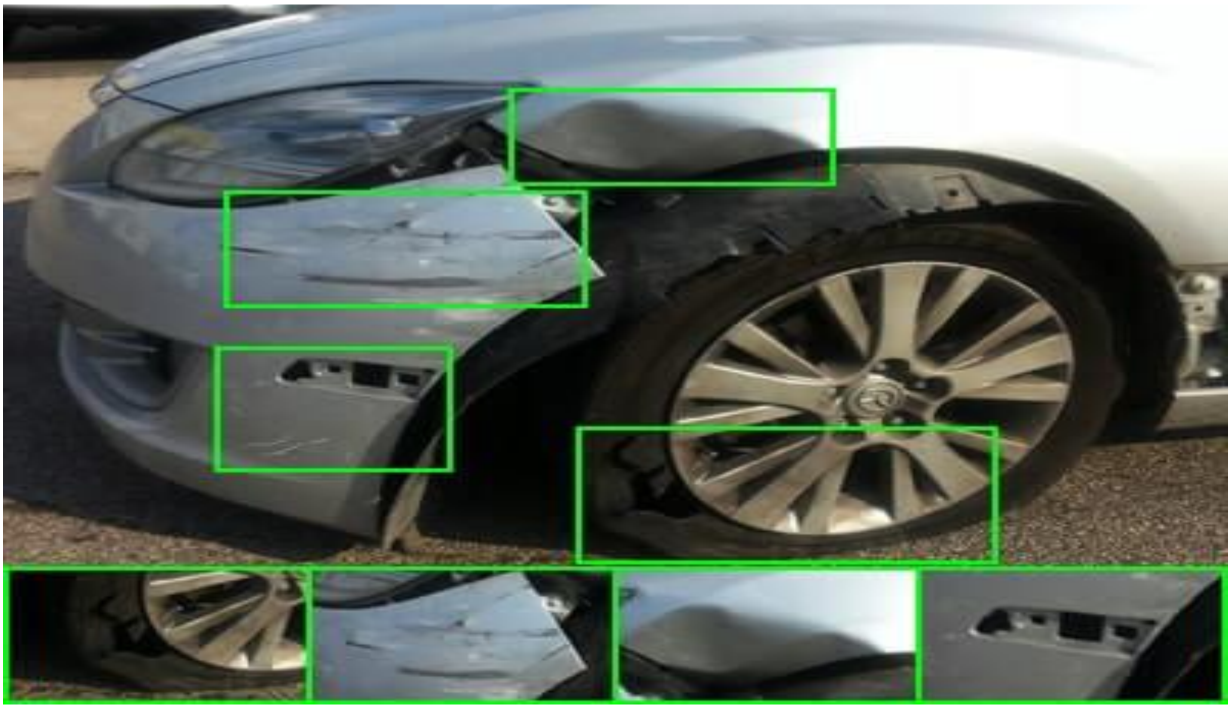


Fig.4.2:Car breaking

Zooming

Image zooming means converting the image into a magnified image. The zoomed version of thermal images can be used for augmentation as they represent the case when images are taken from a close distance from a car

Rotation

Rotation of an image means changing the position of an object about a pivot point at some angle. Image of a damaged car or undamaged car when rotated, will still look like a damaged or undamaged car and would represent the case as if the picture was taken from a different angle.

Contrast enhancement

Contrast enhancement helps in increasing the contrast in the image and thus making the image contain more intensities than just a few. This was applied on images for augmentation since the same image can have better contrast on a sunny day or bright day. Our training set was selected in such a way that it covered a wide range of possibilities.

Salt and pepper noise

Salt and pepper noise is added to an image by randomly changing intensities of some of the pixels of an image to 1 and some other to 0. Salt and pepper noise can represent a case where images were collected on a dusty day, or when the camera has dust on it

4.3 LABELLING AND CLASSIFICATION OF DATA

The images of the first dataset are annotated before loading them into the framework. Image annotation is the process of labeling the objects to be detected. The software used to label the data is called “LabelIng”, an abbreviation of Label Image. The labeling process can be done by drawing a bounding box around the damaged sections of the car.

The position of each damaged part is then converted into a text file with the coordinates of the bounding box.The next step includes annotating all the images in the dataset and labelling the bounding boxes with the labels damaged or not damaged. The images are also labelled according to the class of the damage level as well as damage position.

To start training, a new notebook was created on Colab and the hardware accelerator option is enabled by selecting the GPU option. GPUs are known to be much faster than Central Processing Unit (CPU) for our deep learning application. There are libraries to optimize the use of GPUs in deep learning as the NVIDIA CUDA Deep Neural Network Library (cuDNN), known as a GPU-accelerated library for deep learning

Loading Data To Notebook

The data is transferred to the notebook. The data is then split into train and test data, and then a batch size is set. With the selected batch size, a data loader is created, followed by setting CUDA GPU for faster training. Followed by which, we convert all the images onto the tensor form. All the tensors are then migrated to GPU.

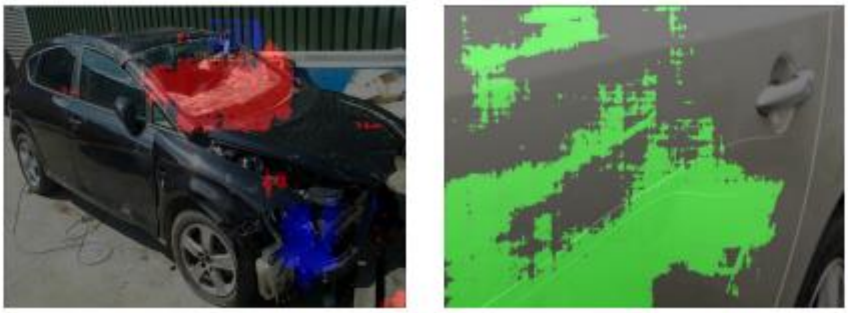


Fig.4.3:Car Damage

Training Of Network

The CNN needs to be trained before it can be used for our classification application. In the training phase, we supply the CNN with different images along with desired output/classes. This process is repeated for a preset number of times to arrive at an optimized network. We have used the approach of transfer learning to reduce the overall training time. we take the pretrained convolution neural network and modify the last classification layer.

These networks are trained on millions of images and classify the data into 1000 or more classes trained over weeks of tie. Since the networks are so optimized, we take them as it is and build our network on top of it. We took pretrained ResNet-50 and modified the last layer to classify the images into three classes instead of one thousand.

4.4 CNN CLASSIFICATION

CNN or ConvNets is one of the deep learning algorithms, designed to extract features of images which help in classifying images. The CNN architecture with different layers is shown and the same is discussed below.

Introduction to the network

Neural networks are mathematical model designed to loosely resemble the human brain. Convolution neural network (CNN) is one of the different types of neural network. CNN specializes in image processing and can be used for image classification, segmentation, object detection etc. Fig. 3 shows a schematic of a simple CNN, which classifies an input image into different category of vehicle present in it.

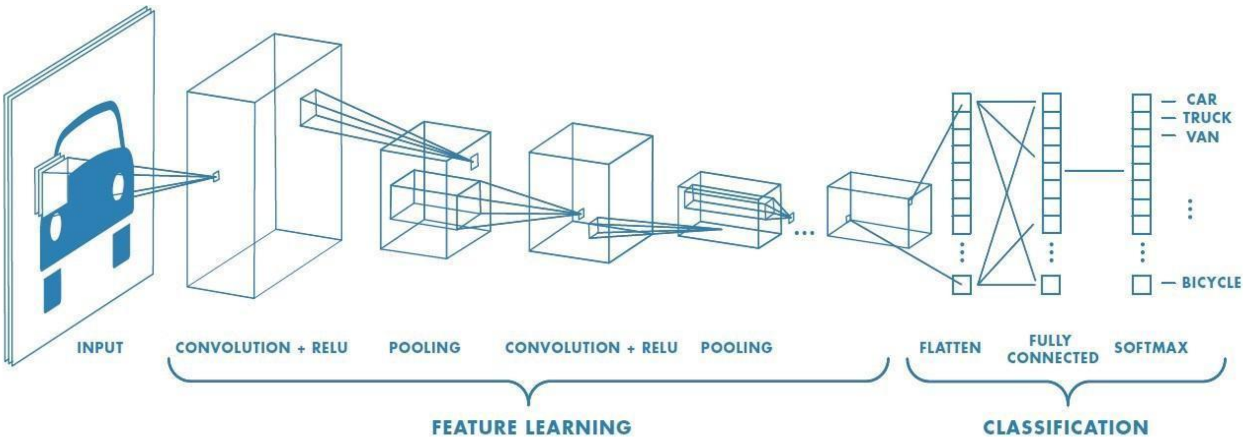


Fig.4.4.CNN Classification

It consists of different categories of layers such as convolution layer, pooling layer, activation layer, etc.Convolutional layer The major part of this layer is carried out by a kernel or filter, which is imposed number of times on the image based on stride length. The kernel is moved over an image to extract the features like color, edges and gradients. The kernel travels through the entire image to extract the features.

Pooling Layer

The Pooling layer is responsible for reducing the spatial size of the Convolved Feature. This is to decrease the computational power required to process the data through dimensionality reduction. Furthermore, it is useful for extracting dominant features which are rotational and positional invariant, thus maintaining the process of effectively training of the model.

It is used to extract features that are invariant to rotation and position. Pooling can be categorised into two types i.e.

1. Max Pooling
2. Avg Pooling.

After the convolution and pooling layer, the model is enabled to understand and extract features from an image. Max Pooling returns the maximum value from the portion of the image covered by the Kernel. On the other hand, Average Pooling returns the average of all the values from the portion of the image covered by the Kernel.

Max Pooling also performs as a Noise Suppressant. It discards the noisy activations altogether and also performs de-noising along with dimensionality reduction. On the other hand, Average Pooling simply performs dimensionality reduction as a noise suppressing mechanism. Hence, we can say that Max Pooling performs a lot better than Average Pooling.

The Convolutional Layer and the Pooling Layer, together form the i-th layer of a Convolutional Neural Network. Depending on the complexities in the images, the number of such layers may be increased for capturing low-levels details even further, but at the cost of more computational power. After going through the above process, we have successfully enabled the model to understand the features. Moving on, we are going to flatten the final output and feed it to a regular Neural Network for classification purposes.

Fully Connected Layer

It will learn non-linear features from the output of a convolution layer. For multi perception, that output should be converted to a column vector and fed to a feed-forward neural network with back propagation in every iteration of training. This helps the model extract the dominant and low-level features of an image.

Adding a Fully-Connected layer is a (usually) cheap way of learning non-linear combinations of the high-level features as represented by the output of the convolutional layer. The Fully-Connected layer is learning a possibly non-linear function in that space.

Now that we have converted our input image into a suitable form for our Multi-Level Perceptron, we shall flatten the image into a column vector. The flattened output is fed to a feed-forward neural network and backpropagation applied to every iteration of training. Over a series of epochs, the model is able to distinguish between dominating and certain low-level features in images and classify them using the Softmax Classification technique.

In turn feature extraction from the model helps to identify and classify images. CNN with ReLu and Dropout ReLu (Rectified Linear Unit) overcomes the gradient problem by propagating a gradient effectively. The ReLu output's the same as the value of the input if it's positive or else its output is zero for negative images. Due to its prominent characteristics, ReLu is used in deep networks. Mathematically, ReLu is represented as shown in the below equation.

$$ReLu(P) = \begin{cases} 0 \; if \; P < 0 \\ P \; if \; P \; > \; = \; 0 \end{cases} \qquad (1)$$

Dropout refers to ignoring hidden or visible units during training for a few sets of neurons by considering them randomly. The units that are ignored will be eliminated in the next layers for forward and backward propagation. Dropout is necessary to avoid inter-dependency between units which over-fit the model

4.5 Creating a RESNET Model

Also, these networks have performed fairly in various deep learning competitions with very less error percentage. They also solve the problem generally observed in large networks that is with increase in depth of networks, saturation and degradation in accuracy start taking place. We have tried to employ the best practices for training the models. We have used cyclic learning rates that change their values cyclically between epochs and differential learning rates that change their values according to layers such that the training rate varies layer-wise as initial layers usually represent primitive features and inner layers represent high-level features. This helps in improving the accuracy greatly and reducing overfitting if any.

Creating a 3 Level Predcition Model

CNN is used as the backbone of our system. We create a 3-step prediction model.

Creating a model for checking ID - damaged or not

Creating model for damage level prediction

Creating model for damage position

Setting up Learning rate and optimizer

The model uses binary cross-entropy as the loss function which is a logarithmic loss function.

Adam optimizer with various parameters like learning rate, decay etc. has been used for optimization.

4.6 Training Model

Training procedure includes adaptive moment estimation (Adam) optimizer that uses exponential weighted moving averaging to find the momentum and the second moment of the gradient. Similarly, different variations of learning rate (LR) has been experimented and 0.001 is found effective with batch size of 64 for the given input frames. Because, if it is very large, optimization diverges, and if it is very small, it takes extended time to train or end up with a trivial outcome. Moreover, activation function ReLU makes model training computationally effective because, the gradient in the positive interval is always 1.

Hence, parameters are not accurately initialized, the sigmoid function might get a gradient of nearly 0, and the model cannot be efficiently trained. Activation score in transitional layers may vary from the input to the output, over time due to the updation in the model's parameters and through nodes in the same layer. This accumulation in the activation distribution can delay the convergence of the network that too may lead to alterations in the LRs for every layer which is computationally expensive. Thus, normalizing the activations of every node through batch normalization has been applied to handle with the mentioned problem and it assists the model to converge at the early stages. Padding and stride are kept at unit movement and Softmax function is utilized in final layer.

The dataset was split into training, validation and test sets in the ratio of 70:20:10. The images were resized to 224×224 dimensions. The modified network has been trained for 10 epochs using batch size of 64 using cyclic learning rate of 0.2. During training of the modified network, discriminative layer learning was used. Initially, the model was trained by keeping initial layers of the model unchanged i.e. no change in values of pre-trained network parameters. Afterwards, model was trained by unfreezing all the layers of the network i.e. parameters values of all layers were changed.

Saving Model Weights For Deployment

Deployment is the method by which you integrate a machine learning model into an existing production environment to make practical business decisions based on data. It is one of the last stages in the machine learning life cycle and can be one of the most cumbersome. In order to start using a model for practical decision-making, it needs to be effectively deployed into production. In order to get the most value out of machine learning models, it is important to seamlessly deploy them into production so a business can start using them to make practical decisions. The deployment for our project is done through streamlit app.

CHAPTER 5

RESULTS AND DISCUSSION

5.1 MODEL TESTING

The purpose of testing is to discover errors. Testing is a process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies, and/or a finished product. It is the process of exercising software with the intent of ensuring that the software system meets its requirements and user expectations and does not fail in an unacceptable manner. Software testing is an important element of software quality assurance and represents the ultimate review of specification, design, and coding. The increasing feasibility of software as a system and the cost associated with the software failures are motivating forces for well-planned testing. Testing Objectives:

There are several rules that can serve as testing objectives they are:

• Testing is a process of executing a program with the intent of finding an error.

• A good test case is the one that has a high probability of finding an undiscovered error

Model Testing: During this phase, the second set of data is loaded. This data set has never been seen by the model and therefore its true accuracy will be verified. After the model training is complete, and it is understood that the model shows the right result, it can be saved by: model.save(“name_of_file.h5”). Finally, the saved model can be used in the real world. The name of this phase is model evaluation. This means a motivating model can be used to evaluate new data.

5.2 RESULTS

The Residual Network architecture, features the usage of residual blocks. The one presented here is in fact the top performing one. In the case of this CNN, the recommended way of training detailed on the paper is to use a single softmax layer with loss. This is the procedure to be used in fine-tuning the network.

They perform especially well on big datasets with a great number of classes such as the Imagenet dataset. The characteristics of the tasks we're going to adapt the architectures to are quite different. The number of classes involved is typically smaller, by a factor of 100, at least, and the differences between the various classes are subtler. These differences in problem characteristics might justify deeper changes in both architectures. As a rule of thumb, problems with less classes, and smaller training samples, often benefit from shallower and thinner architectures. Shallower and thinner architectures typically are faster to train and less prone to overfitting. These advantages come at the cost of a reduced representational power of the net. The decrease in the representational power of the network, materialised in the form of less extracted features from images, might not affect the performance of the system in the new task, especially if the number of classes is reduced too. Making an architecture shallower of thinner has some associated problems. Such deep changes often make the usage of pre-trained models infeasible. This happens because the transfer of weights from the original model to the new one is not easy to do correctly, and not useful at all in some cases because some layers often depend on the previous one.

Without a pre-trained model, the net will have to be trained from scratch, a complicated task to be carried out effectively with a relatively small dataset, like the one we have at our disposal. Deep pre-trained models, especially the ones trained on big datasets such as Imagenet, tend to grasp the feature extraction As to the specific parameters to be used in fine-tuning, the ones mentioned on the paper won't probably be adequate to be used for fine-tuning as they are especially suited for training both nets from scratch. The best meta-parameters for training will have to be identified during the development of the project, possibly making use of parameters used in research where nets were fine-tuned, where the Google LeNet architecture was fine-tuned on a car related dataset.

Data Augmentation It is often necessary to augment the dataset in order to obtain better results from DCNNs. This happens because an image labelled as exhibiting a damage in the right side of the car, actually has a damage on the left side of the vehicle if the image is flipped.. This allows for an augmentation of the number of samples available for training, many times fold. It is also very used, possibly in conjunction with mirroring to boost the performance of already trained models, by getting their prediction for a set of crops from the image to be classified, and averaging them, instead of taking into account only the classification for the whole image. Crop size may vary. It is important to find a size that is big enough to be classified, and small enough so that different crops are significantly different from one another. In this work, and taking into account that we used only pre-trained models, there wasn't much choice regarding crop sizes, as they had to be the same as the ones used by the original model.

While there is a great diversity of ensemble methods, there are two major groups typically used in computer vision related problems. These are boosting, also known as sequential ensemble, and bagging, also known as parallel ensemble. Both have their benefits, their adequacy to the problems addressed in this document is discussed below.

Boosting, sequential ensemble, is a meta-algorithm typically used to correct bias in the predictions of models. These bias and/or variance issues typically arise when the models are not able to grasp the complexity of the classified entities. This may happen when the model is not powerful enough to completely understand the domain they are modelling. The main idea behind boosting is that it is possible to develop a model arbitrarily well-correlated with the true classification (strong learner) by using a combinations of only slightly correlated classifiers (weak classifiers). There are several variations of boosting algorithms the most important of them being AdaBoost (Adaptive Boosting) , Gradient Tree Boosting and XGBoost. For boosting it is desirable to have an usually great number of fast to train, weak learners. DCNNs are usually not fast to train and typically are not weak, highly biased learners and therefore, they are not very adequate to be used as base learnersforBoostingmodels.

CHAPTER 6

CONCLUSION

All models are assessed using validation accuracy and loss metrics. The basis for algorithms used lies in Convolutional Neural Networks, customized to optimize accuracy. Each approach is analyzed and varying degrees of accuracy were achieved across different models deployed ranging from 68% to 87%. Accuracy as high as 87.9% was obtained during the course of experiments. Each algorithm has several parameters that can be tested with different values to increase their accuracy. Static analysis has also proven to be safer and free from the overhead of execution time

Convolutional Neural Networks are accurate at evaluating car damage extent, even when trained on only 300 images per class. With a higher quality dataset that includes pivotal parameters like location information and repair costs, the research could go a step further in predicting the cost of damage repair based on the image. This project opens doors for future collaborations on image recognition projects in general and for the car insurance field in particular. The research successfully recognized the presence of damage, damage location, and extent yielding validation accuracy, avoiding human bias. These can be further improved by incorporating the on the fly data augmentation techniques.

Although various theories and algorithms have been developed for assessing car damage, most of them have been proven that they needed to be reconsidered for reducing problems of overfitting and errors resulting from high-sized datasets. The proposed approach works efficiently and provides us with the most accurate results. The Advantages of proposed system are Secured, Interpret accuracy, Lightweight model & fast processing. The application is beneficial for conditions where data has to be processed in a short time and results are required instantly. It is a fast and quick method and is highly Cost Efficient.

REFERENCES

1] 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

[2] Rakshata P, Padma H V, Pooja M, Yashaswini H V, Karthik V,”Car Damage Detection and Analysis Using Deep Learning Algorithm For Automotive”, Vol 5, Issue 6, International Journal of Scientific Research Engineering Trends (IJSRET), Nov-Dec 2019, ISSN (Online): 2395- 566X

[3] Ren, Shaoqing He, Kaiming Girshick, Ross Sun, Jian. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence. 39. 10.1109/TPAMI.2016.2577031.

[4] X. Zhu, S. Liu, P. Zhang and Y. Duan, ”A Unified Framework of Intelligent Vehicle Damage Assessment based on Computer Vision Technology,” 2019 IEEE 2nd International Conference on Automation, Electronics and Electrical Engineering (AUTEEE), 2019, pp. 124-128, doi: 10.1109/AUTEEE48671.2019.9033150.

[5] 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

[6] Jeffrey de Deijn,"Automatic Car Damage Recognition using Convolutional Neural Networks", 2018 Internship report MSc Business Analytics, 2018

[7] 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), pp. 50-54, 2017.

[8] P Rakshata, H V Padma, M Pooja, H V Yashaswini and V Karthik, "Car Damage Detection and Analysis Using Deep Learning Algorithm For Automotive", International Journal of Scientific Research Engineering Trends (IJSRET), vol. 5, no. 6, Nov-Dec 2019, ISSN 2395566X.

[9] X. Zhu, S. Liu, P. Zhang and Y. Duan, "A Unified Framework of Intelligent Vehicle Damage Assessment based on Computer Vision Technology", 2019 IEEE 2nd International Conference on Automation Electronics and Electrical Engineering (AUTEEE), pp. 124-128, 2019.

[10] Q. Zhang, X. Chang and S. B. Bian, "Vehicle-Damage-Detection Segmentation Algorithm Based on Improved Mask RCNN", IEEE Access, vol. 8, pp. 6997-7004, 2020

[11] Ying Li and Chitra Dorai, “Applying Image Analysis To Auto Insurance Triage: A Novel Application

[12] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li,and L. F. Fei," ImageNet: A Large-Scale Hierarchical Image Database," IEEE, 2019.

[13] Y.-J. Cha, W. Choi, and O. Buyukozturk, “Deep learning-based crack damage detection using convolutional neural networks," Computer-Aided Civil and Infrastructure Engineering, vol. 32, no. 5, pp. 361{378, 2017

[14] R. Singh, M. P. Ayyar, and T. V. S. Pavan, S. Gosain, and R. R. Shah,"Automating Car Insurance Claims Using Deep Learning Techniques," in IEEE fifth International Conference on Multimedia Big Data (BigMM), 2019.

[15] P. Li, B. Y. Shen, and W. Dong, "An Anti-Fraud System for Car Insurance claim

Based on Visual Evidence," arXiv: 1804.11207v1, April, 2018.

[16] M. Dwivedi, M. H. Shadab, S. Omkar, E. B. Monis, B. Khanna, S. R. S. A. Tiwari, and A. Rathi, "Deep Learning Based Car Damage Classification and Detection," DOI:10.13140/ RG.2.2.18702.51525, Sep. 2019.

[17] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," In Proceedings of the IEEE Conference on Computer Vision and Pattern Recogintion, pages 779- 788, 2016.

[18] A. Karpathy, "Course Notes of Convolutional Neural Networks for Visual Networks for Visual Recognition, " Standard University class CS231n. Available: w.w.w.cs231n.github.io, December, 2017.

[19] W. A. R. Harshani, and K. Vidanage, " Image Processing based Severity and Cost Prediction of Damages in the Vehicle Body: A Computational Intelligence Approach," National Information Technology Conference (NITC), 13-15 September, 2017.

[20] N. Dhieb, H. Ghazzai, H. Besbes, and Y. Massoud, "A Very Deep Transfer Learning Model for Vehicle Damage Detection and Localization," in IEEE International Conference on Vehicular Electronics and Safety (ICVES'19), Cairo, Egypt, Sept. 2019

APPENDIX

A. SOURCE CODE

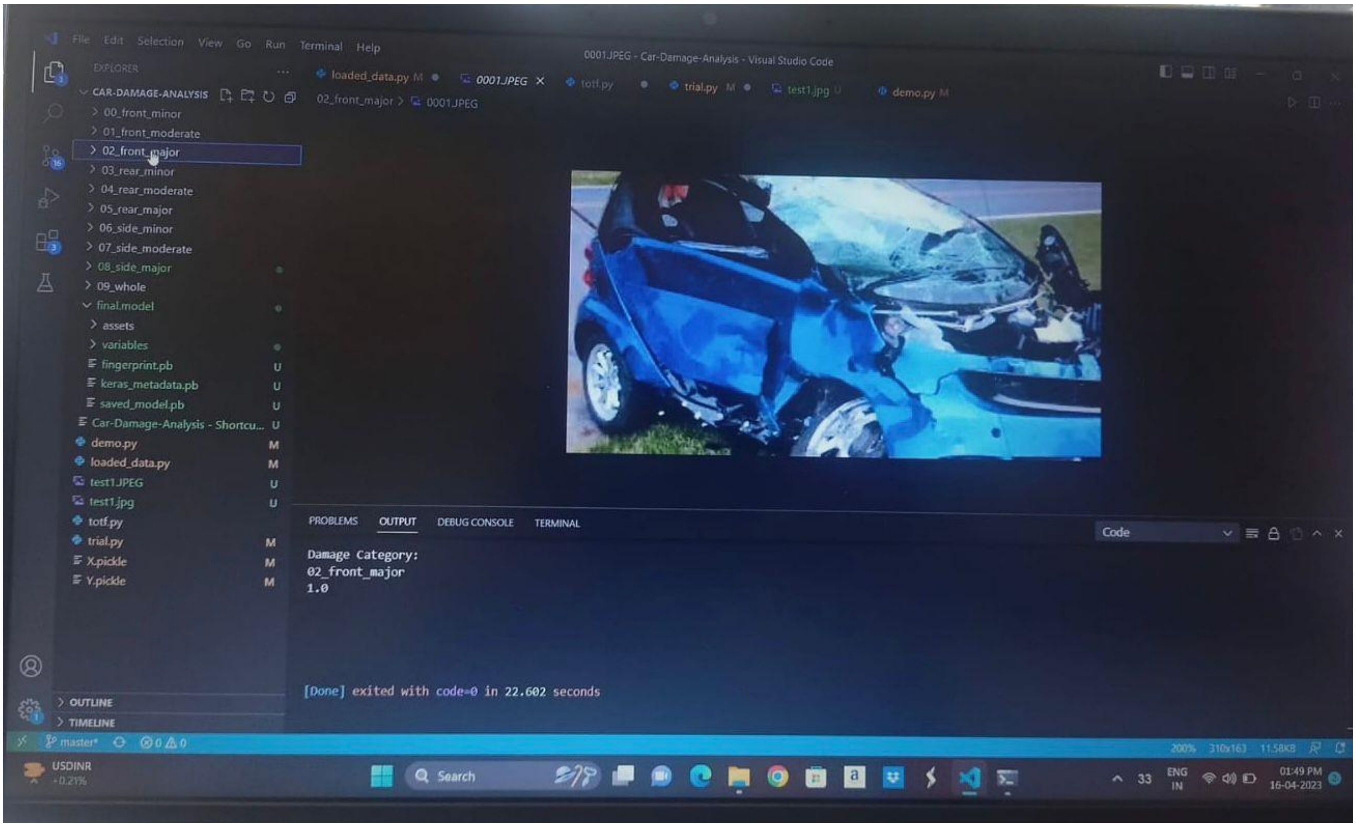
```
import tensorflow as tf
import matplotlib.pyplot as plt
import numpy as np
import random
import os,cv2
import pickle
data_dir= r"./"
categories=["00_front_minor","01_front_moderate","02_front_major","03_rear_minor
"04_rear_moderate", "05_rear_major", "06_side_minor", "07_side_moderate",
          "08_side_major", "09_whole"]
img_size=150
training_data=[]
def create_training_data():
   for category in categories:
      path = os.path.join(data_dir, category)
      class_num =categories.index(category)
      for img in os.listdir(path):
         try:
            img_arr = cv2.imread(os.path.join(path, img), cv2.IMREAD_GRAYSCALE)
            new_arr = cv2.resize(img_arr, (img_size, img_size))
            training_data.append([new_arr, class_num])
         except Exception as e:
            pass

create_training_data()
random.shuffle(training_data)
X=[]
Y=[]
for features, labels in training_data:
   X.append(features)
   Y.append(labels)
```

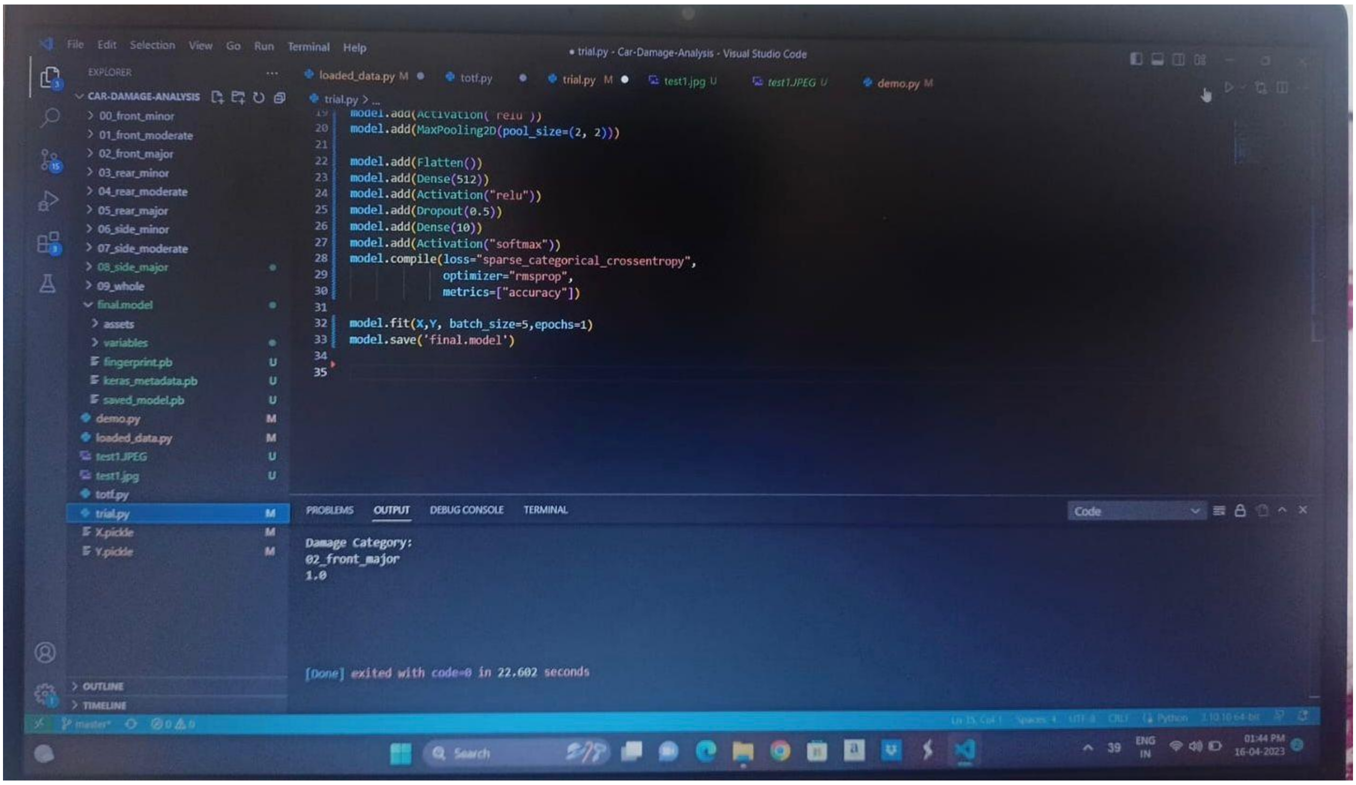
```
X=np.array(X).reshape(-1, img_size, img_size, 1)
pickle_out=open("X.pickle","wb")
pickle.dump(X, pickle_out)
pickle_out.close()
pickle_out=open("Y.pickle","wb")
pickle.dump(Y, pickle_out)
pickle_out.close()
pickle_in=open("X.pickle","rb")
X=pickle.load(pickle_in)
# print(X[1])
# print(Y)
# print(len(X))

import tensorflow as tf
from tensorflow import keras
from keras.models import  Sequential
from keras.layers import Dense, Dropout, Activation, Flatten, Conv2D, MaxPool2D,
MaxPooling2D
import pickle
import numpy as np

X = np.asarray(pickle.load(open("X.pickle", "rb")))
Y = np.asarray(pickle.load(open("Y.pickle", "rb")))

X=X/255
model = Sequential()

model.add(Conv2D(32, (3, 3), input_shape=X.shape[1:]))
model.add(Activation("relu"))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(32, (3, 3)))
model.add(Activation("relu"))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Flatten())
```

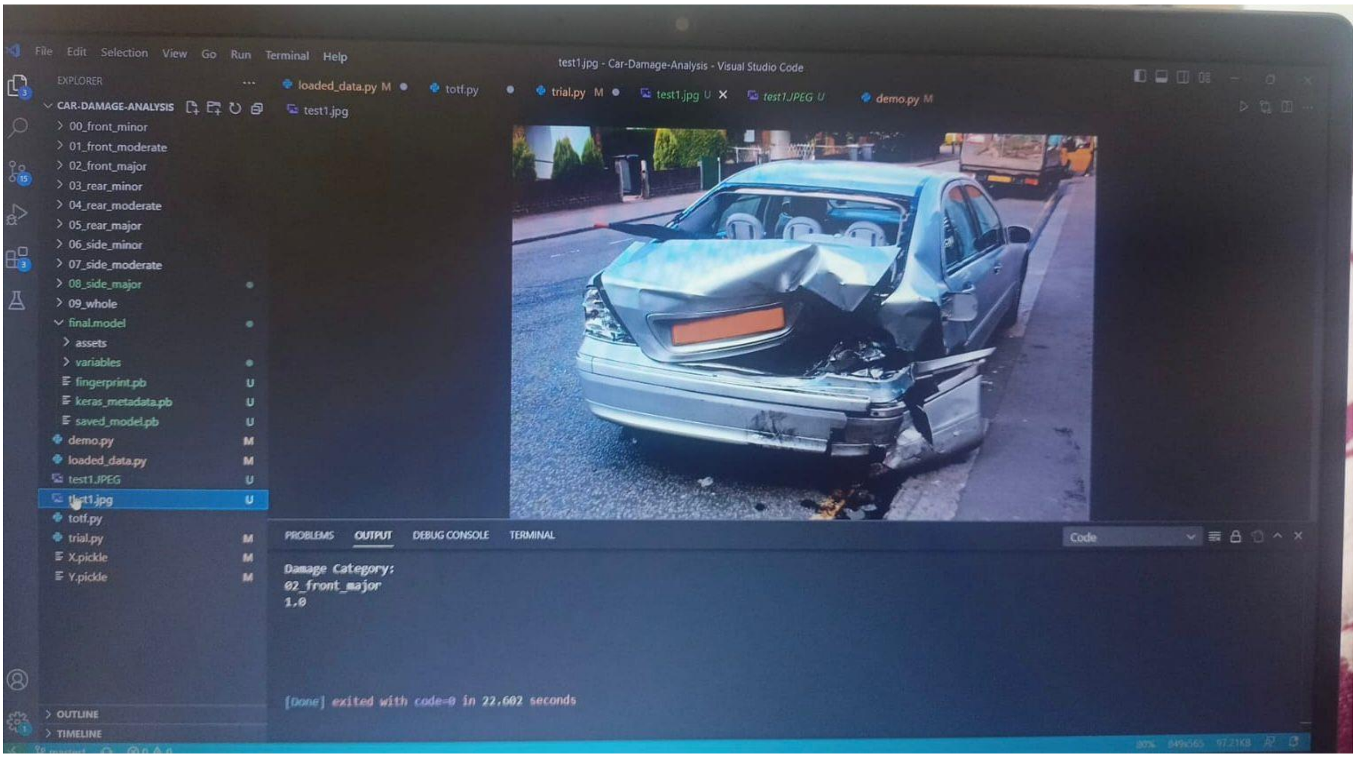
```
model.add(Dense(512))
model.add(Activation("relu"))
model.add(Dropout(0.5))
model.add(Dense(10))
model.add(Activation("softmax"))
model.compile(loss="sparse_catego…

import tensorflow as tf
from tensorflow import keras
from keras.models import  Sequential
from keras.layers import Dense, Dropout, Activation, Flatten, Conv2D, MaxPool2D,
MaxPooling2D
import pickle
import numpy as np

X = np.asarray(pickle.load(open("X.pickle", "rb")))
Y = np.asarray(pickle.load(open("Y.pickle", "rb")))

X=X/255
model = Sequential()

model.add(Conv2D(32, (3, 3), input_shape=X.shape[1:]))
model.add(Activation("relu"))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(32, (3, 3)))
model.add(Activation("relu"))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Flatten())
model.add(Dense(512))
model.add(Activation("relu"))
model.add(Dropout(0.5))
model.add(Dense(10))
model.add(Activation("softmax"))
model.compile(loss="sparse_categorical_crossentropy",
```

```
        optimizer="rmsprop",
        metrics=["accuracy"])

model.fit(X,Y, batch_size=5,epochs=1)
model.save('final.model')
```

B SCREENSHOTS






File Edit Selection View Go Run Terminal Help
test1.jpg - Car-Damage-Analysis - Visual Studio Code
EXPLORER
CAR-DAMAGE-ANALYSIS
00_front_minor
01_front_moderate
02_front_major
03_rear_minor
04_rear_moderate
05_rear_major
06_side_minor
07_side_moderate
08_side_major
09_whole
final.model
assets
variables
fingerprint.pb
keras_metadata.pb
saved_model.pb
demo.py
loaded_data.py
test1.JPEG
test1.jpg
totf.py
trial.py
X.pickle
Y.pickle
OUTLINE
TIMELINE
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
Damage Category:
02_front_major
1.0
[Done] exited with code=0 in 22.602 seconds

C. RESEARCH PAPER

ASSESSING CAR DAMAGE WITH CONVOLUTIONAL NEURAL NETWORK

Varaprasad Seepana[1]
Student, Department of Computer Science and Engineering Sathyabama Institute of Science and Technology, Chennai

V.Santhosh[2]
Student, Department of Computer Science and Engineering Sathyabama Institute of Science and Technology **,Chennai**

Mrs.**Dr.** Bangaru Kamatchi ME,Ph.D **Associate Professor,**
Computer Science and Engineering Sathyabama Institute of Science and Technology,**Chennai**

Abstract-

Visual image classification is a research area that involves both computer vision and machine learning. The task of visually classifying an object consists in assigning an object to a category, or set of categories the object belongs to. Traditionally, visual classification tasks are performed using a two layered system, made up of a first layer featuring an out-of-the-shelf feature extractor and detector, and a second classifier layer. In most recent years, convolutional neural networks have been shown to outperform such previously used systems. Cars have a paramount role in today's world, and being able to automatically classify damages in cars is of great interest specially to the car insurance industry. Car insurance companies deal with car inspections on a daily basis. Such inspections are a manual, lengthy and sometimes faulty processes. Processes that bring costs and inconveniences to costumers and insurance companies alike. Even though the total replacement of such manual inspection processes might still be far away, developing systems to aid, accelerate or enhance the process might be possible with today's technology.

Keywords- Machine Leaming, Data Science, phishing website,

I INTRODUCTION

Insurance is a very old industry and it has been quite resistant to change until recently. Manual validation of large-scale claims is not sufficiently quick in meeting dealing with large numbers of insurance claims. Claim amounts are determined by the type of damage and the affected part of the vehicle and an automated car insurance claim processing system - that efficiently detects and estimates damages - could be of great help. Automatic damage assessment employing analysis of images has proved to be fast and efficient, and will improve further as more and more data is collected. Deep learning enables us to automatically detect scratches, teeth, rust, and breakages, to determine the affected part of the vehicle and to quantify the severity of the damage. After processing, reports can be generated with a list of damages and an estimate of the cost of repair.

Ability to detect damages in dataset taken cars is a subset of image classification because, at its most fundamental level, identifying claims in images entails categorising an image into a particular type as well as collection of classes. There has been a great deal of research on image classification, but there haven't been many works on car visual damage detection. Nonetheless, and be capable of detecting car damage automatically is a research topic with numerous practical uses. Automobile insurance companies deal with car accidents on a daily basis. Cars are frequently required to be serviced for damage, which is problematic for clients as well as extremely expensive for companies. As a result, it's critical to be able to automate car damage detection, making it easy for customers and affordable.

II LITERATURE REVIEW

[1] Deep learning is an efficient method used for classification. Kalpesh Patil, et. al. in [1] have used the concept of deep learning in order to classify car damage. The model used is trained on CNN directly. The preprocessing includes the steps of domain-specific pre-training followed by fine-tuning. The paper has conducted a combined and separate study of Transfer Learning and Ensemble Learning. The research has a setback of unavailability of a proper dataset which has resulted in creation of dataset by annotating images. The use of Convolution Autoencoder based pre-training followed by supervised fine-tuning and transfer learning is a novelty factor of this research.

Deep learning methodology can also be used for detecting presence or absence of damage and conducting further analysis. The researchers in

[2] have applied this in the field of automotives. In this paper, CNN is used for object recognition. The task of classification has been performed on Damaged Vehicle dataset. Mask RCNN is used for segmenting, decomposing and sub-dividing the various instances of Machine Learning. The scope of research is limited to a particular dataset. Extensive research on new data can be performed for testing the quality of the model. Yet the fact that it is an automated system that can classify the damaged vehicle and predict how the damage has occurred remains a unique factor of this research.

The concept of faster R-CNN can be helpful for real-time object detection with Region Proposal Networks. This concept is implemented in
[3 RPN (Region Proposal Network) is trained end-to-end to generate high-quality region proposals, which are used by Fast R-CNN for detection. RPN and Fast R-CNN are merged into a single network by sharing their convolutional features using neural networks with attention mechanisms. The RPN component is essentially used for the unified network to focus on a particular object. The research does not include exploitation and preprocessing on the data. This process could have been used to improve results. The research has built a unified, deep learning based object detection system to run at near real-time frame rates.

Computer Vision Technology can be used for assessment of damage to an object. Xianglei Zhu, et. al. in

[4] have developed an unified intelligent framework based on this concept. This paper uses RetinaNet algorithm to identify damaged parts. The accuracy with this algorithm is improved. Mask R-CNN is adopted for the identification of vehicle parts, the damaged parts are determined by the method of sampling, and the time complexity is greatly reduced. The accuracy achieved in this research can be improved in order to get better results. A combination of characteristics of vehicle damage data and suitable data can further strengthen this system. The research has successfully reduced time complexity in damage detection and the use of Retina Net gives good accuracy in damage detection.

The use of Improved Mask RCNN can be used for vehicle damage detection. In

[5] this approach is followed using Segmentation algorithm. A deep learning approach is used to detect vehicle-damage for compensation problem in traffic accidents. The algorithm has achieved good detection results in different scenarios. Regardless of the strength of the light, the damaged area of multiple cars, or a scene with an overly high exposure, the fitting effect is better and the robustness is strong. The limitation of this research lies in the mask instance segmentation. In many cases the obvious damage is not considered and segmented leading to inaccurate results. This research contributes to detection of damage of vehicles in a more efficient method through improved Mask algorithm.

Convolutional Neural Network (CNN) is a widely used algorithm for the purpose of classification problems. This method is used by Jeffrey de Deijn in [6]). The research was able to detect car damage with fairly accurate results. The type, location and size of damage is detected with moderate accuracy. The addition of Ensemble learning could have further improved the results from this research. The use of ConvNets to detect car damage detection and transfer learning are the novelty factors of this research.

[7] Now days claiming and settlement of vehicle insurance is done through online, where customers are allowed to upload the image of damaged vehicle taken using their mobile phones and request for claim. However there can be chance of repeated claim for same case which can be loss for insurance company. So the main objective is to develop an Anti-Fraud checking system to process the request and speed up the insurance claim process. YOLO detector is used as object detection framework. YOLO detects the damage on vehicle by learning features though regression in 4 coordinates. To extract local features, pre-trained VGG16 object recognition model is employed as a feature extractor. Global deep features and color histogram are present in Global feature. [8] In this paper the authors have suggested an automated system that can classify the damaged vehicle and predict how the damage has occurred. Convolution Neural Network (CNN) can be used for understanding, detecting and analyzing various classes of damage in the minor and major parts of car. The damages can be of any types like bumper dent, door dent, glass shatter, head lamp broken, tail lamp broken, scratch and smash. CNN is used for object recognition task, in the proposed system it is being applied in the specific context of car damage recognition. Classification task is done on Damaged Vehicle dataset. This allows us to separate different objects and give bounding boxes, classes and masks. Once after locating bounding boxes, it can be colored and individually extract the features.

[9] Due to the development of deep learning, in recent years, the field of computer vision grows rapidly. A large amount of computer vision technologies have been applied in actual problems. At present, the industry of vehicle damage assessment requires a lot of manpower, and new automatic intelligent damage assessment technology can greatly reduce industrial costs. In this paper, a framework of intelligent vehicle damage assessment algorithm based on object detection technology and image classification technology is proposed. This algorithm can automatically identify the damage position, type and degree according to photos provided by users, so as to offer appropriate maintenance price and reach the accuracy that can meet actual application requirements.

[10] Traffic congestion due to vehicular accidents seriously affects normal travel, and accurate and effective mitigating measures and methods must be studied. To resolve traffic accident compensation problems quickly, a vehicle-damage-detection segmentation algorithm based on transfer learning and an improved mask regional convolutional neural network (Mask RCNN) is proposed in this paper. The experiment first collects car damage pictures for preprocessing and uses Labelme to make data set labels, which are divided into training sets and test sets. The residual network (ResNet) is optimized, and feature extraction is performed in combination with Feature Pyramid Network (FPN). Then, the proportion and threshold of the Anchor in the region proposal network (RPN) are adjusted. The spatial information of the feature map is preserved by bilinear interpolation in ROIAlign, and different weights are introduced in the loss function for different-scale targets. Finally, the results of self-made dedicated dataset training and testing show that the improved Mask RCNN has better Average Precision (AP) value, detection accuracy and masking accuracy, and improves the efficiency of solving traffic accident compensation problems.

III PROPOSED SYSTEM

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. 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.

IV DESCRIPTION OF THE PROPOSED MODEL/SYSTEM :

MODULE 1 - DATA COLLECTION AND PREPROCESSING

Data Processing is the task of converting data from a given form to a much more usable and desired form i.e. making it more meaningful and informative. Using Machine Learning algorithms, mathematical modeling, and statistical knowledge, this entire process can be automated. The output of this complete process can be in any desired form like graphs, videos, charts, tables, images, and many more, depending on the task we are performing and the requirements of the machine.

The performance and reliability of the models depend upon the dataset used for training. The dataset must contain realistic car damage images. The dataset for car images were collected from a website like Kaggle, a few of the images from Google and other images are collected from Indian traffic management database..

The dataset images are collected from online sources making noisy and low-quality images. The final dataset had about 7000 images. The dataset used for the classification model is preprocessed before it is fed to the model to improve accuracy and increase robustness. The model accepts input in image and video format. The video is divided into frames and the captured frames are stored as images. After all the images are collected, the preprocessing process starts with removal of unwanted car images that are not needed by the system. This step is necessary as the model would work only on a certain set of images in which cars can be identified, classified and so can be the severity of damage. It is followed by removal of noise, unwanted elements from th dataset images. There is also a chance that there are images in the database which are out of focus or we can say the blurred images. It is necessary to remove them as such images provide no actual information to they system and it is necessary to remove them. The next step includes annotating all the images in the dataset and labelling the bounding boxes with the labels damaged or not damaged. The images are also labelled according to the class of the damage level as well as damage position.

Image Pre-processing

After the collection of images of car damage, we performed the pre-processing of the collected image dataset. During pre-processing of the dataset, the following various operations were performed on the images.

Types of Image:

- Binary Image
- Black and White Image
- 8-bit color format
- 16-bit color format

Resizing of images

- First,We will resize all images to same size.
- It will help to reduce computization power.
- The cropped images are resized to 224X224 Pixel, which is the input size required for our pretrained network.

Zooming

Image zooming means converting the image into a magnified image. The zoomed version of thermal images can be used for augmentation as they represent the case when images are taken from a close distance from a car

Rotation

Rotation of an image means changing the position of an object about a pivot point at some angle. Image of a damaged car or undamaged car when rotated, will still look like a damaged or undamaged car and would represent the case as if the picture was taken from a different angle.

Salt and pepper noise

Salt and pepper noise is added to an image by randomly changing intensities of some of the pixels of an image to 1 and some other to 0. Salt and pepper noise can represent a case where images were collected on a dusty day, or when the camera has dust on it.

MODULE 2 - LABELLING AND CLASSIFICATION OF DATA

The images of the first dataset are annotated before loading them into the framework. Image annotation is the process of labeling the objects to be detected. The software used to label the data is called "LabelIng", an abbreviation of Label Image. The labeling process can be done by drawing a bounding box around the damaged sections of the car.

The position of each damaged part is then converted into a text file with the coordinates of the bounding box.The next step includes annotating all the images in the dataset and labelling the bounding boxes with the labels damaged or not damaged. The images are also labelled according to the class of the damage level as well as damage position.

MODULE 3 - CNN CLASSIFICATION

CNN or ConvNets is one of the deep learning algorithms, designed to extract features of images which help in classifying images. The CNN architecture with different layers is shown and the same is discussed below.

Introduction to the network

Neural networks are mathematical model designed to loosely resemble the human brain. Convolution neural network (CNN) is one of the different types of neural network. CNN specializes in image processing and can be used for image classification, segmentation, object detection etc. Fig. 3 shows a schematic of a simple CNN, which classifies an input image into different category of vehicle present in it

It consists of different categories of layers such as convolution layer, pooling layer, activation layer, etc.Convolutional layer The major part of this layer is carried out by a kernel or filter, which is imposed number of times on the image based on stride length. The kernel is moved over an image to extract the features like color, edges and gradients. The kernel travels through the entire image to extract the features.

Pooling Layer

The Pooling layer is responsible for reducing the spatial size of the Convolved Feature. This is to decrease the computational power required to process the data through dimensionality reduction. Furthermore, it is useful for extracting dominant features which are rotational and positional invariant, thus maintaining the process of effectively training of the model.

It is used to extract features that are invariant to rotation and position. Pooling can be categorised into two types i.e.

1. Max Pooling
2. Avg Pooling.

After the convolution and pooling layer, the model is enabled to understand and extract features from an image. Max Pooling returns the maximum value from the portion of the image covered by the Kernel. On the other hand, Average Pooling returns the average of all the values from the portion of the image covered by the Kernel.

Max Pooling also performs as a Noise Suppressant. It discards the noisy activations altogether and also performs de-noising along with dimensionality reduction. On the other hand, Average Pooling simply performs dimensionality reduction as a noise suppressing mechanism. Hence, we can say that Max Pooling performs a lot better than Average Pooling.

The Convolutional Layer and the Pooling Layer, together form the i-th layer of a Convolutional Neural Network. Depending on the complexities in the images, the number of such layers may be increased for capturing low-levels details even further, but at the cost of more computational power. After going through the above process, we have successfully enabled the model to understand the features. Moving on, we are going to flatten the final output and feed it to a regular Neural Network for classification purposes.

Creating a RESNET Model

Also, these networks have performed fairly in various deep learning competitions with very less error percentage. They also solve the problem generally observed in large networks that is with increase in depth of networks, saturation and degradation in accuracy start taking place. We have tried to employ the best practices for training the models. We have used cyclic learning rates that change their values cyclically between epochs and differential learning rates that change their values according to layers such that the training rate varies layer-wise as initial layers usually represent primitive features and inner layers represent high-level features. This helps in improving the accuracy greatly and reducing overfitting if any.

Creating a 3 Level Predcition Model

CNN is used as the backbone of our system. We create a 3-step prediction model.

●Creating a model for checking ID - damaged or not

●Creating model for damage level prediction

●Creating model for damage position

Setting up Learning rate and optimizer

● The model uses binary cross-entropy as the loss function which is a logarithmic loss function.

●Adam optimizer with various parameters like learning rate, decay etc. has been used for optimization.

Training Model

Training procedure includes adaptive moment estimation (Adam) optimizer that uses exponential weighted moving averaging to find the momentum and the second moment of the gradient. Similarly, different variations of learning rate (LR) has been experimented and 0.001 is found effective with batch size of 64 for the given input frames. Because, if it is very large, optimization diverges, and if it is very small, it takes extended time to train or end up with a trivial outcome. Moreover, activation function ReLU makes model training computationally effective because, the gradient in the positive interval is always 1.

Hence, parameters are not accurately initialized, the sigmoid function might get a gradient of nearly 0, and the model cannot be efficiently trained. Activation score in transitional layers may vary from the input to the output, over time due to the updation in the model's parameters and through nodes in the same layer. This accumulation in the activation distribution can delay the convergence of the network that too may lead to alterations in the LRs for every layer which is computationally expensive. Thus, normalizing the activations of every node through batch normalization has been applied to handle with the mentioned problem and it assists the model to converge at the early stages.

Padding and stride are kept at unit movement and Softmax function is utilized in final layer.
The dataset was split into training, validation and test sets in the ratio of 70:20:10. The images were resized to 224×224 dimensions. The modified network has been trained for 10 epochs using batch size of 64 using cyclic learning rate of 0.2. During training of the modified network, discriminative layer learning was used. Initially, the model was trained by keeping initial layers of the model unchanged i.e. no change in values of pre-trained network parameters. Afterwards, model was trained by unfreezing all the layers of the network i.e. parameters values of all layers were changed.

V RESULTS

Machine learning algorithms, such as the random forest algorithm or the support vector machine (SVM), are used to identify and distinguish phishing websites from legitimate ones. Good accuracy in detecting Phishing attacks has been observed

VI CONCLUSION

All models are assessed using validation accuracy and loss metrics. The basis for algorithms used lies in Convolutional Neural Networks, customized to optimize accuracy. Each approach is analyzed and varying degrees of accuracy were achieved across different models deployed ranging from 68% to 87%. Accuracy as high as 87.9% was obtained during the course of experiments. Each algorithm has several parameters that can be tested with different values to increase their accuracy. Static analysis has also proven to be safer and free from the overhead of execution time

REFERENCES:

[1] A. Alswailem, B. Alabdullah, N. Alrumayh and A. Alsedrani, "Detecting Phishing Websites Using Machine Learning," 2019 2nd International Conference on Computer Applications & Information Security (ICCAIS), 2019.

[2] M. Korkmaz, O. K. Sahingoz and B. Diri, "Detection of Phishing Websites by Using Machine Learning-Based URL Analysis," 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2020.

[3] M. H. Alkawaz, S. J. Steven and A. I. Hajamydeen, "Detecting Phishing Website Using Machine Learning," 2020 16th IEEE International Colloquium on Signal Processing & Its Applications (CSPA), 2020,

[4] A. Odeh, I. Keshta and E. Abdelfattah, "Machine LearningTechniquesfor Detection of Website Phishing: A Review for Promises and Challenges," 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC), 2021,

[5] V. Patil, P. Thakkar, C. Shah, T. Bhat and S. P. Godse, "Detection and Prevention of Phishing Websites Using Machine Learning Approach," 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), 2018,

[6] A. Razaque, M. B. H. Frej, D. Sabyrov, A. Shaikhyn, F. Amsaad and A. Oun, "Detection of Phishing Websites using Machine Learning," 2020 IEEE Cloud Summit, 2020.

[7] M. M. Vilas, K. P. Ghansham, S.P.Jaypralash and P. Shila, "Detection of Phishing Website Using Machine Learning Approach," 2019 4th International Conference on Electrical, Electronics, Communication,Computer Technologies and Optimization Techniques (ICEECCOT), 2019,

[8] M. Rastogi, A. Chhetri, D. K. Singh and G. Rajan V, "Survey on Detection and Prevention of Phishing Websites using Machine Learning," 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), 2021,

[9] A. Lakshmanarao, P. S. P. Rao and M. M. B. Krishna, "Phishing website detection using novel machine learning fusion approach," 2021 International

Conference on Artificial Intelligence and Smart Systems (ICAIS), 2021,

[10] W. Bai, "Phishing Website Detection Based on Machine Learning Algorithm," 2020 International Conference on Computing and Data Science (CDS), 2020.