ISSN 2347 - 3983
Volume 8. No. 3, March 2020
International Journal of Emerging Trends in Engineering Research
Available Online at http://www.warse.org/IJETER/static/pdf/file/ijeter41832020.pdf
https://doi.org/10.30534/ijeter/2020/41832020

A Model for Assessing the Nature of Car Crashes using Convolutional Neural Networks

Saiteja Mothe[1], Adusumalli Surya Teja[2], Bhavitha Kakumanu[3], Ravi Kumar Tata[4]
[1]Koneru Lakshmaiah Education Foundation, India, saiteja.mothe6221@gmail.com
[2]Koneru Lakshmaiah Education Foundation, India, suryateja.adusumalli99@gmail.com
[3]Koneru Lakshmaiah Education Foundation, India, bhavithakakumanu@gmail.com
[4]Koneru Lakshmaiah Education Foundation, India, rktata@kluniversity.in

ABSTRACT

The car crash detection and classification is one of the most onerous and tedious tasks to perform. There is a dramatic increase in road crashes nowadays, which brings in the enormous need for the classification of crashes, which helps to alert the emergency services to take action. This paper focuses on building a model for classifying different car crashes with high accuracy, depending upon the severity. By considering the factors such as width, height, orientation, and angles of the images, the model is trained with a large number of images. The model uses Deep learning techniques such as Convolutional Neural Networks for classifying images of different car crashes. Training the model from the enormous data of images, the model can classify the level of the crash with better accuracy by reducing the error of classification. By comparing the accuracy of the model using different Activation functions with the optimizers in CNN tells the best approach in building the model. This paper emphasizes on the analysis of different activation functions and optimizers and their effect on the accuracy of the model in classifying by comparing activation functions with each optimizer.

Key words: Deep Learning, Convolutional Neural Networks, Computer Vision, Image Recognition, Image Classification, Machine Learning.

1. INTRODUCTION

A car crash/accident is the most unwanted thing that happens for any car users. The most hapless thing is that a driver does not pick up from their mistakes on the road. Most of the drivers are quite well aware of the basic rules and safety measures while using vehicles on the road. However, it is only the laxity on the part of road users, which causes accidents and car crashes. The leading cause of accidents and crashes is due to man errors.[2]

Every year nearly 1.3 million people are dead because of road accidents. Around 60%-70% of fatality occurred because of anxiety to reach the destination quickly. In this process, road users are trying to break and are tending to neglect the rules.

When an accident/collision of vehicles takes place on the roadways, the lack of an alarm system is a huge problem that should be considered as a high priority to save the lives of the persons in the crash site. The alarm systems are present in some of the vehicles, which are usually high-end models and require much money to be invested by each individual.[3]

For recognizing and classification of the different types of images, the deep learning techniques give high accuracy and efficient results. Convolutional Neural Networks(CNN) give efficient and better results for the diversified data when trained with a massive amount of data samples.[1]

The model in the paper mainly focuses on the accuracy of the classification of images. To achieve better accuracy using CNN, it requires a great diversity of images and the fine-tuning of the CNN model using different activation functions and optimizers that are available for building. Using different layers that are present in the network helps in giving better accuracy.[1][3]

2. DATASET DESCRIPTION

The dataset consists of about 3700 various images of different cars that are damaged and un-damaged. The images collected are from various sources, such as the Kaggle repository and different search engines. The dataset contains images of different car models with each every angle of the image of the car. There are about 3100 training images and about 700 images for validation and testing purposes. All the images are categorized into two classes- good and damaged. The dataset contains different types of variable resolution images, but the system to process requires a constant input dimensionality. Therefore, pre-processing of the dataset is done to down-sample the images to a fixed resolution.

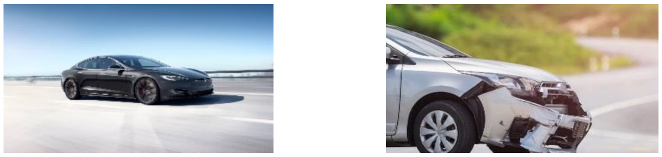


Figure 1: Good and Bad car images in Dataset

3. DATA PRE-PROCESSING

The dataset consists of various categories of car images. Depending upon the severity of the crash, they are categorized. While using Structured data effortlessly, the model is going to train accurately. Different pre-processing techniques used for building model are

3.1 Uniform Aspect Ratio

It is the first step that is followed in the pre-processing of the image data. By applying this technique of uniforming all the images in the dataset, the model can be built seamlessly without any problems. Here each image in the dataset is brought to an equal size and shape for better execution of the model. All the images are brought into the same dimensions.[6]

3.2 Image-Scaling

After uniforming all the image sizes in the dataset, by using the function image data generator from Keras package, the image scaling technique is used. Images are either upscaled or downscaled accordingly with the size fixed in the model.[6]

3.3 Normalizing Image Inputs

This technique ensures the distribution of the data similarly according to the input parameters that are set previously. It helps in faster training of the data. The normalization is performed by deducting the average from each one of the pixels and dividing the obtained result with standard deviation. [6]

4. METHODOLOGY

4.1 Architecture

The proposed model is a Convolutional Neural Network(CNN), which is derived from the Deep Learning Neural Networks. Convolutional Neural Network performs Convolution operation, Max pooling, Flattening, Full connection.

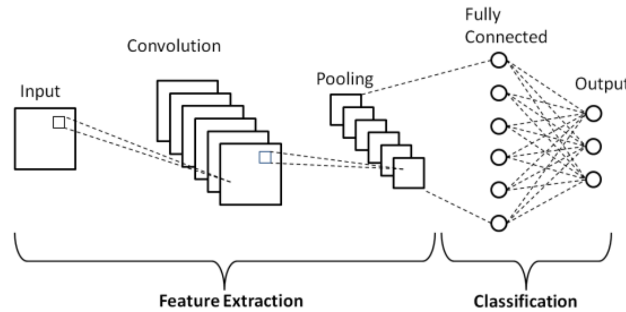


Figure 2: Architecture of the designed CNN model

4.1.1 Convolution Operation

Convolution operation creates a feature map to reduce the size of the image so that the further processing of the image will be quicker and faster. Every image has its features, so to detect the particular feature of an image convolution operation detects features of the image using a feature detector.[7]

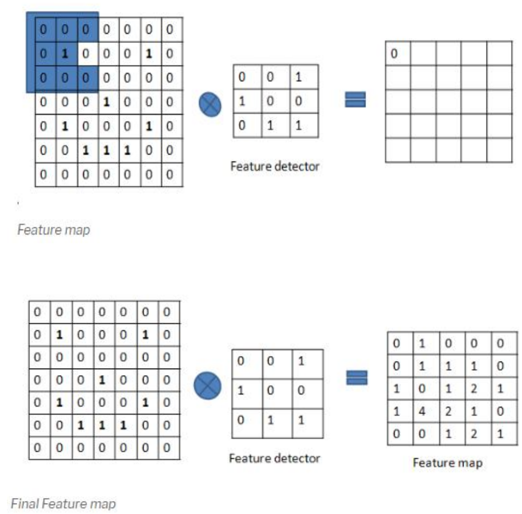


Figure 3: Creation of Feature Map

4.1.2 Max Pooling

A convoluted image can be too large and therefore needs to be reduced. Pooling is mainly used to reduce the image while retaining the critical features or patterns. Max Pooling is one such technique where the features with the highest normalized value are preserved. [7][9]

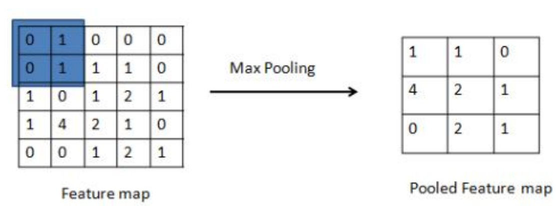


Figure 4: Reduction of Feature Map to Pooled Feature Map

4.1.3 Flattening

For the features of the image to be fed into a neural network, they must be represented in the form of a feature vector. Flattening is the operation that performs this transformation of the 2D feature matrix into a vector. [3][5]

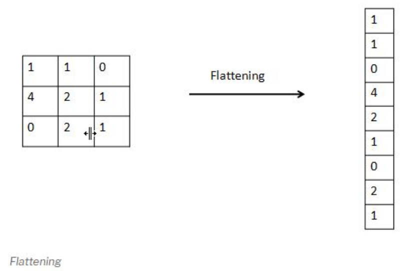


Figure 5: Flattening of the 2D feature matrix into a feature vector

4.1.4 Full-Connection

A full connection establishes a link between the flattened features and the actual neural network.[3][10]

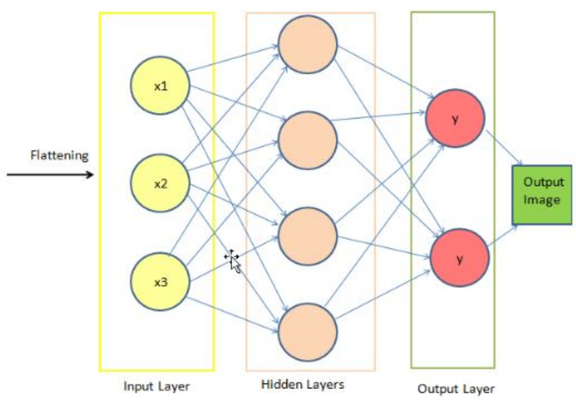


Figure 6: Fully Connected Artificial Neural Network

4.2 Activation Functions

It is a mathematical function that is used to get the output of the node. Also called the Transfer Function. It is utilized to decide the output of the neural networks. The output of the resulting values is mapped as integers, such as between 0 to 1 or -1 to 1. The different types of Activation functions that are used in building the model are

4.2.1 Sigmoid or Logistic Activation Function

It exists between 0 and 1. Therefore, it is mostly used in predicting probability. It is monotonic and differentiable. The convergence rates of this activation function are slow. These are shaped as the S-shape curves when plotted on a graph.[7][8]

$$f(z) = 1 / 1 + \exp(-z)$$

4.2.2 Tanh Activation function

The output values of this function are from -1 to 1. The function is monotonic and differentiable. It is used to classify between two classes.[8][9]

$$f(z)= \tanh(z)$$

4.2.3 ReLU Activation Function

For building the CNN, ReLU function is used in most of the cases. It exists between 0 to infinity. It is monotonic. The function changes all negative values to zero.[8]

$$f(r)= MAX(0 , r) \ ; \ \text{if } r > 0 \Rightarrow f(r) = r, \text{ else } 0$$

4.2.4 Leaky ReLU

The function exists between –infinity to infinity. To overcome the problem with the ReLU, a leak is introduced. For values below 0, $f(x)=ax$ is applied with $a=0.01$. Any other value for a is known as Randomized ReLU.[13]

$$f(r)= MAX(0 , r) \text{ if } r > 0 \Rightarrow f(r) = r, \text{ else } f(r)=ar$$

4.3 Optimizers

Optimization is used for minimizing or maximizing the functions by searching for the parameters which make changes. A certain measure is chosen that tells about the efficiency and correctness of the model by selecting parameters such as accuracy, precision, and recall. To improve the parameter of which is chosen, the optimizers are used. Each model has a different cost function and depends upon the problem. It is designed in a way to indicate how close we are to an ideal solution. Now, finding the minimum for a cost function has its challenges. Gradient comes into play for precisely these kinds of problems.[6]

Using pure gradient descent method also has problems that lead us to use gradient descent with momentum, i.e., when the cost function is close to zero, it leverages exponentially weighted averages. The model uses Adam optimizer, which is also one of the gradient descent with momentum optimizer.[4]

4.3.1 RMSprop

It is known as the Root Mean Squared Propagation. It helps in improving the performance of gradient descent. It uses exponentially weighted means for the calculation. It adapts itself. For each parameter, there is a specific learning rate. The previous gradient values are calculated for the values of specific parameters. For each cost function, the squares of the values are evaluated for every element. The values of the gradients are divided by the square root of the sum of squares.[1][3][10]

4.3.2 Adam

It is also known as Adaptive Moment Estimation. The exponentially decaying mean of the past gradients that are squared is stored. Squared gradients are used to define the learning rate. Instead of using the gradient, momentum is used for the moving average of the gradient. Two moments are used- mean, uncentred variance.[12][14]

$$m = E[X^n]$$

4.3.3 AdaDelta

In AdaDelta, contrary to the precedent squared gradients, gradients are considered as the average of all the previous gradient values. The time of execution and processing depends upon the precedent average of the gradients and the present gradient[11]

5. RESULTS

After training the model with nearly 3100 images, using activation functions ReLU, LeakyReLU, Tanh, Sigmoid, and optimizers Adam, RMSprop, AdaDelta the achieved accuracy using ReLU and Adam is 92%. When comparing with the other functions and optimizers, the accuracy obtained by Adam is high.

Table 1: Accuracy Comparison Table of activation function for each optimizer used

Accuracy	Adam	RMSprop	AdaDelta
ReLU	92	88	85
LeakyReLU	74	72	71
Tanh	86	81	79
Sigmoid	84	80	79

From Table 1, it can be observed that the activation function ReLU with all the optimizers gave higher accuracy when compared with the other functions for the optimizers.

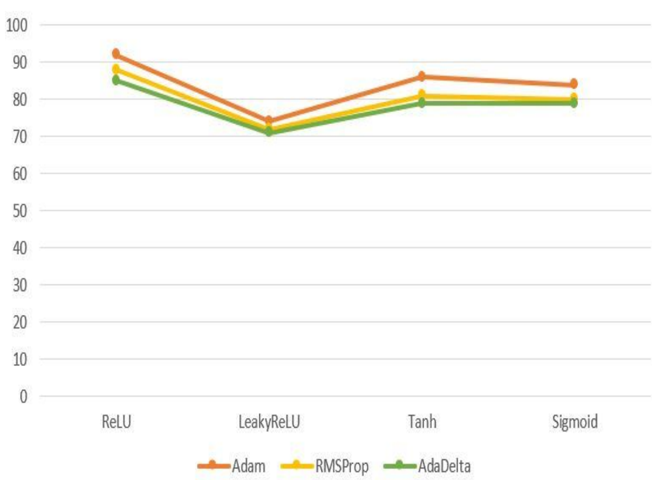


Figure 7: The plot of the Accuracy Comparision

When the model is fed with the 25 iterations(epoch) in the training stage, the accuracy obtained by the ReLU function and the Adam optimizer is 92%, and loss in the model is 20%. ReLU in combination with the optimizer Adam performed well and outperformed all other activation functions

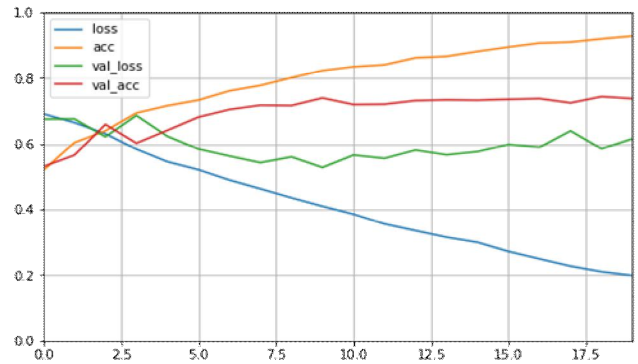
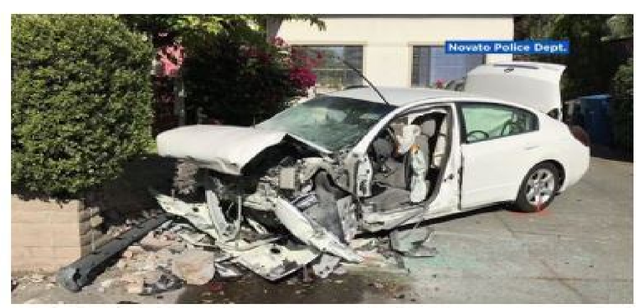


Figure 8: The plot of the Accuracy & Loss obtained by the ReLU function and Adam optimizer

Model when given an image of a car as input, the model classifies and gives the output good or bad, which tells about the level of crash that the car has undergone.

Good

Figure 7: When given an image of a un-damaged car as an input, the output obtained by the model is Good



Bad

Figure 7: When given an image of a damaged car as an input, the output obtained by the model is Bad

6. CONCLUSION

The CNN model, which is built using the activation function ReLU and the optimizer Adam for the dataset, gives better accuracy in classifying the car crashes as good or bad. The model gives 92% accuracy for classifying the images, which are significant and better than the other optimizers such as

RMSprop and AdaDelta. It can be inferred from the results that the model is efficient for classifying the images of car crashes.

6. FUTURE SCOPE

This model can be used for a response measure and to send the status/severity to the authorities. Not confining with cars, this model can be used to train with different types of vehicle images to classify the crash severity. While CNN is one of the best approaches, the local visualization algorithms with 3D images and augmented, reality would be a progressive way to expand this project. Here the image classification can be further dived into levels of severity, which can give the condition and the scale of the collision that occurred to a vehicle.

REFERENCES

1. S. Ghosh, S. J. Sunny and R. Roney, "**Accident Detection Using Convolutional Neural Networks**," 2019 IEEE International Conference on Data Science and Communication (IconDSC), Bangalore, India, 2019, pp. 1-6.
doi: 10.1109/IconDSC.2019.8816881
2. 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), Cancun, 2017, pp. 50-54.
doi: 10.1109/ICMLA.2017.0-179
3. P. Borisagar, Y. Agrawal and R. Parekh, "**Efficient Vehicle Accident Detection System using Tensorflow and Transfer Learning,**" 2018 IEEE International Conference on Networking, Embedded and Wireless Systems (ICNEWS), Bangalore, India, 2018, pp.1-6.
doi:10.1109/ICNEWS.2018.8903938
4. I. A. Tarmizi and A. A. Aziz, "**Vehicle Detection Using Convolutional Neural Network for Autonomous Vehicles**," *2018 International Conference on Intelligent and Advanced System (ICIAS)*, Kuala Lumpur, 2018, pp. 1-5.doi: 10.1109/ICIAS.2018.8540563
5. S. Lee, T. Chen, L. Yu and C. Lai, "**Image Classification Based on the Boost Convolutional Neural Network**," in *IEEE Access*, vol.6, pp. 12755-12768, 2018.
doi: 10.1109/ACCESS.2018.2796722
6. É. Protas, J. D. Bratti, J. F. O. Gaya, P. Drews and S. S. C. Botelho, "**Visualization Methods for Image Transformation Convolutional Neural Networks**," in *IEEE Transactions on Neural Networks and Learning Systems*, vol.30, no.7, pp. 2231-2243, July2019.
doi: 10.1109/TNNLS.2018.2881194
7. S. Albawi, T. A. Mohammed and S. Al-Zawi,"**Understanding of a convolutional neural network**," *2017 International Conference on Engineering and Technology (ICET)*, Antalya, 2017, pp. 1-6.
doi: 10.1109/ICEngTechnol.2017.8308186
8. R. Chauhan, K. K. Ghanshala and R. C. Joshi, "**Convolutional Neural Network (CNN) for Image Detection and Recognition**," *2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)*, Jalandhar, India, 2018, pp. 278-282.
doi: 10.1109/ICSCCC.2018.8703316
9. N. Jmour, S. Zayen and A. Abdelkrim, "**Convolutional neural networks for image classification**," *2018 International Conference on Advanced Systems and Electric Technologies (IC_ASET)*, Hammamet, 2018, pp. 397-402.
doi: 10.1109/ASET.2018.8379889
10. Kollu, Praveen. (2019). "**Intrusion Detection System Using Recurrent Neural Networks and Attention Mechanism**". International Journal of Emerging Trends in Engineering Research. 7. 178-182.
doi: 10.30534/ijeter/2019/12782019.
11. Prasad, M.V.D. & Inthiyaz, Syed & Maddala, Teja & Kunapy, Hari & Manohar, Minna & Kumari, Rupa & Shaik, Hasane. (2019). "**Human activity recognition using Deep Learning**". International Journal of Emerging Trends in Engineering Research. 7. 536-541.
doi: 10.30534/ijeter/2019/227112019.
12. P, Jahnavi. (2019). "**Facial expression detection of all emotions and face recognition system**". International Journal of Emerging Trends in Engineering Research. 7. 778-783.
doi: 10.30534/ijeter/2019/087122019.
13. Ravi Kumar, T., Yasaswini, P., Rafi, G., Krishna, D.V. "**Comparative analysis on job prediction of students based on resume using data mining techniques**". (2018) International Journal of Engineering and Technology(UAE), 7 (2), pp. 1100-1103.
14. Ravi Kumar, T., Sadhana, B., Krishna, D.V. "**Predictions-crime analysis using data mining techniques**". (2018) Journal of Advanced Research in Dynamical and Control Systems, 10 (4 Special Issue), pp. 519-525.