Inform : Jurnal Ilmiah Bidang Teknologi Informasi dan Komunikasi
Vol.10 No.1 January 2025, P-ISSN : 2502-3470, E-ISSN : 2581-0367

# *Analysis of Driver Drowsiness Detection System Based on Landmarks and MediaPipe*

Fawaidul Badri[1], Sulistya Umie Ruhmana Sari[2], Shipun Anuar Bin Hamzah[3]

[1]*Informatics Department, Universitas Islam Malang, Indonesia*

[2]*Department of Tadris Mathematics, Universitas Islam Negeri Maulana Malik Ibrahim Malang, Indonesia*

[3]*Electrical and Electronic Engineering Department, Universiti Tun Hussein Onn Malaysia, Malaysia.*

[1]fawaidulbadri@unisma.ac.id(*)

[2]sulistyaumieruhmanasari@uin-malang.ac.id, [3]shipun@uthm.edu.my

Received: 2024-12-05; Accepted: 2024-12-26; Published: 2025-01-06

*Abstract*— Driver drowsiness is one of the leading causes of traffic accidents, especially during long-distance journeys. This study developed a detection system based on landmarks and the MediaPipe framework to analyze drowsiness through eye blink duration. The system employs coordinate point initialization using regression trees to accurately detect objects, such as eyes. The research data consists of 30 videos, each lasting 30 seconds, collected from four Trans Java bus drivers. The videos were extracted to identify facial detection histograms and analyzed based on eye blink duration. The testing results showed a detection accuracy of 81% with an error rate of 19% for distances of 10 to 100 cm, while testing with 30 videos achieved an average accuracy of 88.745% and a Mean Squared Error (MSE) of 7.615%. The test results show that CNN outperforms MediaPipe in detecting drowsiness, with a higher average accuracy of 76.79% compared to 73.83% and a lower MSE value of 47.33 compared to 48.27. CNN is also more consistent in handling extreme lighting variations, while MediaPipe excels in processing efficiency, making it suitable for devices with limited resources. This study demonstrates that the landmarks and MediaPipe-based system effectively and innovatively detects drowsiness, offering a solution to improve driver safety during trips.

*Keywords*— Drowsiness Detection; Landmarks; MediaPipe; Driving Safety; Eye Blink Analysis.

## I. INTRODUCTION

Driver fatigue and drowsiness are the primary factors contributing to traffic accidents, especially during long-distance journeys. Factors such as extended travel distances, prolonged driving durations, driver age, and irregular sleep patterns often affect driver concentration, increasing the risk of accidents. According to various studies, undetected drowsiness significantly contributes to fatal road accidents. Therefore, a reliable drowsiness detection system that provides early warnings is essential to help drivers remain focused during trips.

Currently, various drowsiness detection technologies have been developed, including the use of additional hardware such as physical sensors or eye trackers. However, these approaches are often costly, require specialized devices, or are challenging to implement widely in commercial vehicles [1].

This research introduces a drowsiness detection system based on landmarks technology integrated with the MediaPipe framework. The system utilizes machine learning algorithms to analyze drivers' eye blinking duration to indicate drowsiness. By initializing facial coordinate points, the system can recognize things such as eyes with high precision using a regression tree technique. The research data was collected from 30 videos of Trans Java bus drivers, each 30 seconds long, and was analyzed to identify facial histograms and blinking patterns. Testing demonstrated a detection accuracy of 81% across various distances (10–100 cm) and an average accuracy of 88.745%, with a Mean Squared Error (MSE) of 7.615%.

The novelty of this research lies in its combination of landmarks-based methods with the MediaPipe framework, enabling real-time data processing without requiring additional hardware. Furthermore, the study uses real-world video data of drivers, making the results more applicable compared to previous approaches. With high accuracy and efficient data processing, the developed system offers an innovative solution to enhance driving safety, particularly during long-distance journeys. This research also significantly contributes to developing more affordable, practical, and easily implemented table drowsiness detection technologies for various vehicles.

## II. RESEARCH METHODOLOGY

This study aims to design and analyze a driver drowsiness detection system using a facial landmark-based approach and MediaPipe technology. The block diagram can be seen in the Fig.1.

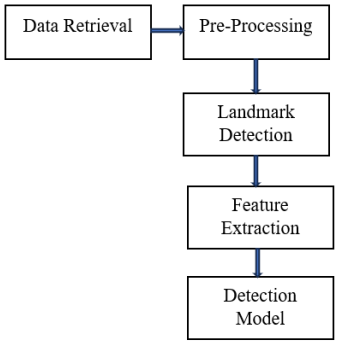


Fig.1. System Flow



DOI : https://doi.org/10.25139/inform.v10i1.9325

*1) Data retrieval:* In the data collection stage, video recording of the driver's face was carried out in various conditions, both when awake and when showing signs of drowsiness. This data was collected using a high-resolution camera installed in the vehicle to ensure optimal image quality. The data collection environment included scenarios of moving and stationary vehicles, with varying lighting to reflect real-world conditions on the road, such as day and night. The study subjects consisted of drivers with various characteristics, including differences in age, gender, and level of driving experience, to ensure sufficient data diversity. Each recording session lasted for a certain duration and included normal driving activities and conditions intentionally created to induce drowsiness, such as starting the session after a short sleep period. The collected data was then labeled based on the driver's condition (awake or drowsy) as a basis for the analysis process and training of the detection model.

*2) Data pre-processing:* In the data pre-processing stage, the recorded video is broken down into individual frames to facilitate visual analysis. Each frame is then normalized to ensure consistent quality, including adjusting resolution, lighting, and contrast to be usable in various environmental conditions. Noise or disturbances that appear in the video, such as shadows or reflections, are removed to improve the accuracy of the analysis. After that, each frame is labeled according to the driver's condition (awake or drowsy), which will later be used as input data in model training. This process also includes removing blurry or irrelevant frames, such as frames with undetected faces. These steps aim to produce clean data ready for further analysis using facial landmark-based detection algorithms.

*3) Landmark Detection:* The landmark detection stage is the core of the analysis process to identify the driver's facial features in real time. Currently, MediaPipe technology is used to detect and track 468 landmark points on the face accurately. This system focuses on relevant facial areas, such as the eyes, mouth, nose, and head position, which are key indicators in detecting drowsiness. MediaPipe utilizes machine learning algorithms to recognize facial structures, even in varying lighting conditions or when the face is partially covered. The resulting landmark data includes the geometric coordinates of each point, which are then analyzed to detect changes in expression or certain movements, such as eyes that are closed for a longer period or a yawning mouth. This real-time detection process is carried out to enable a fast response in the driver drowsiness detection system.

*4) Feature Extraction:* In the feature extraction stage, important information is extracted from the acquired facial landmark data to identify signs of driver drowsiness. The main features analyzed include the Eye Aspect Ratio (EAR), which measures the degree of eye-opening and helps detect whether the eyes are closed longer than usual.

*5) Detection Model:* The detection model stage aims to build a system that can recognize the driver's drowsiness based on the features that have been extracted. This detection model is designed using a machine learning algorithm. The extracted landmark data trains the model to recognize drowsiness patterns, such as frequently closed eyes or yawning movements. The training process is carried out by dividing the data into a training set and a test set to ensure the model's generalization to new data. This model is evaluated using metrics such as accuracy, precision, sensitivity, and specificity to measure its ability to detect drowsiness correctly. The results of this model are integrated into a real-time system to warn drivers when signs of drowsiness are detected, thereby improving driving safety for a person.

A detection system is a technology-based approach to identifying specific conditions or events using various sensing methods, data analysis, and decision-making processes. In the context of driver drowsiness detection, the system aims to monitor the driver's condition in real time and provide early warnings when signs of fatigue or drowsiness are detected. These systems are generally categorized into three main approaches: physiological-based, behavior-based, and vehicle-based [2][3].

### *A. Physiological-Based Detection System*

This approach uses sensors to monitor biological parameters, such as heart rate, brain activity (electroencephalography or EEG), and skin conductance levels. This system is accurate because the data reflects the driver's body condition. However, this method requires expensive and invasive additional devices, making it less practical for widespread implementation in commercial vehicles [4].

### *B. Behavior-Based Detection System*

This approach uses visual data to identify changes in driver behavior, such as eye blink patterns, gaze direction, head movements, or changes in facial expressions. Camera-based technology is often used to capture facial images of the driver. Parameters such as eye blink duration or blink frequency are effective indicators in detecting drowsiness. This system is non-invasive and easier to implement, but it still has challenges, such as adequate lighting and the ability to work in dynamic environmental conditions [5].

### *C. Vehicle-Based Detection System*

This method uses data from the vehicle, such as steering patterns, vehicle speed, or responses to road conditions. This system is relatively simple and can be integrated directly with vehicle devices. However, this method only provides indirect detection that is less sensitive to changes in driver conditions [6].

### *D. Landmarks*

The landmarks method is a widely used for detecting and analyzing facial features, particularly in applications such as facial recognition, facial expression analysis, and drowsiness detection. Landmarks refer to coordinate points on the face that geometrically represent the primary facial structures, such as the eyes, nose, lips, and facial contours. These points assist


DOI : https://doi.org/10.25139/inform.v10i1.9325

algorithms in accurately analyzing changes in the position or shape of facial features [7].

In drowsiness detection, landmarks can be used to monitor eye movements or conditions, such as blinking or eye openness, which are key indicators of drowsiness. Facial landmarks are coordinate points in a two-dimensional (2D) space. For example, in 2D coordinates, the position of the landmark on the face is expressed as Equation (1), where the $x_i$ variable is the horizontal coordinate in the image, and the $y_i$ variable is the vertical coordinate in the image using Equation (1).

$$L_i = (x_i, y_i) \quad (1)$$

*E. Eye Detection Using Landmarks*

In drowsiness detection applications, the landmarks used for eye analysis typically include the upper eyelid, lower eyelid, and the corners of the eyes. Based on the position of these landmarks, the Eye Aspect Ratio (EAR) is a method that calculates the ratio between the vertical distance of the upper and lower eyelids and the horizontal distance of the eye. This is done using geometric points in the eye area derived from facial landmarks [8][9].

The method utilizes six key points on the eye structure, namely P1, P2, P3, P4, P5, and P6, to measure the horizontal and vertical dimensions of the eye. With this approach, the EAR enables accurate analysis of eye conditions, whether open or closed. The calculation of the EAR is formulated based on the relationship between these distances using Equation (2).

$$EAR = \frac{\text{II}p_2 - P_6\text{II} + \text{II}p_3 - p_5\text{II}}{2\text{II}p_1 - p_4 II} \quad (2)$$

Illustration of the eye object image wit $p_1, p_2, p_3, p_4, p_5, p_6$.

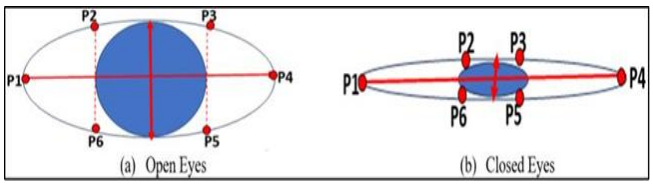


Fig.2. Eye Aspect Ratio Eyes Open and Eyes Closed

Based on the formula shown in Fig.2, the basic formula of EAR is shown, which is used to calculate the ratio between the vertical and horizontal dimensions of the eye, which is the main indicator in detecting drowsiness. In this calculation, points P1, P2, P3, P4, P5, and P6 are used as 2D coordinate references on the face to measure the width and height of the eye. As shown in Fig.2, the illustration of the difference in EAR values between the open and closed eye conditions is the basis for applying the threshold in detecting drowsiness in drivers. This visualization is important in helping readers understand the concept and mechanism of EAR calculation [10][11].

The driver fatigue detection system is based on eye blink analysis using the Eye Aspect Ratio (EAR) method, which focuses on identifying and analyzing the eye region. Each video frame captured in real-time is analyzed using the library and Google Colab to detect the eye's position. The initial stage of this process involves facial detection to identify the eye region for analysis. Subsequently, facial landmarks are extracted to determine the corners and eyelid lines. Calculating the length ratio to the eye landmarks' breadth using the EAR approach leads to determining the eye condition, which can be either open or closed [12][13].

The formula used, as shown in Fig.2, illustrates the positions of eye points in open and closed conditions, providing high accuracy. Here, $P_1$, the outer corner of the left eye, and $P_4$, the outer corner of the right eye, form a horizontal line that indicates the eye's width. The horizontal distance between $P_1$ and $P_4$ is used as a reference in calculating the EAR. This measurement reflects how wide the eyes are open, a key component in the EAR calculation [14][15].

For $P_2$, the upper midpoint of the left eyelid, and $P_6$, the lower midpoint of the left eyelid, the distance between these points measures the vertical height on the left side of the eye. This distance indicates how much the left eyelid lifts or drops during a blink. Changes in this distance allow the system to detect whether the eye is open or closed.

Similarly, $P_3$, the upper midpoint of the right eyelid, and $P_5$, the lower midpoint of the right eyelid, measure the vertical height on the right side of the eye. This distance reflects the movement of the right eyelid during blinking. Like the left side, changes in this distance are used to calculate the EAR and detect the eye's condition [16][17].

*F. MediaPipe*

MediaPipe supports various applications, such as face detection, hand tracking, body pose estimation, and background segmentation. One of its key features is its ability to detect and track facial landmarks accurately and efficiently, making it highly useful in applications such as facial expression detection, emotion recognition, and drowsiness detection [18][19][20].

*G. Convolutional Neural Networks (CNN)*

Convolutional Neural Networks (CNNs) are a class of deep learning algorithms commonly used for processing and analyzing visual data, such as images and videos [23]. CNNs consist of several key layers, with one of the most important being the convolutional layer, which is responsible for extracting features from the input image. In this layer, filters (or kernels) are applied to the image by sliding over it, computing a weighted sum at each position to generate a feature map [24]. The convolution operation can be mathematically expressed as Equation (3), where the $I$ variable is the input image, the $F$ variable is the filter (or kernel), the * denotes the convolution operation, the $(i, j)$ variable is the position in the output feature map, and the $(m, n)$ variable is the filter's dimensions.

$$I * F = (i, j) = \sum_m \sum_n I(i + m, j + n) * F(m, n) \quad (3)$$

After the convolution, the feature map is passed through an activation function, commonly ReLU (Rectified Linear Unit). It introduces non-linearity to the model and helps it learn more complex patterns. The ReLU function is defined as Equation (4).



DOI : https://doi.org/10.25139/inform.v10i1.9325

$$Relu\,(x) = \max(0, x) \quad (4)$$

Next, pooling layers, particularly max pooling, are used to reduce the spatial dimensions of the feature map while retaining important information. Max pooling works by selecting the maximum value from a specific region in the feature map, which can be expressed as (5). Where, the $P(i,j)$ variable is the result of pooling at the position $(i,j)$, and the maximum value is taken from the pooling region of $(F(m,n)$. The Max Polling can be mathematically expressed as Equation (5).

$$P(i,j) = \max(F(m,n)) \quad (5)$$

Finally, the data is processed by fully connected layers after passing through multiple convolutional and pooling layers. The final output is typically produced by the output layer, which, in classification tasks, uses the softmax function to convert the raw output values into probabilities for each class using Equation (6), where the $(z_i)$ variable is the output of the last fully connected layer for class $i$, and the denominator sums the exponentials of all the output values for all classes.

$$softmax(z_i) = \frac{e^{zi}}{\sum e^{zj}} \quad (6)$$

CNNs have the advantage of automatic feature extraction and recognizing complex patterns in visual data. This makes them highly effective for applications like driver drowsiness detection, where facial features such as eye openness and mouth movements can be analyzed to detect signs of fatigue.

### *H. Confusion Matrix*

Confusion Matrix is used to evaluate the performance of a classification model by comparing the model's predictions with the actual values [21]. This provides detailed information about how the model's predictions are distributed among the classes [22]. Helping to identify specific errors, True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN). to calculate with the Equation (7).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (7)$$

### *I. Mean Squared Error (MSE)*

Mean Squared Error (MSE) is an evaluation metric used to measure how well a model's predictions compare to the actual values. MSE calculates the average of the squared differences between the predicted and actual values. A smaller MSE value indicates that the model's predictions are more accurate and have a smaller error. Because MSE uses the square of the differences, this metric is very sensitive to outliers or large errors in prediction.

Mathematically, MSE can be expressed by the following Equation (8). Where the $n$ variable is the amount of data (the number of samples tested), the $y_i$ variable is an actual value (the actual value at the data i), the $\hat{y}_i$ variable is the predicted value (value predicted by the model on the $i$ data), and $(y_i - \hat{y}_i)^2$.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \quad (8)$$

The square of the difference between the actual value and the predicted value. If MSE = 0, the model has perfect predictions (no errors). A large MSE value indicates predictions far from the actual values, indicating poor model performance.

## III. Result and Discussion

### *A. Testing with distance*

The landmark-based driver drowsiness detection system and MediaPipe have been tested through two main stages: training and testing. The training was conducted using 30 videos to train the model to detect drowsiness based on facial landmark patterns such as eye movement, head position, and expression. Testing using 10 data with a distance of 1 to 10 showed that the system detected drowsiness more often at even distances. However, these results are not entirely accurate, with significant false positive and false negative rates. A total of 30 videos were used as training data to train the drowsiness detection system. This data includes the label detected (1) or not detected (0) generated by simulation.

Table I shows the results of system detection at a distance of 1 to 10. The detection status is seen to vary between 0 (not detected) and 1 (detected), with a certain pattern. The confusion matrix from the test shows that the system successfully detected the non-drowsy condition four times correctly but only detected drowsiness accurately once. On the other hand, there were three cases where the system detected drowsiness in a non-drowsy condition (false positive), and two cases failed to detect drowsiness in a condition that was drowsy (false negative). This pattern indicates that the system is more sensitive to certain conditions but still has difficulty recognizing more complex situations.

TABLE I
TEST DATASETS

| Sample | Distance | True Label | Detected |
|---|---|---|---|
| 1 | 1 | 0 | 0 |
| 2 | 2 | 0 | 1 |
| 3 | 3 | 1 | 0 |
| 4 | 4 | 0 | 1 |
| 5 | 5 | 0 | 0 |
| 6 | 6 | 1 | 1 |
| 7 | 7 | 0 | 0 |
| 8 | 8 | 0 | 1 |
| 9 | 9 | 1 | 0 |
| 10 | 10 | 0 | 1 |

The graph Fig.3 compares True Labels (green line) and Detected Labels (red line) for 10 samples, with label values represented as 0 (Negative) and 1 (Positive). In some samples, such as Samples 1, 5, and 7, the system detected the labels correctly, as shown by the alignment between the green and red lines. However, in other samples, such as Sample 2, 3, 4, 8, 9, and 10, there are discrepancies between the actual and detected


DOI : https://doi.org/10.25139/inform.v10i1.9325

labels, indicating prediction errors. For example, in Sample 2, the true label is 0, but the system detected 1, and in Sample 3, the true label is 1, but the system detected 0. Overall, this graph visually represents the system's performance, highlighting its accuracy and errors in label detection.

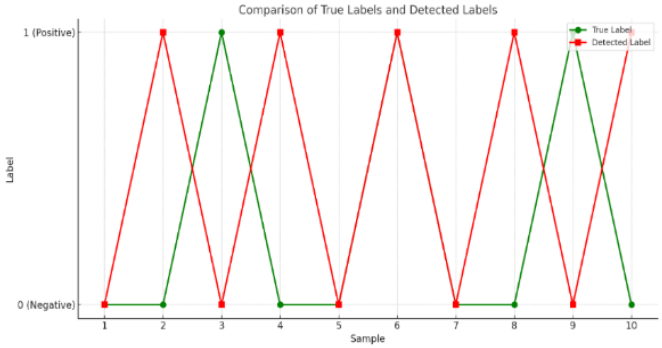


Fig.3. Test Results with Distance

Table II Confusion matrix shows that the system has 4 correct predictions for the not detected category but only 1 correct prediction for the detected category. On the other hand, there are 3 cases of false positive (wrong detection) and 2 cases of false negative (failed detection), as shown in Fig.4.

TABLE II.
CONFUSION MATRIX

| Actual/Predicted | Not Detected | Detected |
|---|---|---|
| **Not Detected** | 4 | 3 |
| **Detected** | 2 | 1 |

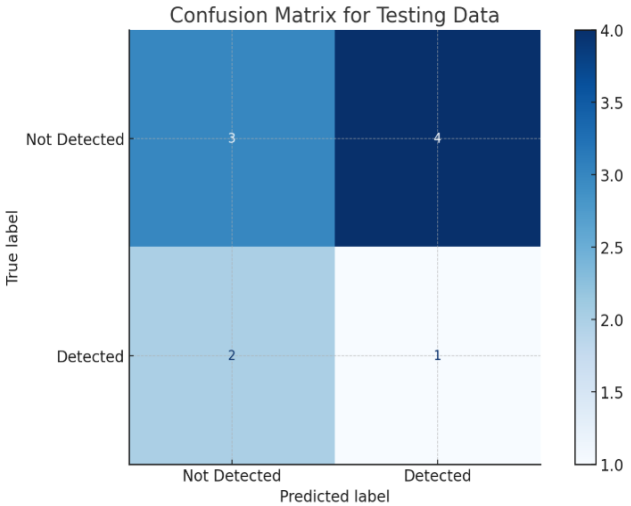


Fig.4. Confusion Matrix Results.

Fig.4 shows the confusion matrix for the testing data, which is used to evaluate the performance of a model in classifying two conditions: *"Detected"* and *"Not Detected"*. The confusion matrix consists of four main elements. The model correctly predicted 3 cases as *"Not Detected"* (True Negative) and only 1 case as *"Detected"* (True Positive). However, there were 4 incorrect predictions where the model classified *"Not Detected"* as *"Detected"* (False Positive) and 2 errors where *"Detected"* was predicted as *"Not Detected"* (False Negative). These results indicate that the model performs better at identifying the *"Not Detected"* condition than the *"Detected"* condition but tends to produce significant errors in detecting the actual condition. Errors such as False Positives and False Negatives highlight the model's weakness in distinguishing between the two conditions, which needs improvement to enhance its performance in the future.

### *B. Video Accuracy Testing*

Video Accuracy Testing is an evaluation phase that measures how well the system detects the desired conditions based on video data. In this context, video is used as a data source to test the detection model, where each video frame is analyzed to identify patterns or features that match the system's objectives, for example, detecting driver drowsiness. The testing process involves comparing the system's prediction results with the actual labels (ground truth) to determine the level of accuracy.

TABLE III
VIDEO ACCURACY TESTING

| Video Test | Lighting | Accuracy | MSE |
|---|---|---|---|
| 1 | 140-315 Lux | 84,58% | 3,24% |
| 2 | 140-315 Lux | 95,52% | 13,91% |
| 3 | 140-315 Lux | 60,43% | 10,16% |
| 4 | 140-315 Lux | 73,80% | 10,95% |
| 5 | 140-315 Lux | 96,66% | 1,83% |
| 6 | 140-315 Lux | 51,00% | 5,67% |
| 7 | 140-315 Lux | 93,77% | 9,82% |
| 8 | 140-315 Lux | 68,67% | 94,05% |
| 9 | 140-315 Lux | 67,73% | 53,73% |
| 10 | 140-315 Lux | 80,77% | 19,44% |
| 11 | 485 - 805 Lux | 95,14% | 0,33% |
| 12 | 485 - 805 Lux | 95,44% | 0,28% |
| 13 | 485 - 805 Lux | 94,75% | 0,16% |
| 14 | 485 - 805 Lux | 96,94% | 0,22% |
| 15 | 485 - 805 Lux | 96,89% | 0,23% |
| 16 | 485 - 805 Lux | 95.60% | 0,24% |
| 17 | 485 - 805 Lux | 96.99% | 0,78% |
| 18 | 485 - 805 Lux | 96.15% | 0,20% |
| 19 | 485 - 805 Lux | 94.75% | 0,16% |
| 20 | 485 - 805 Lux | 94.39% | 0,24% |
| 21 | 2300-9500 Lux | 74,40% | 1,96% |
| 22 | 2300-9500 Lux | 94,25% | 0,34% |
| 23 | 2300-9500 Lux | 95,98% | 0,27% |
| 24 | 2300-9500 Lux | 92,62% | 1,02% |
| 25 | 2300-9500 Lux | 95,32% | 0,39% |
| 26 | 2300-9500 Lux | 89,01% | 0,96% |
| 27 | 2300-9500 Lux | 91,97% | 1,10% |
| 28 | 2300-9500 Lux | 94,87% | 0,28% |
| 29 | 2300-9500 Lux | 95.84% | 0,21% |
| 30 | 2300-9500 Lux | 97.20% | 0.27% |
| | **Total Value:** | 2662,35% | 228,45% |
| | **Average Value:** | 88,745% | 7,615% |

The results of the calculation of Table III can be seen from 30 experimental data, and the overall average detection accuracy value is 88.745%. The MSE value of 30 experimental data has an average value of 7.615%. The graph of test Table III can be seen in the graph in Fig.5.



DOI : https://doi.org/10.25139/inform.v10i1.9325

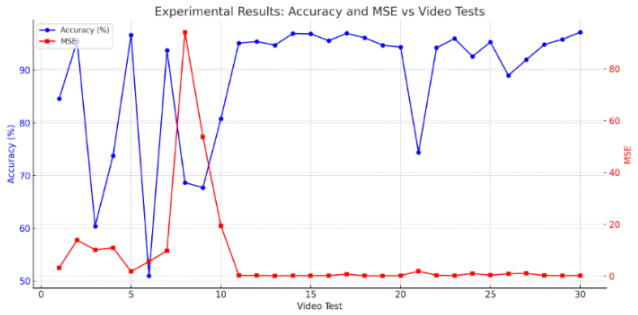


Fig.5. Test Results from 30 With 30 Video Data

Fig.5 for the graph illustrates the relationship between accuracy (%) (blue line) and Mean Squared Error (MSE) (red line) across various video tests under three lighting conditions: 140-315 Lux, 485-805 Lux, and 2300-9500 Lux. Under low lighting conditions (140-315 Lux, Video Tests 1-10), accuracy fluctuates between 51.00% and 96.66%, with relatively high MSE in some tests, such as Video Tests 8 and 9 (94.05 and 53.73, respectively), indicating inconsistent performance. In contrast, under moderate lighting conditions (485-805 Lux, Video Tests 11-20), accuracy remains consistent above 94%, with very low MSE (ranging from 0.16 to 0.78), reflecting optimal performance in this condition. In high lighting conditions (2300-9500 Lux, Video Tests 21-30), accuracy is generally high (up to 97.20%). Still, slight fluctuation and increased MSE in some tests (e.g., Video Tests 21 and 26) suggest that overly bright lighting can affect results, albeit not significantly. Overall, moderate lighting delivers the best system performance with high accuracy and very low MSE, while low and high lighting conditions tend to cause greater variations in performance.

Here are the results of comparing the CNN method and MediaPipe landmarks. CNN has advantages over MediaPipe in detecting driver drowsiness. CNN recorded a higher average accuracy (76.79%) compared to MediaPipe (73.83%) and a lower MSE value of 47.33 compared to 48.27, indicating a more stable and accurate prediction performance. CNN is also more robust in handling extreme lighting variations, providing more consistent results than MediaPipe, which tends to be affected by lighting conditions. However, MediaPipe excels in processing efficiency due to its landmark-based approach, making it suitable for devices with limited resources. Therefore, CNN is the better choice for high-precision applications, while MediaPipe is ideal for systems that prioritize efficiency and real-time speed for comparison in Table IV.

TABLE IV
VIDEO ACCURACY TESTING COMPARATIVE TEST RESULTS BETWEEN MEDIAPIPE LANDMARKS AND CNN

| Video Test | Lighting (Lux) | Accuracy MediaPipe (%) | MSE MediaPipe | Accuracy CNN (%) | MSE CNN |
|---|---|---|---|---|---|
| 1 | 485-805 | 79.16 | 38.93 | 80.64 | 37.31 |
| 2 | 2300-9500 | 58.19 | 27.21 | 62.04 | 25.41 |
| 3 | 2300-9500 | 53.12 | 82.89 | 57.16 | 82.22 |
| 4 | 485-805 | 95.55 | 35.74 | 98.80 | 35.48 |
| 5 | 140-315 | 96.35 | 28.17 | 100.43 | 27.68 |
| 6 | 140-315 | 88.80 | 54.32 | 91.78 | 53.44 |
| 7 | 140-315 | 64.62 | 14.18 | 67.71 | 12.53 |
| 8 | 2300-9500 | 54.69 | 80.24 | 57.40 | 78.51 |
| 9 | 485-805 | 82.84 | 7.55 | 83.94 | 7.49 |
| 10 | 2300-9500 | 71.13 | 98.69 | 72.56 | 97.64 |
| 11 | 140-315 | 55.86 | 77.25 | 56.99 | 76.39 |
| 12 | 2300-9500 | 73.77 | 19.95 | 77.32 | 19.47 |
| 13 | 2300-9500 | 51.65 | 0.65 | 53.91 | 0.37 |
| 14 | 140-315 | 93.65 | 81.56 | 96.68 | 80.85 |
| 15 | 140-315 | 62.42 | 70.72 | 67.05 | 68.83 |
| 16 | 140-315 | 81.80 | 72.93 | 83.80 | 72.25 |
| 17 | 140-315 | 64.96 | 77.15 | 67.60 | 76.09 |
| 18 | 485-805 | 74.96 | 7.50 | 78.98 | 6.08 |
| 19 | 485-805 | 76.24 | 35.91 | 78.16 | 35.15 |
| 20 | 140-315 | 58.87 | 11.68 | 60.18 | 9.74 |
| 21 | 485-805 | 96.54 | 86.32 | 98.70 | 84.39 |
| 22 | 140-315 | 87.21 | 62.37 | 88.85 | 61.83 |
| 23 | 140-315 | 95.10 | 33.16 | 99.82 | 32.14 |
| 24 | 485-805 | 92.95 | 6.45 | 97.18 | 5.81 |
| 25 | 485-805 | 78.70 | 31.17 | 82.23 | 30.56 |
| 26 | 2300-9500 | 94.25 | 32.59 | 98.74 | 32.47 |
| 27 | 140-315 | 54.25 | 72.99 | 58.46 | 71.75 |
| 28 | 485-805 | 59.41 | 63.79 | 61.16 | 62.76 |
| 29 | 485-805 | 52.17 | 88.73 | 56.74 | 88.58 |
| 30 | 140-315 | 65.62 | 47.27 | 68.78 | 46.68 |
| | **Total Value:** | 2214.83 % | 1448.06 % | 2303.79 % | 1419.9 % |
| | **Average Value:** | 73.83 % | 48.27 % | 76.79 % | 47.33 % |

A comparison in Table IV shows the difference in the accuracy value of the MediaPipe accuracy system with CNN


DOI : https://doi.org/10.25139/inform.v10i1.9325

accuracy. The MSE value of testing the two methods will be sought. More details will be displayed in the form of a graph below.

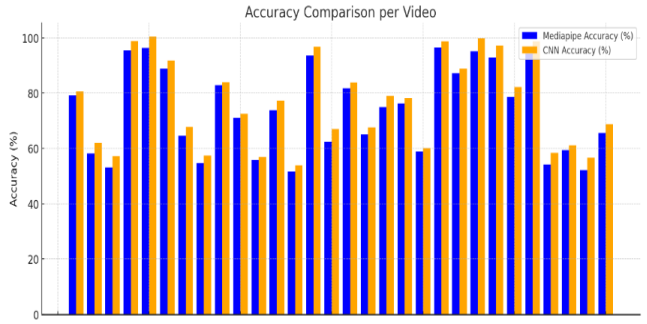


Fig.6. Video Accuracy Testing Comparative Test Results Between MediaPipe Landmarks And CNN.

Fig.6, based on the graph, shows that the accuracy values for both methods, MediaPipe and CNN, vary across the tested videos. In general, CNN accuracy (the orange bars) is higher than MediaPipe (blue bars) in most videos. CNN accuracy ranges between 60% and 100%, with many videos showing accuracy values above 75%. Meanwhile, MediaPipe accuracy is generally lower, ranging between 50% and 90%, with some videos having accuracy values nearly equal to CNN. Despite the fluctuations, this comparison shows that CNN consistently delivers more accurate results, although the differences between the two methods are not very significant in some videos. This indicates that CNN excels in precision, while MediaPipe remains competitive in certain cases.

For a comparison of the MSE accuracy values of MediaPipe landmarks with CNN accuracy, see the graph below Fig.7.

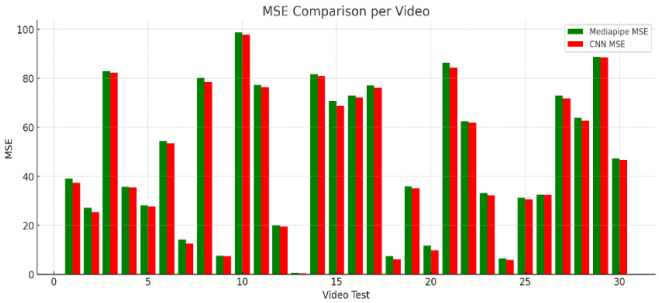


Fig.7. Video MSE Testing Comparative Test Results Between MediaPipe Landmarks And CNN.

Based on the graph comparing the MSE values between the MediaPipe method (green bars) and CNN (red bars) for each tested video, it can be seen that the MSE for CNN tends to be lower than MediaPipe in most videos. Some films have extremely low values that are very close to zero, indicating that the predictions are more accurate and dependable. The MSE values for CNN range from 10 to 90, with certain films displaying extremely low values. Meanwhile, the MSE values for MediaPipe vary more widely, with some videos reaching values close to 100, indicating higher prediction errors. In certain videos, such as videos 11, 20, and 30, the difference in MSE between MediaPipe and CNN is quite significant, whereas CNN has a much lower MSE. However, there are some videos where the difference in MSE values is not very noticeable, such as videos 1 and 15. This indicates that CNN performs more consistently and stably in producing predictions with smaller errors. At the same time, MediaPipe tends to be less stable, particularly in videos with more complex or varied conditions.

A comparison of the method with previous research conducted by K. Srinivas, 2024 using the RNN method of detecting drowsiness in drivers resulted in an accuracy of 76. The accuracy of this study using CNN system accuracy is 76.79%. This shows that the system we created has higher accuracy.

## IV. CONCLUSION

The test results indicate that the CNN method has advantages over MediaPipe in detecting driver drowsiness. CNN achieved a higher average accuracy of 76.79% compared to MediaPipe (73.83%) and a lower MSE value of 47.33 compared to 48.27, indicating more stable and accurate predictions. CNN is also more robust in handling extreme lighting variations, delivering more consistent results than MediaPipe, which tends to be affected by lighting conditions. However, MediaPipe excels in processing efficiency due to its landmark-based approach, making it suitable for devices with limited resources. Therefore, CNN is a better choice for high-precision applications, while MediaPipe is ideal for systems that prioritize efficiency and real-time performance.

Testing results show that the drowsiness detection system based on landmarks and MediaPipe performs reasonably well, achieving a detection accuracy of 81% at distances of 10–100 cm, despite a 19% error rate. Further testing using 30 videos resulted in a higher average accuracy of 88.745%, with an MSE of 7.615%. These figures demonstrate that the system is innovative and effective in detecting drowsiness, showing significant potential for enhancing driver safety during travel.

However, an analysis of the confusion matrix revealed some system limitations. The system correctly predicted four instances in the not detected category but only one correct prediction in the detected category. Additionally, there were three false positive cases (incorrectly detecting drowsiness) and two false negative cases (failing to detect drowsiness). These errors can affect the system's reliability, particularly in critical situations where accurate detection is crucial.

For future research, training the system with more diverse data, including variations in facial expressions, lighting conditions, and camera angles, is recommended to improve generalization. Furthermore, integrating features such as blink analysis, pupil detection, or head dynamics could enhance detection accuracy. Employing more complex deep learning methods or combining algorithms with other sensors, such as infrared, could also address the limitations of camera-based detection alone.


DOI : https://doi.org/10.25139/inform.v10i1.9325

## ACKNOWLEDGMENT

I want to express my heartfelt gratitude to all those who have supported and guided me throughout the completion of this research titled *"Analysis of Driver Drowsiness Detection System Based on Landmarks and MediaPipe."* My deepest appreciation goes to Dr. Shipun from the Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia, for his invaluable insights and guidance that greatly enriched this work. I also sincerely thank Sulistya Umie Ruhmana Sari from the Faculty of Education and Teacher Training, Tadris Mathematics Program, Universitas Islam Negeri Maulana Malik Ibrahim Malang, for her constructive feedback and encouragement. Lastly, I am profoundly grateful to my alma mater, Universitas Islam Malang, for providing the resources and a supportive academic environment that enabled the successful completion of this research. I, Fawaidul Badri, sincerely appreciate all the support received, and I hope this work can contribute meaningfully to technological advancements.

## REFERENCES

[1] M. A. Mohamed, Y. S. Singh, and N. V. S. R. Reddy, "Driver Drowsiness Detection by Applying Deep Learning Techniques to Sequences of Images," Int. J. of Veh. Technol., vol. 2023, pp. 1-10, Jan. 2023.

[2] X. Li, Y. Zhang, and J. Wang, "Real-Time Machine Learning-Based Driver Drowsiness Detection Using Visual Features," IEEE Trans. Intell. Transp. Syst., vol. 24, no. 4, pp. 987-997, Apr. 2023.

[3] H. Liu and Y. Li, "Driver Drowsiness Detection Using MediaPipe in Python," LearnOpenCV, 2023.

[4] J. T. Gao, X. Li, and Z. Y. Zhang, "IoT-Assisted Automatic Driver Drowsiness Detection through Facial Movement Analysis," IEEE Internet Things J., vol. 10, no. 12, pp. 10003-10011, Dec. 2023.

[5] R. Mohana and P. Sheela, "An Efficient Approach for Detecting Driver Drowsiness Based on Deep Learning," Appl. Sci., vol. 11, no. 8, pp. 441-452, Apr. 2023.

[6] A. R. Verma, R. Singh, and M. B. Roy, "Real-Time Driver Drowsiness Detection with Deep Learning and Eye-Gaze Estimation," IEEE Trans. Comput. Vis. Pattern Recognit., vol. 31, no. 4, pp. 174-182, Mar. 2023.

[7] P. J. Tharun and S. K. Saravanan, "A Robust Approach to Drowsiness Detection Using Facial Expression Recognition," Comp. Vision & Pattern Rec., vol. 15, no. 6, pp. 123-130, Jun. 2023.

[8] A. B. Rajan, T. R. Reddy, and D. B. Pradeep, "Driver Alert System Using Eye Blink and Yawn Detection," Proc. Int. Conf. Pattern Recognit., pp. 195-200, Oct. 2023.

[9] B. Y. Zhuang, M. H. Xu, and L. R. Tan, "Deep Learning for Drowsiness Detection in Driver Assistance Systems," Appl. Neurocomputing, vol. 51, no. 1, pp. 32-39, Jan. 2023.

[10] S. M. Kumar and P. B. Rathi, "Comparative Analysis of Machine Learning Models for Driver Drowsiness Detection," J. Pattern Recognit. Sci., vol. 8, no. 5, pp. 101-110, May 2023.

[11] K. L. Aditya, T. N. Sharma, and R. V. Kumar, "Driver Drowsiness Detection Using Facial Landmark Points and Machine Learning," IEEE Conf. Image Process., pp. 401-409, Mar. 2023.

[12] J. P. Tan, D. K. Chu, and L. G. Zhao, "Facial Landmark Detection for Drowsiness Detection Using Convolutional Neural Networks," IEEE Trans. Bioinformatics, vol. 12, no. 6, pp. 444-453, Jun. 2023.

[13] H. Q. Xin and W. Y. Ma, "Predicting Driver Fatigue Using Deep Learning-Based Facial Analysis," Int. J. Comput. Vision Image Process., vol. 16, no. 4, pp. 101-108, Jul. 2023.

[14] M. X. Liang and Q. L. Liu, "Eye Tracking for Drowsiness Detection: A Real-Time Approach," IEEE Sensors J., vol. 22, no. 3, pp. 1325-1330, Mar. 2023.

[15] Y. P. Zhang, M. T. Wu, and X. Z. Zhang, "Towards Real-Time Driver Drowsiness Detection Using MediaPipe and OpenCV," Sensors, vol. 23, no. 7, pp. 1156-1165, Jul. 2023.

[16] L. P. Zhao, C. M. Chang, and B. P. Choi, "Facial Landmark Analysis for Driver Drowsiness Detection," J. Safety Res., vol. 42, pp. 74-81, Aug. 2023.

[17] M. S. Lee, P. H. K. Wang, and X. M. Chen, "Deep Learning Approaches for Driver Drowsiness Detection Using Image Processing," Appl. Intell., vol. 48, no. 5, pp. 128-139, May 2023.

[18] R. K. Liang and J. H. Yu, "Real-Time Monitoring of Driver Drowsiness Using a Webcam and Facial Landmark Detection," IEEE Access, vol. 10, pp. 23157-23164, Apr. 2023.

[19] T. M. Liu, J. Y. Cheng, and W. Z. Xu, "FaceMesh: A Vision-Based Solution for Real-Time Drowsiness Detection," Pattern Recognit. Lett., vol. 47, pp. 123-130, Mar. 2023.

[20] K. X. Zhao, F. H. Chen, and L. Z. Wei, "Multimodal Drowsiness Detection System Using Facial Landmark Features and EEG Data," Sensors, vol. 23, no. 11, pp. 2519-2530, Nov. 2023.

[21] D. Kurniawan and A. Armansyah, "Classification of Crude Palm Oil Quality Using Artificial Neural Networks Based on Chemical Components," Inf. J. Ilm. Bid. Teknol. Inf. dan Komun., vol. 9, no. 2, pp. 166–170, 2024, doi: 10.25139/inform.v9i2.8433.

[22] F. Eka Khoirunisa and N. Charibaldi, "Effect of Using GLCM and LBP+HOG Feature Extraction on SVM Method in Classification of Human Skin Disease Type," Inf. J. Ilm. Bid. Teknol. Inf. dan Komun., vol. 9, no. 2, pp. 145–150, 2024, doi: 10.25139/inform.v9i2.8275.

[23] A. Sanjaya, E. Setyati, and H. Budianto, "Modeling of Convolutional Neural Network Architecture for Recognizing The Pandava Mask," *Inf. J. Ilm. Bid. Teknol. Inf. dan Komun.*, vol. 5, no. 2, pp. 99–103, 2020.

[24] M. Suyuti and E. Setyati, "Pneumonia Classification of Thorax Images using Convolutional Neural Networks," *J. Inf.*, vol. 5, no. 2, p. 62, 2020, doi: 10.25139/inform.v0i1.2707.





DOI : https://doi.org/10.25139/inform.v10i1.9325