Reducing Road Mishaps: A CNN Model for Driver Fatigue Detection

[1]Adedayo A. Sobowale, [2]T.A. Abdul-Hameed, [3]Peace O. Sobowale and [4]Adeshola B. Ogunleye,
[1,3,4] Department of Computer Engineering, Federal University, Oye-Ekiti
[2]Department of Electrical and Electronics Engineering, Federal Polytechnic Ayede.
sobowaleadedayo@gmail.com|abdulhameedat@federalpolyayede.edu.ng |nnamanipeace@gmail|comogunade98@gmail.com

Received: 15-MARCH-2024; Reviewed: 21-MARCH-2024; Accepted: 25-MARCH-2024
https://dx.doi.org/10.4314/fuoyejet.v9i1.9

ORIGINAL RESEARCH

Abstract— Over the last few decades, there have been substantial developments in a variety of domains, including computer science, artificial intelligence, and machine learning, which has accelerated the evolution and application of intelligent systems in specific areas such as transportation. One way these systems are used in transportation is through fatigue detection, and to enable this technology, Convolutional Neural Networks serve as the foundation on which this system is built. CNNs are frequently employed in computer vision applications because they can automatically extract pertinent features from input data without the need for manual feature engineering. This CNN-based fatigue detection system was built in-order to address the repercussions of fatigued driving by monitoring the driver's facial features in real-time so as to predict the fatigue levels and send out an alert to restore the driver's alertness level so as to reduce road mishap. The system relies on visual input from a camera, which sends this input to a fatigue detection program installed on a Raspberry Pi microprocessor and sends out via alerts via a display and audio alerts via a piezoelectric buzzer. The model shows excellent performance, consistently achieving high accuracy, peaking at 97.34% on the 64th epoch, and consistently maintaining high validation accuracy, reaching 96.02% in the 67th epoch, indicating its ability to function effectively.

Keywords— Fatigue Detection, Convolutional Neural Network, Drowsiness, Image Processing, Road Mishaps

1 INTRODUCTION

Technology that uses machine learning to address a particular problem is now referred to as an "intelligent system." Artificial Intelligence is present in these systems, but its goals are more practical (Sarker, 2022). The capacity of intelligent systems to adapt and gain knowledge from experience is one of their main characteristics. These programs analyze data and look for patterns and trends using machine learning algorithms. They can then make predictions or decisions in the present using this information. A few categories of intelligent systems that are frequently employed are Expert Systems, Decision Support Systems (DSSs), Fuzzy Logic Systems (FLSs), Bayesian Networks, Genetic Algorithm Systems (GANs), and Artificial Neural Networks (ANNs) (Agarwal and Singholi, 2018).
Artificial Neural Networks (ANNs) are a form of machine learning method that use interconnected nodes, or "neurons," to describe complex structures in data. ANNs are able to generalize to new, untested examples and learn from data. They have numerous uses in fields including robotics, natural language processing, and image and audio recognition (LeCun et al., 2015). The biological makeup and operations of the human brain served as the inspiration for a class of machine learning techniques known as ANNs. LeCun et al (2015) describes Artificial Neural Networks (ANNs) are a form of machine learning method that uses nodes with associations that generalize to new, untested examples and learn from data. They have numerous uses in fields including robotics, natural language processing, and image and audio recognition.
In order to handle data with a grid-like architecture, such as photos, movies, and audio spectrograms, Convolutional Neural Networks (CNNs) were developed. They are made up of many layers of neurons that carry out convolutional operations, pooling operations, and nonlinear transformations. CNNs have demonstrated impressive performance in image processing while performing tasks like picture classification, object detection, segmentation, and recognition. CNNs are highly suited for a wide range of applications, including medical imaging, , autonomous cars, surveillance systems, and facial recognition due to their capacity to automatically learn and extract complicated information from images (Goodfellow et al., 2016).
Convolutional Neural Networks (CNNs) have shown great promise in detecting driver fatigue through analysis of facial features and changes in driving patterns. By analyzing images of drivers' faces and movements, CNNs can identify signs of drowsiness or fatigue and alert drivers to take a break or stop driving. Fatigue detection is one of the many practical applications of CNNs in the field of transportation. In one study, Li et al. (2020), employed a CNN-based method to identify driver weariness in real-time. To assess the degree of driver sleepiness, the system was built to evaluate face traits such eye, head, and mouth states. The outcomes demonstrated the potential for CNNs to be applied in real-world applications such in-vehicle tiredness detection systems, with the CNN-based system achieving an accuracy of 96.55% in identifying driver drowsiness.
In general, the application of CNNs for driver tiredness identification in transportation has considerable promise for enhancing road safety and lowering the likelihood of

*Corresponding Author

Section B- ELECTRICAL/COMPUTER ENGINEERING & RELATED SCIENCES
Can be cited as:
Sobowale A.A., Abdul-Hameed T.A., Sobowale P.O. and Ogunleye A.B. (2024). Reducing Road Mishaps: A High-Accuracy CNN Model for Driver Fatigue Detection, FUOYE Journal of Engineering and Technology (FUOYE**JET**), **9(1)**, 56-61. https://dx.doi.org/10.4314/fuoyejet.v9i1.9

accidents brought on by driver drowsiness. By preventing accidents and saving lives, CNN-based fatigue detection systems may eventually be a standard component of automobiles (Wang et al., 2019).

2 Literature Review

2.1 Review Of Existing Fatigue Detection Systems And Their Limitations

There are many different systems for detecting fatigue in both developed and developing nations. In this part, a few of these systems will be discussed, highlighting their strengths and limitation, as well as how they relate to this research.

Jiang & Yang (2023) developed a system that assesses the driver's fatigue through physiological signals These devices collect data continuously, allowing real-time monitoring of the driver's condition and thus early detection of signs of fatigue. However, physiological signals can be affected by many factors, including individual differences, environmental factors, and emotional states Xiao et al. (2021) proposed a fatigue detection method for drivers using face recognition under a single sample condition. Optimal camera parameters were calculated and the images were refined using the maximum likelihood method. However, due to time constraints, the problem of model checking under single-sample conditions has not been further studied.

Zhang et al. (2018) used a convolutional neural network to create a real-time eye blink detection system for tiredness detection. They were successful in recognizing eye blinks, which are a sign of exhaustion, with a high accuracy of 97.5% using an Inception-v3 architecture to extract features from eye images. It's limitation was in that the dataset used in the study was limited to a specific age range and ethnicity. The system developed in the course of this project tries to address this by making the facial recognition more adaptive to the local populace.

Yan et al. (2018) employed infrared thermal imaging to identify driver drowsiness based on variations in facial temperature. Facial thermal pictures was gathered from 36 volunteers these images were classified as exhausted or not using a convolutional neural network. Using a network with 4 convolutional layers and 3 fully connected layers, they were able to reach a high accuracy of 91.8%. However it is limited to infrared thermal imaging which incurs more cost, and may not fully represent the variations in real-world conditions. The system developed in the course of this project did not require any special infrared thermal imaging system and hence was not limited by cost, environmental dependence and limited visibility through materials such as glass.

Wang et al. (2019) proposed a feature-fusion and convolutional neural network-based tiredness detection method. In order to extract information from photos of the eyes and faces, they employed a network with four convolutional layers and two fully linked layers. They successfully detected fatigue from eye and facial movements with a high accuracy of 97.6%. However, this work only used facial features to detect fatigue which may not be sufficient to accurately gauge fatigue under certain circumstances and is why the system developed in the course of this project uses both the eyes and mouth.

Singh et al. (2018) - developed deep convolutional neural network to analyze EEG signals to identify driver drowsiness. They extracted characteristics from EEG data using a 3-layer convolutional neural network, and their accuracy at identifying sleepiness in EEG signals was 95.1%. However this research proposes a system that may be obstructive or uncomfortable for the driver as it requires them to put on sensors while driving while also being limited to healthy individuals

3 Methodology

The architectural framework of the fatigue detection and alerting system is consists of a camera module for capturing the driver's facial features, a microprocessor, a screen for video output, an alarm, and a computer program written in Python to monitor and detect fatigue in real time. Convolutional Neural Networks (CNN) was used to create a fatigue detection program that can detect and warn when a person is getting tired when driving a vehicle, or engaging in any other task that needs constant focus and attention.

The program was written in Python language because python offers powerful libraries such as OpenCV, which offers computer vision functionalities perfect for image and video processing tasks often required in fatigue detection systems. It was also selected because of its clear and concise syntax and because of its integration with other tools. The process involves data collection, training the model with mathematical expressions that will calculate the Eye Are Region and Mouth Area Region while crosschecking it against a specified threshold, and then validating the system.

A generic USB camera from Dingdao Technology, also known as a webcam, is used to capture and record the face of the driver in real time. Essentially, the driver's blinking pattern and yawn pattern are being monitored by the camera. The video feed from the camera will serve as the input for the microprocessor. A Raspberry Pi 4 Model B microprocessor was used for the implementation of this project. The fatigue detection program was placed onto the Raspberry Pi 4 which will be running Raspbian OS and this microprocessor was chosen because of its speed and flexibility. The program processes the video from the camera frame-by-frame and apply a mathematical expression to calculate the Eye Aspect Region (EAR) and the Mouth Aspect Region (MAR) in-order to deduce the fatigue level of the driver.

Once the system detects that the driver is fatigued while driving, a piezoelectric buzzer will be used to generate audio alerts. The audio is expected to restore the alertness of the driver while they are the road. Figure 1a and 1b, depicts how the components are related functionally and the flowchart of the processes involved in the fatigue detection process.

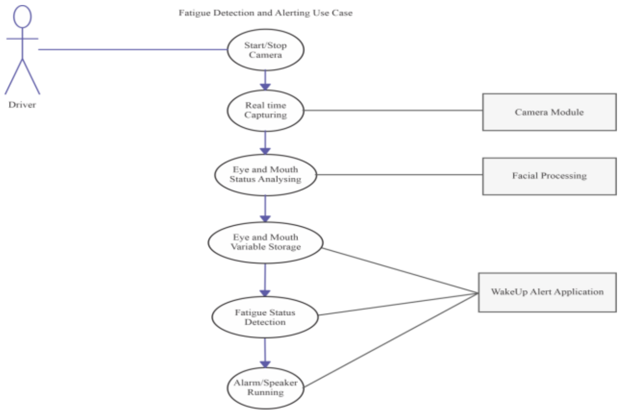

Figure 1: Use Case Diagram of the Fatigue Detection System

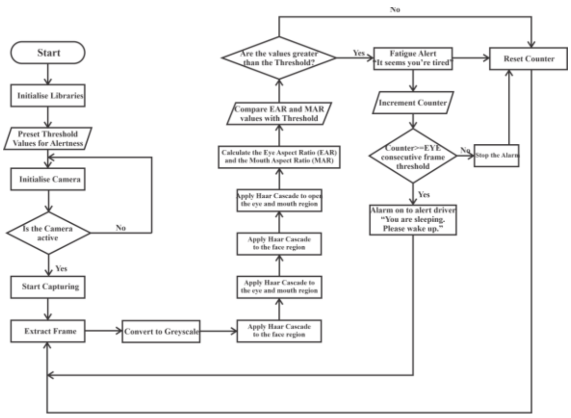

Figure 2: Flowchart Diagram of the Fatigue Detection System

The model we used in this research consists of several layers that work together to process the input image and make predictions. The initial layers focus on extracting features through convolutions and reducing spatial dimensions through max pooling. The first convolutional layer takes the input image of size 145x145 with 3 color channels (RGB) and applies 16 filters of size 3x3. This layer employs the ReLU activation function to introduce non-linearity. The subsequent batch normalization layer normalizes the outputs, stabilizing and accelerating the training process. The max pooling layer reduces the spatial dimensions of the feature maps while retaining the most important information. These operations result in an output shape of (71, 71, 16)

Following the initial layers, the model repeats a similar pattern of convolution, batch normalization, max pooling, and dropout operations to extract deeper features. The second convolutional layer applies 32 filters of size 5x5 to the previous layer's outputs, followed by batch normalization and max pooling. The third convolutional layer increases the number of filters to 64, using larger filters of size 10x10. Again, batch normalization and max pooling are applied. These layers progressively reduce the spatial dimensions, resulting in an output shape of (12, 12, 64). The structure of our model and its component layers is shown figure 3.

After the convolutional layers, the model transitions into fully connected dense layers to perform classification. The fourth convolutional layer employs 128 filters of size 12x12 and is followed by batch normalization. The subsequent flatten layer reshapes the tensor to a 1-dimensional form, preparing it for the dense layers. The first dense layer, with 128 neurons, applies the ReLU activation function. A dropout layer with a rate of 0.25 is inserted for regularization. The second dense layer reduces the number of neurons to 64, again using ReLU activation.

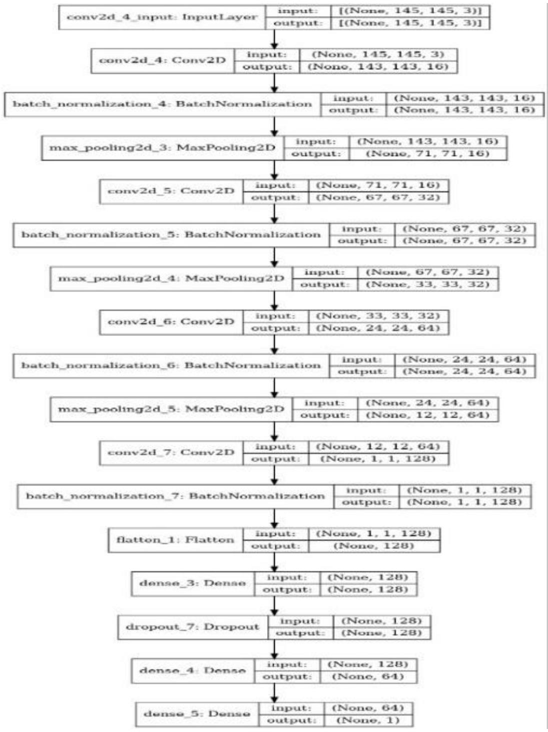

Figure 3: Organisational Structure of the Model

The output layer consists of a single neuron with a sigmoid activation function. It produces a binary classification output, representing the predicted class (0 or 1). The model is compiled with the binary cross-entropy loss function, accuracy as the metric, and the Adam optimizer. The summary of the model (table 4.1) provides a comprehensive overview of the layers, including the input and output shapes of each layer, aiding in understanding the model's architecture and parameter sizes.

The Eye Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR) are calculations used in computer vision to analyze facial features, particularly in applications like drowsiness detection. Eye Aspect Ratio (EAR) is calculated based on the distances between specific points around the eye. The formula used is:

$$EAR = (||p2-p6||+||p3-p5||)/(2*||p1-p4||)$$

Where:

|| || represents the Euclidean distance between two points.

p1, p2, p3, p4, p5, and p6 are specific points on the face

https://dx.doi.org/10.4314/fuoyejet.v9i1.9 engineering.fuoye.edu.ng/journal

identified by facial landmark detection algorithms, in this case the dlib68 point model.
Moving on, Mouth Aspect Ratio (MAR) uses a similar concept to EAR but focuses on the mouth. The formula used is:

$$MAR = (||p2-p6||+||p3-p5||)/||p1-p4||)$$

Where: p2 and p6 correspond to the outer corners of the mouth.
p3 and p5 correspond to the inner corners of the mouth.
p1 and p4 might be placed on the upper lip.

Table 1: Summary of the Model

Layer Type	Output Shape	Param #
conv2d_4	(None, 143, 143, 16)	448
batch_normalization_4	(None, 143, 143, 16)	64
max_pooling2d_3	(None, 71, 71, 16)	0
conv2d_5	(None, 67, 67, 32)	12832
batch_normalization_5	(None, 67, 67, 32)	128
max_pooling2d_4	(None, 33, 33, 32)	0
conv2d_6	(None, 24, 24, 64)	204864
batch_normalization_6	(None, 24, 24,64)	256
max_pooling2d_5	(None, 12, 12, 64)	0
conv2d_7	(None,1, 1, 128)	1179776
batch_normalization_7	(None, 1, 1, 128)	512
flatten_1	(None, 128)	0
dense_3	(None, 128)	1651
dropout_7	(None, 128)	0
dense_4	(None, 64)	825
dense_5	(None, 1)	65
Total Parameters		**1423713**
Trainable parameters		**1423233**
Non-trainable parameters		**480**

The details of all submission requirements must be according to the specified rules and guidelines stipulated within the 'GfA'.

4 Results And Discussion

The model shows excellent performance, achieving high accuracy throughout the training process, with a peak accuracy of 97.34% during the 64th epoch. Validation accuracy also remains high, reaching up to 96.02% in the 67th epoch. The model's strong performance is attributed to its architectural choices, including convolutional layers with varying filter sizes, batch normalization after each convolutional layer, and dropout layers. These features help capture spatial hierarchies, extract meaningful features, stabilize the learning process, and prevent over-fitting. The model's robustness and ability to generalize to unseen data is also noteworthy.

The model uses a sigmoid activation function in the output layer to produce a probability-like output, indicating the likelihood of input belonging to a class. The binary cross-entropy loss function measures the dissimilarity between predicted and true labels for binary classification. The Adam optimizer optimizes this loss function, improving accuracy. The model's architecture, regularization techniques, and activation and loss functions demonstrate strong performance. Figure 2a and 2b shows the graph of accuracy and loss obtained from the training of our model.

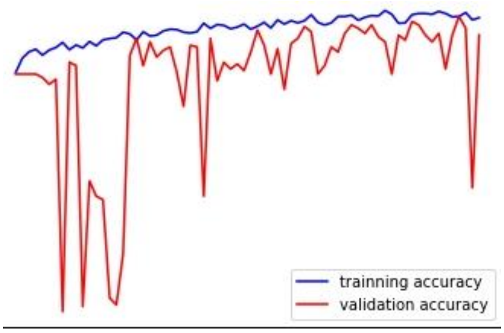

Figure 4: Graph of model accuracy over 70 epochs

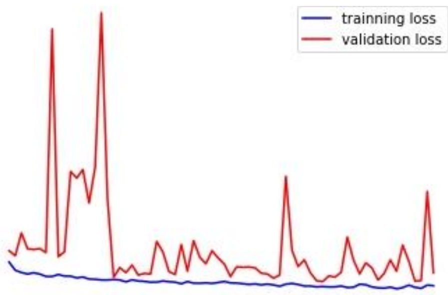

Figure 5: Graph of model loss

From the graphs above, the Training Accuracy illustrates how the accuracy of the model improves over epochs as it learns from the training data. For our system. it started low and increased gradually. While the Validation Accuracy Curve demonstrates how well the model generalizes to unseen data. It usually follows a similar pattern to the training accuracy but may plateau or even decline if the model starts to overfit. During the course of the project, it increases and stabilizes, indicating good generalization. Furthermore, the Training Loss Curve illustrates how the loss (error) of the model decreases over epochs as it learns from the training data and the Validation Loss Curve demonstrates how well the model generalizes to unseen data in terms of loss.

4.1 Confusion Matrix

The confusion matrix (given in table 2) is a table that represents the performance of a classification model. We used it to summarize the results of classification in terms of True Positives (TP), False Positives (FP), True

https://dx.doi.org/10.4314/fuoyejet.v9i1.9 engineering.fuoye.edu.ng/journal

Negatives (TN), and False Negatives (FN).

Table 2: Confusion Matrix

	Predicted No Drowsiness	Predicted Drowsiness
Actual No Drowsiness	3427	252
Actual Drowsiness	44	2331

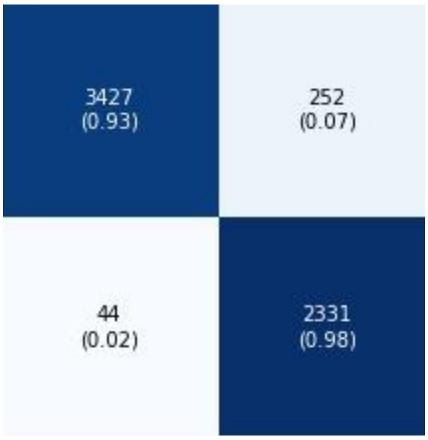

Figure 6: Confusion matrix of the model

4.2 F1 Score

The F1 score is a metric that combines precision and recall into a single value. It is calculated as the harmonic mean of precision and recall. The formula for F1 score is:

F1 = 2 * (precision * recall) / (precision + recall)

To calculate precision and recall, we need to use the values from the confusion matrix:

Precision = TP / (TP + FP) = 3427 / (3427 + 44) ≈ 0.9874

Recall = TP / (TP + FN) = 3427 / (3427 + 252) ≈ 0.9313

Now we to calculate the F1 score:
F1 = 2 * (0.9874 * 0.9313) / (0.9874 + 0.9313) ≈ 0.9584

4.3 Support

Support refers to the number of occurrences of each class in the actual data. In this case, the support for the "No Drowsiness" class is 3427 + 252 = 3679, and the support for the "Drowsiness" class is 44 + 2331 = 2375.

4.4 Recall

Recall, also known as sensitivity or true positive rate, measures the model's ability to correctly identify positive samples. It is calculated as TP / (TP + FN). The recall for the "No Drowsiness" class is:

3427 / (3427 + 252) ≈ 0.9313,

and the recall for the "Drowsiness" class is:

2331 / (44 + 2331) ≈ 0.9818.

4.5 Precision

Precision measures the model's ability to correctly identify positive predictions. It is calculated as TP / (TP + FP). The precision for the "No Drowsiness" class is:

3427 / (3427 + 44) ≈ 0.9874

and the precision for the "Drowsiness" class is:

2331 / (2331 + 252) ≈ 0.9022.

4.6 Accuracy

Model accuracy refers to the number of classifications a model correctly predicts divided by the total number of predictions made. The accuracy of our model is calculated as:

Accuracy = (TP + TN) / (TP + FP + TN + FN) 'Accuracy = (3427 + 2331) / (3427 + 44 + 2331 + 252)

Accuracy ≈ 0.9115 or 91.15%.

Table 3: Result Summary:

Metric	Model Performance
Training Accuracy	95.67%
Testing accuracy	91.15%
Validation Accuracy	91.15$%
Drowsiness precision	90'22%
No Drowsiness Precision	98.74%
Drowsiness Recall	98.18%
No Drowsiness Recall	93.13%
F1-score	95.84%

5 Conclusion And Recommendation

The development and implementation of a fatigue detection and alerting system is a crucial step towards enhancing road safety and preventing accidents caused by drowsy driving. Throughout the development and implementation of this research project, the objectives highlighted in chapter one have been achieved. The calculated F1 score of 0.9584 indicates a balance between precision (0.9874) and recall (0.9313). In practical terms, this means that our model is both reliable (it often makes correct predictions, as indicated by the high precision) and robust (it captures a large proportion of positive instances, as indicated by the high recall). The model's robustness and generalizability are attributed to the ReLU architecture's activation function to introduce non-linearity and architectural features, including convolutional layers, batch normalization, and dropout layers, which effectively capture spatial hierarchies.

A major limitation observed with the system is lightening. Because the system was developed with a webcam, the accuracy is dependent on adequate environmental lightening. If the driver is in a poorly lit environment or it driving at night, accuracy might be reduced. To aid this, we used a camera with an inbuilt LED but it is safe to say

https://dx.doi.org/10.4314/fuoyejet.v9i1.9 engineering.fuoye.edu.ng/journal

that proper lightening is paramount in order to get the best results from the system.

5.2 RECOMMENDATIONS

In relation to the results and outcome of this research project, some recommendations include partnering with automotive manufacturers, insurance companies, and government agencies to encourage the use of fatigue detection systems and potentially provide incentives for their implementation. Another is incorporation various alert mechanisms like audible alarms, visual warnings, and haptic feedback to cater to diverse driver preferences and those with hearing impairments. Finally, there is cost consideration and also testing the system rigorously under various driving conditions and scenarios to ensure its reliability. The flexibility of this system also indicates that it can be adapted for other applications such as heavy machinery operations and other forms of transportation beyond driving.

REFERENCES

Agarwal, D., & Singholi, K. S. (2019). Review of Expert Systems and Its Application in Robotics. International Journal of Computer Applications, 10.1007/978- 981-10-5903-2_131.

Xiao, C., Han, L., & Chen, S. (2023). Automobile Driver Fatigue Detection Method Based on Facial Image Recognition under Single Sample Condition. Symmetry 2021, 13, 1195. https://doi.org/10.3390/sym13071195.

Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. (2015). Internet of things: A survey on enabling technologies, protocols, and applications. IEEE Communications Surveys & Tutorials, 17(4), 2347-2376.

Buolamwini, J., Raji, I., Gebru, T., Mitchell, M., Lee, J., & Denton, E. (2020) Saving Face: Investigating the Ethical Concerns of Facial Recognition Auditing. In Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society (AIES '20). Association for Computing Machinery, New York, NY, USA, 145–151. https://doi.org/10.1145/3375627.3375820

Jiang, M., & Xinran, Y., (2021). Study on Fatigue Driving Detection based on Physiological Characteristics of Drivers. Applied and Computational Engineering. 32.223-231.10.54254/2755-2721/32/20230215.

Chen, X., Yu, B., Gong, X., & Huang, Z. (2020). An intelligent system for pedestrian tracking and behavior analysis using computer vision and deep learning. IEEE Transactions on Intelligent Transportation Systems, 21(4), 1444-1454.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT press.

He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask R-CNN. In Proceedings of the IEEE international conference on computer vision (pp. 2980-2988).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).

Huang, Y., Yu, L., Liu, M., & Guo, L. (2020). A Multi-Task Learning Framework for Driver Drowsiness Detection based on Facial Features. IEEE Access, 8, 48998-49007.

Karray, F., Alemzadeh, M., & Kamel, M. S. (2018). Soft computing and intelligent systems for machine learning. Springer.

Kim, K. H., Bang, J. H., & Lee, S. Y. (2020). An effective driver fatigue detection system using a deep neural network with multi-level fusion. Sensors, 19(6), 1456.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436- 444.

Li, H., Zhang, Y., & Sun, Z. (2020). A deep learning-based driver fatigue detection system using multi-modal features. Sensors, 20(10), 2881.

Li, K., Gao, Y., Li, X., & Zhang, Y. (2019). A review of vision-based driver fatigue detection systems. IEEE Access, 7, 16345-16359.

Li, Y., Zhang, X., & Chen, J. (2019). Driver Drowsiness Detection based on Convolutional Neural Network using Facial Landmarks. IEEE Access, 7, 142246-142256.

Liu, J., Zhang, Y., & Cheng, L. (2021). Deep neural networks: A comprehensive survey. Neurocomputing, 460, 1-22.

Litman, T. (2019). Autonomous vehicles and the future of urban transportation. International Journal of Transportation Science and Technology, 8(2), 101-107.

Russell, S., & Norvig, P. (2018). Artificial Intelligence: A Modern Approach (3rd ed.). Pearson.

Sarker, I.H. AI-Based Modeling: Techniques, Applications and Research Issues Towards Automation, Intelligent and Smart Systems. SN COMPUT. SCI. 3, 158(2022). https://doi.org/10.1007/s42979-022-01043-x

Tan, M., & Le, Q. V. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. Proceedings of the 36th International Conference on Machine Learning, PMLR 97:6105-6114.

T.F. Scott Darling, III, Federal Motor Carrier Safety Administration. (2021). Hours of Service. Retrieved from https://www.fmcsa.dot.gov/regulations/hours-service/hours-service-drivers

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J.,Gomez, N., Gomez, L., & Polosukhin, I. (2017). Attention is all you need. Proceedings of NeurIPS, pp. 5998-6008

Wang, H., Xue, Y., & Tang, J. (2020). A review on driver fatigue detection using physiological signals. IEEE Access, 8, 5147-5164.

Wu, Z., Wei, W., Chen, X., Chen, Y., & Chen, L. (2020). A review of intelligent logistics systems: Concepts, technologies, and applications. IEEE Access, 8, 185912-185926.

Yang, M., Liu, Z., Zhou, X., Huang, J., & Wang, Y. (2019). Driver fatigue detection using multi-feature fusion based on a hierarchical deep neural network. IEEE Access, 7,130807-130817.

Zhang, J., Williams, S., & Wang, H. (2018). Intelligent computing system based on pattern recognition and data mining algorithms. Sustainable Computing: Informatics and Systems, Volume 20, 2018, Pages 192-202, ISSN 2210-5379

Zhang, X., Li, Y., & Sun, Z. (2020). A review of driver fatigue detection systems. IEEE Access, 8, 45617-45634.