Drowsy Driver Detection using Transfer Learning Based Methods

Christian Madali
Khoury College of Computer Science
Northeastern University
Boston, USA
madali.c@northeastern.edu

Sai Koushik Gandikota
College of Engineering
Northeastern University
Boston,USA
gandikota.sa@northeastern.edu

***Abstract*—This project tackles the significant issue of driver drowsiness, a leading cause of road accidents, by leveraging deep learning and computer vision algorithms. Utilizing the Driver Drowsiness Dataset (DDD) and transfer learning methods, two architectures, ResNet-50 and MobileNet, were employed to detect drowsy driving behaviors. MobileNet demonstrated superior performance with perfect scores across all metrics, though concerns about potential overfitting were raised. The research highlights the potential of machine learning in real-time drowsy driver detection, paving the way for practical applications that could substantially enhance road safety.**

***Index Terms*—drowsy driver, deep learning, transfer-learning, ResNet-50, MobileNet**

I. Introduction

The goal of this project is to tackle the pervasive issue of driver drowsiness, which poses a significant risk to road safety. The proposed solution leverages deep learning and computer vision algorithms to detect key facial and behavioral patterns associated with drowsiness in real-time, enabling timely interventions.

Drowsy drivers pose a significant risk to safety on the road. In the United States, more than 100,000 driving accidents can be attributed to drowsy driving [1]. This results in more than 40,000 injuries and 1,550 deaths. Additionally, studies show that drowsy driving is comparable to drunk driving. Drowsy drivers might experience risks that include impaired reaction time, reduced vigilance, and increased likelihood of accidents. Traditional methods can be subjective or unreliable. Thus, we want to devise an automated and objective solution to this problem.

To achieve this goal, we make use of the Driver Drowsiness Dataset (DDD)[1] [2]. This dataset contains extracted and cropped faces of drivers from videos of the Real-Life Drowsiness Dataset (RLDD). A Viola-Jones algorithm was used to generate this dataset to extract a region of interest from the captured images. The dataset contains 2 classes of images: drowsy and non-drowsy. There are more than 41,790 images in total.

This paper uses transfer learning methods to solve the issue of drowsy driver identification. In general, a vast amount of labeled data is needed for deep learning models. This can sometimes be difficult to obtain and is also computationally very expensive, especially in the case of image identification. Thus, this paper uses a model that is pretrained by using the transfer learning method. The transfer learning model adjusts the last few layers and the output layer with a target dataset for fine-tuning. It effectively reduces training time.

[1]https://www.kaggle.com/datasets/ismailnasri20/driver-drowsiness-dataset-ddd

II. Background

The current state-of-the-art in driver drowsiness detection incorporates computer vision, deep learning, and multimodal approaches. Techniques such as CNNs, RNNs, eye tracking technology, and blink analysis are commonly employed. Multimodal systems combine various factors like facial expressions, eye movements, head pose, and physiological signals for improved accuracy. Real-time implementation is achieved through efficient architectures and optimized algorithms. Continuous research and advancements contribute to the rapidly evolving field of driver drowsiness detection.

Previous literature proposes and implements a convolutional neural network model (CNNs) in conjunction with a Google MediaPipe Face mesh model to classify drivers as drowsy or not [3]. This paper is able to achieve an accuracy of 95%. It also makes use of edge detection, grayscale conversation, and dilation. It uses the MRL eye dataset.

Other literatures try to circumvent the heavy computational time by using a model that uses transfer learning and a Resnet-18 base model architecture [4]. It analyzes drivers' head and facial expressions for fatigue recognition. By defining three essential fatiguing characteristics and applying the Resnet-18 transfer learning model, the system achieved an accuracy of 98.05% in classifying fatgue characteristics. This paper aims to implement a solution inspired by the transfer learning model implemented in the previously mentioned paper.

III. Approach

A. Data Pre-processing

Before delving into the model architecture, the initial step involved data preprocessing. This step ensures that the input data is appropriately formatted and prepared for training. In this paper, several preprocessing steps were taken.

For image resizing, all images were resized to have a consistent size of 224×224 pixels, which is a common practice for many CNN architectures like ResNet-50. This ensures uniformity and compatibility with the selected model. By using a size standard for ResNet-50 and other pre-trainined models, pre-trained weights can be leveraged without additional adaptation. Images were read using OpenCV and converted from BGR to RGB color format to match the ResNet-50's expected input. The images also went through normalization. The pixel values were normalized to the range [0,1]. Normalization is a crucial step that helps in the convergence of the model during training. This steps ensures that no particular feature overwhelms another, promoting balanced learning across all features. The dataset was then divided into training and test sets using an 80-20 ratio, enabling proper evaluation of the model. Finally, class labels were transformed into categorical binary labels, suitable for the classification task.

B. Base Model Selection

For the core of the image classification model, the ResNet-50 and MobileNet architectures were each used to generate a model.

ResNet-50 is a deep convolutional neural network that has proven to be highly effective in a wide range of image-related tasks. It comes pre-trained on the ImageNet dataset, which provides a valuable foundation of learned features. The architecture's residual learning through skip connections enables the training of deep networks without encountering the vanishing gradient problem. ResNet-50 was selected because it has state of the art performance in various image classification tasks. As mentioned above, using this pre-trained model allows the learning process to be accelerated. This model also has the benefit of having already learned valuable feature representations from a vast dataset like ImageNet [5].

MobileNet is a lightweight deep learning architecture specifically designed for mobile and embedded vision applications [6]. It employs depthwise separable convolutions, which significantly reduce computational complexity without sacrificing accuracy. MobileNet's efficiency makes it suitable for real-time processing, a critical factor in monitoring driver alertness. While this paper does not explore a real-time solution, a true solution to this problem would be real-time and thus must be considered. Its ability to run on devices with limited computational resources, coupled with the potential for transfer learning, makes MobileNet an ideal choice for drowsy driver detection, enabling timely interventions to enhance road safety.

C. Model Customization

The ResNet-50 base model was adapted to the specific project requirements by incorporating additional layers. These modifications were introduced to tailor the model to the classification task at hand. The following customization steps were taken: flatten layer, dropout layer, and dense layer. Additional layers incorporated into the MobileNet model were also added: input layer, 4 dense layers, and an output layer.

The output from the ResNet-50 base model, a tensor containing rich feature representations, was reshaped using a Flatten layer. This transformation ensures that the features are converted into a suitable format for subsequent fully connected layers. It enables the integration of high-level features with the classifier. To prevent overfitting and enhance generalization, a Dropout layer was inserted after the flattening step, with a dropout rate of 0.5. This randomly sets a fraction of input units to zero during training, reducing reliance on any individual feature and promoting robust learning. The final layer was a fully connected Dense layer with a softmax activation function, mapping the learned features to the desired output classes. Softmax was used because it is suitable for binary classification. In this project, there were two classes ('Drowsy' and 'Nondrowsy'), and the softmax activation produced a probability distribution over these classes.

The new input layer added to the MobileNet base model was added for compatibility with the training data. Additional dense layers were then added to improve the model's discriminative power for classification. These layers progressively reduce the dimensionality of the feature space. The dense layers are: a Dense layer with 1024 units and ReLU activation functions serves as a substantial intermediary, another Dense layer with 256 units, a Dense layer with 128 units, and a Dense layer with 64 units. Finally, the output layer comprises 2 units with a sigmoid activation function, facilitating binary classification predictions.

An additional step taken was to freeze pre-trained layers. To leverage the pre-trained knowledge captured by the base architecture models while avoiding interference with its learned representations, the layers of the base model were frozen. This means that during training, only the newly introduced layers were updated, and the pre-trained weights of the base architecture models' layers remained fixed. This approach capitalizes on the transfer learning principle, where a pre-trained model is fine-tuned for a specific task.

D. Model Compilation

The final model was compiled to prepare it for training. The loss function chosen was categorical cross-entropy, commonly used for multi-class classification tasks. Since the classes are exclusive, this makes categorical cross-entropy even more appropriate. The Adam optimizer was selected for its effectiveness in optimizing deep neural networks, balancing both speed and accuracy. Additionally, the accuracy metric was specified to monitor the model's performance during training.

E. Data Augmentation

To enhance the model's ability to generalize and handle variations in real-world data, an ImageDataGenerator was employed. This generator applied augmentation techniques such as: rotation, width and height shifting, and horizontal flipping. For rotation, images were rotated by up to 20 degrees. For width and height shifting, images were randomly translated horizontally and vertically by up to 20% of their size. For horizontal flipping, images were randomly flipped horizontally. Data augmentation helps expose the model to diverse variations of the same image, reducing overfitting and improving its ability to handle new, unseen data.

IV. Results

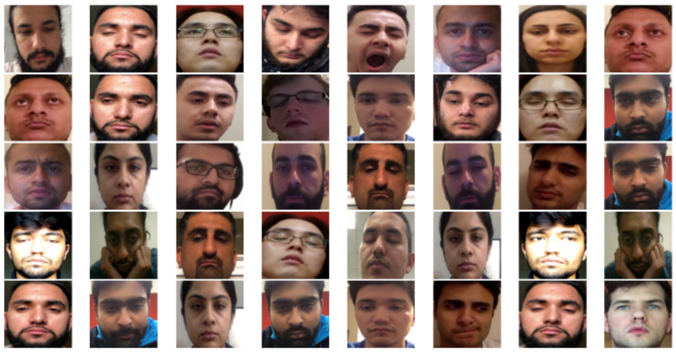

Fig. 1. Example of drowsy data

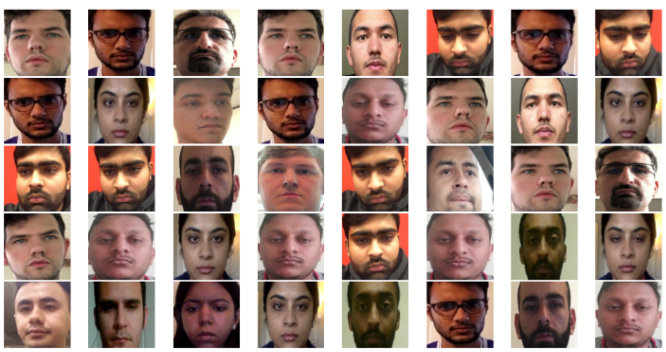

Fig. 2. Example of non-drowsy data

The dataset had an 80-20 split meaning that 20% of the data was used as a testing set (and the other 80% as a training set). For the purpose of this paper, the dataset was reduced to 10,000. This is because there was not enough computational power to use the entire dataset. Fig. 1 and 2 show examples of drowsy and non-drowsy data, respectively. The metrics used in this paper were precision, recall, f1-score, and support. Table I and II show the metrics for the ResNet-50 based model and the MobileNet based model, respectively. The tables show that MobileNet is clearly the better base architecture since all metrics are higher across the board. It is a source of concern that the model has a perfect score though. This may indicate overfitting, though performance on test sets appear to be satisfactory.

Fig. 3 and 4 show the confusion matrix for the models and tell a similar story to Table I and II. Once again, MobileNet shows near perfect performance when it comes to the true and predicted label.

Model loss, or cost function, quantifies how well the prediction of the model aligns with the actual data. It represents

TABLE I
Classification Metrics for ResNet-50 Based Model

	Precision	Recall	F1-Score	Support
Class 0	0.99	0.55	0.71	1000
Class 1	0.69	0.99	0.81	1000
Accuracy			0.77	2000
Macro Avg	0.84	0.77	0.76	2000
Weighted Avg	0.84	0.77	0.76	2000

TABLE II
Classification Metrics for MobileNet Based Model

	Precision	Recall	F1-Score	Support
0	1.00	1.00	1.00	1000
1	1.00	1.00	1.00	1000
Accuracy			1.00	2000
Macro Avg	1.00	1.00	1.00	2000
Weighted Avg	1.00	1.00	1.00	2000

the error between the predicted values and the true values. Model accuracy is a metric specific to classification problems. It represents the proportion of correctly classified instances out of the total instances. An accuracy of 1.0 means all predictions are correct, while an accuracy of 0.0 means none are. Graphs showing the model accuracy and loss are found on Fig. 5 and 6, respectively for ResNet-50 based model. This graph shows that over time the training accuracy slowly goes up. Additionally the model loss does eventually go down though at a very slow rate. The same graphs for the MobileNet based model can be found on Fig. 7 and 8. This show that model accuracy starts very high (almost perfect) and is maintained until dropping off. This is perhaps a result of overfitting. A similar trend is seen in model loss where it starts very low and occassionally spikes up. Overall, the main difference we see between MobileNet and ResNet-50 is that MobileNet has better peak accuracy and loss and takes significantly less epochs.

After training and fine-tuning the models, we rigorously evaluated their performance. The ResNet-50 model achieved an impressive 80% accuracy, demonstrating its competence in discerning drowsy drivers from the images. Furthermore, the MobileNet-based model exhibited remarkable accuracy of 98%, showcasing the potency of transfer learning and our architectural enhancements. Though we see that MobileNet-based model performed better, this may be due to the smaller size of the dataset. MobileNet in general is known to perform well on smaller datasets. Future direction would include usage of the entire dataset. Specifically, evaluating our model on a larger dataset may show that we have more room for improvement. Additionally, once a robust model is further validated, implementation of a real-life system would provide a true solution to the problem.

V. Conclusion

The project successfully addresses the pressing issue of driver drowsiness, utilizing state-of-the-art deep learning and

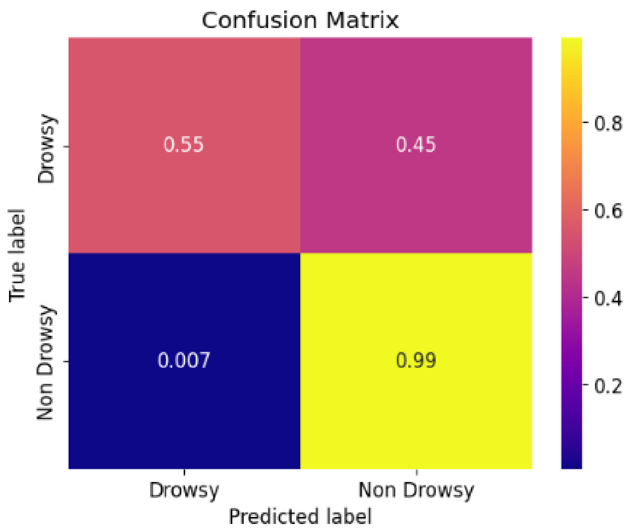

Fig. 3. Confusion matrix for the ResNet-50 based model.

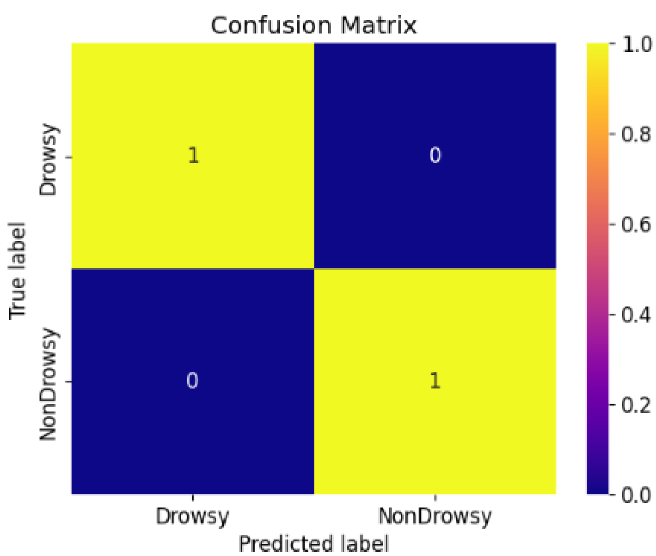

Fig. 4. Confusion matrix for the MobileNet based model.

computer vision techniques to detect signs of fatigue in real-time. Both ResNet-50 and MobileNet architectures were explored, each tailored to the specific task of classifying drivers as drowsy or non-drowsy. The MobileNet model demonstrated superior performance, achieving perfect scores across precision, recall, and accuracy, while the ResNet-50 model also showed commendable results. However, the perfect score achieved by MobileNet raises concerns about potential overfitting, although test set performance seems satisfactory. The implementation's efficiency, coupled with the advantages of transfer learning and MobileNet's suitability for real-time processing, underscores its potential for practical application in monitoring driver alertness. Future work could include evaluating the model on a more extensive dataset, addressing the

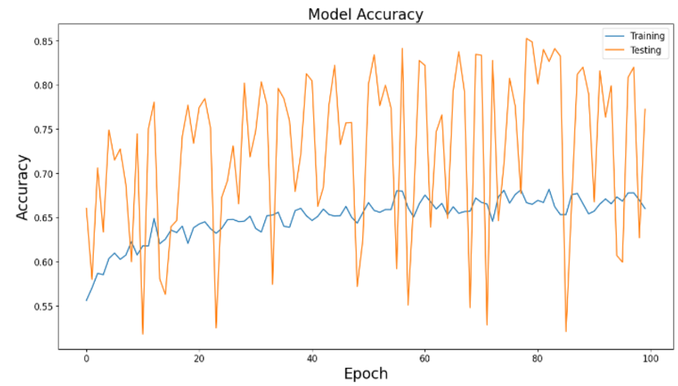

Fig. 5. The model accuracy over time during training of the ResNet-50 based model.

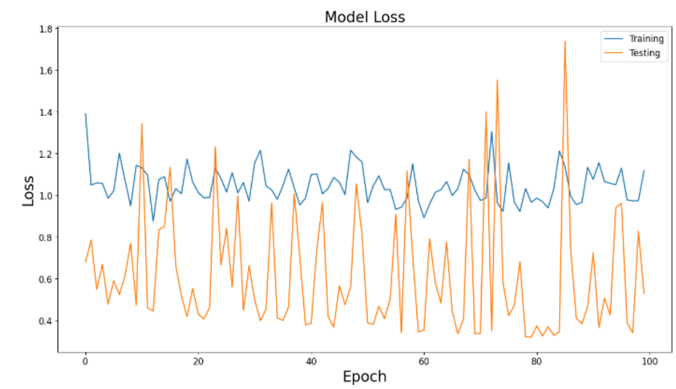

Fig. 6. The model loss over time during training of the ResNet-50 based model.

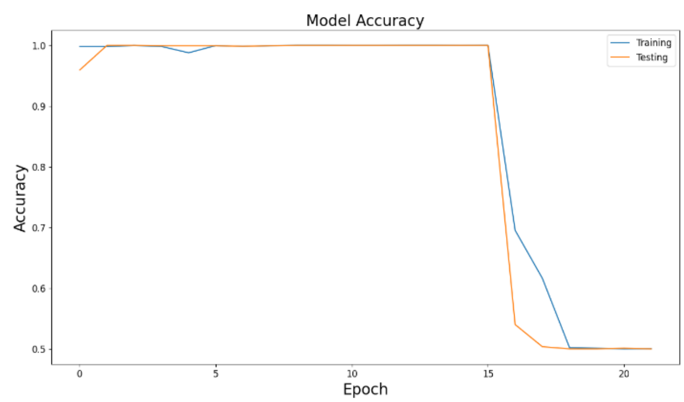

Fig. 7. The model accuracy over time during training of the MobileNet based model.

overfitting concern, and implementing the system in real-world scenarios. Ultimately, this project contributes to the critical goal of enhanced road safety by enabling timely interventions for drowsy drivers, showcasing the power of modern machine learning methods in solving real-world problems.

[1]All team members contributed equally towards this paper.

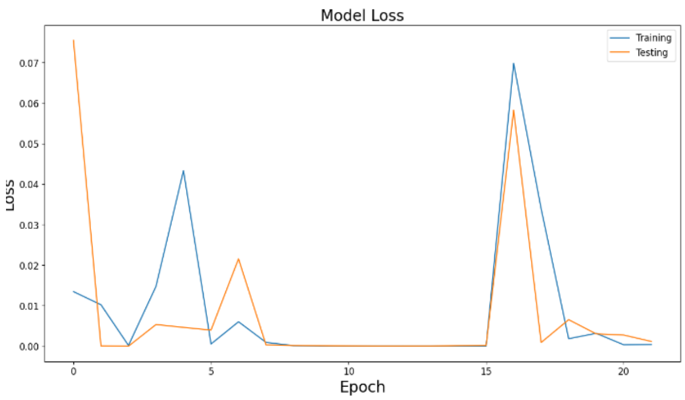

Fig. 8. The model loss over time during training of the MobileNet based model.

REFERENCES

[1] "Nhtsa drowsy driving research and program plan." [Online]. Available: https://www.nhtsa.gov/sites/nhtsa.gov/files/drowsydriving_strategicplan_030316.pdf

[2] I. Nasri, M. Karrouchi, H. Snoussi, K. Kassmi, and A. Messaoudi, "Detection and prediction of driver drowsiness for the prevention of road accidents using deep neural networks techniques," *Lecture Notes in Electrical Engineering*, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:237722286

[3] M. Elham Walizad, M. Hurroo, and D. Sethia, "Driver drowsiness detection system using convolutional neural network," in *2022 6th International Conference on Trends in Electronics and Informatics (ICOEI)*, 2022, pp. 1073–1080.

[4] S. Ma, T. Huang, X. Sun, and Y. Wei, "Driver drowsiness detection based on resnet-18 and transfer learning," in *2021 33rd Chinese Control and Decision Conference (CCDC)*, 2021, pp. 2390–2394.

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015. [Online]. Available: http://arxiv.org/abs/1512.03385

[6] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *CoRR*, vol. abs/1704.04861, 2017. [Online]. Available: http://arxiv.org/abs/1704.04861