International Journal of Scientific Research in Computer Science, Engineering and Information Technology
ISSN : 2456-3307 (www.ijsrcseit.com)
doi : https://doi.org/10.32628/CSEIT217287

Detecting Safe and Not Safe Driving Actions using Convolutional Neural Network

Saurabh Takle[1], Shubham Desai[2], Sahil Mirgal[3], Ichhanshu Jaiswal[4]
[1-3]Department of Information Technology, Vidyalankar Institute of Technology, Mumbai, Maharashtra, India
[4]Assistant Professor, [4]Department of Information Technology, Vidyalankar Institute of Technology, Mumbai, Maharashtra, India

Article Info
Volume 7, Issue 2
Page Number: 372-378

Publication Issue :
March-April-2021

Article History
Accepted : 15 April 2021
Published : 22 April 2021

ABSTRACT

The main cause of accidents is due to Manual, Visual or Cognitive distraction out of these three Manual distractions are concerned with various activities where “driver’s hands are off the wheel”. Such distractions include talking or texting using mobile phones, eating and drinking, talking to passengers in the vehicle, adjusting the radio, makeup, etc. To solve the problem of manual distraction, the Convolutional Neural Network (CNN) model of ResNet-50 using transfer learning with 23,587,712 parameters was used. The dataset used is from State Farm Distracted Driver Detection Dataset. The training accuracy is 97.27% and validation accuracy is 55%. Further the model works on detecting real-time distractions on a video feed for this purpose the system uses OpenCV and the model is integrated with the frontend using the flask.

Keywords : Distracted Driver Detection, InceptionV3, VGG16, Resnet50, Image Classification, Computer Vision.

I. INTRODUCTION

According to the World Health Organization (WHO) survey [1], 1.3 million people worldwide die in traffic accidents each year, making them the eighth leading cause of death, and an additional 20-50 million are injured/ disabled. As per the report of the National Crime Research Bureau (NCRB), [2] Govt. of India, Indian roads account for most fatalities in the world. There has been an endless increase in road crash deaths in India. According to the report compiled by the ministry Transport Research Wing of India, there has been a 3.2% rise in road fatalities which corresponds to the death of 1,50,785 people across the country in 2016. A total of 4,67,044 road accidents are reported by States and Union Territories within the year 2018, taking 1,51,417 lives and injuring 4,69,418 persons,” the report said and driver error is the most common cause behind these traffic accidents. The number of accidents because of distracted drivers has been increasing for few years.

The National Highway Traffic Safety Administration [3] is an agency of the U.S. federal government that describes distracted driving as “any activity that diverts the attention of the driver from the task of driving” which can be classified into Manual, Visual or Cognitive distraction. As per the definitions of the Center for Disease Control and Prevention (CDC), cognitive distraction is basically “driver’s mind is off the driving” [4]. In other words, even though the driver is in a safe driving posture, he is mentally distracted from the task of driving. Driver might be lost in thoughts, daydreaming, etc. Distraction due to inattention, sleepiness, fatigue, or drowsiness falls into visual distraction class where “drivers’ eyes are off the road”. Manual distractions are concerned with various activities where “driver’s hands are off the wheel”. Such distractions include talking or texting using mobile phones, eating, and drinking, talking with the passengers within the vehicle, adjusting the radio, makeup, etc.

To solve the problem of Manual distraction and to detect whether the driver is driving safe or not there is a need to develop a CNN model. Out of the three models InceptionV3, ResNet-50, and VGG-16 by transfer learning method, ResNet50 model proved its state-of-the-art performance. Further to make a real-time prediction on the video feed, OpenCV was used, which captures a frame of the video and sends it to neural network model and makes the prediction and message is displayed on video in real-time of Safe and Not Safe driving. To integrate CNN model with frontend, flask was used. Further, the images in the dataset were of left-hand driving so to make the predictions on right-hand driving data augmentation was introduced and the images were flipped horizontally. Also, the model was giving wrong predictions in spite of achieving high accuracy. It was due to the problem caused by repetition of images in dataset, which is commonly referred to as the Data Leakage problem. So, to solve this problem the repeated images were eliminated manually and the accuracy got lowered and the predictions were more accurate.

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a ConvNet is much lower as compared to other classification algorithms. While in primitive methods filters are hand-engineered, with enough training, ConvNets have the ability to learn these filters/characteristics. A CNN is well suited for processing of 2-D data such as images, it requires very less human supervision and detects feature automatically. Full connected Neural Network is more complex than CNN which reduces overfitting.

II. RELATED WORK

The project was started with a literature survey, which included going through various research papers associated with distracted driving. The gist of all the papers is as follows:

The first paper [7], written by Qingzhi Bu, Jun Qiu, Hao Wu, Chao Hu proposed the distracted driver detection system based on the histogram of oriented gradient (HOG) and support vector machine (SVM). HOG is used to extract different features of driver’s distracted behavior and SVM classifier to classify decentralized behavior. The paper included various algorithms with their accuracy such as SVM linear, SVM RBF, LBP+SVM, HOG+SVM linear. The experimental results show that using HOG+SVM and optimizing SVM parameters has a good effect on the recognition rate of this dataset, with an average recognition rate of 93.33%. The paper also mentioned that there was a need to improve the training and recognition speed.

In the second paper [5], Maitree Leekha, Mononito Goswami proposed a system for distracted driving detection using foreground extraction and Convolutional Neural Network (CNN). The dataset was available from the State Farm Distracted Driver

Detection dataset (SFD3) and the AUC Distracted Driver dataset (AUCD2) which gave test accuracy of 98.48% and 95.64% respectively. Their model also suggested that incorporating features such as posture through foreground extraction using Grab Cut improves performance. Grab Cut helped in eliminating background noise in the images and extract their foreground image as the input images were frames of the video. These images were then processed by CNN. Their ConvNet model had significantly fewer parameters (0.5M). The model has three convolutional blocks. Each block is composed of a convolutional layer, followed by ReLU nonlinear activation function, a max-pooling layer, and a dropout layer. All convolutional layers have 3*3 filters. The convolutional layer in the first block has 64 kernels, the second has 128, and the third has 256. The convolution blocks are followed by a dense layer and a SoftMax layer, both separated by a dropout layer. The dense layer has 192 neurons and ReLU as the activation function.

The third paper [6], written by Bhakti Baheti, Suhas Gajre, and Sanjay Talbar proposed a system based on the VGG-16 convolutional neural network model. They propose a thinned version of VGG-16 with just 15M parameters and still achieving satisfactory classification accuracy compared to an older version of VGG-16 with 140 parameters. The system processes 42 images per second on average. Performance of the system is significantly improved with the addition of dropout, L2 weight regularization, and batch normalization which results in 96.31% accuracy on the test set.

And the paper [8], written by Ketan Ramesh Dhakate and Ratnakar Dash they proposed a system, where they have used ensembling of different CNN models to predict the distraction. Accuracy-97% is achieved. Trained five different pre-trained CNN models: VGG-16, VGG-19, InceptionV3, Resnet-50, Xception Models. They have used Categorical Cross Entropy as loss function, SoftMax as activation function, Adam optimizers as an optimization parameter. The different postures resulted in an incorrect prediction also model often gets confused due to slight change in body movement also and if the driver is driving in the night and the car cabin is in the dark, then this model may not work properly.

III. METHODOLOGY

The project is Distracted Driver Detection which would classify the driver's action into safe and not safe driving based on the actions of the driver. The actions are talking on the phone, texting, drinking, operating the radio, reaching behind. The dataset was obtained from Kaggle which was a competition and the dataset was provided by State Farm.

For the implementation of the project, a CNN (Convolutional Neural Networks) model for image classification was build, and then for the prediction purpose, it is tested on the video feed of the driver's actions. The video part is implemented by OpenCV and the backend model is integrated with the video prediction using flask on a webpage. Some of the phases of implementation are data cleaning, model building, video feed part, integration using flask.

Data Cleaning

This was the first step in building our CNN model. There was a data leakage problem in the dataset because of which the model was giving a very high training and validation accuracy but the predictions initially were slightly incorrect. The data leakage problem was that the images were repeated again and again in the dataset because of which while building the model during 80-20 validation split or 70-30 validation split the same images were loaded in training and validation sets because of which the accuracy would be very high but the predictions would go wrong several times. The problem of data leakage was solved by manually removing the repeated images, then the accuracy reduced and predictions were becoming right.

Model Building

The model is implemented using TensorFlow Keras API. The images in the dataset were for left hand driving so for building the model for right hand driving data augmentation was performed on the images. Various models were built by using transfer learning. The architectures used are inception v3, resnet50 models. The best working model was of resnet50.

Prediction on Video Feed

The driver's actions were predicted as safe or not safe on a continuous video feed. The prediction was done using CNN model at the backend. For prediction purpose on video each frame of video was fetched using cv2 of OpenCV and sent to the backend model which would give the prediction and the safe or not safe message would be displayed on video in continuous time. The backend CNN model was integrated with the video feed using flask.

IV. IMPLEMENTATION OF CNN MODEL

For building the CNN (Convolutional Neural Networks) model, the transfer learning method was used in all the models that were built for testing. The best working model was Resnet50. So, the implementation of the Resnet50 model is as follows.

Data Cleaning

The dataset of State Farm had a data leakage problem which means many images got repeated and when there was a validation split of 70-30 or 80-20 same images were repeated both in the training set and validation split which gave high accuracy but poor prediction to solve this, the repeated images were removed and then the dataset was used for training.

Data Augmentation

The images in the dataset were left-hand driving to solve the problem for righthand driving, a horizontal flip was implemented using Keras pre-processing.

Architecture

In the resnet50 architecture, the layers after flatten were modified according to the project need, which consisted of 512 filters, dropout, 512filters, and last layer sigmoid function. The total parameters were 23,587,712. The model was of binary classification which would classify the images as safe and not safe driving.

RESNET50: Deep Residual Network is almost similar to the networks which have convolution, pooling, activation and fully-connected layers stacked one over the other. The only construction to the simple network to make it a residual network is the ***identity connection*** between the layers.

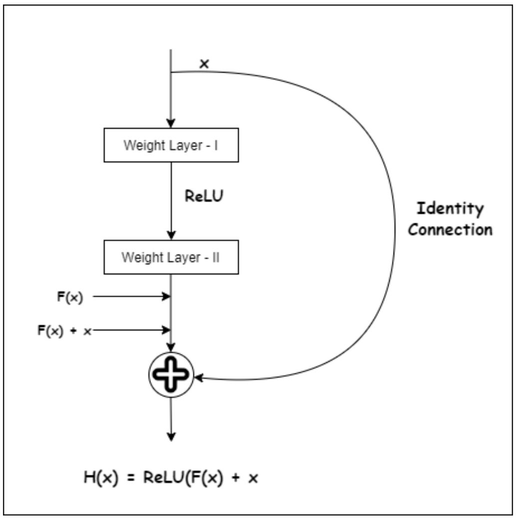

Fig 1 : Identity Connection between Layers

Fig 2 : If necessary, the images can be extended both columns

V. EVALUATION

Dataset

The dataset was from the State Farm Distracted Driver Dataset which consisted of 22,414 training labeled images. The images were classified into 10 classes.

The 10 classes to predict are:

c0: safe driving

c1: texting - right

c2: talking on the phone - right

c3: texting - left

c4: talking on the phone - left

c5: operating the radio

c6: drinking

c7: reaching behind

c8: hair and makeup

c9: talking to the passenger

Out of which, 4691 images were used while training the model. Where class [0] was of safe driving and class [1] was of not safe driving. Class [0] was of safe driving images and class [1] was a combination of nine classes (c1 – c9) of the dataset of not safe driving.

Result

The following table represents the models and their parameters.

Table 1 : Number of parameters

Model	No of parameters
Resnet50	23,587,712
InceptionV3	22,857,002
VGG16	21,140,042

Adam optimizer was better than all it was fast and gave better result as compared to other.

Realtime Working

The video feed was used to demonstrate the real-time application of the system. The CNN model used is ResNet-50, which is integrated with frontend using flask and captured each frame of the video using OpenCV

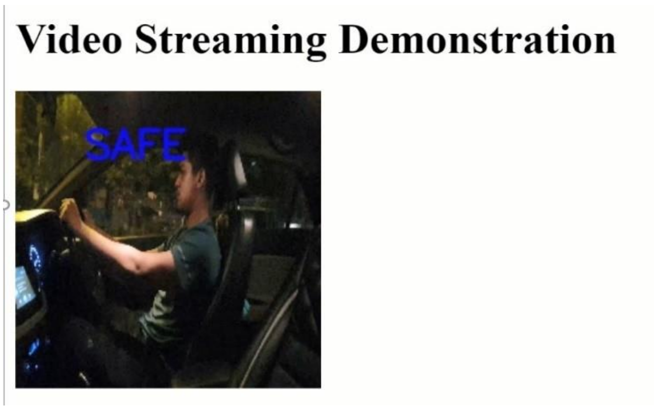

Fig 3 : Detecting Safe Driving

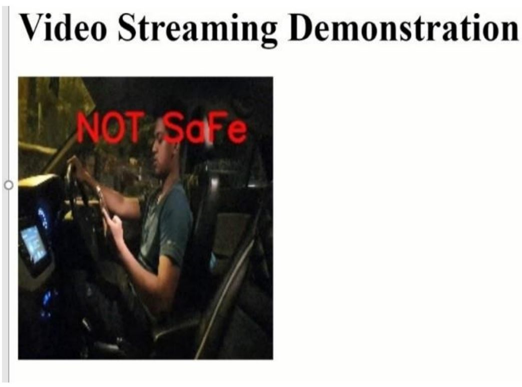

Fig 4 : Detecting Not Safe Driving

The following table shows the model performance of our project which includes training accuracy, validation accuracy, training loss, validation loss and optimizer used.

Table 2 : Model Summary

Model	Training accuracy	Validation accuracy	Training loss	Validation loss	optimizer
InceptionV3	0.6680	0.2500	0.8984	5.2075	adam
VGG16	0.2802	0.6625	1.8479	1.3501	Adam(lr=0.001)
Resnet50	0.9727	0.5500	0.2712	7.5636	adam

VI. CONCLUSION & FUTURE SCOPE

Implementation of a greater number of CNN architecture through transfer learning could be possible. Also, to improve the accuracy the model ensembling method could be used. Increasing the processing speed for the prediction can be taken into account. Finding a greater number of distracting actions and including them into the dataset could result in determining a greater number of not safe driving actions.

VII. ACKNOWLEDGMENTS

We express our deepest gratitude towards our project guide Prof. Ichhanshu Jaiswal for her valuable and timely advice during the various phases of our project. We would also like to thank her for providing us with all the proper facilities and support as the project co-coordinator. We would like to thank him for his support, patience, and faith in our capabilities and for giving us flexibility in terms of working and reporting schedules.

VIII. REFERENCES

[1]. https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries“Road Traffic Injuries”.

[2]. “Poor enforcement, training: The reasons why there are so many road accidents in India”

[3]. https://scroll.in/article/944201/poor-enforcement-training-the-reasons-why-there-are-so-many-road-accidents-in-india

[4]. https://www.nhtsa.gov/risky-driving/distracted-driving

[5]. “Distracted Driving”

[6]. "Top 10 causes distracted driving and what they all have common”

[7]. https://safestart.com/news/top-10-causes-distracted-driving-and-what-they-all-have-common/

[8]. M. Leekha, M. Goswami, R. R. Shah, Y. Yin and R. Zimmermann, "Are You Paying Attention? Detecting Distracted Driving in Real-Time," 2019 IEEE Fifth International Conference on Multimedia Big Data (BigMM), Singapore, Singapore, 2019, pp. 171-180, doi: 10.1109/BigMM.2019.00-28.

[9]. Detection of Distracted Driver Using Convolutional Neural Network B. Baheti, S. Gajre and S. Talbar, "Detection of Distracted Driver Using Convolutional Neural Network," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Salt Lake City, UT, 2018, pp. 1145-11456, doi: 10.1109/CVPRW.2018.00150.

[10]. Research on Driver’s Distracted Behavior Detection Method Based on Multiclass Classification and SVM Q. Bu, J. Qiu, H. Wu

and C. Hu, "Research on Driver’s Distracted Behavior Detection Method Based on Multiclass Classification and SVM," 2019 IEEE International Conference on Robotics and Biomimetics (ROBIO), Dali, China, 2019, pp. 444-448, doi: 10.1109/ROBIO49542.2019.8961551.

[11]. Distracted Driver Detection using Stacking Ensemble K. R. Dhakate and R. Dash, "Distracted Driver Detection using Stacking Ensemble," 2020 IEEE International Students' Conference on Electrical,Electronics and Computer Science (SCEECS), Bhopal, India, 2020, pp. 1-5, doi: 10.1109/SCEECS48394.2020.184.

[12]. Computer Vision: https://en.wikipedia.org/wiki/Computer_vision

[13]. https://youtu.be/gT4F3HGYXf4

[14]. Building model State Farm’s distracted driver detection

[15]. https://www.kaggle.com/c/state-farm-distracted-driver-detection

[16]. Y. Abouelnaga, H. M. Eraqi, and M. N. Moustafa. Real-time distracted driver posture classification. CoRR, abs/1706.09498, 2017.

[17]. N. Das, E. Ohn-Bar, and M. M. Trivedi. On performance evaluation of driver hand detection algorithms: Challenges, dataset, and metrics. In 2015 IEEE 18th International Conference on Intelligent Transportation Systems, pages 2953– 2958, Sept 2015.

Cite this article as :

Saurabh Takle, Shubham Desai, Sahil Mirgal, Ichhanshu Jaiswal, "Detecting Safe and Not Safe Driving Actions using Convolutional Neural Network", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 7 Issue 2, pp. 372-378, March-April 2021. Available at

doi : https://doi.org/10.32628/CSEIT217287

Journal URL : https://ijsrcseit.com/CSEIT217287