



Lightweight Facial Feature-Based Drowsiness Detection Using Haar Cascade, EAR, MAR, and KNN

Asri Mulyani¹, Erick Fernando², Dede Kurniadi³, Adinda Jaida Fauziyah⁴

^{1,3,4}Department of Computer Science, Institut Teknologi Garut, Garut, Indonesia

²Faculty of Engineering and Informatics, Universitas Multimedia Nusantara, Tangerang, Indonesia

*asrimulyani@itg.ac.id

Abstract

Vehicle accidents globally are heavily caused by driver fatigue, highlighting the need for automated fatigue detection systems. This research introduces an efficient method to detect driver fatigue using facial geometric features. The proposed method integrates Haar Cascade to locate faces, followed by the extraction of Eye Aspect Ratio (EAR) together with Mouth Aspect Ratio (MAR) metrics, which are then classified through the K-Nearest Neighbor (KNN) algorithm. The experiment utilized a total collection of 2,468 facial samples structured across Open, Closed, Yawn, and No Yawn categories, which were subsequently grouped into "Drowsy" and "Not Drowsy" classes. Model evaluation was performed via Stratified 10-Fold Cross-Validation. The results show that the classifier obtained a mean accuracy rate of $88.2\% \pm 2.6\%$ with an AUC of 0.93, confirming robust classification effectiveness. These findings position the integrated geometric-feature approach as a systematically validated classical baseline for driver monitoring research. While this study does not empirically measure hardware latency, the stable classification performance suggests that EAR and MAR features provide a reliable foundation for evaluating the performance-to-complexity trade-offs of more advanced models in resource-constrained environments.

Keywords: drowsiness detection; eye aspect ratio; Haar cascade; k-nearest neighbor; mouth aspect ratio

How to Cite: A. Mulyani, E. Fernando, D. Kurniadi, and A. J. Fauziyah, "Lightweight Facial Feature-Based Drowsiness Detection Using Haar Cascade, EAR, MAR, and KNN", *J. RESTI (Rekayasa Sist. Teknol. Inf.)*, vol. 10, no. 5, pp. 1435–1446, Oct. 2026, doi: [10.29207/resti.v10i5.7190](https://doi.org/10.29207/resti.v10i5.7190).

Permalink/DOI: <https://doi.org/10.29207/resti.v10i5.7190>

Received: September 19, 2025

Accepted: May 23, 2026

Available Online: October 13, 2026

This is an open-access article under the CC BY 4.0 License

Published by Ikatan Ahli Informatika Indonesia

1. Introduction

Driver fatigue is recognized as one of the major contributors to road traffic accidents worldwide. The World Health Organization (WHO) and several previous studies have reported that drowsy driving contributes substantially to traffic accidents, with estimates indicating that approximately 20% of all road crashes are associated with driver fatigue. [1]. Current drowsiness detection still relies on subjective self-assessment, which is often ineffective, thus highlighting the need for accurate and automated systems.

Various studies have been conducted using vision-based approaches. Ramadana Lubis et al [2] achieved 92.5% accuracy with Haar Cascade, while Vishesh et al. [3] employed a CNN-based approach using MobileNetV2 for eye blink detection, achieving an accuracy of 97%. Mujahidin et al. [4] demonstrated the use of Haar Cascade for facial region detection before CNN-based drowsiness classification, while Arun Babu

et al. [5] reported that detection performance decreased under lower illumination and greater camera distance. Altameem et al [6] investigated driver drowsiness detection using a hybrid machine learning approach based on visual facial features for real-time monitoring. Recent studies have also explored deep learning models, including convolutional neural networks (CNNs) and transfer learning-based architectures, for driver fatigue detection, achieving high predictive performance at the expense of greater computational demands and the need for larger annotated datasets. [7]. While impressive in accuracy, most deep learning architectures inherently demand significant computational resources, which may pose integration challenges for cost-sensitive, low-resource embedded deployment. Rather than claiming definitive superiority in performance, this research addresses the need for a validated classical baseline by exploring the feasibility of a lightweight approach using geometric features. Therefore, several researchers have shifted their

attention toward lightweight computer vision techniques that utilize facial geometric descriptors, particularly the Eye Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR), as they provide efficient computation while preserving reliable detection performance [8]. This resource constraint represents a key gap that our lightweight approach addresses.

This study proposes a lightweight approach for detecting driver drowsiness by integrating the Haar Cascade algorithm to identify facial regions, followed by geometric feature extraction through the Eye Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR), before performing classification with the K-Nearest Neighbor (KNN) algorithm. The model was assessed through a 10-fold cross-validation procedure using a dataset of 2,468 facial images, resulting in an average accuracy of 88.2% and an AUC of 0.93. These findings suggest that simple geometric features combined with a classical classifier can provide reliable performance for fatigue detection tasks.

In contrast, facial geometric descriptors, including the Eye Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR), offer an efficient computational alternative for identifying fatigue-related behaviors, particularly eye closure and yawning. Similarly, the K-Nearest Neighbor (KNN) classifier offers simplicity and low resource usage while maintaining competitive accuracy [7], [9]. These characteristics make them suitable for embedded or low-resource driver monitoring systems.

Eye Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR) remain widely adopted and well-established geometric features in driver drowsiness detection because of their low computational cost, straightforward implementation, and reliable performance under various operating conditions. For instance, Kannan et al. [10] reported that EAR effectively identifies eye closure without requiring computationally intensive image processing, making it appropriate for real-time driver monitoring and implementation on devices with limited computational resources. Similarly, MAR effectively captures yawning behavior with minimal computational cost by calculating the vertical versus horizontal distances between mouth landmarks, as shown in several visual-feature-based models that achieve high accuracy while preserving low latency [7].

Meanwhile, the K-Nearest Neighbor (KNN) algorithm has been widely adopted because of its straightforward implementation and modest memory requirements, making it well suited for deployment in edge computing and embedded systems. [11], [12]. Recent research has also demonstrated the applicability of KNN for driver drowsiness detection, with KNN achieving strong classification performance when combined with image-based facial feature extraction [9].

This study introduces a lightweight method for driver drowsiness detection by integrating geometric features

(EAR and MAR) with a classical K-Nearest Neighbor (KNN) classifier.

Unlike previous studies that mainly rely on deep learning models with high computational requirements, this study focuses on evaluating the effectiveness of simple geometric features for fatigue detection using a lightweight machine learning model. The proposed method demonstrates that a combination of EAR and MAR features with KNN can achieve reliable classification performance while maintaining model simplicity and interpretability.

The primary contribution of this study lies in the systematic evaluation of a classical baseline for driver drowsiness detection using EAR and MAR geometric features. Unlike previous studies that may prioritize absolute accuracy through complex architectures, this work focuses on establishing the performance boundaries of a lightweight framework using a KNN classifier. By providing a stable reference through 10-fold cross-validation, this study establishes a benchmark for future comparisons with deep learning methods, allowing researchers to evaluate whether their higher computational requirements correspond to meaningful improvements in classification performance. Demonstrating that classical machine learning methods can provide stable and competitive performance for fatigue detection tasks. The experimental evaluation demonstrates that the proposed approach achieved an average accuracy of $88.2\% \pm 2.6\%$ and an AUC of 0.93, indicating that the method provides a reliable baseline for driver monitoring applications, especially for deployment on platforms with constrained computational capabilities. To ensure academic rigor and clarity of scope, this study strictly delineates its focus to the computational analysis of visual fatigue indicators. This research does not perform, nor does it intend to substitute, clinical diagnoses of sleep disorders or physiological medical assessments. The methodology is confined to the domain of behavioral computer vision, where observable facial behaviors, specifically prolonged eye closure and yawning, are measured as indirect indicators of driver drowsiness. By establishing this boundary, the study avoids overgeneralization into the clinical medical field.

2. Methods

2.1. Proposed Framework

This research began with a literature review to explore previous studies on driver drowsiness detection through facial expression analysis. This literature review formed the basis for designing a reliable and non-intrusive detection method, specifically targeting operational efficiency for low-resource embedded platforms. The proposed methodology integrated three main components: facial region detection through the Haar Cascade algorithm, feature extraction of Eye Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR) values, and classification using the K-Nearest Neighbor (KNN)

algorithm. The selection of these classical methods was based on a design rationale that prioritizes algorithmic simplicity, a critical consideration for potential in-vehicle deployment where minimizing model complexity is often preferred over incremental gains in classification accuracy.

In software engineering, K-Nearest Neighbor (KNN) identifies similar projects by measuring the distance between their attributes, allowing information from neighboring projects to be used for missing-data imputation and effort prediction [13], [14]. This research develops a facial expression-based driver fatigue detection model, starting from face detection using Haar Cascade to isolate facial areas automatically [15]. Next, EAR and MAR geometric features are extracted to represent eye and mouth conditions, which are used as input for the K-Nearest Neighbor (KNN) classification algorithm. Haar Cascade was selected as the face detection method to identify the facial region of interest before subsequent drowsiness analysis [16]. The combination of these methods makes the drowsiness detection system computationally efficient and can be applied to monitoring drivers of private and commercial vehicles.

Classification is a machine learning task that assigns data to predefined categories, such as identifying the driver's eye state as open or closed for drowsiness detection [17]. In this research, driver fatigue detection is performed by analyzing geometry-based facial features, specifically the eye and mouth conditions via Eye Aspect Ratio (EAR) alongside Mouth Aspect Ratio (MAR) metrics, which represent the conditions of the eyes and mouth. The initial facial image is processed through Haar Cascade feature detection to isolate the face area, and then the EAR and MAR features are extracted from the image. These extracted feature values are subsequently categorized via the K-Nearest Neighbor (KNN) classifier to identify if the driver status is "Drowsy" or "Not Drowsy". This research is not conducted in real time and does not include an automated warning system. Instead, it focuses on evaluating the predictive accuracy of classification algorithms using previously gathered and prepared datasets. This approach aims to simplify the analysis without sacrificing detection accuracy.

2.2. Geometric Feature Extraction

Feature extraction serves as a vital process within computer vision applications and machine learning techniques, enabling the gathering and isolation of relevant information from raw data to facilitate easier analysis. This step aims to reduce data dimensionality while preserving the essential characteristics of the original features, thereby reducing computational cost and improving processing efficiency [18]. Three common types of features are color (RGB, HSV), shape (length-to-axis ratio), and texture (GLCM, LBP). In this study, the features used are the geometric metrics of Eye Aspect Ratio (EAR) together with Mouth Aspect Ratio (MAR), which represent the conditions of ocular and

oral states, respectively, for efficient and accurate detection of driver fatigue.

Accurate spatial coordinates of facial landmarks are determined using the Dlib shape predictor tool. Consequently, drowsiness can be detected through the monitoring of multiple parameters, including eye and mouth aspect ratios as well as head inclination angles. [8].

Evaluating ocular openness, the Eye Aspect Ratio (EAR) measures eye openness by utilizing six distinct facial landmarks positioned surrounding each eye. Specifically, it calculates how vertical dimensions compare relative to horizontal length. When a driver's eyes begin to close due to drowsiness, the EAR value decreases dramatically, even approaching zero upon being fully shut. When a driver's eyes begin to close due to drowsiness, the EAR value decreases, allowing predefined thresholds to be used to distinguish between open and closed eye states [7].

$$EAR = \frac{\|P_2 - P_6\| + \|P_3 - P_5\|}{2\|P_1 - P_4\|} \quad (1)$$

P_1 , P_2 , P_3 , P_4 , and P_6 are key points around the eye area, as shown in Figure 1, where the symbol $\|$ denotes determining the Euclidean spacing connecting the pair of coordinates.

Figure 1 illustrates the location of six facial landmark points positioned around the eye region, capturing both horizontal and vertical distances. The outer corners of the eye act as the horizontal reference points, while the upper and lower eyelid points define the vertical distance used in the EAR calculation. These points are then mathematically formulated in Equation 1, in which comparing vertical ocular opening against horizontal eye span determines the overall Eye Aspect Ratio (EAR). A higher ratio suggests fully unclosed eyes, while a lower value, approaching zero, signals that the eyes have become nearly shut, which is widely recognized to be a key visual marker for driver drowsiness.

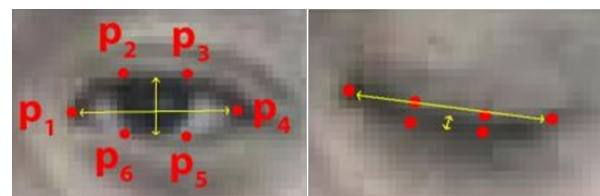


Figure 1. Location of the six key points in the eye

The Mouth Aspect Ratio (MAR) quantifies mouth opening and can be used to identify yawning as a visual indicator of drowsiness. Combined with EAR, MAR provides complementary information about eye closure and mouth opening, enabling facial visual features to be used for driver drowsiness detection [7].

$$MAR = \frac{\|P^2 - P^6\| + \|P^3 - P^5\|}{2\|P_1 - P_4\|} \quad (2)$$

P_1 to P_6 are key points around the mouth, as in Figure 2, with measurements taken on the interior rim of the

labial region in order to improve accuracy. The symbol $\|$ indicates the Euclidean distance between points.

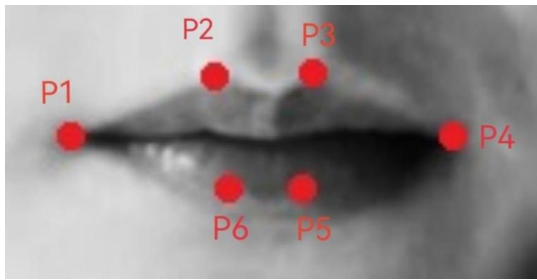


Figure 2. Location of the six facial landmarks within the oral region

As depicted in Figure 2, six landmark positions around the oral area are identified, where the mouth corners represent the horizontal reference, while vertical spacing separating the upper and lower labial borders represents mouth opening. These points are formulated in Equation 2 for determining the Mouth Aspect Ratio (MAR). An elevated MAR metric reflects a wider oral aperture, often linked to yawning, which is commonly recognized as an indicator of driver drowsiness. By combining MAR with EAR, the system improves the ability to detect fatigue by capturing both eye closure and yawning behavior through facial expression analysis. By focusing on inner lip landmark points, MAR provides a reliable yet lightweight indicator of yawning behavior, ensuring the method remains computationally efficient while maintaining detection accuracy.

2.3. K-Nearest Neighbor (KNN)

The K-Nearest Neighbor (KNN) algorithm serves as an effective distribution-free method for imputing missing data and predicting software project effort based on similarity. KNN calculates project similarity using the Manhattan distance, with $K=1$ selecting the closest project to support missing-data imputation and subsequent software effort prediction [13]. And the Euclidean distance measurement is formulated as Equation 3.

$$D = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2} \quad (3)$$

Description: Let X represent the sample instance, Y denote the test vector, and D is the distance between the two.

In addition to Manhattan distance, K-Nearest Neighbor (KNN) can also use Euclidean distance, as shown in Equation 3. This formula determines the direct spatial separation connecting a pair of data points across multidimensional dimensions by summing their squared differences of their coordinates and taking the square root. In the context of classification, Euclidean distance provides a reliable method for measuring similarity, where smaller distances indicate greater similarity between data points.

Evaluating driver status classification using a K-Nearest Neighbor-based feature selection strategy, this approach developed and evaluated attribute

combinations that provide critical insights into how influence drowsiness affects blinking patterns and head orientation of drivers [19]. KNN was selected for its proven accuracy in classification tasks and its low computational overhead, rendering it ideal for live deployment on resource-constrained hardware, such as in-vehicle monitoring systems.

2.4. Performance Evaluation

To assess the accuracy of classification algorithms like artificial neural networks, a confusion matrix compares model predictions with actual data in a matrix format. The top part of the matrix displays the predicted class, while the left part displays the actual class [20]. In this work, multiple performance evaluation metrics are adopted, including sensitivity, specificity, precision, F1 Score, MCC, kappa, and AUC. Figure 7 illustrates the performance metrics obtained by the ERS-KNN model, metrics. Specifically, the model achieved identical scores of 0.99 across all evaluated parameters: sensitivity, specificity, precision, F1-score, MCC, kappa, and AUC[21].

Equation 4 outlines the mathematical expression for accuracy, serving as a general indicator of overall model effectiveness. This metric quantifies the ratio of true classifications generated by the model relative to total predictions made [22]. It is computed by comparing the sum of correct positive and negative outcomes (TP and TN) against the complete dataset size. [23].

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \quad (4)$$

Equation 5 details the evaluation of precision, reflecting how reliably the model identifies positive outcomes. Mathematically, it represents the ratio of True Positives relative to all positive classifications made by the model, encompassing both True Positives and False Positives. Consequently, this metric highlights the proportion of instances accurately labeled as positive out of every instance categorized as positive.[24].

$$Precision = \frac{TP}{TP + FP} \quad (5)$$

Equation 6 outlines the formula to assess recall, demonstrating the sensitivity of the framework in capturing true positive cases. Recall is expressed as the proportion of True Positives compared to all existing positive samples, comprising both True Positives and False Negatives. [25].

$$Recall = \frac{TP}{TP + FN} \quad (6)$$

Equation 7 presents the mathematical expression for the F1 score, an evaluation metric that integrates precision and recall into a unified score to assess classifier effectiveness, particularly in scenarios with class imbalance. To balance these two, the F1 score serves as a harmonic average calculated between precision and recall values.

$$F1 - Score = \frac{Precision \times Recall}{Precision + Recall} \quad (7)$$

The ROC-AUC metric (Receiver Operating Characteristic - Area Under the Curve) serves as an evaluation framework for two-class models that assesses a classifier's capability to differentiate between positive and negative categories through a ROC curve visualization, which graphs the True Positive Rate against the False Positive Rate across different classification thresholds [26]. AUC scores lie between 0 and 1, where an AUC of 1 represents an optimal model, and AUC=0.5 equivalent to a random guess. The advantages of ROC-AUC are that it does not depend on a specific threshold, is suitable for comparing models, and reflects overall performance. However, it can be biased in the presence of unbalanced data. Based on ROC curve evaluations presented in the study by Fouad and Labib (2022) [27], the areas under the curves (AUC) of the applied the Area Under the Curve (AUC) metrics across the evaluated models, including KNN with N=5, demonstrated strong outcomes surpassing 0.90, indicating strong model discrimination performance. The KNN classifier, in particular, attained an accuracy reaching 92.69%, alongside precision at 86.38% and sensitivity of 100%, demonstrating its reliability in classifying drowsiness by leveraging facial indicators including ocular and oral regions. According to ROC analysis, AUC values exceed 0.9 and approach 1.0, indicating a superior degree of classifier performance when differentiating between fatigued and non-fatigued states. This suggests that the model employed possesses excellent discriminative ability across various threshold settings[28].

The K-Fold Cross-Validation technique functions as a data partitioning strategy which allows the entire dataset to be utilized effectively in both model training and testing phases. The data is partitioned into k equal subsets (folds), where k-1 subsets serve for training while the remaining fold is reserved for testing, repeated k times iteratively. This guarantees every sample is evaluated precisely once, thereby mitigating bias and variance in the evaluation process. The technique yields a more stable performance estimation than a conventional train-test split. [29].

K-Fold cross-validation reduces the dependence of model evaluation on a single data split by repeatedly training and validating the model across different subsets of the dataset. Stratified K-Fold is particularly useful when the class distribution is imbalanced because it maintains a similar proportion of each class across the folds. In this study, Stratified K-Fold cross-validation was employed to maintain consistent class proportions across the validation folds and obtain a more reliable estimate of model performance [30].

2.5. SEMMA Framework

The SEMMA methodology (Sample, Explore, Modify, Model, Assess) is adopted in this study framework exclusively to serve as a structural heuristic for organizing the data processing pipeline. While acknowledging that SEMMA is traditionally associated with business data mining, it is employed here to

provide a systematic methodology for the transition from raw image sampling to feature-based classification assessment. This ensures that the computer vision workflow remains organized and reproducible, without merging into business-centric analytics [31].

At the Sample stage, a dataset of 2,468 images from Kaggle was classified into binary categories, namely Drowsy and Not Drowsy. Subsequently, the Explore stage was conducted to analyze the data distribution and image quality. The Modify stage involves preprocessing steps, including face detection using the Haar Cascade, resizing to 224×224 pixels, augmentation, and extraction and normalization of EAR and MAR features. To avoid potential data leakage during model evaluation, data expansion techniques were strictly restricted to the training portion of every fold during a cross-validation process, preserving the original integrity of the validation subset. This strategy ensures that the validation data is not influenced by augmented samples derived therefrom. Next, the Model phase employs the K-Nearest Neighbor (KNN) classifier. Finally, the Assess stage is performed with the Confusion Matrix and ROC-AUC evaluation.

3. Results and Discussions

3.1. Sample

During the Sample phase, the data utilized for this research was acquired via the open-source repository Kaggle. The dataset consists of 2,468 images divided into four main categories: Open, Closed, Yawn, and No Yawn. These categories represent eye and mouth conditions that are commonly used as indicators of fatigue.

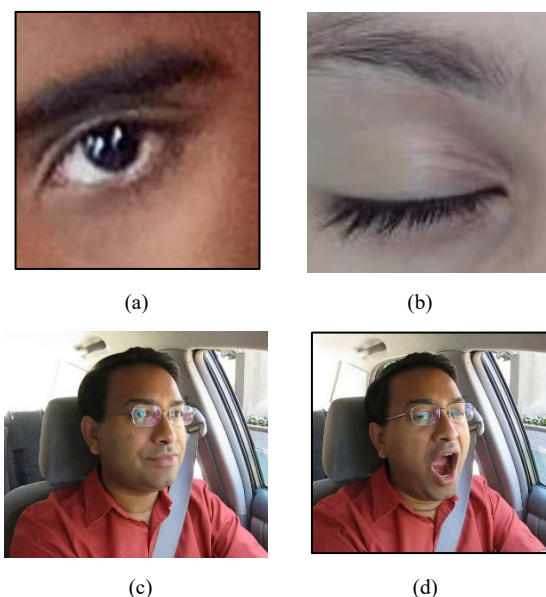


Figure 3. Data Sample

Figure 3 presents illustrations of the data categories analyzed in this research Image (a) shows the *Open* category, where the driver's eyes are open, representing an alert condition. Image (b) illustrates the *Closed*

category, where the eyes are shut, indicating drowsiness. Image (c) belongs to the *No Yawn* category, showing a driver with a neutral mouth condition, which is also considered an alert expression. Lastly, image (d) depicts the *Yawn* category, where the driver is captured while yawning, a strong indicator of fatigue.

3.2. Explore

The Explore stage involves exploring and visualizing the dataset's distribution to ensure class balance and data quality. This analysis uses graphs to detect imbalances or anomalies, such as blurry images. Findings derived from this analysis serve to guide subsequent decisions on whether data augmentation or rebalancing is needed.

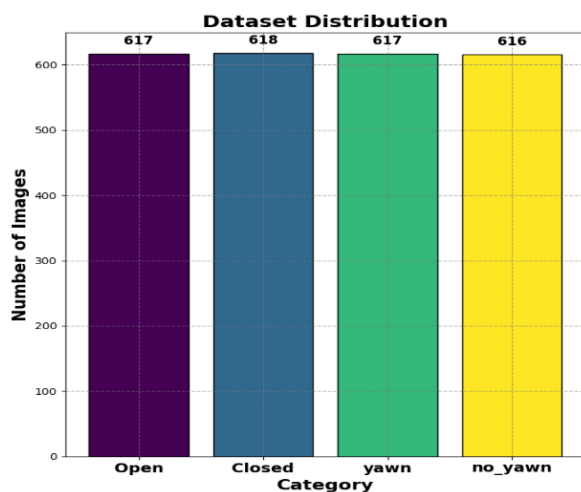


Figure 4. Dataset Distribution

As depicted in Figure 4, the data distribution comprises of four main classes: Open, Closed, Yawn, and No Yawn, with approximately 616-618 images in each category. The dataset is balanced, with no significant class imbalance, which reduces the risk of bias in the model. This supports a more accurate classification process in the next stage.

3.3. Modify

The Modify stage included preprocessing aimed at optimizing model accuracy. The Haar Cascade algorithm was employed for face detection in the Yawn and No Yawn categories. Furthermore, every image underwent resizing to 224×224 pixels, a common benchmark in feature extraction. Augmentation was performed with rotation, flipping, and contrast adjustment techniques. Two geometric features, specifically the Eye Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR), were extracted using key landmark detection and calculated by Euclidean distance, which were then normalized.

Face detection was first executed via the Haar Cascade method to localize the driver's face prior to the feature extraction phase. Figure 5 presents the results of the face detection process.

As illustrated in Figure 5, the results of the Haar Cascade face detection process are shown. Image (a) shows the original input image taken from the Yawn category. In image (b), the facial area is accurately identified by the framework, as highlighted by the green bounding box. Finally, image (c) presents the cropped version of the detected face, isolating the relevant area for subsequent feature extraction. This step demonstrates the effectiveness of Haar Cascade in accurately locating and segmenting facial regions. This process ensures that the facial features used in the next stage are the relevant parts, thereby improving the data quality for feature extraction and modeling. This accurate detection also helps to reduce noise from the image background.



Figure 5. Haar Cascade Face Detection Result

Following face detection, all images were standardized to dimensions of 224 × 224 pixels and expanded through rotational transformations, horizontal flips, and contrast adjustment to increase data diversity and improve model robustness. The augmentation process increased the number and diversity of training samples, producing a total of 3,085 images from the original 2,468-image dataset.

Figure 6 presents examples of the resized and augmented images from the four categories: (a) Closed, (b) Open, (c) No Yawn, and (d) Yawn. The augmentation process improves data diversity and strengthens the model's resilience to variations in facial orientation and illumination [32].

Following image preprocessing, geometric indicators comprising the Eye Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR) features were extracted via the identified face landmarks. These geometric features represent eye closure and mouth opening, respectively, enabling the classifier to differentiate "Drowsy" from "Not Drowsy" categories. Lower EAR values generally indicate eye closure associated with drowsiness, whereas higher MAR values represent yawning behaviour. Therefore, the combination of EAR and MAR provides complementary information for fatigue detection.

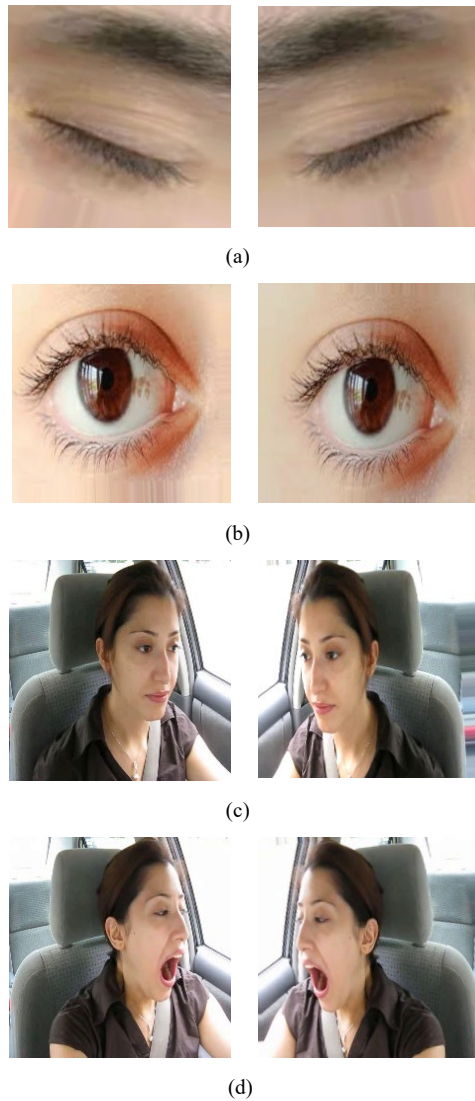
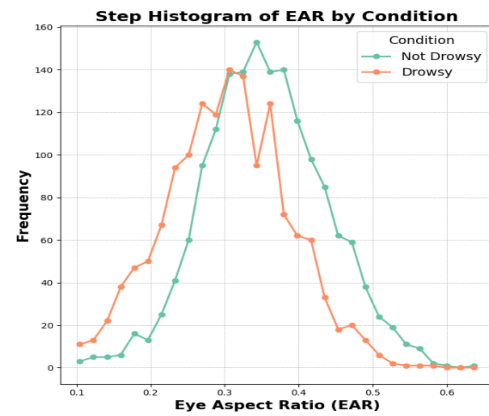


Figure 6. Resize and Augmentation Result

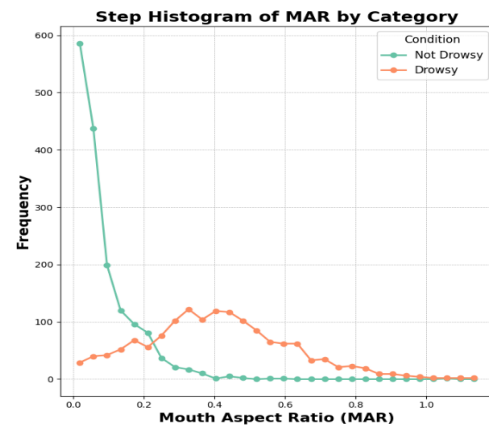
Figures 7(a) and 7(b) present the distributions of EAR and MAR for the “Drowsy” and “Not Drowsy” classes. The histograms show a clear separation between the two classes, where lower EAR values and higher MAR values are predominantly associated with drowsiness.

Figure 7(a) shows that the “Not Drowsy” class is concentrated at higher EAR values (approximately 0.30–0.45), indicating open eyes, whereas the “Drowsy” class is primarily distributed below 0.30, reflecting eye closure due to fatigue. Similarly, Figure 7(b) demonstrates that the “Not Drowsy” class is concentrated at lower MAR values, whereas the “Drowsy” class exhibits higher MAR values corresponding to yawning behaviour. Overall, these distributions confirm that EAR and MAR provide complementary geometric information for distinguishing between alert and drowsy driver conditions.

By combining both geometric features, the proposed model captures multiple visual indicators of fatigue simultaneously, improving classification robustness compared with relying on a single facial feature.



(a)



(b)

Figure 7. EAR & MAR Distribution (a) EAR distribution, (b) MAR distribution

Figure 8 presents the joint distribution of EAR and MAR values. Samples labelled “Drowsy” are mainly concentrated in regions with lower EAR and higher MAR values, whereas “Not Drowsy” samples occupy regions with higher EAR and lower MAR values. This clear separation demonstrates that the combined use of EAR and MAR provides discriminative feature representations for the subsequent KNN classification.

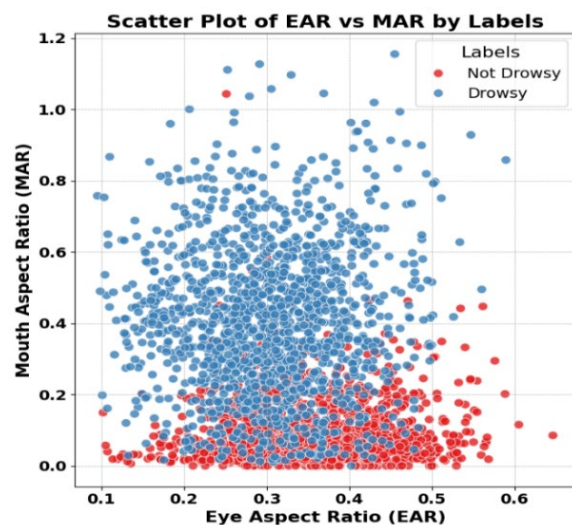


Figure 8. Visualization of EAR & MAR Features Based on Labels

3.4. Model

The Model stage evaluates the proposed K-Nearest Neighbor (KNN) classifier using Stratified 10-Fold Cross-Validation. Different values of K were tested to identify the optimal neighborhood size based on the average classification accuracy obtained across all validation folds. The assessment findings are illustrated across Figure 9 and detailed in Table 1.

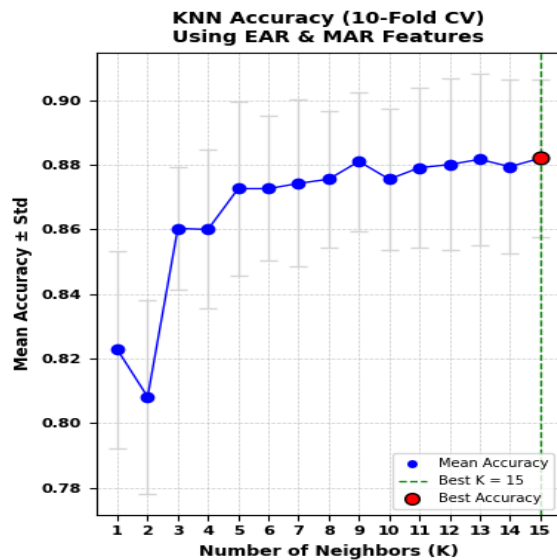


Figure 9. KNN Accuracy

Figure 9 illustrates the average classification performance achieved by the developed K-Nearest Neighbor classifier across various settings of K, evaluated via Stratified 10-Fold Cross-Validation. The graph summarizes the mean accuracy obtained across all folds during validation, with error bars representing the associated standard deviations.

Classification accuracy generally increased as the value of K increased before reaching a stable region. The highest performance was obtained at K = 15 with an average accuracy of 0.88 ± 0.02 , indicating that this neighborhood size provides an optimal trade-off between model stability and predictive generalization.

These findings demonstrate that combining EAR and MAR features with the KNN algorithm offers a highly efficient and resource-friendly lightweight approach for identifying driver fatigue. The proposed framework therefore offers potential for deployment in resource-constrained driver monitoring systems.

Table 1. Cross-Validation Results (10-Fold) with Best K

Fold	Train Set Size	Validation Set Size	Accuracy
1	2776	309	0.93
2	2776	309	0.83
3	2776	309	0.89
4	2776	309	0.87
5	2776	309	0.87
6	2777	308	0.88
7	2777	308	0.85
8	2777	308	0.89
9	2777	308	0.88
10	2777	308	0.88

Table 1 summarizes the predictive accuracy recorded across all individual iterations of the Stratified 10-Fold Cross-Validation procedure using an optimal K value. The results provide an overview regarding the stability of the classifier's performance across varied validation partitions. The obtained accuracies range from 0.83 to 0.93 across the ten validation folds, indicating relatively stable classification performance despite variations in the validation subsets. The slight differences in the training and validation set sizes are expected because the total number of samples cannot be divided equally across the ten folds. These differences do not significantly affect the evaluation results. Among the ten validation folds, Fold 1 yielded the peak performance (93.20%), whereas Fold 2 recorded the minimum accuracy (83.49%). Nevertheless, most folds produced accuracies close to the overall average of $88.2\% \pm 2.6\%$, indicating consistent classification results regardless of variations in the data splits.

This minimal fluctuation in accuracy indicates that the proposed KNN classifier generalizes well across different validation subsets. This consistency indicates that the selected EAR and MAR features provide reliable discriminative information for driver drowsiness detection and that the classifier avoids over-reliance on any individual data split.

3.5. Assess

In the Assess phase, performance evaluation for the proposed KNN classifier is conducted via a confusion matrix alongside ROC curve analysis. These evaluation methods offer a comprehensive assessment of how effectively the algorithm discriminates "Drowsy" from "Not Drowsy" categories. The corresponding outcomes are presented in Figures 10 and 11. Figure 10 displays the confusion matrix visualization for the KNN classifier based on the aggregated results of Stratified 10-Fold Cross-Validation. The model correctly classified 1,247 drowsy samples as True Positives (TP) and 1,473 non-drowsy samples as correctly identified negatives (TN), while 223 misclassified negative instances (FN) and 142 incorrectly assigned positive instances (FP) were observed.

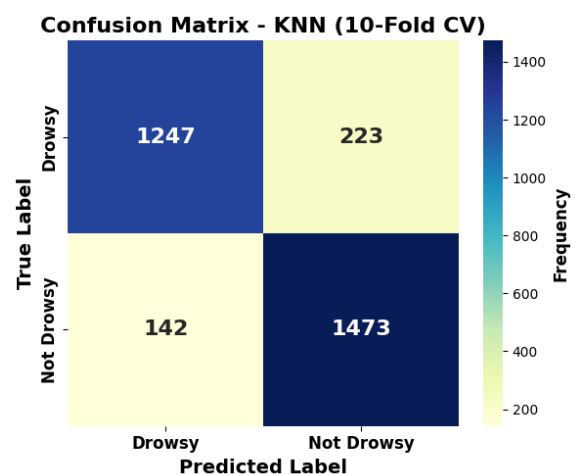


Figure 10. Confusion Matrix

The confusion matrix indicates that most samples were correctly classified, demonstrating balanced classification performance for both classes. Although the number of False Negatives (223 cases) is slightly higher than False Positives (142 cases), the overall results indicate that the proposed KNN classifier provides reliable discrimination between "Drowsy" and "Not Drowsy" conditions.

The higher number of False Negatives compared with False Positives indicates that several drowsy samples were still classified as "Not Drowsy." Despite this limitation, the variance observed between both error types is relatively small, indicating that the proposed classifier maintains balanced performance without exhibiting substantial bias toward either class.

These findings confirm that the proposed KNN classifier, combined with EAR and MAR geometric features, provides reliable performance for driver drowsiness detection. The proposed framework evidences that a computationally efficient traditional machine learning approach is capable of attaining consistent classification outcomes while remaining adaptable for integration into computationally restricted driver monitoring applications.

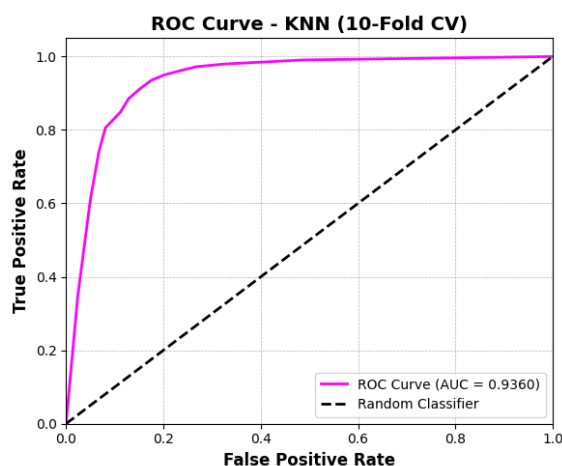


Figure 11. ROC - AUC

Figure 11 illustrates the Receiver Operating Characteristic (ROC) curve associated with the proposed KNN classifier using the aggregated results of Stratified 10-Fold Cross-Validation. The ROC analysis evaluates the classifier's capacity to discriminate "Drowsy" and "Not Drowsy" categories over various decision thresholds.

The ROC analysis produced an Area Under the Curve (AUC) value of 0.93, indicating that the proposed KNN classifier achieves strong discriminative performance in distinguishing between the "Drowsy" and "Not Drowsy" classes across different classification thresholds.

The obtained AUC value demonstrates that the combination of EAR and MAR features provides effective discriminative information for driver drowsiness detection. These complementary geometric

features enable the proposed KNN classifier to achieve reliable classification performance across different decision thresholds.

In summary, the ROC evaluation validates that the KNN classifier consistently distinguishes between the "Drowsy" and "Not Drowsy" classes across different decision thresholds. Together with the confusion matrix and cross-validation results, the AUC value of 0.93 further demonstrates the effectiveness of combining EAR and MAR features for driver drowsiness detection.

Although the proposed approach demonstrates reliable classification performance, this study does not directly evaluate computational metrics such as inference time, memory consumption, or power usage. Therefore, the proposed framework should be considered a validated lightweight classical baseline rather than a computationally superior alternative to deep learning methods.

Future work will focus on implementing the proposed framework on embedded hardware platforms and comparing its computational efficiency with lightweight deep learning models under real-time operating conditions.

Table 2 summarizes the classification performance of the proposed KNN classifier using precision, recall, F1-score, and support for each class. These metrics provide a thorough evaluation of how effectively the model can distinguish between drowsy and non-drowsy driver conditions. This suggests that, although the model accurately identifies drowsy conditions, a small number of fatigue instances may still remain undetected.

Table 2. Classification Report

Metric	Drowsy	Not Drowsy	Weighted Average
Precision	0.90	0.87	0.88
Recall	0.85	0.91	0.88
F1-score	0.87	0.89	0.88
Support	1470	1615	3085
Accuracy			0.88
ROC-AUC			0.93

The classification results indicate balanced performance across both classes. Although the "Not Drowsy" class achieved slightly higher recall than the "Drowsy" class, the differences in precision, recall, and F1-score remain relatively small, demonstrating that the proposed classifier performs consistently without significant class bias.

The overall results demonstrate stable classification performance, achieving an accuracy of 88.2% with consistent macro and weighted average scores. Together with the ROC-AUC value of 0.93, these findings indicate that the proposed classifier generalizes well across different evaluation metrics.

These findings confirm that the proposed KNN classifier, using EAR and MAR geometric features, provides reliable performance for driver drowsiness detection. Rather than claiming computational

superiority over deep learning approaches, this study establishes a lightweight classical machine learning baseline that achieved 88.2% classification accuracy with an AUC of 0.93. This validated baseline provides a meaningful reference for future studies to evaluate the trade-off between classification performance and computational complexity when developing more advanced driver monitoring systems.

3.6. Detection Result

The following section presents several examples of driver drowsiness detection using the system developed in this study. These results demonstrate the model's ability to analyze facial landmarks and assess the driver's condition in real-world scenarios. Initially, the dataset was preprocessed using the Haar Cascade method to detect facial regions, particularly for classifying images into "Yawn" and "No Yawn" categories. This step ensured that relevant facial features were accurately localized before feature extraction. Afterward, the framework calculated the geometric EAR and MAR values from the detected facial landmarks to quantify signs of fatigue, such as prolonged eye closure and mouth opening.

Using these metrics, the trained K-Nearest Neighbors (KNN) model then classified the driver's state as either "Drowsy" or "Not Drowsy." The visual examples below illustrate how the model interprets these facial dynamics and generates a classification decision. These outputs demonstrate the effectiveness of the feature-based approach and offer an intuitive view of how the system responds to various expressions related to drowsiness.

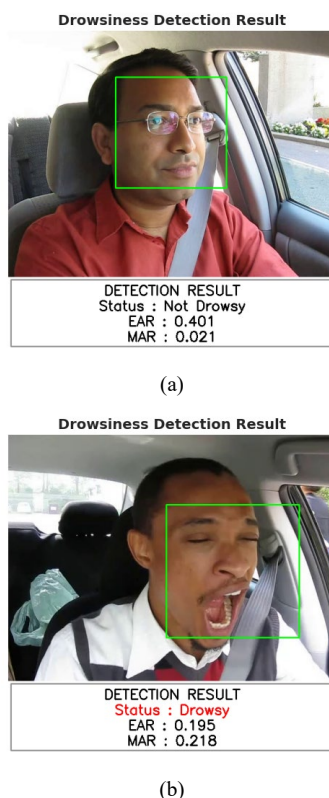


Figure 12. Detection Result (a) Not Drowsy (b) Drowsy

As illustrated in Figure 12, the detection results of the proposed system under two different driver conditions are shown using facial landmark-based classification. In both images, the green rectangular box indicates that the system detected and localized the driver's face region using Haar Cascade before feature extraction. This face detection step ensures that the Eye Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR) can be accurately computed from the cropped facial area.

In image (a), the driver's face is clearly detected despite eyeglasses, with the green box properly enclosing the face. The EAR value of 0.40 and MAR value of 0.02 indicate open eyes and a closed mouth, which leads the classifier to accurately classify the driver as Not Drowsy. In contrast, image (b) captures a driver mid-yawn, which is effectively highlighted by the system. The EAR drops to 0.19 (indicating possible eye closure) and MAR rises to 0.21 (indicating a wide mouth opening), leading to a correct classification of Drowsy. These examples validate the system's robustness in detecting varying facial expressions and support its ability to classify drowsiness status accurately in different real-world driving scenarios.

3.7. Limitations and Future Work

The scope of this study remained restricted to static facial images and did not implement real-time detection. As a result, the system may not fully capture the dynamic changes in driver behavior that occur during continuous driving. In addition, the classification framework was exclusively evaluated against a solitary dataset with predefined categories, which may limit its generalizability to real-world conditions with diverse lighting and facial orientations.

Responding to the need to strengthen the originality claim and validate practical advantages, future research must explore a robust real-time embedded evaluation using embedded systems or mobile applications to enhance practical usability. Specifically, a direct comparative study will be conducted to formally establish the technological novelty, demonstrating clear latency and power advantages over compact deep baselines (e.g., Tiny YOLO) on a dedicated resource-constrained platform. Furthermore, testing the model on more expansive and varied benchmark datasets, including video-based inputs and variations in environmental conditions, is recommended to improve system stability as well as the adaptability of the drowsiness monitoring system.

4. Conclusions

The present study introduces an efficient framework for monitoring driver drowsiness by combining the Haar Cascade algorithm for facial identification, along with Eye Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR) metrics to extract features, alongside a K-Nearest Neighbor (KNN) algorithm to handle classification. Using a dataset of 2,468 facial images divided into four categories (Open, Closed, Yawn, and

No Yawn), the model yielded a mean accuracy of $88.2\% \pm 2.6\%$ and an AUC of 0.93 under a 10-fold cross-validation setup. The low standard deviation confirms predictive consistency alongside robust generalization. These outcomes indicate that the model performs reliably with balanced precision and recall across both “Drowsy” and “Not Drowsy” classes.

The contribution of this study lies in evaluating the effectiveness of simple geometric facial features (EAR and MAR) combined with a K-Nearest Neighbor (KNN) classifier for driver drowsiness detection. The experimental results show that this lightweight approach can achieve stable classification performance, with an average accuracy of $88.2\% \pm 2.6\%$ and an AUC of 0.93 using 10-fold cross-validation.

These findings indicate that classical machine learning methods based on geometric facial features can serve as a reliable baseline for driver fatigue detection, while maintaining model simplicity and interpretability for potential implementation in resource-constrained monitoring systems.

However, the system has not yet been validated under real-world driving conditions and remains sensitive to variations in lighting and camera positioning. Future work should address these challenges by testing real driving scenarios, utilizing infrared (IR) cameras for low-light environments, and exploring deployment on edge computing platforms, such as Raspberry Pi. Overall, the proposed method contributes a robust and interpretable classical baseline for computer vision-based driver monitoring. This research formally establishes that geometric facial features, when evaluated through a rigorous cross-validation framework, provide reliable discriminative power, thereby serving as a necessary technological reference for the development of more complex fatigue detection systems.

Declarations

The authors provide the following declarations regarding the preparation of this manuscript.

AI Usage Statement

The authors used Grammarly solely to improve the manuscript's readability and language; all scientific content, analysis, and conclusions are the authors' own.

Author Contributions

Conceptualization, Asri Mulyani and Dede Kurniadi; methodology, Asri Mulyani and Dede Kurniadi; software, Asri Mulyani and Adinda Jaida Fauziyah; validation, Asri Mulyani, Dede Kurniadi, and Adinda Jaida Fauziyah; formal analysis, Asri Mulyani and Dede Kurniadi; investigation, Asri Mulyani and Dede Kurniadi; resources, Asri Mulyani, Erick Fernando, and Dede Kurniadi; data curation, Asri Mulyani and Adinda Jaida Fauziyah; writing—original draft preparation, Asri Mulyani; writing—review and editing, Asri Mulyani, Erick Fernando, Dede Kurniadi, and Adinda

Jaida Fauziyah; visualization, Asri Mulyani and Adinda Jaida Fauziyah; supervision, Dede Kurniadi; project administration, Asri Mulyani; funding acquisition, Asri Mulyani, and Erick Fernando.

Funding Statement

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors. All work was carried out using the authors' own resources.

Conflict of Interest Statement

The authors state that there are no financial or personal conflicts of interest regarding this publication. Our appreciation is also extended to Institut Teknologi Garut and Universitas Multimedia Nusantara for their unwavering support.

Ethical Clearance

The facial images analyzed in this work were sourced from an open-access repository hosted on the Kaggle platform for research purposes. The dataset was used in accordance with its applicable terms of use. This study neither gathered new human participant information nor attempted to identify any individual represented in the dataset, and no personally identifiable information was collected or disclosed.

Data and Code Availability

The dataset used in this study is publicly available on [Kaggle](#). The source code developed for this study is available from the corresponding author upon reasonable request.

References

- [1] C. Dewi, R. C. Chen, C. W. Chang, S. H. Wu, X. Jiang, and H. Yu, “Eye Aspect Ratio for Real-Time Drowsiness Detection to Improve Driver Safety,” *Electron.*, vol. 11, no. 19, 2022, doi: <https://doi.org/10.3390/electronics11193183>.
- [2] A. A. Ramadana Lubis, S. I. Purnama, and M. A. Afandi, “Haar Cascade-Based Drowsiness Detection System for Computer Vision Applications,” *Techno.Com*, vol. 22, no. 3, pp. 589–598, 2023, doi: <https://doi.org/10.33633/tc.v22i3.8425>.
- [3] P. Vishesh, S. Raghavendra, S. K. Jankatti, and V. Rekha, “Eye Blink Detection Using CNN to Detect Drowsiness Level in Drivers for Road safety,” *Indones. J. Electr. Eng. Comput. Sci.*, vol. 22, no. 1, pp. 222–231, 2021, doi: <http://doi.org/10.11591/ijeecs.v22.i1.pp222-231>.
- [4] S. Mujahidin, A. Ripaldi, B. Nugroho, and R. Paninggali, “Development of Drowsiness Detection System for Drivers using Haar Cascade Classifier and Convolutional Neural Network,” *IJCSAM (International J. Comput. Sci. Appl. Math.)*, vol. 10, no. 2, pp. 108–112, 2024, doi: <http://dx.doi.org/10.12962%2Fj24775401.v10i2.21955>.
- [5] B. Y. P. Arun Babu, G. Vidya Vyshnavi, D. Sai Kumar, S. Sameer Basha, “Detecting Human Driver Drowsiness,” vol. 13, no. 4, pp. 1077–1082, 2024, doi: <http://dx.doi.org/10.17148/IJARCCCE.2024.134159>.
- [6] A. Altameem, A. Kumar, R. C. Poonia, S. Kumar, A. Khader, and J. Saudagar, “Early Identification and Detection of Driver Drowsiness by Hybrid Machine Learning,” *IEEE Access*, vol. 9, pp. 162805–162819, 2021, doi: <https://doi.org/10.1109/ACCESS.2021.3131601>.
- [7] Y. Albadawi, A. AlRedhaei, and M. Takruri, “Real-Time Machine Learning-Based Driver Drowsiness Detection Using Visual Features,” *J. Imaging*, vol. 9, no. 5, 2023, doi:

- <https://doi.org/10.3390/jimaging9050091>.
- [8] S. Baul, M. R. Rana, and F. Bente Alam, "A Real-time Light-weight Computer Vision Application for Driver's Drowsiness Detection," *Int. J. Eng. Manuf.*, vol. 14, no. 2, pp. 23–33, 2024, doi: <https://doi.org/10.5815/ijem.2024.02.02>.
 - [9] S. Winarno, F. Alzami, M. Naufal, H. Al Azies, and M. A. Soeleman, "EfficientNet-KNN for Real-Time Driver Drowsiness Detection via Sequential Image Processing," *IJETA*, vol. 30, no. 7, pp. 1703–1713, 2025, doi: <https://doi.org/10.18280/isi.300703>.
 - [10] R. Kannan, P. Jahnavi, and M. Megha, "Driver Drowsiness Detection and Alert System based on Visual Analysis using Machine Learning," *2023 IEEE Int. Conf. Integr. Circuits Commun. Syst. ICICACS 2023*, no. April, 2023, doi: <https://doi.org/10.1109/ICICACS57338.2023.10100316>.
 - [11] I. Veritawati, F. Maspiyanti, R. Mastra, E. Fernando, and A. Murtako, "Comparison of Classification Algorithms in Bamboo Distribution Mapping for Identification of Industrial Supporting Raw Materials," *JOIV Int. J. Informatics Vis.*, vol. 9, no. 4, p. 1792, Jul. 2025, doi: <http://dx.doi.org/10.62527/joiv.9.4.3072>.
 - [12] K. Eldora, E. Fernando, and W. Winanti, "Comparative Analysis of KNN and Decision Tree Classification Algorithms for Early Stroke Prediction: A Machine Learning Approach," *J. Inf. Syst. Informatics*, vol. 6, no. 1, pp. 313–338, Mar. 2024, doi: <https://doi.org/10.51519/journalisi.v6i1.664>.
 - [13] S. Handayaningsih and A. Ardiansyah, "Missing Data Imputation using K-Nearest Neighbour for Software Project Effort Prediction," *J. Inform.*, vol. 16, no. 1, p. 19, 2022, doi: <https://doi.org/10.26555/jifo.v16i1.a23494>.
 - [14] D. Kurniadi, E. Fernando, A. Fauziyah, and A. Mulyani, "Improving Low-Light Face Recognition using DeepFace Embedding and Multi-Layer Perceptron," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 9, no. 5, pp. 1047–1055, Oct. 2025, doi: <https://doi.org/10.29207/resti.v9i5.6797>.
 - [15] D. R. Rochmawati, "Deteksi Wajah Dengan Metode Haar Cascade Menggunakan Opencv (Face Detection With Haar Cascade Method Using Opencv)," *J. Tekomin*, 2024, doi: <https://doi.org/10.59820/tekomin.v3i1.293>.
 - [16] M. N. Andrean et al., "Comparing Haar Cascade and YOLOFACE for Region of Interest Classification in Drowsiness Detection," *J. Media Inform. Budidarma*, vol. 8, pp. 272–281, 2024, doi: <https://doi.org/10.30865/mib.v8i1.7167>.
 - [17] R. Florez, F. Palomino-quispe, R. J. Coaquira-castillo, J. C. Herrera-levano, T. Paixão, and A. B. Alvarez, "A CNN-Based Approach for Driver Drowsiness Detection by Real-Time Eye State Identification," *MDPI*, 2023, doi: <https://doi.org/10.3390/app13137849>.
 - [18] W. Jia, M. Sun, J. Lian, and S. Hou, "Feature dimensionality reduction : a review," *Complex Intell. Syst.*, vol. 8, no. 3, pp. 2663–2693, 2022, doi: <https://doi.org/10.1007/s40747-021-00637-x>.
 - [19] M. Dreisig, M. H. Baccour, T. Schack, and E. Kasneci, "Driver Drowsiness Classification Based on Eye Blink and Head Movement Features Using the k-NN Algorithm," *2020 IEEE Symp. Ser. Comput. Intell. SSCI 2020*, pp. 889–896, 2020, doi: <https://doi.org/10.1109/SSCI47803.2020.9308133>.
 - [20] R. K. Dinata, H. Akbar, and N. Hasdyna, "Algoritma K-Nearest Neighbor dengan Euclidean Distance dan Manhattan Distance untuk Klasifikasi Transportasi Bus," *Ilk. J. Ilm.*, vol. 12, no. 2, pp. 104–111, 2020, doi: <https://doi.org/10.33096/ilkom.v12i2.539.104-111>.
 - [21] M. Rashid, M. Mustafa, N. Sulaiman, N. R. H. Abdullah, and R. Samad, "Random subspace K-NN based ensemble classifier for driver fatigue detection utilizing selected EEG channels," *Trait. du Signal*, vol. 38, no. 5, pp. 1259–1270, 2021, doi: <https://doi.org/10.18280/ts.380501>.
 - [22] M. Sari and E. R. Jamzuri, "Hand Sign Recognition of Indonesian Sign Language System (SIBI) Using Inception V3 Image Embedding and Random Forest," *J. RESTI*, vol. 9, no. 2, pp. 258–265, 2025, doi: <https://doi.org/10.29207/resti.v9i2.6156>.
 - [23] H. Fonda, "Klasifikasi Batik Riau Dengan Menggunakan Convolutional Neural Networks (Cnn)," *J. Ilmu Komput.*, vol. 9, no. 1, pp. 7–10, 2020, doi: <https://doi.org/10.33060/JIK/2020/Vol9.Iss1.144>.
 - [24] M. R. A. Yudianto, K. Kusriani, and H. Al Fatta, "Analisis Pengaruh Tingkat Akurasi Klasifikasi Citra Wayang dengan Algoritma Convolutional Neural Network," *J. Teknol. Inf.*, vol. 4, no. 2, pp. 182–191, 2020, doi: <https://doi.org/10.36294/jurti.v4i2.1319>.
 - [25] M. Azhari, Z. Situmorang, and R. Rosnelly, "Perbandingan Akurasi, Recall, dan Presisi Klasifikasi pada Algoritma C4.5, Random Forest, SVM dan Naive Bayes," *J. Media Inform. Budidarma*, vol. 5, no. 2, p. 640, 2021, doi: <https://doi.org/10.30865/mib.v5i2.2937>.
 - [26] T. Yang and Y. Ying, "AUC Maximization in the Era of Big Data and AI: A Survey," *ACM Comput. Surv.*, vol. 55, no. 8, 2022, doi: <https://doi.org/10.1145/3554729>.
 - [27] I. A. Fouad and F. E. Z. M. Labib, "Role of Deep Learning in Improving the Performance of Driver Fatigue Alert System," *Trait. du Signal*, vol. 39, no. 2, pp. 577–588, 2022, doi: <https://doi.org/10.18280/ts.390219>.
 - [28] L. Yu, X. Yang, H. Wei, J. Liu, and B. Li, "Driver fatigue detection using PPG signal, facial features, head postures with an LSTM model," *Heliyon*, vol. 10, no. 21, 2024, doi: <https://doi.org/10.1016/j.heliyon.2024.e39479>.
 - [29] V. Teodorescu and L. Obreja Braşoveanu, "Assessing the Validity of k-Fold Cross-Validation for Model Selection: Evidence from Bankruptcy Prediction Using Random Forest and XGBoost," *Computation*, vol. 13, no. 5, 2025, doi: <https://doi.org/10.3390/computation13050127>.
 - [30] S. Widodo, H. Brawijaya, U. Bina, S. Informatika, and U. N. Mandiri, "Stratified K-fold cross validation optimization on machine learning for prediction," *Sinkron*, vol. 6, no. 4, pp. 2407–2414, 2022, doi: <https://doi.org/10.33395/sinkron.v7i4.11792>.
 - [31] K. A. Zweig, *Big Data, Data Mining, and Machine Learning Value Creation for Business Leaders and Practitioners*. 2022. doi: <https://doi.org/10.7551/mitpress/13915.003.0010>.
 - [32] M. A. Saleem et al., "Enhancing stroke risk prediction through class balancing and data augmentation with CBDA-," pp. 1–19, 2025, doi: <https://doi.org/10.1038/s41598-025-07350-6>.