2022 25th International Conference on Computer and Information Technology (ICCIT)
17-19 December, Cox's Bazar, Bangladesh

An Ensemble Approach for Identification of Distracted Driver by Implementing Transfer Learned Deep CNN Architectures

Md. Abdur Rakib Mollah*, Azmain Yakin Srizon† and Mohiuddin Ahmed‡
Department of Computer Science & Engineering
Rajshahi University of Engineering & Technology, Rajshahi, Bangladesh
Emails: *rakib1703115@gmail.com, †azmainsrizon@gmail.com, ‡mohiuddin.nirob.mn@gmail.com

Abstract—**Road accidents are common issues caused by overpopulation in almost all countries around the world. In developed and developing nations, one of the main causes of road accidents is driver distraction. Due to the distraction of the drivers, the time required for decision-making can be reduced significantly and result in catastrophic damage. Texting or talking over the phone while driving, drinking, reaching behind, hair and makeup, operating the radio, and talking to the passenger are common causes of driver distraction happens. Previously, many works have been conducted in this domain as the lives of passengers depend on the successful driving of drivers. Diverse proposals have been introduced throughout the last decade for driver distraction detection including machine and deep learning approaches. Although some methods seem promising, most of them couldn't achieve the desired performance due to the lack of important feature extraction. The recent advances in transfer learning, however, created an opportunity for contribution in this sector as these approaches are capable of extracting deep features. In this study, we have considered 10 states of driver distraction and utilized two transfer learning approaches i.e., DenseNet121 and MobileNet for successful recognition of driver distraction. However, further application of ensemble learning produced a better accuracy than DenseNet121 and MobileNet alone. The proposed ensemble model achieved an overall accuracy of 99.81% whereas DenseNet121 and MobileNet achieved an overall accuracy of 99.66% and 99.73% respectively.**

Index Terms—**Distracted Driving, Computer Vision, Deep Convolutional Neural Network, MobileNet, DenseNet121, SoftMax Averaging**

I. Introduction

The world's population is rapidly increasing, as are the traffic demands on roads and highways. One problem that population growth creates is increased traffic demands on transportation networks, which can be devastating [1]. Around the world, 79.1 million automobiles were produced in 2021, a rise of 1.3% over 2020. [2]. The number of cars on the roads is increasing and as a result, driving has become riskier and, if improperly handled, may lead to tragic accidents. Despite efforts made in terms of improving roads and law enforcement, road traffic fatalities increased gradually over time, from 1.15 million in 2000 to 1.35 million in 2018 [3]. The severity and frequency of automobile accidents are increasing throughout the world.

In one study it is estimated that, overall, 58% of accidents were caused by drivers who were seen engaged in dangerously distracting actions while driving [4]. With the rapid development of technology, it is getting harder for drivers to focus on traffic. This situation increases driving errors and can lead to accidents. There are three basic categories of distraction: visual, manual, and cognitive [5]. All activities that require the driver to glance aside from the road are considered to be a visual distraction, like texting and leaning over to the back. Manual distraction includes all activities that demand the driver to take their hands off the wheel, such as using the radio, applying makeup, and having a drink. When a motorist is thinking about anything other than driving, like as conversing on the phone or engaging with a passenger, this is known as cognitive distraction [6].

Researchers have returned to the distracted driving issue multiple times over the years as a result of the creation of fresh datasets and the development of machine learning and deep learning techniques. Berri et al. in their 2014 paper, used a dataset that comprises a driver's frontal vision view [7]. Ohn et al. used a dataset that contained images focusing on wheel and hand movements. The dataset contained six classes [8].

Among the distracted driver detection datasets, the State Farm dataset is widely used and re-examined several times over the past few years due to the variety of classes it contains. This was the first publicly accessible dataset that took a wide range of distractions into account. One of the very first attempts to classify State Farm distracted drivers was done by Torres et al. in 2017 [9]. They used only five of the ten classes of the State Farm dataset in their paper for classification.

Abouelnaga et al. implemented a weighted ensemble of multiple CNNs in 2017, achieving 95.28% accuracy [10]. The images were preprocessed by the authors by adding skin, face, and hand segmentation, and a weighted ensemble of five distinct convolutional neural networks was suggested as the solution. The CNN computations required by this method must be completed in real time, which is not possible for embedded devices.

In 2019, Leekha et al. utilized a preprocessing idea to eliminate the background noise of the images. To increase performance, they deployed Grab Cut for foreground extraction. And the system reached an accuracy of 98.48% [11].

Chen et al. used a finely tuned vision transformer that searches for long-range dependencies on images instead of

979-8-3503-4602-2/22/$31.00 ©2022 IEEE

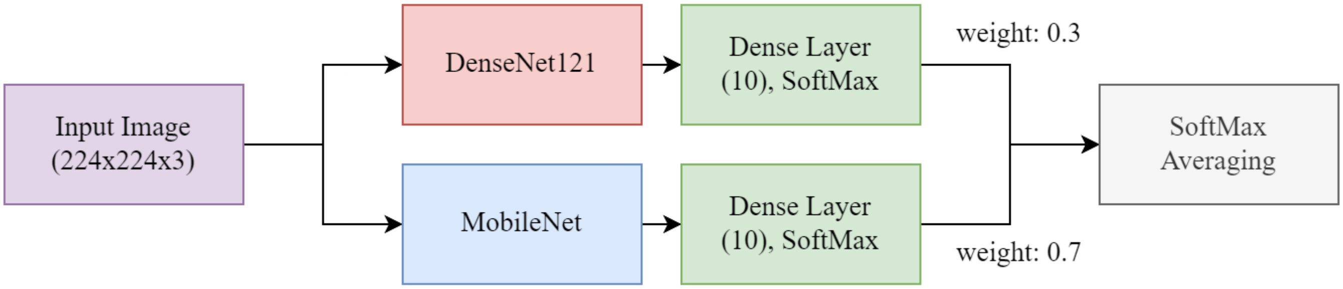

Fig. 1: Proposed ensembled learning of MobileNet and DenseNet121 (weighted SoftMax averaging)

finding local features from an image. This technique acquired 97.5% accuracy [12].

In 2021, Al-Doori et al. used transfer learning architecture to extract features from images and machine learning algorithms for classification. This process obtained 98.1% accuracy [13].

Bahari et al. implemented ResNet50 architecture and got accuracy of 94% in 2022 [14].

Ma et al. proposed a bi-linear fusion network with the addition of an attention module for classification with an accuracy of 95.08% [15].

In 2021, Sajid et al. acquired 99.16% accuracy after implementing a modified EfficientNet on the dataset [16]. But they reduced the classes from ten to eight.

In this paper, we suggest a weighted ensemble approach to classify all ten classes in the dataset using pre-trained transfer learning models. We selected parameter-efficient models MobileNet and DenseNet121 for transfer learning. Some of the earlier works have simply used a single model on the dataset and have not employed an ensemble. Dropout, a regularization strategy that helps minimize overfitting, has not been adopted by others. Additionally, it was discovered that certain earlier implementations lacked batch normalization, which normalizes the output of the preceding activation layer. These shortcomings are overcome in our proposed solution which achieved an accuracy of 99.66% on DenseNet121 and 99.73% on MobileNet. And a weighted ensemble of the two models reached an accuracy of 99.81% which surpassed the previous studies by a great margin.

II. Materials and Methods

A. Dataset Description

While conducting our research, the State Farm distracted driver detection dataset—which had 10 separate graphical groups or classes—was taken into consideration. [11]. The dataset included a standard amount of samples (22424 in total). The variety of picture sizes and formats, such as grayscale and RGB, ensured that the dataset's diversity was up to par. Images captured by a 2D dashboard camera constitute the dataset. The 10 classes to predict are:

c0: normal 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 passenger

B. Convolutional Neural Network

One of the most popular components of deep learning modules including convolution, pooling, and fully connected layers is the convolutional neural network, sometimes abbreviated as CNN. Convolutional neural networks have been used for years in applications such as medical image identification, natural language processing, Internet of Things, self-driving automobiles, and financial data analysis. By identifying the optimal filters, convolution layers often extract useful characteristics from the input data. Convolutions are used, pooling layers are used for reshaping, and fully linked layers are used for multilayer perceptrons. CNN is regarded as one of the most effective technologies for picture categorization since it doesn't require extra feature engineering.

C. Transfer Learning

Transfer learning seeks to advance traditional machine learning by applying knowledge gained from one or more source activities to a related target activity. The development of methods for knowledge transfer is a step toward making machine learning as effective as human learning. Image classification i.e. recognition of cats or dogs in an image is one area where machine learning thrives. To distinguish an animal as a cat or a dog, it is important to know that it has whiskers, paws, fur, and other features. Nevertheless, this knowledge is not useful for solving other common vision problems, such as attribute detection, scene identification, image retrieval, or fine-grained recognition. However, representations that capture the broad strokes of an image's composition, including the combinations of edges and patterns it contains, might be helpful. Thus, we can easily develop a new model using the extracted features from a state-of-the-art CNN that has already been trained on ImageNet to perform a new task. In order to prevent unlearning prior knowledge, we either maintain the pre-trained parameters fixed or adjust them with a slow

TABLE I: Class-wise accuracy, precision, recall, f1-score and support illustration for the ensemble model

Class	Accuracy	Precision	Recall	F1-Score	Support
0	0.99	1.00	1.00	1.00	499
1	1.00	1.00	1.00	1.00	454
2	1.00	1.00	1.00	1.00	464
3	0.99	1.00	1.00	1.00	470
4	0.99	1.00	1.00	1.00	466
5	0.99	1.00	1.00	1.00	463
6	1.00	1.00	1.00	1.00	465
7	0.99	1.00	1.00	1.00	401
8	0.99	1.00	0.99	0.99	383
9	1.00	1.00	1.00	1.00	427

learning rate. The image categorization issue can be resolved by transfer learning from a deep learning perspective. In fact, a number of state-of-the-art outcomes in picture categorization are based on transfer learning techniques.

D. MobileNet Architecture

In their paper titled MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications, a team of Google engineers presented MobileNet at CVPR 2017. The neural network model is shrunk by MobileNet's proposed depthwise separable convolution architecture, which enables it to address the resource-constrained concerns of edge devices. There are two primary components to the MobileNet architecture: (a) Depthwise separable convolution and (b) 1x1 pointwise convolution.

E. DenseNet121 Architecture

DenseNet is a more advanced architecture of CNN for visual object identification that uses fewer parameters to provide state-of-the-art performance. DenseNet and ResNet are quite similar, with a few key differences. While ResNet blends the output from the previous layer with the subsequent layers using an additive attribute (+), DenseNet employs concatenated (.) attributes to mix the output from the previous layer with the subsequent layer. Through tightly linking all levels, the DenseNet Architecture seeks to address this issue. This study used the DenseNet-121 architecture, which is one of three DenseNet models (DenseNet-121, DenseNet-160, and DenseNet-201). Details of the DenseNet-121 are as follows: 1 classification layer (16), 2 dense blocks (1x1 and 3x3 Conv), 5 convolution and pooling layers, 3 transition layers (6,12,24), and 1 transition layer.

F. Weighted SoftMax Averaging

MobileNet and DenseNet121 are different kinds of deep convolutional neural networks. Therefore, they extract different types of features from the same image. The two models were combined into a weighted average ensemble in order to benefit from both CNNs. We multiplied the probabilities of MobileNet and DenseNet121 with 0.7 and 0.3, respectively, to predict any class of an image. MobileNet was assigned the higher weight in the ensemble since it performed marginally better when used as a single model. Figure 1 illustrates the weighted SoftMax averaging technique used in this study.

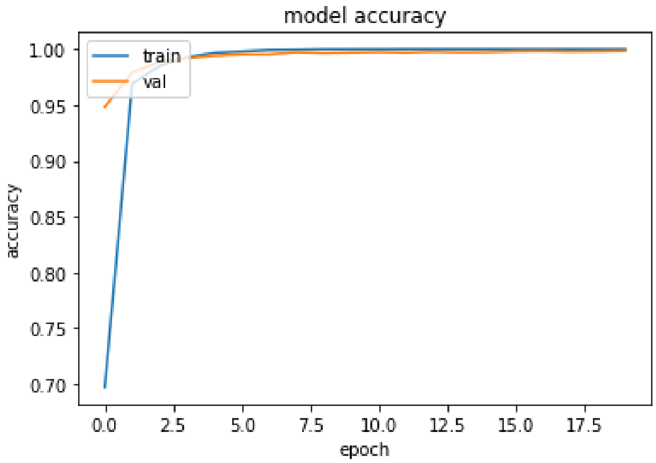

Fig. 2: Training accuracy and validation accuracy of MobileNet architecture

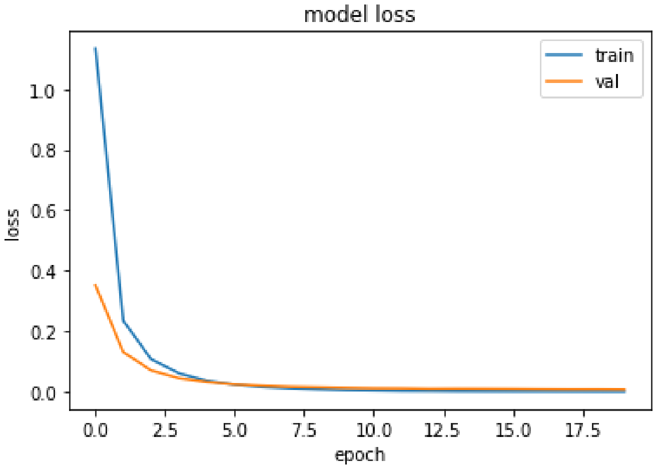

Fig. 3: Training loss and validation loss of MobileNet architecture

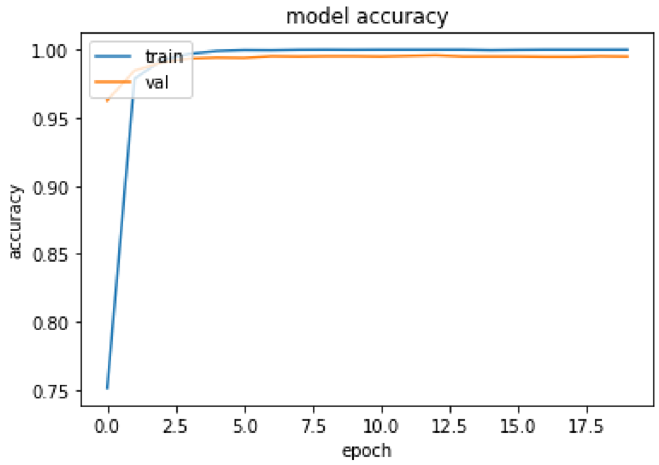

Fig. 4: Training accuracy and validation accuracy of DenseNet121 architecture

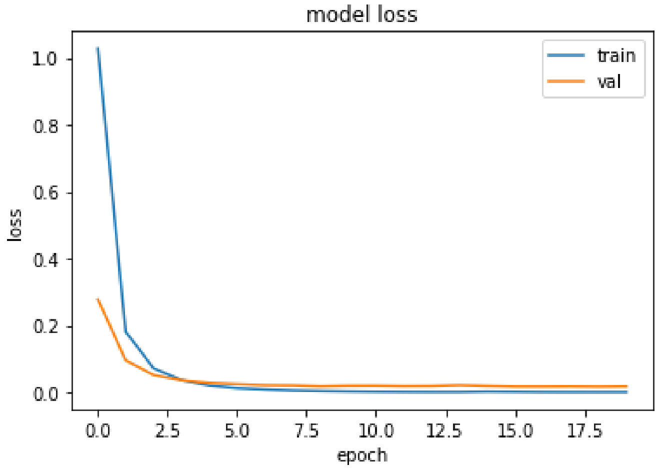

Fig. 5: Training loss and validation loss of DenseNet121 architecture

III. Experimental Analysis

A. Preprocessing

Since convolutional neural networks are capable of extracting important features on their own, little feature engineering is necessary. It was necessary to resize the images because MobileNet and DenseNet121 both capture images with an input size of 224x224x3. After dividing the dataset into the train, validation, and test sets, 15692 photos from the training set that belonged to 10 classes and 4481 images from the validation set that belonged to 10 classes were discovered. A test set of 2251 photos from 10 classes was employed.

B. Experimental Design

The MobileNet and DenseNet121 architecture were both trained in batches of 24 images, the maximum that our computer system could operate. 20 epochs were iterated to run the models. Dropouts were implemented and tuned for both models to avoid overfitting. For the loss function, a categorical cross-entropy function was applied. The learning rate was locked to 0.0000000001. It was optimized by employing 'Adam' optimizer. Since the "Adam" optimizer makes use of both of their components, momentum and RMSprop optimizations were excluded. All convolutional layers used ReLU activation.

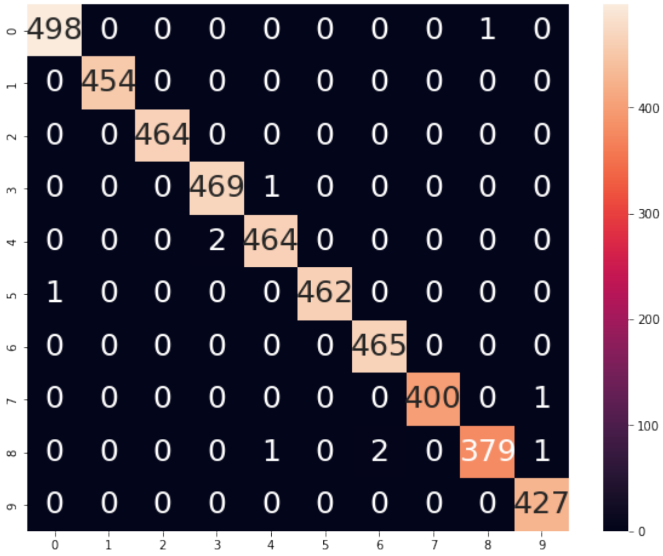

Fig. 6: Confusion Matrix of the ensemble of MobileNet and DenseNet121

TABLE II: Comparison of performances among our proposed approach and previous approaches

Work Reference	No. of Classes	Overall Accuracy
Abouelnaga et al. [10]	10	94.29%
Leekha et al. [11]	10	98.48%
Chen et al. [12]	10	97.50%
Al-Doori et al. [13]	10	98.10%
Bahari et al. [14]	10	94.00%
Ma et al. [15]	10	95.08%
Sajid et al. [16]	8	99.16%
Proposed Ensemble	**10**	**99.81%**

C. Result Analysis

The dataset was divided into 80:20 ratios at the beginning of the procedure, with 60% of the data being maintained for training and 20% being preserved as the validation set. A test set of 20% of the data was retained. The validation set differs from the test set in that it is a component of the training phase, where the objective is to simultaneously reduce training and validation loss. The test set, on the other hand, is completely autonomous and has no bearing on training. MobileNet and DenseNet121 were used following the split. DenseNet121 had a 99.66% accuracy and MobileNet had a 99.73% accuracy. By a narrow margin, they both effectively exceeded the earlier works.

But in order to further improve the overall performance, we employed a weighted ensemble and further made use of the softmax-averaging method, resulting in a 99.81 percent overall accuracy. Figure 2 and Figure 3 displays the training and validation accuracy and loss curves for the MobileNet architecture during the training phase. On the other hand, Figure 4 and Figure 5 depicts the training and validation accuracy and loss curves for the DenseNet121 architecture during the training phase. TABLE II shows the comparison of the overall accuracy of our proposed models to earlier efforts, whereas TABLE I shows the class-wise accuracy, precision, recall, f1-score, and support. It is obvious that our suggested models have outperformed the earlier efforts by a significant margin.

IV. Conclusion

The State Farm dataset from Kaggle was used in this study. We used MobileNet, DenseNet121, and Weighted SoftMax Averaging approach to recognize objects with a respective overall accuracy of 99.73%, 99.66%, and 99.81%. When the findings were compared, it became clear that our models had significantly outperformed the earlier studies. Additionally, MobileNet and DenseNet121 use 3.2 million and 7 million parameters, for a combined total of 10.2 million parameters, which makes our model more economical to apply. In the future, we'll strive to employ augmentation to reduce overfitting even more. Additionally, there is a plan to minimize parameters by modifying the CNN models.

References

[1] D. Y. Yang and D. M. Frangopol, "Societal risk assessment of transportation networks under uncertainties due to climate change and population growth," *Structural Safety*, vol. 78, pp. 33–47, 2019.

[2] O. Analytica, "Automotive output is falling but innovation is rising," *Emerald Expert Briefings*, no. oxan-db, 2022.

[3] F.-R. Chang, H.-L. Huang, D. C. Schwebel, A. H. Chan, and G.-Q. Hu, "Global road traffic injury statistics: Challenges, mechanisms and solutions," *Chinese journal of traumatology*, vol. 23, no. 4, pp. 216–218, 2020.

[4] C. Carney, D. McGehee, K. Harland, M. Weiss, and M. Raby, "Using naturalistic driving data to assess the prevalence of environmental factors and driver behaviors in teen driver crashes," 2015.

[5] Y. Zhang, D. B. Kaber, M. Rogers, Y. Liang, and S. Gangakhedkar, "The effects of visual and cognitive distractions on operational and tactical driving behaviors," *Human factors*, vol. 56, no. 3, pp. 592–604, 2014.

[6] Y. Liang and J. D. Lee, "Combining cognitive and visual distraction: Less than the sum of its parts," *Accident Analysis & Prevention*, vol. 42, no. 3, pp. 881–890, 2010.

[7] 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.

[8] 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.

[9] R. Torres, O. Ohashi, E. Carvalho, and G. Pessin, "A deep learning approach to detect distracted drivers using a mobile phone," in *International Conference on Artificial Neural Networks*. Springer, 2017, pp. 72–79.

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

[11] M. Leekha, M. Goswami, R. R. Shah, Y. Yin, and R. Zimmermann, "Are you paying attention? detecting distracted driving in real-time," in *2019 IEEE Fifth International Conference on Multimedia Big Data (BigMM)*. IEEE, 2019, pp. 171–180.

[12] H. Chen, H. Liu, X. Feng, and H. Chen, "Distracted driving recognition using vision transformer for human-machine co-driving," in *2021 5th CAA International Conference on Vehicular Control and Intelligence (CVCI)*. IEEE, 2021, pp. 1–7.

[13] S. K. S. Al-Doori, Y. S. Taspinar, and M. Koklu, "Distracted driving detection with machine learning methods by cnn based feature extraction," *International Journal of Applied Mathematics Electronics and Computers*, vol. 9, no. 4, pp. 116–121, 2021.

[14] M. S. H. S. Bahari and L. Mazalan, "Distracted driver detection using deep learning," in *2022 IEEE 18th International Colloquium on Signal Processing & Applications (CSPA)*. IEEE, 2022, pp. 198–203.

[15] C. Ma, H. Wang, and J. Li, "Driver behavior recognition based on attention module and bilinear fusion network," in *Second International Conference on Digital Signal and Computer Communications (DSCC 2022)*, vol. 12306. SPIE, 2022, pp. 381–386.

[16] F. Sajid, A. R. Javed, A. Basharat, N. Kryvinska, A. Afzal, and M. Rizwan, "An efficient deep learning framework for distracted driver detection," *IEEE Access*, vol. 9, pp. 169 270–169 280, 2021.