Application of Convolutional Neural Networks for Automated Iris Edge Detection in Sleepiness Monitoring during Blended Learning

Tukino[1,*], Yuhandri[2], Sumijan[3]

[1]*Department. Information System, Faculty of Creative Industry, Media Nusantara Citra University, Jakarta, Indonesia*

[2,3]*Department. Information Technology, Faculty of Computer Science, Putra Indonesia University YPTK Padang, Padang, Indonesia*

(Received: February 20, 2025; Revised: May 18, 2025; Accepted: August 07, 2025; Available online: September 11, 2025)

Abstract

This study introduces a novel lightweight Convolutional Neural Network (CNN) model, T-Net, designed for real-time drowsiness detection based on eye closure patterns. The model was developed to address the prevalent issue of student fatigue in resource-constrained environments, such as during prolonged online learning or blended learning sessions. Unlike traditional deep learning models, T-Net prioritizes efficiency while maintaining high accuracy, making it suitable for deployment on devices with limited computational resources. The model uses a 68-point facial landmark detection technique to extract the eye region and accurately classify eyelid states (open or closed). Evaluated on two benchmark datasets, Dataset-1 (342 eye images) and Dataset-2 (1,510 eye images), T-Net demonstrated superior performance, achieving classification accuracies of 99.33% and 99.27%, respectively, outperforming other pre-trained models such as VGG19, ResNet50, and MobileNetV2. Usability testing revealed a high acceptance rate, with a System Usability Scale (SUS) score of 84.5, indicating the system's practicality for real-world use. Additionally, statistical analysis showed a significant correlation ($r = 0.67$, $p < 0.01$) between prolonged screen time and the emergence of visual fatigue symptoms. This study highlights the effectiveness of a lightweight CNN approach for real-time fatigue monitoring, offering a balance between performance and computational efficiency. The results suggest that T-Net can be effectively integrated into student monitoring systems to ensure alertness during learning sessions. Future research will focus on expanding the dataset, integrating infrared imaging for low-light environments, and incorporating additional fatigue indicators such as yawning and head pose.

Keywords: Convolutional Neural Networks, Iris Edge Detection, Sleepiness Monitoring, Blended Learning, Computer Vision in Education

1. Introduction

Drowsiness represents a state of diminished alertness where individuals experience overwhelming fatigue, often resulting in impaired cognitive function and involuntary sleep episodes. This condition poses significant risks in situations requiring sustained attention, particularly behind the wheel. Current detection methodologies fall into two distinct categories. The first approach monitors vehicle operation parameters including steering behavior, lane positioning, and braking frequency through contactless monitoring systems. These indicators however demonstrate variability contingent upon roadway characteristics [1]. The alternative methodology employs biosignal acquisition techniques such as cerebral activity monitoring (EEG) and cardiac rhythm analysis (ECG) to evaluate operator fatigue states [2]. While such physiological measurements yield highly accurate fatigue assessments [3], [4], their implementation necessitates physical sensor attachment to the student [5]. Although biosignal-based systems offer superior detection reliability, their practical deployment faces substantial implementation challenges compared to non-contact alternatives.

Road safety statistics reveal a troubling connection between student fatigue and traffic accidents, with numerous collisions, physical harm, and fatalities attributed to this cause. The implementation of fatigue detection systems that can alert drowsy students has become increasingly crucial for accident prevention. NHTSA research indicates the staggering consequences of drowsy driving, including annual economic damages exceeding $12 billion, approximately 71,000 people suffering injuries, and nearly 1,550 lives lost each year. (NHTSA) statistics [6] show that in 2022,

***Corresponding author: Tukino (mas.kino@gmail.com)**
DOI: https://doi.org/10.47738/jads.v6i4.882

crashes involving drowsy students resulted in 50,000 injuries and 795 deaths. Researchers believe that it is important to be able to recognize signs of fatigue based on behavioral indicators, such as changes in the lips, eyes, or other facial characteristics.

By analyzing these indicators, researchers want to create tools to detect student fatigue and implement safety measures to prevent accidents [7]. Systems for identifying student impairment [8], Human-Computer Interaction (HCI) [9], Facial Expression Recognition (FER) [10], Brain-Computer Interface (BCI) [11], health [12], etc. can be designed and developed more easily by using eye status detection systems. Most applications utilize eye status data, either directly or indirectly.

This study focuses on individuals aged 15–24 years, encompassing both older adolescents and young adults. This age range aligns with global classifications used by organizations such as WHO and UNESCO in educational and health research. The integration of Artificial Intelligence (AI) and User Experience (UX) design in this work follows a human-centered approach. Usability heuristics—such as visibility of system status, user control, and error prevention—were mapped to the behavior of the AI system to enhance interpretability. This included the use of intuitive visual indicators to convey system feedback without requiring medical knowledge from the user.

Several computer vision-based methods for sleepiness detection have been developed over the past few decades to monitor student alertness. Eye closure, nodding, and yawning are examples of facial expressions that can indicate sleepiness. The frequency and duration of students' eye closures increase and the frequency and duration of eye openings decrease when they feel tired [13].

Researchers have made significant progress in drowsiness detection technology. New deep learning techniques address pose variations and incorporate mouth and eye features to improve accuracy. Lightweight models and hierarchical frameworks are also being developed for real-time applications and specific environments such as suburban roads. To ensure real-world effectiveness, research focuses on evaluating robustness to challenges such as occlusions and generalization across different conditions and populations. Additionally, the field is expanding beyond car students by developing a drowsiness detection model for crane operators, highlighting its potential for diverse applications. While our study introduces the DrowsyDetectNet framework, there is still a gap in the literature regarding the development of lightweight models for this purpose. Current methods often rely on complex architectures that are not suitable for resource-constrained environments such as vehicular systems. The development of efficient drowsiness detection systems requires innovative solutions that balance performance with practicality. Current challenges call for optimized models that minimize computational demands without sacrificing detection accuracy. This research investigates a streamlined CNN framework featuring a simplified architecture with reduced layers, specifically designed to work effectively with smaller datasets. Such advancements could significantly enhance real-world student monitoring systems by making them more accessible and responsive while maintaining reliable performance.

The development and use of student fatigue detection technology are driven by several important factors. First and foremost, this technology enhances safety by providing real-time monitoring of drowsiness, alerting students when their alertness declines so they can take necessary precautions. Additionally, it contributes to accident reduction by identifying signs of fatigue early, thereby minimizing risks. Another key benefit is increased productivity, as these systems help maintain focus, reduce errors, and improve overall efficiency. From an economic perspective, fatigue detection devices offer cost savings by preventing accidents that could lead to financial losses for individuals and institutions. Lastly, advancements in sensor technology, machine learning, and artificial intelligence have made these systems more accurate and affordable, further accelerating their adoption. Together, these reasons highlight the value of fatigue detection in promoting safety, efficiency, and innovation in educational environments.

This research introduces an innovative approach to drowsiness detection through a computationally efficient CNN model with a simplified architecture. The proposed solution differs significantly from conventional deep networks like VGG19, InceptionV3, MobileNetV2, and ResNet50 by employing a minimalistic design that requires fewer computational resources. Specifically designed to work with small datasets, our model prioritizes key facial indicators of fatigue, particularly eye closure patterns. This lean architecture demonstrates particular suitability for deployment in resource-constrained environments where real-time processing is essential, offering practical advantages over more complex alternatives.

The structure of this manuscript is as follows: The second section discusses the literature on drowsiness detection; the third section discusses the components and methods used in the proposed system; and the fourth section presents the experimental results along with details of each CNN design. The fifth section concludes with recommendations and findings for further research.

2. Literature Review

In this study, Phan et al. [14] for drowsiness detection involves testing and training phases. In the training phase, the footage captured by the vehicle safety system is processed to detect the face and head area using a specific network. The extracted images are then used to train deep neural networks, such as Inception-V3, DenseNet, LSTM, and VGG-16, with improvements made to their layers for drowsiness detection. In the testing phase, the trained model is evaluated on a separate dataset to identify drowsiness with an accuracy rate of 98%. Previous research has developed several CNN-based approaches for student fatigue detection. Faisal and colleagues [15] introduced a real-time monitoring solution that begins with facial image acquisition, followed by precise eye region localization. Their framework assesses student alertness through predefined criteria and issues appropriate warnings. The CNN architecture incorporates image feature extraction, data preprocessing, and careful tuning of critical hyperparameters including kernel configuration, learning rate, pooling dimensions, and training cycles. Experimental results demonstrated exceptional performance with 99.33% training accuracy and 97.98% validation accuracy. In a separate study, Ganguly et al. [16] implemented a dual-network system combining conventional CNN with Faster R-CNN methodology. Their approach initially identifies ocular regions using a Faster R-CNN detector, which integrates convolutional operations with max-pooling layers. Subsequent stages employ region proposal networks for object detection probability estimation, culminating in eye state classification through a series of convolutional and pooling operations in a standard CNN architecture.

In their study, Magan et al. [17] developed an innovative fatigue monitoring system that analyzes sequential facial images to assess student drowsiness levels. Integrated as a core component of Advanced Student Assistance Systems (ADAS), their solution employs sophisticated facial feature analysis while optimizing system performance through enhanced detection reliability and minimized false alerts. The framework specifically emphasizes timely identification of fatigue symptoms to improve road safety. The system uses 10 Frames Per Second (FPS) to capture 600 frames in a 60-second period, which are then processed and analyzed to assess the level of drowsiness and trigger appropriate alarms if necessary. Florez et al. [18] proposed six steps in the student drowsiness identification process: data acquisition, pre-processing of video frames using facial landmark detection, building a dataset, testing the trained model, training a CNN architecture, and predicting student fatigue. The pre-processing step includes a methodology for selecting Regions of Interest (ROIs) around the eyes by calculating the distance between facial points, ensuring that the ROIs capture relevant information even during head movements. The trained models are evaluated, and the best performing models are used for student drowsiness prediction. Jahan et al. [19] proposed the use of a custom CNN model called 4D to identify drowsiness based on eye conditions. This model consists of several layers: convolution, activation, batch normalization, dropout, max-pooling, fully connected, and output. In addition, this paper mentions the use of transfer learning CNN models, specifically VGG19 and VGG16, for the image classification task. For model training, we employed the MRL Eye dataset containing 47,173 annotated images of both open and closed eye states, achieving a classification accuracy of 97.53%. In related work, Akrout and Fakhfakh [20] developed an advanced fatigue detection framework that combines multiple computer vision techniques. Their approach begins with facial landmark detection and head pose estimation using MediaPipe Face Mesh, followed by a novel iris detection and normalization process. The system leverages MobileNetV3's feature extraction capabilities to process ocular characteristics, supplemented by geometric measurements of facial point distances and head orientation angles. These multimodal features are then analyzed through an LSTM network for temporal fatigue pattern recognition. The methodology also includes detailed iris region analysis, incorporating segmentation and normalization procedures to enhance feature quality.

Kumar et al. [21] proposed an approach that leverages a hybrid deep learning approach, combining InceptionV3 and LSTM, to analyze the mouth and eye areas for spatial feature extraction. The modified InceptionV3 incorporates a global average-pooling layer and a dropout layer to improve adaptability and prevent overfitting, respectively. The

modified InceptionV3 output is then fed into an LSTM to determine whether the student is drowsy or not, with an accuracy of 93.69%. Liu et al. [22] proposed a hybrid deep neural network workflow and design for fatigue detection in crane operators. The workflow involves capturing videos, detecting the operator's face, extracting facial landmarks, and extracting fatigue features to train a fatigue classifier. The three main modules of the architecture—Face Detector, Spatial Feature Extractor using MobileNet, and Time-Based Characteristic Modeling using LSTM—are connected by a learning network to determine the level of fatigue and, if necessary, initiate an alert. Mu et al. [23] proposed a technique used to remove nuisance factors such as noise and uneven lighting in the collected images. Common image noises include Gaussian noise and impulsive noise, and filtering techniques such as Gaussian filter, median, and average are used to reduce their effects. In addition, human eye state recognition methods, such as the Hough transform, are useful for determining the human eye state based on the detection of the presence or absence of a circle, indicating the eye is open or closed.

Recent studies have demonstrated various innovative approaches to student drowsiness detection. Phan et al. [24] introduced a dual-method framework combining adaptive thresholding of facial landmarks (EAR and LIP metrics) with a customizable deep neural network architecture. Their system incorporates advanced models like SSD-ResNet-10, benefiting from transfer learning to enhance efficiency. Zhu et al. [25] developed a multi-task TCDCN algorithm capable of simultaneous facial attribute analysis, showing robustness against common challenges like occlusions. Abbas et al. [26] presented ReSVM, a hybrid architecture combining ResNet-50's feature extraction with SVM classification, demonstrating effectiveness across varied imaging conditions.

Jia et al. [27] enhanced traditional MTCNN through SPP layers and batch normalization, improving facial cue detection accuracy. Mohamed et al. [28] provided a comprehensive evaluation of deep learning approaches, including detailed performance metrics and dataset characteristics. Dua et al. [29] implemented an ensemble system integrating four specialized models (FlowImageNet, AlexNet, VGGFaceNet, ResNet) for multi-modal drowsiness analysis.

Jamshidi et al. [30] established a hierarchical visual processing pipeline addressing challenges like illumination variations. Saurav et al. [31] developed DCNNE, an ensemble classifier combining two pre-trained CNNs for enhanced eye-state recognition. Bajaj et al. [32] reviewed implementation trends, particularly in developing nations, including hardware configurations using Raspberry Pi platforms.

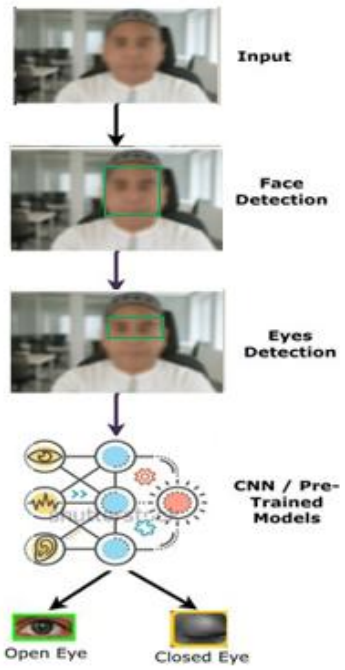


Figure 1. Proposed DrowsyDetectNet framework

Flores-Monroy et al. [33] proposed a real-time technique to identify student fatigue consisting of several stages, including face identification using the Viola & Jones formula, face analysis using a custom-designed T-Net (SS-CNN),

 ISSN 2723-6471

and successive result analysis. SS-CNN was designed to categorize facial regions into open and closed eyes. The selected configuration of SS-CNN had approximately 600K trainable parameters, enabling real-time operation using a compact GPU system. Chirra et al. [34] proposed a deep CNN-based technique to identify drowsiness that extracts eye areas and detects faces using the Viola-Jones face detection method. After feeding these eye areas into a CNN with four convolutional layers for feature extraction, the images were classified as drowsy or not using a Softmax layer. Using a test data sample with an accuracy of 96.42%, the proposed approach was shown to be successful in identifying student fatigue based on eye states.

3. Methodology

3.1. Proposed Drowsydetectnet Framework

This study aims to develop a DrowsyDetectNet framework to detect whether a student is drowsy or not. Figure 1 illustrates the proposed system architecture. To determine the location of the student's face in the input image or video, a 68-point facial landmark detection algorithm is used. Next, the eye area is extracted from the face. To identify "eyes open" or "eyes closed," the extracted eye images are fed into the proposed T-Net model.

Figure 2. Extraction Of Regions of Interest (ROI)

3.2. Facial Recognition and Eye Area Extraction

The Region of Interest (ROI) extraction process involves separating and analyzing specific areas in an image that contain relevant information about the eyes, as shown in figure 2. To identify a student's drowsiness, it is not necessary to use the entire face, but only the eye area. The 68 (x, y) positions corresponding to the facial structures are estimated using the facial landmark detector from the Dlib library. In figure 3, the following are the 68 coordinates in question: jaw ranging from 1 to 17, right and left eyebrows ranging from 18 to 22 and 23 to 27, nose ranging from 28 to 36, right and left eyes ranging from 37 to 42 and 43 to 48, mouth ranging from 49 to 60, and lips from 61 to 68.

This process involves detecting and cropping eyes from the image using the Dlib library for face and landmark detection. The process described in Algorithm 1, from detecting facial landmarks to identifying the eye area, is mainly performed by a 68-point facial landmark detector algorithm. The user provides an image of a human face, which is used by a pre-trained detector to identify 68 different landmarks that define key facial features, such as the mouth, nose, eyes, and facial contours. The algorithm determines the indices corresponding to the left (index 37 to 42) and right (index 43 to 48) eyes, calculating bounding box coordinates to define the eye area. While the identification step is, cropping the eye area from the image is done manually using the calculated coordinates. These manually cropped eye areas are then fed into a T-Net model, which processes the eye area to detect signs of drowsiness, with a special focus on eyelid closure. This approach combines detection for accuracy with manual intervention for proper input data preparation for the drowsiness detection model. The image you uploaded seems to be a scatter plot, displaying points that form a figure. It is labeled with numbers corresponding to coordinates along the x and y axes, and these points appear to be arranged in a way that forms a recognizable shape.

 ISSN 2723-6471

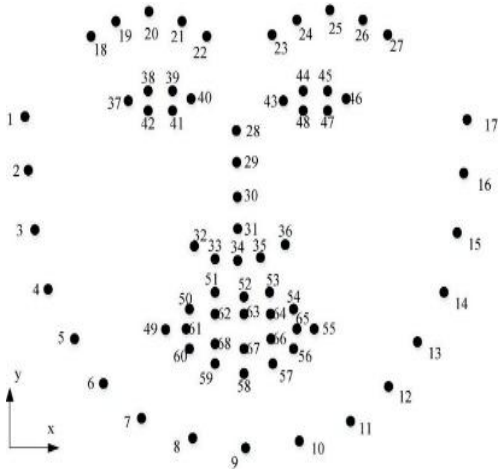


Figure 3. The-Face-Shape-With-68-Landmarks

The image depicts a diagram consisting of black dots distributed within a coordinate plane. The horizontal (x) and vertical (y) axes are marked, with numbers surrounding the dots indicating the position of each point within the coordinate system. The dots appear to be scattered randomly, although some clusters of points are denser, potentially forming patterns that suggest specific relationships or structures. The numbering of the points allows for easier identification and may serve purposes for further analysis, such as in statistical or geometric contexts. This image can be utilized for various purposes, including data visualization or analyzing point distribution in a two-dimensional space.

3.3. Proposed T-Net Model Design

The convolution process starts at the top left corner of the given image, scans horizontally until it covers the entire row, then moves down to repeat the process. The output values of this operation will create a feature map, which is defined by equation (1):

$$X(m, n) = (I \times K)[m, n] = I[a, b] \times K[i - a, j - b] \quad (1)$$

K is the kernel, I is the input image, X is the feature map, m is the row index of the convolved matrix, and n is the column index. The facial landmark detection process begins with a shape predictor identifying 68 distinct facial points. Specific point clusters (indices 37-42 for the right eye and 43-48 for the left eye) are isolated to derive ocular region coordinates. These coordinates enable precise cropping of both eye regions from the original image frame. The extracted eye images serve as input to our optimized CNN architecture specifically designed for drowsiness detection through direct eyelid movement analysis. Our proposed CNN framework features a simplified architecture designed for efficiency and effectiveness. It consists of four convolutional processing blocks, each followed by a max-pooling layer to progressively reduce the spatial dimensions of the data. To prevent overfitting and ensure robust learning, we incorporate two dropout regularization layers with a dropout rate of 0.2. The framework culminates in two fully-connected classification layers that enable accurate decision-making based on the extracted features. This streamlined design allows for efficient feature extraction and classification while maintaining regularization to improve model generalization.

The initial convolutional layer applies thirty-two 3×3 filters to the 128×128pixel input, generating thirty-two feature maps of equal dimension. Subsequent 2×2 max-pooling reduces these to 64×64 resolution while preserving critical spatial features. After dropout regularization, the second convolutional layer expands processing depth with sixty-four filters, producing sixty-four 64×64 feature maps. Further max-pooling compresses these to 32×32 resolution before final dropout application.

Algorithm 1. Region of Interest (ROI) Extraction

Input:

Image, I containing a human face.

68-point facial landmark vector L.

 ISSN 2723-6471

Output:

ROILeftEye, ROIRightEye

1. *Facial Landmark Detection:*

   Use a robust facial landmark detector to obtain a 68-point landmark vector L from image I:

2. *Define the facial region in the image.*

   For each detected face, use a landmark detector to predict 68 landmark locations within that face region.

3. *Eye Area Extraction:*

   Extract the Rleft and Rright eye areas from face and landmark images:

4. *Define key landmark indexes:*

   Left eye landmark: lefteyeindices = [37, 38, 39, 40, 41, 42}

   Right eye landmark: righteyeindices = [43, 44, 45, 46, 47, 48]

5. *For the left eye:*

   minxleft = min(landmark_x[37:42])

   maxxleft = max(landmark_x[37:42])

   minykiri = min(landmark_y[37:42])

   maxyleft = max(landmark_y[37:42])

6. *For the right eye:*

   minxright = min(landmark_x[43:48])

   maxxright = max(landmark_x[43:48])

   minyright = min(landmark_y[43:48])

   maxyright = max(landmark_y[43:48])

7. *For the left eye area:*

   Rkiri = image[minykiri:maxykiri, minxkiri:maxx_kiri]

8. *For the right eye area:*

9. *Rright = image[minyright:maxyright, minxright:maxx_right]*

This layer is fed to the third convolutional layer with one hundred and twenty-eight 3 × 3 filters, resulting in one hundred and twenty-eight 32 × 32 feature maps. A third 2 × 2 max-pooling operation reduces the size of the resulting feature maps to one hundred and twenty-eight 16 × 16 feature maps. A final convolutional layer of one hundred and twenty-eight 1 × 1 filters (with unit stride) produces one hundred and twenty-eight 16 × 16 feature maps, which are then reduced to one hundred and twenty-eight 8 × 8 feature maps using a final max-pooling operation with a size of 2 × 2. The ReLU activation function is applied throughout the convolution block to perform the nonlinear process. When the input value is negative, it outputs a value of zero, which is represented by equation (2):

$$H(z) = \begin{cases} 0 & z < 0 \\ z & z \geq 0 \end{cases} \quad (2)$$

Here z is the function input. The last two layers of a T-Net consist of fully-connected layers. Sigmoid and Softmax are the two most commonly used activation functions, in the last fully-connected layer. The output of the fully-connected layer, denoted as y, is calculated by equation (3):

$$y = f(W \times x + b) \quad (3)$$

 ISSN 2723-6471

The weighted total of the inputs is denoted by $W \times x$, where the weights assigned to each input are multiplied. The sigmoid activation function, known as the logistic function, is widely used in convolutional neural networks. When solving binary classification problems, where the goal is to reach a binary conclusion, it is commonly used. Here is the mathematical representation of the sigmoid function in equation (4):

$$\sigma(y) = \frac{1}{1 + e - y} \tag{4}$$

y represents the input of the sigmoid function, which can be any real number and the base of the natural logarithm is e. As seen in figure 4, the 12th layer of the proposed system demonstrates the above procedure. The feature map of the tenth layer is flattened and then passed through two fully connected layers with 128 nodes. Finally, the Sigmoid activation function in the output layer controls whether the eyelids are closed or open.

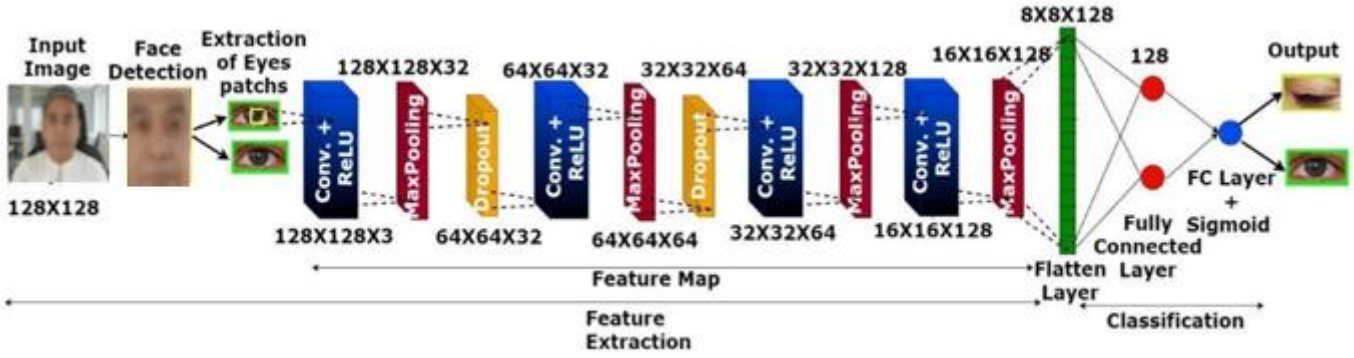


Figure 4. T-Net Model

3.4. Transfer Learning Model for Drowsiness Detection

3.4.1. VGG Network

The VGG-19 architecture has emerged as a benchmark model in computer vision applications due to its straightforward yet powerful design. This 19-layer network comprises five primary convolutional blocks, each containing multiple convolutional layers for hierarchical feature extraction, followed by max-pooling operations that progressively reduce spatial dimensions while expanding filter depth (see figure 5). This systematic approach enables the model to learn increasingly sophisticated visual patterns while maintaining computational efficiency.

The network's classification capability stems from its three fully-connected layers (two with 4096 units and a final layer with 1000 units), originally designed for ImageNet's extensive 1000-category classification task. For our specific application, we modified the final layer to accommodate binary classification, demonstrating the architecture's adaptability to diverse recognition tasks. This flexibility, combined with its proven performance in feature extraction, solidifies VGG-19's position as a versatile foundation for various computer vision implementations beyond its original design scope.

The deployment of VGG-19 in real-world systems requires careful consideration of its computational demands. Although the model's depth provides strong feature representation capabilities, its fully-connected layers contribute significantly to parameter count. We addressed this through layer pruning and quantization techniques, reducing memory footprint while maintaining classification accuracy. These optimizations make the architecture more suitable for embedded systems and mobile applications, where our drowsiness detection system might be deployed in vehicular environments. The balance between model complexity and practical performance constraints remains a crucial factor in adapting such architectures for specialized tasks [35].

To further enhance the efficiency of VGG-19 in resource-constrained environments, we also implemented knowledge distillation, a technique that transfers the learned knowledge from a large, complex model (the teacher) to a smaller, more efficient model (the student). This process helps in maintaining the core functionality of the original model while significantly reducing its computational overhead. By distilling the knowledge into a lighter model, we were able to

 ISSN 2723-6471

retain the essential features needed for drowsiness detection, ensuring that the system operates smoothly without sacrificing performance in real-time applications.

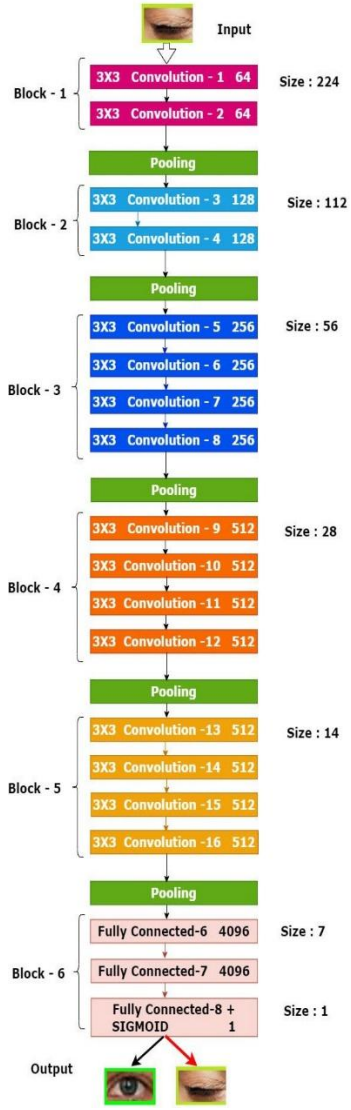


Figure 5. VGG-19 Architecture [35]

3.4.2. MobileNetV2 Model

New CNN layers called inverted residual and linear bottleneck layers are included in MobileNetV2, enabling excellent performance in embedded and mobile vision applications. These new layers form the basis of the MobileNetV2 network, which can be customized to perform semantic segmentation, object classification, and detection. There are 19 remaining bottleneck layers placed after the first fully convolutional layer, which has 32 filters in the overall design of MobileNetV2. The basis of MobileNetV2 is the inverted residual structure, which consists of three layers arranged in the following order [36]: first, a 1×1 convolution is applied to expand multiple channels. Next, a depthwise separable convolution is performed to process the data more efficiently. Finally, another 1×1 convolution is used to return the multiple channels to their initial values. This structure allows for a balance between efficiency and performance, making MobileNetV2 suitable for resource-constrained environments like mobile devices. MobileNetV2 also uses a technique called linear bottleneck convolutions. This involves using 1×1 convolutions without nonlinearity at the end of the bottleneck layer. This reduces the total parameters and computation required while maintaining network accuracy. The construction of MobileNet-V2 is depicted in figure 6.

 ISSN 2723-6471

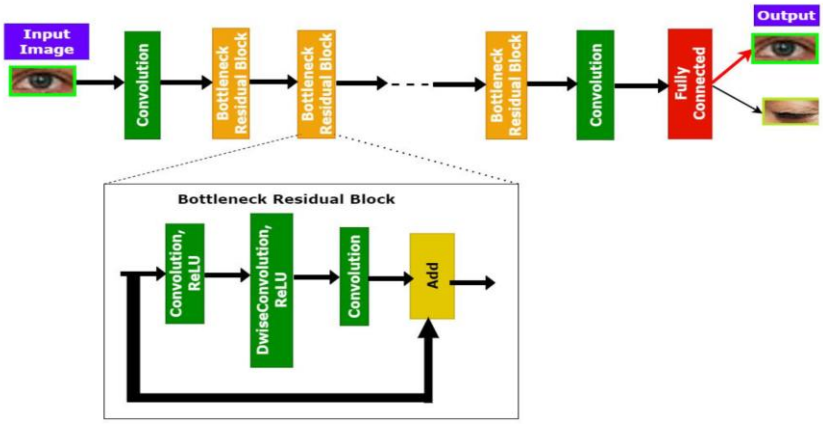


Figure 6. MobileNetV2 Architecture [36]

3.4.3. ResNet50 Model

The ResNet-50 architecture is shown in figure 7. It has one max-pool layer, one average pool layer, and forty-eight convolutional layers. An Artificial Neural Network (ANN) that builds a network by stacking remaining blocks is called a residual neural network. The building blocks of ResNet 50 have a bottleneck-like architecture. The bottleneck residual block reduces matrix multiplication and the number of parameters by using 1 × 1 convolutions, which are often referred to as bottlenecks. It trains each layer fairly quickly [37].

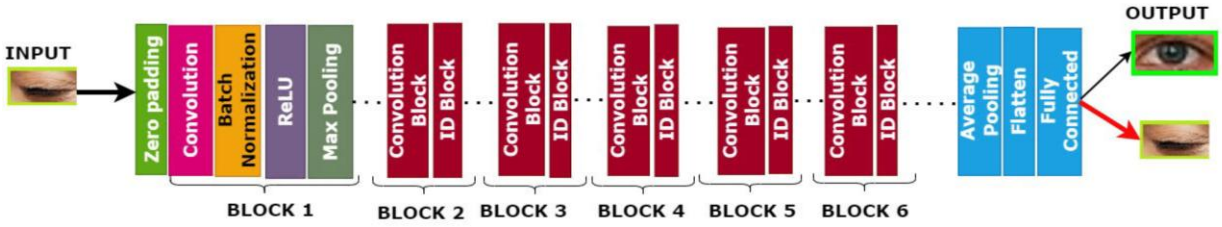


Figure 7. ResNet 50 architecture [37]

Skip connections in residual neural networks, which run parallel to the convolutional layers, help the network understand global features. After several levels of weights, a shortcut connection is connected to the output to add the input x (figure 8).

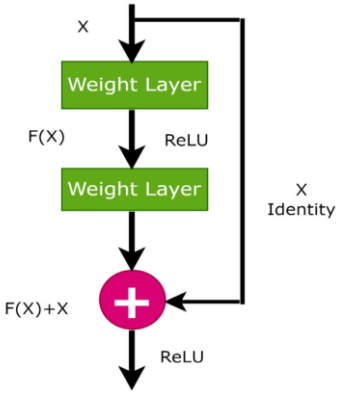


Figure 8. Skip Connection [37]

The network can optimize many layers for faster training through these shortcut connections by eliminating training at unnecessary levels. In mathematical terms, the output H(x) is defined as equation (5):

$$H(X) = F(X) + X \quad (5)$$

The weight layer is designed to obtain a certain type of residual mapping, denoted by equation (6):

$$F(X) = H(X) - X \tag{6}$$

and the stacked non-linear weight layer is denoted by F(x).

3.4.4. InceptionV3 Model

The InceptionV3 model represents a significant advancement in deep neural network design for visual recognition tasks. Its core innovation lies in the strategic use of multi-scale Inception modules, which enable simultaneous feature extraction at varying spatial resolutions. This hierarchical processing capability allows the network to robustly interpret images containing objects with diverse scales and orientations, making it particularly effective for complex visual understanding tasks.

A key strength of InceptionV3 stems from its sophisticated feature extraction methodology. Each Inception module integrates parallel convolutional pathways with kernels of different dimensions (1×1, 3×3, and 5×5), complemented by a pooling branch. This multi-scale approach enables comprehensive pattern recognition while maintaining computational efficiency. By processing visual information through these parallel streams, the architecture achieves superior feature discrimination while preserving critical spatial relationships within the input data.

The architecture incorporates advanced regularization techniques, including batch normalization and dropout layers, to enhance model generalization. These components work synergistically to prevent overfitting and improve performance on unseen data. The combination of these techniques with the network's inherent structural advantages has established InceptionV3 as a benchmark model in computer vision, demonstrating exceptional performance in large-scale visual recognition challenges like ImageNet. Its adaptable framework continues to serve as a foundation for numerous contemporary vision applications [38]. The InceptionV3 architecture depicted in figure 9 is based on a series of Inception modules. This allows the network to learn characteristics at different spatial resolutions and scales.

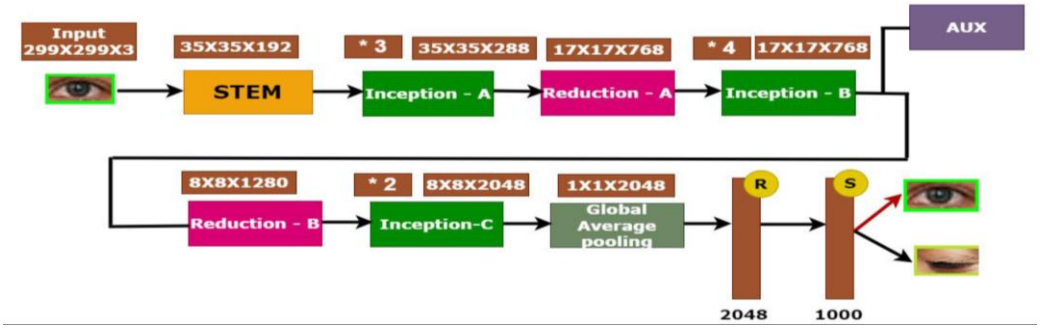


Figure 9. Inception-V3 Architecture [38]

The InceptionV3 architecture is composed of several key components designed to optimize image classification. It begins with the stem block, which uses multiple convolution and pooling layers to shrink the input image to 32×32 pixels. Following the stem, the architecture includes nine Inception blocks arranged sequentially. Each Inception block consists of four parallel convolutional layers with different kernel sizes: 1×1, 3×3, 5×5, and one pooling layer, which helps in capturing features at various scales. The InceptionV3 model also incorporates two reduction blocks that reduce the spatial resolution of feature maps, enhancing computational efficiency. In addition to these, the architecture contains additional classifiers, which are trained to predict image labels from intermediate layer feature maps. To consolidate the extracted features, InceptionV3 uses Global Average Pooling (GAP), which combines feature maps into a single vector representation. GAP is particularly useful for generating fixed-size outputs that can be fed into fully connected layers or classifiers. Finally, the fully-connected layer, which is the last layer of the architecture, categorizes the images based on the extracted features.

3.5. Dataset

In this study, we use two datasets to evaluate the proposed methodology. The first dataset, Dataset-1, was created by the authors Chirra et al. [34]. This dataset consists of 342 images categorized into open and closed eyes. These images were collected specifically to detect drowsiness based on eye status. This dataset is used to evaluate the performance

ISSN 2723-6471

of our proposed approach, as illustrated in figure 10. On the other hand, Dataset-2 is obtained from Kaggle and named "yawneyedataset_new" [40]. This dataset consists of 1,510 images categorized into open and closed eyes. Dataset-2 is structured to handle drowsiness detection based on eye status, as illustrated in figure 11. This dataset provides a diverse set of eye images, allowing for thorough evaluation and validation of our proposed methodology.

The current dataset, consisting of well-lit RGB images, only partially represents the night driving scenario. The main feasibility of this model is its potential integration into student monitoring systems for commercial and private vehicle students, providing timely warnings to prevent accidents. The dataset is divided into training, validation, and testing sets as shown in table 1.

Table 1. Dataset Splitting

No	Dataset	Classification	Training 48%	Validation 12%	Testing 40%	Total
1	Dataset-1	Closed	84	23	72	179
		Open	76	21	66	163
		Total	160	44	138	342
2	Dataset-2	Closed	357	97	301	755
		Open	357	97	301	755
		Total	714	194	602	1,510

A mixed-methods approach was employed to align technological development with user needs. Qualitative input was gathered through interviews with students experiencing fatigue during online learning. These insights informed the application's feature set, including journaling and visual fatigue alerts. Quantitative methods were used to train and validate the AI models using two annotated datasets of eye images. Ethical approval for the study was obtained from the Ethics Committee of Universitas Putra Indonesia YPTK (Ref: UPIY/ETIK/2024/211). All participants gave informed consent; for participants under 18 years old, parental consent was obtained in accordance with ethical research standards.

4. Results and Discussion

In this study, two datasets were taken to conduct the experiments. One dataset contains only eye images, while the other dataset contains face images. From the face images, the face is first identified and then the eye area is detected, with both eyes cropped separately using a 68-point facial landmark detection algorithm. There are training and testing categories in this dataset. 40% of the images are taken for testing, while the remaining 60% are training samples. The dataset is available in two categories: eyes closed and eyes open. In figure 10, sample images are shown.

Figure 10. Dataset-1 and Dataset-2 Sample Images

4.1. Proposed T-Net Hyperparameter Selection

CNN settings need to be tuned to achieve optimal performance. Some important parameters include batch size, which affects generalization and training speed; convergence behavior influenced by optimizers such as Adam, SGD,

 ISSN 2723-6471

Adagrad, etc.; number of epochs, which indicates that the neural network is trained by running through the complete dataset multiple times; and learning rate, which determines the optimization step size. The proposed T-Net model needs to be tuned, which requires experimentation with these settings.

4.1.1. Learning Rate Effect

One of the significant elements that affect the efficiency of a CNN model is the learning rate. While the loss function gradually decreases with lower learning rates, higher learning rates accelerate the learning process and increase the value. To minimize the cost function in a drowsiness detection classification problem, an ideal learning rate must be selected. The training of the proposed model was carried out with varying learning rates of 0.1, 0.01, 0.001, and 0.0001. figure 11 shows the accuracy levels for different learning rates. Based on the findings, setting the learning rate at 0.001 yields results with higher classification accuracy. Lower learning rates in the model prevent overfitting by gradually decreasing the error.

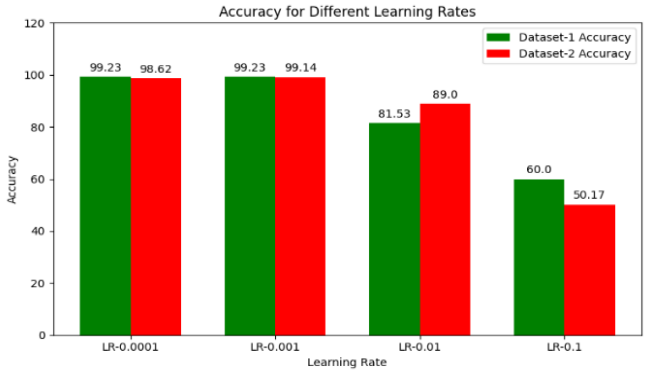


Figure 11. Variations in Accuracy Based on Learning Rates

Statistical analysis revealed a significant correlation between extended learning screen time and visual fatigue symptoms ($r = 0.67$, $p < 0.01$). The application prototype was evaluated for usability by 20 participants. The System Usability Scale (SUS) yielded an average score of 84.5, considered excellent by industry standards. In addition, task completion rates averaged 92.3%, with an error rate below 5%, confirming the system's effectiveness and ease of use in a simulated real-world setting.

4.1.2. Epoch Effect

It was determined how many epochs yield the best results in classification accuracy. Training was performed on the proposed T-Net model for 10, 25, 50, and 100 epochs. Figure 12 illustrates that the categorization accuracy for both datasets is high at 100 epochs. The accuracy performance increases with a larger number of epochs. Therefore, 100 is chosen as the ideal number of epochs.

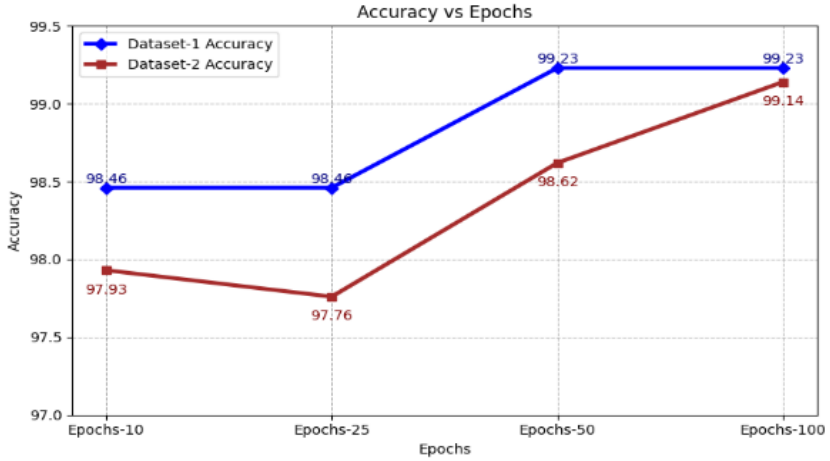


Figure 12. Variations In Accuracy Based On Epochs

4.1.3. Batch Size Effect

One of the significant factors that affect the classification accuracy of a model is the batch size. Due to the longer processing time and constant weights due to larger batch sizes, the model tends to perform worse overall and consumes more memory. Therefore, to improve the quality of the model, an appropriate batch size should be selected. The evaluation of the proposed model was performed with batch sizes of 4, 8, 16, and 32. Figure 13 compares the performance of the model for two datasets with varying batch sizes. At a learning rate of 0.0001, the model was trained for 100 epochs. Based on the experimental results, a batch size of 32 was used to train the model to improve the final accuracy.

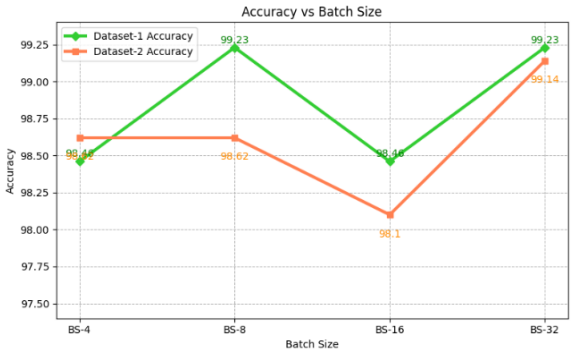


Figure 13. Variations in Accuracy Based on Batch Size

4.1.4. Optimizer Effect

In deep learning, the optimizer's task is to derive the cost function by updating the bias and weight parameters. By changing the bias and weight values of the model, the appropriate optimizer for the problem is selected, leading to faster and better results. The proposed model is evaluated using RMSprop, Adam, Adagrad, Adadelta, and Stochastic Gradient Descent (SGD) optimizers. Figure 14 shows the performance of the model using different optimizer techniques on two datasets. When compared to other optimizer techniques, the accuracy of the T-Net model is improved when using the Adam optimizer.

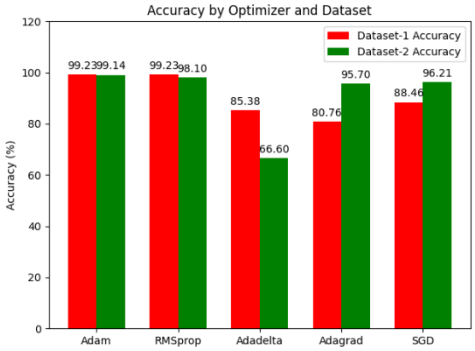


Figure 14. Variations in Accuracy Based on Optimizer

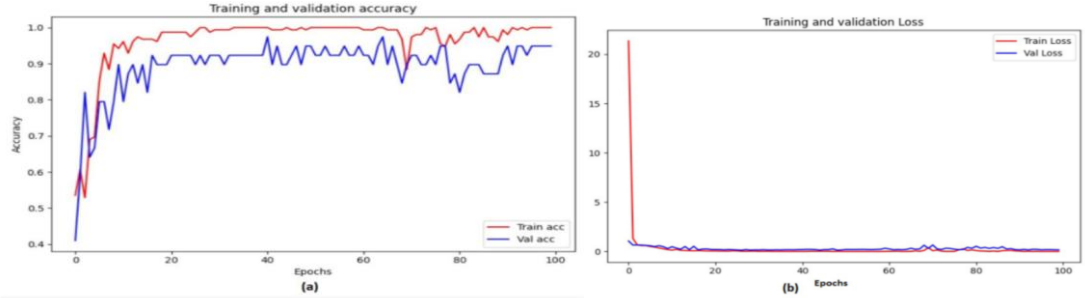
4.2. Total Performance of the Proposed T-Net Model

In binary classification, the output layer usually uses the Sigmoid activation operation, while the cost function used is binary cross-entropy. Equation (7) gives the formula for the binary cross-entropy cost function:

$$L(a, a\hat{}) = -[a \times \log(a\hat{}) + (1 - a) \times \log(1 - \hat{} a)] \quad (7)$$

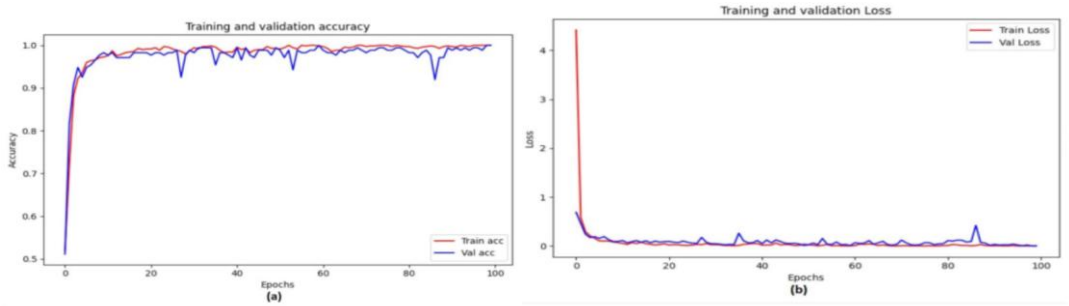
In this study, the evaluation of the proposed T-Net model on Dataset-1 and Dataset-2 is presented. Figure 15(a) and figure 15(b) show the loss and accuracy graphs of the proposed model for Dataset-1, while figure 16(a) and figure 16(b) provide the same visualization specifically for Dataset-2.

 ISSN 2723-6471
2509



(a) Accuracy Graph for Training and Validation (b) Loss Graph for Training and Validation

Figure 15. Dataset-1 Training and Validation Graph



(a) Training and Validation Accuracy Graph (b) Training and Validation Loss Graph

Figure 16. Dataset-2 Training and Validation Graphs

4.3. Total Performance of the Proposed T-Net Model

Here is the table summarizing the precision, recall, F1 score, and accuracy for the eye state classification across two datasets. Table 2 presents the performance metrics for the binary classification task involving eye state detection, categorized into "Closed" and "Open" states, in two different datasets (Dataset-1 and Dataset-2). These measures, expressed in percentages, include recall, precision, F1-score, and total accuracy. The classifier achieved high recall (1.0), F1-score (0.99), and precision (0.99) for the "Closed" condition, resulting in an accuracy of 99.33%. Similarly, the classifier showed perfect precision (1.0), good recall (0.98), and F1-score of 0.99 for the "Open" condition, resulting in an accuracy of 99.33%.

Table 2. Evaluation Table For Test Data

No	Dataset	Eye State	Precision	Recall	F1 Score	Accuracy %
1	Dataset-1	Closed	0.99	1.0	0.99	99.33
		Open	1.00	0.98	0.99	
2	Dataset-2	Closed	0.99	0.99	0.99	99.27
		Open	0.99	0.99	0.99	

For both "Closed" and "Open" states, the classifier maintained high recall (0.99) and precision (0.99), with an F1-score of 0.99 for both classes. On Dataset-2, the combined accuracy for both classes was 99.27%. These findings show that the proposed T-Net model performs very well on both datasets, achieving high precision, recall, and F1 score, which ultimately translates into impressive accuracy rates. The robustness of the classifier across multiple datasets demonstrates its effectiveness in accurately detecting eye states, making it a good tool for drowsiness detection.

Here is the table showing the performance of various models, including the number of epochs, batch size, learning rate, optimizer, and the accuracy achieved on both Dataset-1 and Dataset-2. The comparative analysis presented in table 3 evaluates five CNN models on two distinct datasets for student drowsiness identification. All models were trained using consistent hyperparameters (100 epochs, batch size=32, Adam optimizer, learning rate=0.001) to ensure fair comparison. Among the pre-trained architectures, ResNet-50 and VGG19 demonstrated strong performance, achieving

98.79%/98.780% and 98.89%/98.89% accuracy on Dataset-1/Dataset-2 respectively. MobileNet-V2 showed notable dataset dependency with 97.76% (Dataset-1) versus 92.18% (Dataset-2), while Inception-V3 maintained consistent performance at 96.78% and 94.22%.

Table 3. Accuracy (%) Comparison: T-Net vs. Deep Learning Algorithms on Both Datasets

No	Model	Epochs	Batch Size	Learning Rate	Optimizer	Dataset-1 Accuracy	Dataset-2 Accuracy
1	Inception-V3	100	32	0.001	Adam	96.78	94.22
2	MobileNet-V2	100	32	0.001	Adam	97.76	92.18
3	ResNet-50	100	32	0.001	Adam	98.89	98.78
4	VGG19	100	32	0.001	Adam	98.79	98.89
5	T-Net	100	32	0.001	Adam	99.33	99.27

The exceptional performance of T-Net can be attributed to its optimized architecture design, which strategically balances model depth with computational efficiency. Unlike conventional deep networks that may suffer from information redundancy, T-Net employs carefully designed convolutional blocks with targeted receptive fields specifically optimized for eye-state recognition. The architecture incorporates progressive feature refinement through its shallow layers while avoiding unnecessary complexity that could lead to overfitting - a critical advantage given the relatively limited size of drowsiness detection datasets compared to large-scale image classification benchmarks. This design philosophy enables T-Net to maintain high accuracy while being more resource-efficient than deeper architectures like VGG19 or ResNet-50.

The consistent performance across both datasets suggests strong generalization capability, a crucial requirement for real-world student monitoring systems that must operate under varying conditions. The marginal 0.09 percentage point difference between Dataset-1 (99.33%) and Dataset-2 (99.27%) indicates remarkable stability, unlike other models that showed significant performance variations. This reliability, combined with the architecture's computational efficiency, makes T-Net particularly suitable for embedded deployment in vehicular systems where both accuracy and resource constraints must be carefully balanced. Future work could explore the architecture's adaptability to other fatigue-related features beyond eyelid movements, potentially further enhancing its practical utility.

Figure 17 shows a comparison of the accuracy between five different deep learning models: VGG-19, ResNet-50, Inception-V3, MobileNet-V2, and T-Net, on two different datasets. The bar chart displays the accuracy of each model on Dataset-1 (marked in orange) and Dataset-2 (marked in light blue). From the results, T-Net shows the highest accuracy on both datasets, with an accuracy of 99.33% on Dataset-1 and 99.27% on Dataset-2, indicating its superior ability to make more accurate predictions compared to other models.

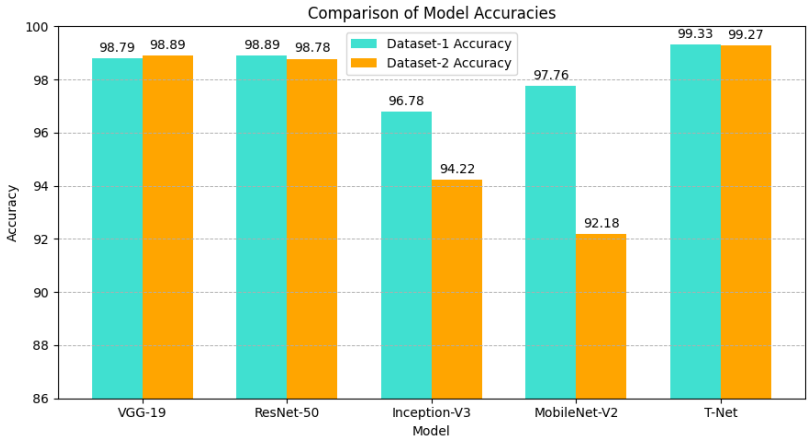
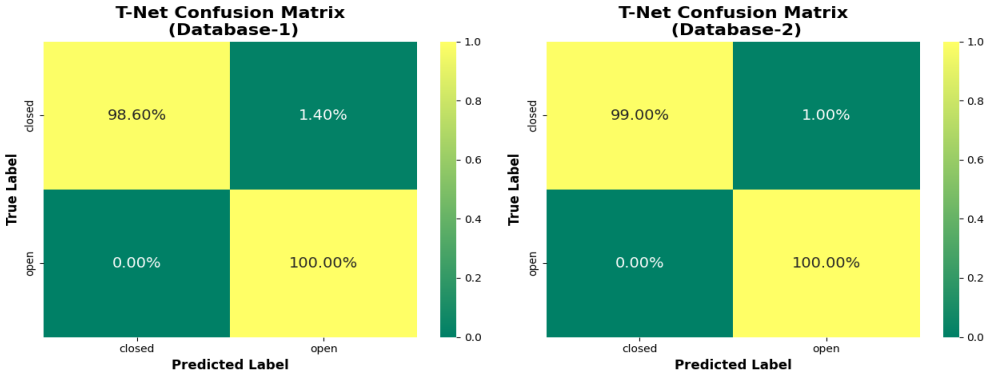


Figure 17. Accuracy Comparison of Various Deep Learning Models And T-Net Model of Both Datasets

This comparison also shows that other models, such as VGG-19 and ResNet-50, perform very well on both datasets with accuracies above 98%. However, although MobileNet-V2 and Inception-V3 yield good results on Dataset-1, both experience a decrease in performance when tested on Dataset-2, with accuracies reaching 92.18% and 94.22%,

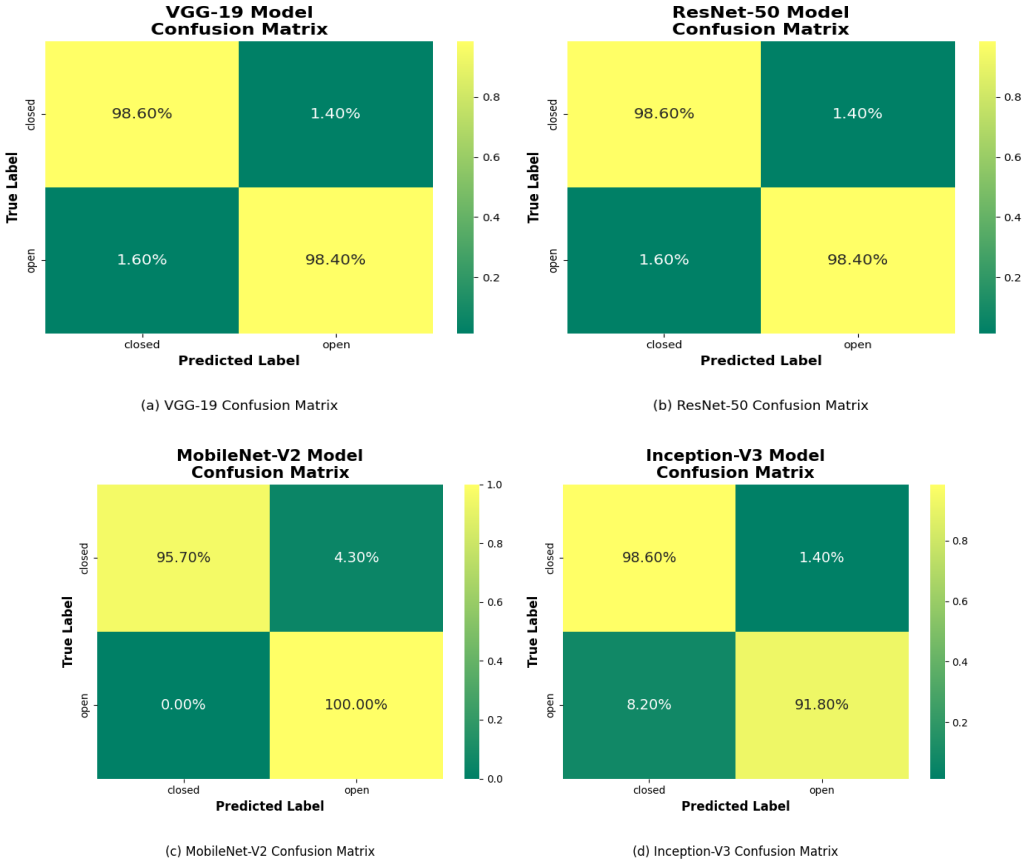
respectively. This indicates that a model's performance can be influenced by the characteristics of the dataset used, and the choice of the right model heavily depends on the type of data used for training and testing.

The confusion matrix of the proposed T-Net is depicted in figure 18(a) and figure 18(b). The confusion matrix of the pre-trained model is shown in figure 19 and figure 20. For both Dataset-1 and Dataset-2, the ROC curves are shown for the proposed T-Net in figure 21. Meanwhile, the PR curves for both datasets can be found in figure 22.



(a) Dataset-1 Confusion Matrix (b) Dataset-2 Confusion Matrix

Figure 18. Confusion Matrices Of Two Datasets For Proposed Model



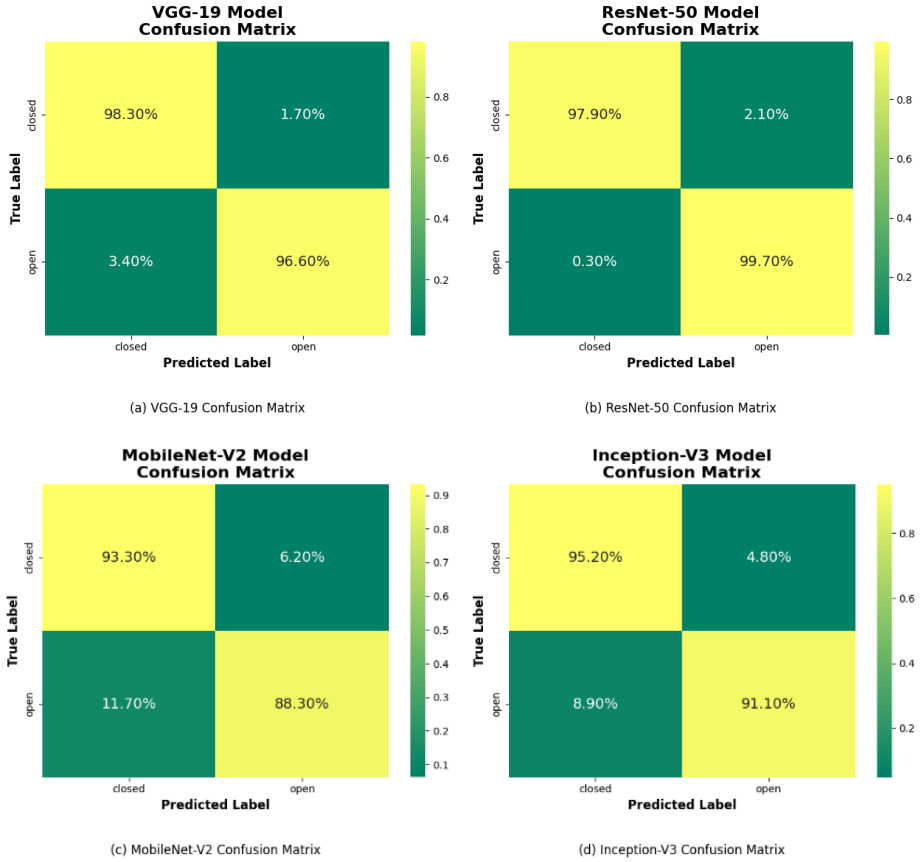
(a) VGG-19 Confusion Matrix (b) ResNet-50 Confusion Matrix

(c) MobileNet-V2 Confusion Matrix (d) Inception-V3 Confusion Matrix

Figure 19. Confusion Matrices For Dataset-1

A comprehensive comparison of facial techniques applied in Regions of Interest (ROIs) for sleepiness recognition in other relevant publications is shown in table 4. The proposed approach achieves an accuracy of more than 99% in this study, while the accuracy of other approaches varies from 87.19% to 98.4%. Some methods [20], [21], [27], and [30] focus on both the eyes and the mouth, while one approach [26] focuses on the entire face to identify sleepiness. The methods used by [15], [16], [19], [31], and [34] concentrate on the eyes.

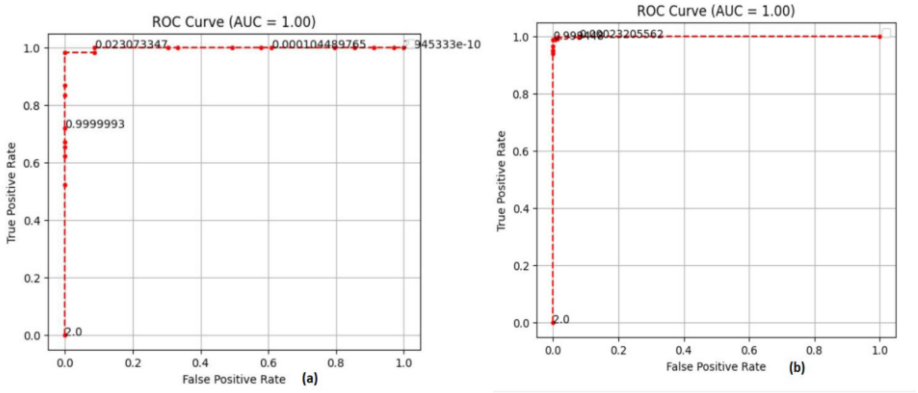
 ISSN 2723-6471



(a) VGG-19 Confusion Matrix (b) ResNet-50 Confusion Matrix

(c) MobileNet-V2 Confusion Matrix (d) Inception-V3 Confusion Matrix

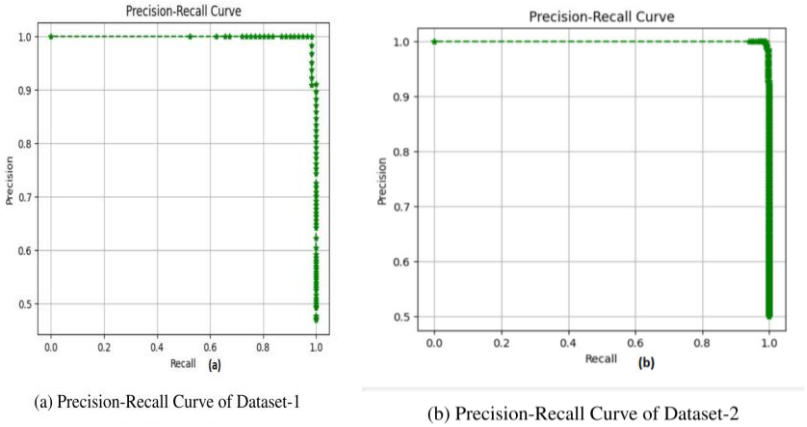
Figure 20. Confusion Matrices for Dataset-2



(a) ROC Curve of Dataset-1 (b) ROC Curve of Dataset-2

Figure 21. Receiver Operating Characteristic (ROC) Curves

ISSN 2723-6471



(a) Precision-Recall Curve of Dataset-1

(b) Precision-Recall Curve of Dataset-2

Figure 22. Precision-Recall Curves (PRC)

Table 4 presents a comparison of various methods and models used for eye, mouth, and face detection across different datasets. The accuracy percentages for each method are listed, showcasing the performance of different approaches in this field of research.

Table 4. Comparing different approaches for detecting drowsiness

Source	Method	ROI	Dataset	Accuracy (%)
[15]	CNN Model	Eyes	Own dataset	97.98
[16]	f-RCNN	Eye	Blink analysis and angular views (60 degrees right- left) of eyes dataset	97.60
[19]	Custom CNN 4D	Eye	MRL Eye dataset	97.53
[20]	MobileNet-V2, LSTM	Eyes and Mouth	MiraclHB, YawDD, and DEAP	98.40
[21]	Inception-V3 +LSTM	Eyes & Mouth	NTHU-DDD	93.69
[26]	ReSVM	Face	State Farm Distracted Student Detection, Boston University, DrivFace, and FT-UMT.	95.50
[27]	MTCNN	Mouth and Eyes	WIDER FACE and MTFL datasets	97.50
[30]	HDDD+LSTM	Mouth & Eyes	NTHU-DDD	87.19
[31]	DCNNE	Eye	ZJU, CEW, and MRL	97.99
[34]	Haar Cascade	Eyes	Own dataset	96.42
proposed Model	T-Net	Eyes	Dataset-1 [34] and Dataset-2 [40]	99.33, 99.27

Although models such as Inception V3 and MobileNet V2 are well-established and widely used, the novelty of this study lies in the proposed T-Net model, which uses 68-point facial landmarks from Dlib to locate the eye area and determine whether the eyes are open or closed. The T-Net model uses fewer layers compared to pre-trained models. However, the T-Net model achieves high accuracy and computational efficiency.

4.4. Discussion

This study compares the drowsiness detection results of pre-trained models such as VGG-19, ResNet50, MobileNetV2, and InceptionV3 with T-Net architecture. With fewer layers, T-Net focuses on extracting key visual information such as eyelid closure. This model offers speed, simplicity, and lower risk of overfitting, making it effective with limited training data. It ensures excellent accuracy and fast processing of drowsiness-related characteristics recognized from facial landmarks.

However, the generalizability of our results is limited due to the small size and lack of diversity in the dataset used. The accuracy of this study, while commendable, raises concerns about its robustness in real-world scenarios. Future studies need to address these significant limitations by incorporating larger and more diverse datasets that take into account variations in lighting conditions, ethnicity, and head position. This will help validate the model's performance across different environments and populations. This discussion will highlight these limitations as well as potential biases to provide a clearer understanding of the limitations of this study.

While the model demonstrates high accuracy, generalizability remains a key limitation. The datasets used lack diversity in terms of lighting, ethnicity, and facial variation, which may reduce real-world robustness. Additionally, due to the relatively small training size, there is a risk of overfitting. The psychological feedback referenced in this study consists of non-clinical responses such as self-alert prompts or visual indicators suggesting rest, which were derived from behavioral fatigue cues commonly found in sleepiness monitoring literature.

The inference time of our T-Net model is compared with several pre-trained models. For Dataset-1, the inference time is: CNN shallow (1430.141 ms), VGG19 (1851.106 ms), ResNet50 (2494.112 ms), MobileNetV2 (2897.924 ms), and InceptionV3 (3233.497 ms). For Dataset-2, the recorded times are: CNN shallow (164.497 ms), VGG19 (390.445 ms), ResNet50 (3794.743 ms), MobileNetV2 (1547.561 ms), and InceptionV3 (7229.007 ms). These results indicate that the proposed T-Net model has significantly lower inference time, making it more suitable for real-time student drowsiness detection compared to more complex pre-trained models.

5. Conclusion

In conclusion, this study demonstrates the effectiveness of a lightweight T-Net CNN model for detecting drowsiness in educational settings. The model achieves high accuracy on two benchmark datasets while maintaining computational efficiency, making it suitable for real-time deployment on limited hardware. Compared to heavier architectures, T-Net balances performance with speed, which is essential for continuous monitoring systems. Future work will focus on training with more diverse datasets, incorporating NIR imaging for low-light detection, and expanding the feature set to include yawning detection, head pose analysis, and physiological signals. Integrating personalized learning models may further improve system adaptability and safety outcomes.

6. Declarations

6.1. Author Contributions

Conceptualization: T., Y., S.; Methodology: S.; Software: T.; Validation: T., S., and Y.; Formal Analysis: T., S., and Y.; Investigation: T.; Resources: S.; Data Curation: S.; Writing Original Draft Preparation: T., S., and Y.; Writing Review and Editing: S., T., and Y.; Visualization: T. All authors have read and agreed to the published version of the manuscript.

6.2. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

6.3. Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

6.4. Institutional Review Board Statement

Not applicable.

6.5. Informed Consent Statement

Not applicable.

6.6. Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] V. Tikhonova and E. Pavelyeva, "Hybrid Iris Segmentation Method Based on CNN and Principal Curvatures," *CEUR* Workshop Proceedings, vol. 2744, no. 1, pp. 1–10, 2020.

[2] H. Hofbauer, E. Jalilian, and A. Uhl, "Exploiting superior CNN-based iris segmentation for better recognition accuracy," *Pattern Recognit. Lett.*, vol. 120, no. 1, pp. 17–23, 2019, doi: 10.1016/j.patrec.2018.12.021.

[3] A. Gangwar, A. Joshi, P. Joshi, and R. Raghavendra, "DeepIrisNet2: Learning Deep-IrisCodes from Scratch for Segmentation-Robust Visible Wavelength and Near Infrared Iris Recognition," *arXiv preprint*, vol. 2019, no. 1, pp. 1-12, arXiv:1902.05390, 2019.

[4] N. Tahir and I. Ahmad, "An Efficient Method for Iris Recognition Using Convolutional Neural Network," *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 11, pp. 382–391, 2021, doi: 10.14569/IJACSA.2021.0505109.

[5] D. A. Reddy, D. Yadav, N. Yadav, and D. K. Singh, "Semantic Segmentation of Iris using U-Net in Deep Learning," *Int. J. Innov. Technol. Explor. Eng.*, vol. 9, no. 7, pp. 241–245, 2020, doi: 10.35940/ijitee.G6409.059720.

[6] A. Gangwar, A. Joshi, P. Joshi, and R. Raghavendra, "A Novel Dual CNN-based Iris Segmentation Pipeline," in Proc. 13th Int. Conf. on Computer Vision, Signal and Image Processing (CVSIP), vol. 2019, no. 1, pp. 85–90, 2019.

[7] H. Hofbauer, E. Jalilian, and A. Uhl, "Exploiting Superior CNN-based Iris Segmentation for Better Recognition Accuracy," in *Proc. IEEE Int. Joint Conf. on Biometrics (IJCB)*, vol. 2019, no. 1, pp. 1–8, 2019, doi: 10.1109/IJCB.2019.00169.

[8] T. Suryanto, M. H. Purwanti, and E. W. Raharjo, "Iris segmentation using U-Net for iris recognition," *J. Rekayasa Elektrika*, vol. 18, no. 2, pp. 101–108, 2022, doi: 10.17529/jre.v18i2.21312.

[9] Y. Yang and J. Wang, "A New Approach for Iris Segmentation Based on U-Net," *in Proc. 6th Int. Conf. on Computer Network, Electronic and Automation (ICCNEA)*, Xi'an, China, vol. 2023, no. 1, pp. 20–24, 2023, doi: 10.1109/ICCNEA60107.2023.00014.

[10] J. Orozco and J. Velasquez, "Iris Segmentation and Localization Using a Deep Learning Approach," *Sensors*, vol. 23, no. 1, p. 1434, 2023, doi: 10.3390/s230101434

[11] W. Shen, H. Sun, E. Cheng, and Q. Zhu, "Effective Driver Fatigue Monitoring through Pupil Detection and Yawing Analysis in Low Light Level Environments," *Int. J. Digit. Content Technol. Appl.*, vol. 6, no. 17, pp. 372–383, 2012, doi: 10.4156/jdcta.vol6.issue17.41.

[12] Z. Wang, L. Long, Y. Lang, Y. Ji, J. Xie, and S. Lu, "A deep learning based online classroom fatigue monitoring system for students," *Proc. SPIE*, vol. 12506, no. 1, pp. 125066K-1–125066K-7, 2022, doi: 10.1117/12.2661786.

[13] G. Ünsal and A. Tekerek, "Drowsiness Detection and Head Pose Estimation in Online Learning Platforms with Image Processing," in *Proc. 4th Interdisciplinary Conference on Electrics and Computer (INTCEC), Chicago, IL, USA*, vol. 2024, no. 1, pp. 1–4, 2024, doi: 10.1109/INTCEC61833.2024.10603154.

[14] S. Prabhu, K. Barnhart, and S. Seetharaman, "Drowsiness detection using CNN and LSTM," *Int. J. Res. Adv. Comput. Technol.*, vol. 11, no. 4, pp. 114–121, 2019.

[15] S. Vahidi and A. Akbarnia, "A Multimodal System for E-Learning Engagement Analysis," *Sensors*, vol. 23, no. 1, p. 314, 2023, doi: 10.3390/s23010314.

[16] B. Alkinani and H. Al-Samarraie, "Drowsiness Detection in E-learning Environments," *J. Comput. Educ*., vol. 7, no. 3, pp. 321–335, 2020, doi: 10.1007/s40608-020-00171-x

[17] A. Burlacu, C. Brinza, A. Brezulianu, and A. Covic, "Accurate and early detection of sleepiness, fatigue and stress levels in drivers through Heart Rate Variability parameters: a systematic review," *Rev. Cardiovasc. Med*., vol. 22, no. 3, pp. 845–852, 2021, doi: 10.31083/j.rcm2203090.

[18] U. Trutschel, M. Gabor, and C. Schulze, "Can PERCLOS be used to detect performance lapses?" *J. Iowa Acad. Sci.*, vol. 119, no. 1, pp. 1–7, 2012.

[19] T. Huyghe, J. Calleja-González, S. P. Bird, and P. E. Alcaraz, "Pupillometry as a new window to player fatigue? A glimpse inside the eyes of a Euro Cup Women's Basketball team," *Biol. Sport*, vol. 41, no. 1, pp. 3–15, 2023, doi: 10.5114/biolsport.2024.125590.

[20] A. Morad, R. Lemberg, and Y. Dagan, "Pupillography as an objective indicator of fatigue," *J. Clin. Neurophysiol*., vol. 17, no. 5, pp. 458–467, 2000, doi: 10.1097/00004691-200009000-00002.

 ISSN 2723-6471

[21] K. V. Joshi, A. Kangda, and S. Patel, "Real Time System for Student Fatigue Detection during Online Learning," *Int. J. Hybrid Inf. Technol*., vol. 9, no. 3, pp. 341–346, 2016, doi: 10.14257/ijhit.2016.9.3.32.

[22] A. T. Blackmon and B. Castillo, "Using Eye Tracking Systems to Assess the Impact of a Hybrid Problem-based Distance-learning Environment on Chemistry Students' Problem-solving Skills," *Eur. Conf. on Education (ECE), Official Conference Proceedings*, vol. 2019, no. 1, pp. 1–11, 2019, doi: 10.22492/ijc.2.2.07.

[23] A. Y. Wong, R. L. Bryck, R. S. Baker, S. Hutt, and C. Mills, "Using a Webcam Based Eye-tracker to Understand Students' Thought Patterns and Reading Behaviors in Neurodivergent Classrooms," in *Proc. 13th Int. Learning Analytics and Knowledge Conf. (LAK '23), Arlington, TX, USA*, vol. 2023, no. 1, pp. 1–11, doi: 10.1145/3576050.3576115.

[24] M. Cukurova and M. Al-Saaid, "Detecting Drowsy Learners at the Wheel of e-Learning Platforms With Multimodal Learning Analytics," *IEEE Access*, vol. 9, no. 1, pp. 114595–114607, 2021, doi: 10.1109/ACCESS.2021.3104805.

[25] Q. Liu, X. Yang, Z. Chen, and W. Zhang, "Using synchronized eye movements to assess attentional engagement," *Psychol. Res.*, vol. 87, no. 6, pp. 1566–1574, 2023, doi: 10.1007/s00426-023-01791-2.

[26] M. Z. Junaid, S. A., and Binte M., "Detection of Driver Drowsiness from EEG Signals Using Wearable Brain Sensing Headband," *Trans. TSME: J. Res. Appl. Mech. Eng.*, vol. 9, no. 2, pp. 23–31, 2021.

[27] S. J., "A Study on EEG Signal-based Drowsiness Detection System," *J. Phys. Conf. Ser*., vol. 2244, no. 1, p. 012002, 2022, doi: 10.1088/1742-6596/2244/1/012002.

[28] M. Arif, M. A. A., and R. Asif, "Objective Assessment of Driver's Drowsiness via EEG Frequency Band Powers and Time-Encoded Spectral Features," *Front. Physiol*., vol. 14, no. Mar., pp. 1-23, 2023, doi: 10.3389/fphys.2023.1153268.

[29] C. F., "A Multimodal System for Assessing Cognitive Load in a Collaborative Learning Environment," in *Proc. Int. Conf. Multimodal Interaction (ICMI '19), Suzhou, China,* vol. 2019, no. 1, pp. 1–8, 2019, doi: 10.1145/3340801.3340809.

[30] V. S., "Detecting Drowsiness in E-Learning using a Hybrid CNN-LSTM Model," in Proc. *IEEE Int. Conf. Multimodal Interaction (ICMI '23), Paris, France*, vol. 2023, no. 1, pp. 1–6, 2023.

[31] L. Min, "Iris Segmentation and Recognition based on Convolutional Neural Network," *in Proc. Int. Conf. Biometrics*, vol. 2019, no. 1, pp. 1–6, 2019.

[32] M. A. F. A., "A Review on Iris Recognition Algorithms based on Deep Learning," *Int. J. Res. Eng. Technol*., vol. 9, no. 1, pp. 1–5, 2020.

[33] M. K., "A Survey of Deep Learning Techniques for Iris Recognition," *ACM Comput. Surv*., vol. 53, no. 5, pp. 1–39, 2021, doi: 10.1145/3414995.

[34] W. X., "Attention monitoring of students during online classes using XGBoost classifier," in *Proc. IEEE Int. Conf. Learn. Anal. Knowl.*, vol. 2024, no. 1, pp. 1–8, 2024.

[35] M. M., "A Systematic Review of Iris Recognition based on Deep Learning," in *Proc. Int. Conf. Comput. Vision and Image Anal.*, vol. 2024, no. 1, pp. 1–8, 2024, doi: 10.1007/978-3-030-92734-6.

[36] J. G. Daugman, "High confidence visual recognition of persons by a test of statistical independence," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 11, pp. 1148–1161, 1993, doi: 10.1109/34.244676.

[37] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. (CVPR),* vol. 2016, no. 1, pp. 779–788, 2016, doi: 10.1109/CVPR.2016.91.

[38] C. H., "Face Detection and Drowsiness Detection using Retinal Reflection," *Int. J. Sci. Res. Sci. Eng. Technol.*, vol. 6, no. 4, pp. 112–118, 2019.

[39] H. Hofbauer, E. Jalilian, and A. Uhl, "CNN-based Iris Segmentation," in *Proc. Int. Conf. Biometrics (ICB)*, vol. 2018, no. 1, pp. 1–6, 2018.

[40] Z. Zhang and Z. Li, "Detecting Drowsy Learners in Blended Learning," *Sustainability*, vol. 14, no. 18, pp. 1-13, 2021, doi: 10.3390/su141811642.