Distracted Driver Detection using Stacking Ensemble

Ketan Ramesh Dhakate
Computer Science Department
National Institute of Rourkela
Rourkela,India
ketandhakate1020@gmail.com

Ratnakar Dash
Computer Science Department
National Institute of Rourkela
Rourkela,India
ratnakar@nitrkl.ac.in

***Abstract*—Distracted driving is one of the primary causes of car crashes. While driving the vehicle, drivers frequently perform secondary activities that distract driving. A decrease in driver distraction is a critical aspect of the smart transportation system. To decrease accidents and improve safety, this paper proposes a distracted driver detection system that classifies various types of distracted activities using ensemble techniques. Different convolutional networks had been trained on images by eliminating the final layer to get there feature vectors. By using the stacking ensemble technique, we stack all the feature vectors to train it on a convolutional network. This stacking technique, which is used to detect the distracted driver posture, achieves 97% accuracy. The study shows how models predict the desired classes. The model proposed in this paper can be used in a real- time environment to detect activities done by the driver.**

***Index Terms*—stacking ensemble, distracted driver, CNN**

I. Introduction

The number of road accidents around the world is increasing in recent years. According to data provided by the Government of India in 2017, there were around five lakh road accident incidents in India, which killed around one and a half lakh people, and around five lakh people got injured. The majority of the accident was due to the usage of wireless devices like mobile phones, Bluetooth devices while driving the vehicle. Majorly the distracted driving can be distinguished in four types:

- Biomechanical Distraction: The driver performs some physical movements, e.g., drinking.
- Visual Distraction: Driver takes eyes off the road, e.g. Reading
- Cognitive Distraction: Driver takes the mind off the road,e.g. Talking
- Auditory Distraction: Driver takes ears off the road,e.g. Listening to music

Many activities can be performed by distracted driver such as a) Talking on a cellphone, b) eating something, c) talking to co-passengers, d) tuning radio or stereo system. The major cause of distraction is from inside of the car. Most car companies provide very advanced features in there premium segment which distract drivers from adjusting and operating those features.

The advancement of self-driving vehicles still requires human intervention. Thus while driving, a driver has to pay attention to a vehicle so that it can be prevented from fatal accidents. So detection of distracted driver is important to avoid possible accidents.

The paper is organized as follows. In Section-2, the recent development and research are described. The proposed method is discussed in Section 3. Section 4 describes the experimental setup and presents the results. The final section gives concluding remarks on some future research directions.

II. Literature Review

We try to summarize some current research works used to detect distracted driving posture. Berri and silva et al.[1] proposed a dataset of a front view of the driver's face and detect the use of mobile phones using the SVM-based model. Hoang NganLe et al.[2] proposed a method to detect the driver using a cellphone while having "hands on the wheel" using Faster CNN.

Zhao et al.[3] designed a dataset by taking images of the distracted driver during other activities while driving. They use different classifiers such as k-nearest neighbor classifier (KNN), Random Forest, Multi-Layer Perceptron classifier. Random Forest (RF) gives an accuracy of 90.50%. Zhao et al. [2012] improve MLP classifiers using some combined features of Pyramid Histogram of Oriented Gradient (PHOG) and spatial scale feature extractors. Yehya et al. [4][2018] prepare a new dataset and propose a model for estimating posture classification. They propose a weighted ensemble using a genetic algorithm for classification. They, too, explained the effect of different visual elements using hand and face localizations and achieved 94.29% classification accuracy. Colborn et al.[5] uses a pre-defined VGG-16 model for the classification of a distracted driver and achieves an accuracy of 80%. Abouelnag et al. proposed a real-time distraction detection approach using a combination of AlexNet and GoogleNet models with hand, face, and skin features. It achieved 95% classification accuracy. Funda G¨unes¸ et al. [6][2017] describe how to efficiently

978-1-7281-4862-5/20/$31.00©2020 IEEE

Fig. 1: Statefarm distracted driving dataset

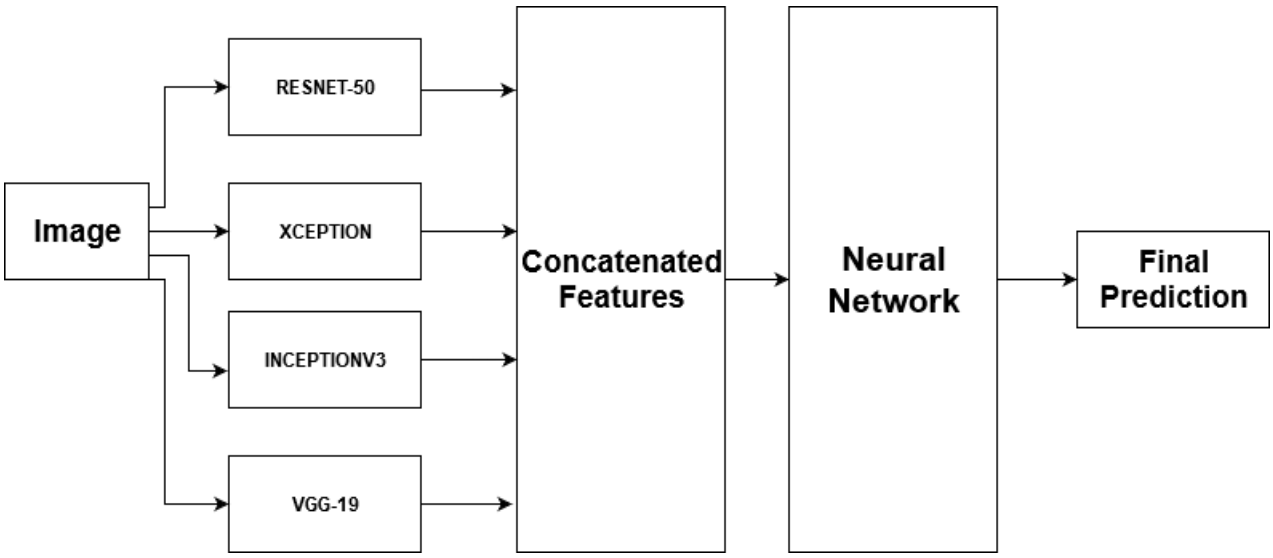

Fig. 2: Stacking Ensemble Model

perform an ensemble that performs better than naive models and helps to achieve better results. Zhou et al. has described Class Activation Map (CAM) to demonstrate how CNN model decides and watches the images.

The recent models we have discussed take more time to train the model and are not robust. This model also requires high computation to execute. The accuracy we got so far can be further enhanced to deploy in real-world applications.

III. Proposed Model

In the proposed scheme, we use the ensembling of different CNN models to predict the distraction. The concept of transfer learning is used to design an ensemble model. Transfer Learning is a concept of using Convolutional neural network (CNN) models, which have been pre-trained on the different datasets as initialization. It speed-up the training time and performance of the model.

CNN is specially designed to have an image as input. They are very much similar to neural networks. Layers of CNN transform one volume layer to another. For our proposed model, we trained five different pre-trained CNN models: VGG-16, VGG-19, InceptionV3, Resnet-50, Xception Models. These models were trained in recent researches, and each has given significant accuracy. All models have different directions to look into images and predict different classes for a given image. Thus to predict a model using a single classifier is not an optimal solution. So, the stacking ensembling technique is used on these CNN models to get better classification accuracy.[Fig 2]

As described in Algorithm 1,Stacking is a technique that

Algorithm 1 Stacking Ensemble

Input: Data for Training $D = \{x_j, y_j\}_{j=1}^{m}$

Output: Ensemble classifier E.

Step 1 : *learn base level classifier.*
for i = 1 to T **do**
    learn e_i based on D.
end for
Step 2 : *construct new data set of feature vectors.*
for j = 1 to m **do**
    $D_h = \{x'_j, y'_j\}$ *where* $x'_j = \{e_1(x_j), \ldots\ldots, e_T(x_j)\}$
end for
Step 3 : *learn a meta-classifier*
learn E based on D_e.
return E.

TABLE I: Performance Comparison of Different Models

MODEL	LOSS (categorical cross entropy)	ACCURACY(%)
VGG-16	1.3055	58.30
VGG-19	1.4248	55.7
INCEPTIONV3	0.30	92.90
RESNET-50	0.9973	82.50
XCEPTION	0.4296	90.00
STACKING ENSEMBLE MODEL 1 (XCEPTION+INCEPTIONV3)	**0.861**	**73.01**
STACKING ENSEMBLE MODEL 2 (RESNET-50+XCEPTION+ INCEPTIONV3 +VGG19)	**0.1154**	**97.00**

ensemble multiple classifications or regression models by a meta-classifier or meta-regressor. Different base-level models have been trained on the complete training set, then the meta-classifier or meta-regressor is trained on the output of the base-level model as features. The base-level often consists of different learning algorithms, and therefore stacking ensembles are often heterogeneous. Ensemble stacking can be referred to as blending because all the numbers are blended to produce a prediction or classification.

The input layer for all the CNN models is the pixel values of the input images. Our dataset consists of full-color images, so the input layer is 640 480 3×There×are three different channels:- Red channel, Blue Channel, Green Channel. The dataset has been trained on pre-trained CNN models by elimi- nating the final layer. Each output from the model is appended to make a feature vector. This feature vector is further trained on the neural network. Adaptive Moment Estimation (Adam) is an optimization method used to determine learning rates for each parameter. It stores the exponentially decaying average of past squared gradient-like RMSprop or Adadelta, but also keeps an exponentially decaying average of past gradients, like momentum. Categorical Cross-entropy loss is used to train these models. It is comprises of Softmax activation plus a Cross-Entropy loss. It is used for the multi-class classification problem.

$$\text{Category Cross Entropy} = \sum_{c=1}^{M} u \log(p) + (1-y)\log(1-p) \quad (1)$$

where y - binary indicator (0 or 1) if class label

c is the correct classification for observation o

p - predicted probability observation of class c

M - Number of classes

We have proposed two stacking ensemble model. In First Model ,feature vector of Xception and InceptionV3 are con-catenate.We have used this two pre-trained convolutional mod-els because they have given maximum accuracy.In Second Model ,feature vector of Xception, ResNet-50, VGG-19 and InceptionV3 is used.This model is proposed to get maximum from all pre-trained models and to yield better classification accuracy.

IV. Experiment

We divided the dataset into a 3:1 ratio for training and testing, respectively. Taking inputs from the training dataset, we train the pre-trained CNN model with the initial weights of imagenet. The final layer from the CNN models is not considered and saved the weights in a file. The new dataset is formed by combining the weights from different CNN models and train this on a simple neural network model with 10 neurons in output layer. While training the model, we have used Categorical Cross Entropy as loss function, Softmax as activation function, adam optimizers as an optimization parameter. Four thousand different images were used to test the proposed model. [Fig 3] To analyze how the model is understanding our problem and how it behaves in different scenarios, we plot Class Activation Mapping on test images. We train and test Model using a Tesla K80 GPU and 12 GB of RAM, Intel(R) Core(TM) i7-5960X CPU @2.70 GHz.

A. Dataset

Distracted driver dataset was published by Statefarm in which side view of the each driver in a car while doing some task in the car. The goal of this dataset is to predict the likelihood of driver posture. There are around 13K training images,4K validation images, and 4K testing images. In Fig 1 shows how distracted driver performs secondary activities while driving.

The dataset cam be categorized into 10 classes described below.

- Safe Driving (c0)
- Texting-Right (c1)
- Talking on the phone-right (c2)
- Texting-Left (c3)
- Talking on the phone-Left (c4)
- Operating the Radio (c5)
- Drinking (c6)
- Reaching Behind (c7)
- Hair and Makeup (c8)
- Talking to Passengers (c9)

V. Analysis

After the analysis,In Table 1 we observe that classifica- tion accuracy was 97 % of the Stacked Ensemble Model 2(ResNet50 + Xception + InceptionV3 + VGG-19) .Classi-fication accuracy of Stacked Ensemble Model 1(Xception +

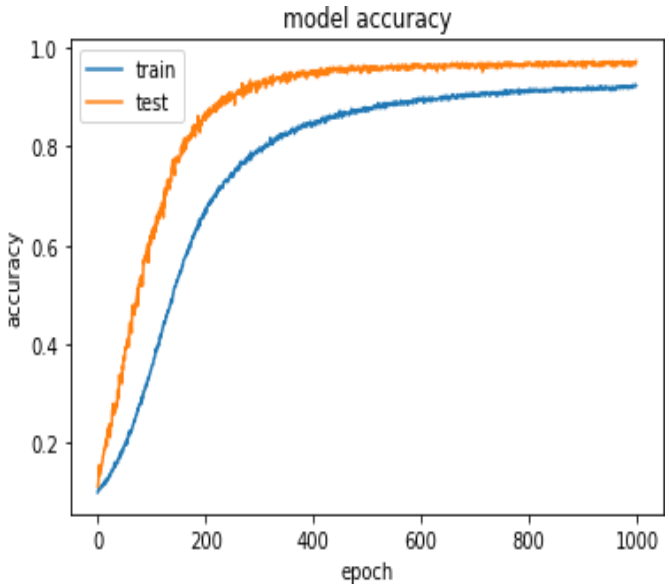

Fig. 3: Stacked Ensemble Model 2 Accuracy

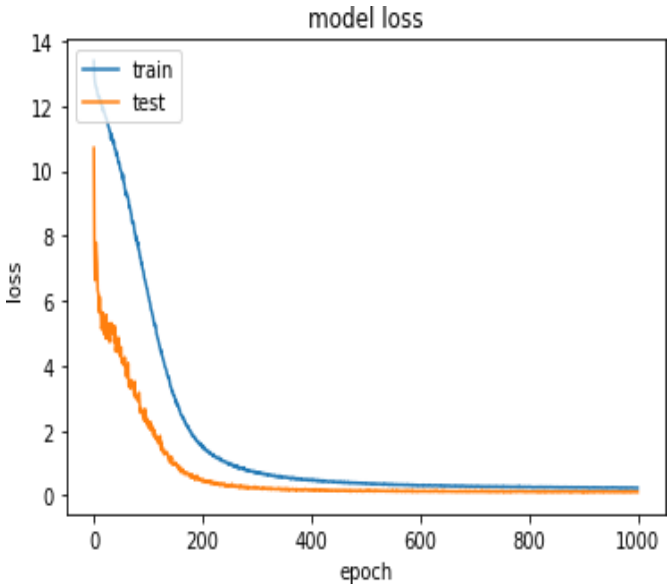

Fig. 4: Stacked Ensemble Model 2 Loss

InceptionV3) is 73%.Thus Stacked Ensemble Model 2 gives more accuracy than other models.From Figure 3 also shows that the proposed model is not overfitting. Initially , training accuracy was much less as compared to testing accuracy but after 600 epochs both accuracy not provide further increment. The categorical cross-entropy loss[Fig 4] for the training dataset is 0.2354, and for the testing, the dataset is 0.1154.Loss of both training and testing dataset didn't show any improvement after 600 ecpochs. From Table II We noted that safe driving is the most confusing posture in the dataset. This is due to the lack of temporal context in static images. Another class of Hair and Makeup got less accuracy of 0.93 because of the confusion caused to the model why the hand is raised. It can be raised for any of the other classes, also like talking on the phone or drinking water. We also required less computational time(approx 22 sec per epochs) to train our proposed model and to test our model. While training the ensemble model, the model was overfitting for initial epochs, but after certain epochs it model performed very well and gave a much better result.[Fig 3 and Fig 4]

From Table III, Talking on the phone-left (C4) and operating the radio (C5) are most precise classes to classify while Texting-Right (C1) is least precise. Recall of Texting-Right (C1) and Reaching Behind (C7) is maximum while Safe Driving(C0) and Hair and Makeup (C8) is minimum. Reaching Behind gives more f1-score than all other classes,

We also performed Class Activation Mapping[Fig 5] on the testing image to analyze how CNN mode sees the image and how it predicts the desired class from that image. It also shows how CNN works for pattern prediction problems. The region with red marking is more weighted as compared to other areas. For example, while safe driving, it will give more weights on are near to steering wheels and hands, while texting phone it will give weight on the area of hands by whom you are using a mobile phone.

TABLE II: Classification Report of Proposed Model

class	precision	recall	f1-score
C0	0.96	0.93	0.94
C1	0.93	1.00	0.96
C2	0.97	0.99	0.98
C3	0.95	0.99	0.97
C4	1.00	0.96	0.98
C5	1.00	0.98	0.99
C6	0.98	0.97	0.98
C7	1.00	1.00	1.00
C8	0.96	0.93	0.94
C9	0.95	0.95	0.95

TABLE III: Confusion Matrix of Proposed Model

ACTUAL \ PREDICTED	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9
C0	**0.93**	0.012	0.004	0.0034	0.002	0	0.002	0	0.02	0.014
C1	0	**1**	0	0	0	0	0.0022	0	0	0
C2	0	0.0022	**0.99**	0	0	0	0.0022	0	0.0022	0
C3	0.0021	0.0043	0.0021	**0.99**	0	0	0	0	0.0021	0
C4	0.0086	0.0021	0.0021	0.015	**0.96**	0	0.0021	0.0021	0.0021	0.0064
C5	0.0065	0.0022	0.0043	0.0022	0.0022	**0.98**	0	0	0	0.0022
C6	0	0.022	0.0043	0	0	0	**0.97**	0	0.0065	0
C7	0	0	0	0	0	0	0	**1**	0.005	0
C8	0.0052	0.016	0.021	0.0026	0	0	0.0079	0	**0.93**	0.021
C9	0.023	0.012	0	0	0	0	0.0023	0	0.016	**0.95**

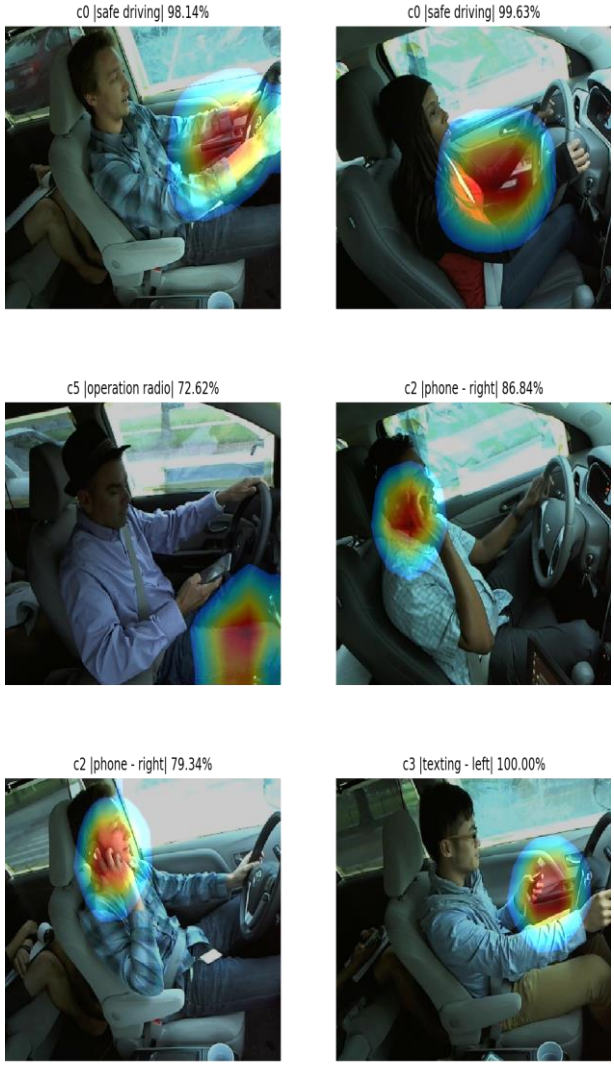

Fig. 5: Class Activation Mapping

VI. Conclusion

This study represents a distracted driver detection, which is based on different CNN architectures, which include VGG-19,InceptionV3, Xception, and ResNet50. The proposed model performs better than pre-trained models and takes less computational time also. Thus stacked ensemble approach achieves better performance than other model presented in earlier research. This system has the potential to be implemented in real cars to prevent road accidents. There are some areas where future research can be done.For example, The similarity between the different postures resulted in an incorrect prediction. Model often get confused due slightly change in body movement also. Another problem is if the driver is driving in the night and the car cabin is in the dark, then this model may not work properly, and according to survey, accident occur in the night are much more in numbers than an accident in the day. The future work can be done by using some sensors and actuators which provide us more accurate data. We can take photos from different angles, putting some sensors on the steering wheel, or microphones to record voice in the car.

References

[1] R. A. Berri, A. G. Silva, R. S. Parpinelli, E. Girardi, and R. Arthur, "A pattern recognition system for detecting use of mobile phones while driving," in *2014 International Conference on Computer Vision Theory and Applications (VISAPP)*, vol. 2. IEEE, 2014, pp. 411–418.

[2] T. Hoang Ngan Le, Y. Zheng, C. Zhu, K. Luu, and M. Savvides, "Multiple scale faster-rcnn approach to driver's cell-phone usage and hands on steering wheel detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016, pp. 46–53.

[3] N. Zhao, B. Reimer, B. Mehler, L. A. D'Ambrosio, and J. F. Coughlin, "Self-reported and observed risky driving behaviors among frequent and infrequent cell phone users," *Accident Analysis & Prevention*, vol. 61, pp. 71–77, 2013.

[4] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang *et al.*, "End to end learning for self-driving cars," *arXiv preprint arXiv:1604.07316*, 2016.

[5] Y. Abouelnaga, H. M. Eraqi, and M. N. Moustafa, "Real-time distracted driver posture classification," *arXiv preprint arXiv:1706.09498*, 2017.

[6] A. Rayes and S. Salam, "Internet of things—from hype to reality," *Springer*, 2017.

[7] N. Moslemi, R. Azmi, and M. Soryani, "Driver distraction recognition using 3d convolutional neural networks," in *2019 4th International Conference on Pattern Recognition and Image Analysis (IPRIA)*. IEEE, 2019, pp. 145–151.

[8] B. Huval, T. Wang, S. Tandon, J. Kiske, W. Song, J. Pazhayampallil, M. Andriluka, P. Rajpurkar, T. Migimatsu, R. Cheng-Yue *et al.*, "An empirical evaluation of deep learning on highway driving," *arXiv preprint*

arXiv:1504.01716, 2015.

[9] D. Held, J. Levinson, and S. Thrun, "A probabilistic framework for car detection in images using context and scale," in *2012 IEEE International Conference on Robotics and Automation*. IEEE, 2012, pp. 1628–1634.

[10] D. Wang and F. Qi, "Trajectory planning for a four-wheel-steering vehicle," in *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No. 01CH37164)*, vol. 4. IEEE, 2001, pp. 3320–3325.

[11] C. Craye and F. Karray, "Driver distraction detection and recognition using rgb-d sensor," *arXiv preprint arXiv:1502.00250*, 2015.

[12] E. Ohn-Bar, S. Martin, and M. Trivedi, "Driver hand activity analysis in naturalistic driving studies: challenges, algorithms, and experimental studies," *Journal of Electronic Imaging*, vol. 22, no. 4, p. 041119, 2013.