A Comparative Evaluation of Driver Drowsiness Detection Techniques

Wedad M. Albalawi
Computer Science and Artificial Intelligence Department, University of Jeddah, Jeddah, Saudi Arabia
2200453@uj.edu.sa (corresponding author)

Sahar Sabbeh
Information Systems and Technology Department, University of Jeddah, Jeddah, Saudi Arabia
sfsabbeh@uj.edu.sa

Nuha Zamzami
Computer Science and Artificial Intelligence Department, University of Jeddah, Jeddah, Saudi Arabia
nezamzami@uj.edu.sa

Received: 26 January 2026 | Revised: 21 February 2026 and 9 March 2026 | Accepted: 13 March 2026

 DOI: https://doi.org/10.48084/etasr.17766

ABSTRACT

Drowsy driving is a leading cause of fatal road accidents, underscoring the need for accurate, efficient automated detection systems. Despite significant breakthroughs in deep learning approaches, there is still no clear consensus in the scientific community on whether end-to-end deep learning models or hybrid approaches, which decouple feature extraction from classification, offer more efficient performance. The present study aims to fill this gap by providing a comparative analysis of multiple algorithmic configurations, considering combinations of deep pre-trained feature extractors, such as Visual Geometry Group-16 (VGG16), Mobile Networks (MobileNet), and Extreme Inception (Xception), and traditional feature extractors, such as Histogram of Oriented Gradients (HOG), Local Binary Patterns (LBP), Scale-Invariant Feature Transform (SIFT), and Gabor filters, with a wide variety of classifiers, such as machine learning-based classifiers like Support Vector Machines (SVMs), Random Forest (RF), and Naïve Bayes (NB), and deep learning-based classifiers like LeNet, VGG16, AlexNet, and Convolutional Neural Networks (CNNs). All experiments were conducted on three benchmark datasets: National Tsing Hua University-Driver Drowsiness Detection (NTHU-DDD), Yawning Detection Dataset (YAWDD), and Yawn and Eye Close. The experimental results reveal two main findings. First, hybrid architectures consistently outperform fully end-to-end deep models. For example, the VGG16 network combined with a LeNet classifier achieved 99.32% accuracy on the controlled YAWDD dataset and 98.69% on the real-world NTHU-DDD dataset, whereas MobileNet combined with SVM achieved 99.31% and 98.83%, respectively. Second, incorporating multimodal features (eye and mouth regions) significantly improves performance compared to using either modality alone, highlighting the importance of feature complementarity in drowsiness detection.

Keywords-CNN; MobileNet; Xception; machine learning; feature extraction; driver drowsiness; deep learning

I. INTRODUCTION

Despite advancements in vehicle safety systems and road infrastructure, the number of fatal traffic accidents remains alarmingly high, with several of them being caused by sleeping drivers. It is estimated that more than 40,000 individuals are injured annually in approximately 56,000 sleep-related accidents in the United States alone [1]. Additionally, numerous studies, including those conducted by the United Kingdom's Center for Sleep Research, have highlighted the significant impact of sleep deprivation on driving performance and traffic accidents. In addition to fatigue, driver distraction and inattention pose major contributing factors to road accidents [2]. These risks spurred researchers to develop methods for the early detection of driver drowsiness. Drowsiness (or sleepiness), defined as the transition state between wakefulness and sleep, impairs cognitive and motor responses, reducing a driver's ability to react promptly. Common indicators include frequent yawning, difficulty maintaining alertness, head nodding, and lane deviation [3]. The various detection approaches that have been proposed are broadly categorized into i) camera-based methods, which analyze facial and eye movement patterns; ii) vehicle-based methods, which monitor driving behavior; iii) physiological measurements capturing drivers' bio-signals via sensors; and iv) hybrid approaches, which combine multiple modalities [4].

These data sources form the basis for a wide range of methodologies that employ i) traditional feature extraction with machine learning, ii) deep learning techniques, or iii) hybrid frameworks integrating both machine learning and deep learning approaches.

A. Traditional Feature Extraction with Machine Learning

Research has explored the use of machine learning classifiers and traditional feature extraction methods for detecting driver fatigue. Authors in [5] demonstrated that Histogram of Oriented Gradients (HOG) outperforms several traditional feature extraction techniques, including Canny edge detection, Local Binary Patterns (LBP), Gabor filters, and standard grayscale-based features. Authors in [6] proposed a hybrid approach that integrates an improved HOG descriptor with a Naïve Bayes (NB) classifier, achieving enhanced performance. Authors in [7] focused on extracting eye and mouth features using HOG, followed by classification with Support Vector Machines (SVMs). Furthermore, authors in [8] introduced a non-intrusive system that combines HOG-based region-of-interest detection with an ensemble learning framework incorporating stacked deep Convolutional Neural Networks (CNNs), Random Forest (RF), CNNs, and linear SVM classifiers. Authors in [9] investigated facial feature detection and extraction using a Unified Local Binary Pattern (ULBP), followed by SVM-based classification, while authors in [10] employed similar methodologies with comparable outcomes. Although these traditional methods are computationally light and efficient, they often lack generalization across diverse driving scenarios with varying lighting conditions and backgrounds.

B. Deep Learning

Deep learning techniques have also been adopted for identifying driver sleepiness. Authors in [11] used the Viola-Jones face identification technique in combination with a stacked deep CNN structure, while authors in [12] developed an alternative method that improved recognition stability and accuracy using a factorized bilinear feature fusion model, along with CNNs. Additionally, authors in [13] demonstrated that combining the outputs of deep CNN models, including AlexNet, Visual Geometry Group (VGG)-FaceNet, FlowImageNet, and Residual Network (ResNet), results in high accuracy scores. Authors in [14] developed a method based on a 4D CNN model that outperformed previous algorithms in predicting the driver's eye condition, while authors in [15] introduced a non-invasive approach that detects driver tiredness through facial recognition, examining iris size with Mobile Networks V3 (MobileNetV3), and employing Long Short-Term Memory (LSTM) networks for classification. Authors in [16] used Multi-Task Cascaded Convolutional Networks (MTCNN) for face detection and introduced the Eye Aspect Ratio (EAR) for eye analysis, followed by SVM classification. Meanwhile, recent research shows an inclination toward more intricate architectures to address temporal dependencies. The SAFE-DRIVE-AI framework in [17] incorporates a hybrid model combining CNN, LSTM, and an attention mechanism, effectively addressing the temporal aspect of drowsiness by focusing on critical frames in a sequence. However, despite the promising results of deep learning approaches, these methodologies require high computational cost, along with high memory requirements and large data demands, which hinder their deployment in resource-constrained vehicular edge devices.

C. Combining Machine Learning and Deep Learning

In order to mitigate the individual drawbacks of machine learning and deep learning approaches, several studies have opted for combined approaches for the detection of driver fatigue. Authors in [18] proposed and compared two approaches: a transfer learning approach using fine-tuned AlexNet and a hybrid approach combining AlexNet as a feature extractor with SVM classification, with the latter achieving higher accuracy. Furthermore, authors in [19] presented a pre-trained You Only Look Once (YOLO) model for face detection and a pre-trained ResNet-50 model for deep feature extraction, followed by a hybrid feature selection method based on feature clustering, with the resulting feature set fed into an SVM classifier. Authors in [20] used Deep Cascade Convolutional Neural Networks (DCCNN) to detect the face region and focus on the eyes, introducing the EAR parameter for drowsiness evaluation and classification using SVM. Finally, authors in [21] combined Scale-Invariant Feature Transform (SIFT), Fast Library for Approximate Nearest Neighbors (FLANN), and CNN features, achieving higher accuracy than standard CNNs while enabling fast detection within milliseconds.

Although these hybrid models demonstrate strong potential in effectively balancing between computation speed and accuracy, most of the studies investigate only a limited set of feature-classifier pairs. At the same time, few standardized studies on driver drowsiness detection systematically investigate optimal feature extraction and classification techniques and their combinations. Therefore, there is a scientific need for a comprehensive study that systematically compares and identifies the most effective feature extraction method, classification method, and combination techniques, as well as the most influential features.

This study provides a comparative analysis of traditional and deep feature extraction methods using both machine learning and deep learning classifiers for driver drowsiness detection. Unlike previous studies that evaluate limited configurations, 23 feature-classifier combinations are compared under identical experimental conditions across multiple public benchmark datasets, including both controlled and real-world driving scenarios. Furthermore, this work empirically demonstrates that combining deep feature extraction with lightweight classifiers, such as SVM and LeNet, achieves higher accuracy than end-to-end deep learning models. In addition, the effect of multimodal feature fusion is analyzed by comparing eye-based, mouth-based, and combined eye-mouth features, showing consistent improvement with feature fusion.

II. METHODOLOGY

The driver fatigue classification system proposed in this study is divided into two phases, training and testing, as shown in Figure 1. Images from the dataset are fed into the system during the training phase for preprocessing. In addition, several traditional and deep feature extraction techniques are applied to the preprocessed data. The resulting feature vectors are then

used with multiple machine learning and deep learning classification techniques. The resulting models are subsequently used in the testing phase to identify driver fatigue. The goal of this study is to determine the most effective combination of extraction and classification methods (Figure 2) by applying the proposed strategy and evaluating the results according to specific criteria.

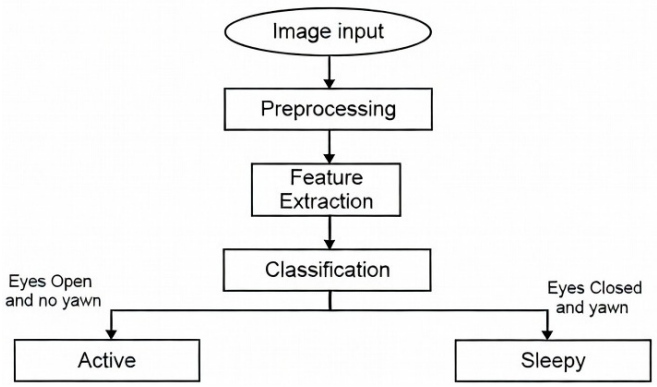

Fig. 1. The proposed methodology.

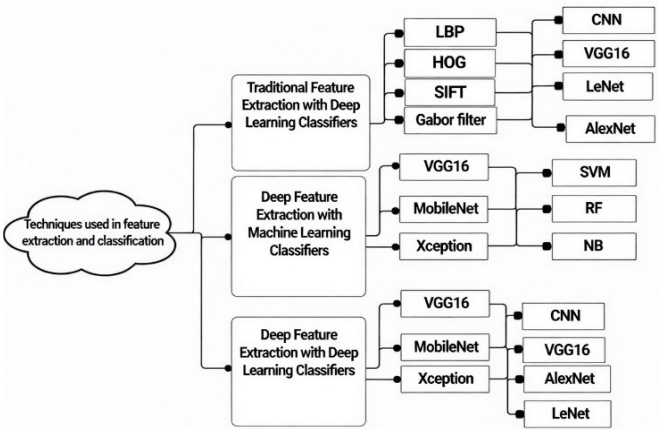

Fig. 2. List of techniques used in feature extraction and classification.

A. Dataset

Three driver fatigue datasets were used in this study: the Drowsiness Dataset, the Yawn and Eye Close dataset, and the National Tsing Hua University-Driver Drowsiness Detection (NTHU-DDD) dataset.

1) Drowsiness Dataset

The first dataset used in this work is the publicly available Drowsiness Dataset from Kaggle [22], which contains both eye and mouth images for drowsiness detection. Specifically, the Eye1 subset, which was used, consists of 726 closed-eye and 726 open-eye images utilized in all experiments related to eye features, split into 1,234 for training and 218 for testing. The Yawning Detection Dataset (YAWDD), containing 725 no-yawn and 723 yawn images originally derived from the YAWDD dataset [23], is partitioned into 1,233 for training and 215 for testing. For multimodal experiments using both eye and mouth features, the Combined Eye1 and YAWDD dataset was employed. In this way, a unified naming convention was applied across all experiments and analyses, with the data source remaining the Drowsiness Dataset from Kaggle.

2) NTHU-DDD Dataset

The efficiency of the proposed models was tested on the publicly available NTHU-DDD dataset [24], which consists of videos recorded in a simulated environment with male and female participants of various appearances and races. The dataset captures a range of driving behaviors, including yawning, gesturing, looking away, chatting, laughing, having tired eyes, being drowsy, and remaining still. It also includes different conditions such as BareFace, Glasses, Sunglasses, Night-BareFace, and Night-Glasses. In addition, the dataset provides annotations at the framework level for sleepiness, mouth movement, and eye movement. The NTHU-DDD dataset consists of 356 training videos across 18 subjects and 20 test videos from 4 subjects. After frame extraction and filtering, the Yaw subset contains 3,364 images, 2,354 for training and 1,010 for testing, the Eye subset contains 7,757 images, 5,429 for training and 2,328 for testing, and the Combined Yaw+Eye subset contains 9,940 images, 7,952 for training and 1,988 for testing.

3) Yawn and Eye Close Dataset

The Yawn and Eye Close dataset [25] consists of labeled images capturing yawn and eye closure events of diverse participants with different facial expressions, poses, and lighting conditions. Each image is labeled to indicate the presence of a yawn or closed eyes, enabling supervised learning for fatigue detection, alertness monitoring, and facial behavior analysis. The dataset is organized to support model training, validation, and testing, and is hosted on Roboflow Universe, ensuring accessibility and reproducibility. For this study, a total of 3,180 images were used, divided into 2,226 for training and 954 for testing. Table I lists the abbreviated names used to describe the subsets utilized in the analysis.

TABLE I. DESCRIPTION OF THE DATASETS AND FEATURE MODALITIES

Abbreviation	Description
M-YAWDD	Mouth-only features from YAWDD dataset.
EYE1	Eye-only features from Drowsiness dataset.
E+M-YAWDD+E1	Eye and Mouth features from Drowsiness dataset.
M-NTHU	Mouth-only features from NTHU-DDD dataset.
E-NTHU	Eye-only features from NTHU-DDD dataset.
E+M-NTHU	Eye and mouth features from NTHU-DDD dataset.

B. Preprocessing

Each of the images used underwent a preprocessing pipeline that removed noise and unnecessary regions from them, normalized brightness, and resized all images to a standardized format before classification [26, 27]. Specifically, in the case of integrating conventional feature extraction with deep learning classifiers, the images were resized to 150 × 150 and transformed into grayscale channels (150, 150, 1). In all other cases, involving deep learning techniques, the images were resized to 224 × 224 and transformed into Red Green Blue (RGB) channels (224, 224, 3). The pixel value range for all images was normalized to [0, 1].

C. Data Augmentation

To improve model robustness and generalization, extensive data augmentation was performed using Keras ImageDataGenerator. The applied transformations included rotation up to 25°, horizontal and vertical shifts of 10%, shear, zoom up to 20%, and horizontal flipping with nearest-fill mode. Three augmented images were generated for each original image, significantly expanding the dataset size.

D. Feature Extraction Technique

To effectively monitor driver fatigue, appropriate feature extraction methods are required to retrieve relevant information from input images or sensor data. Several methodologies were employed for the purposes of this study, including both traditional and deep feature extraction methods.

1) Traditional Feature Extraction

Traditional feature extraction techniques provide valuable information about the characteristics of input data by extracting features related to key points, gradients, textures, and shapes, which allow subsequent classification algorithms to make more accurate predictions [22]. The feature extraction methods used included HOG, LBP, SIFT, and Gabor filters.

a) Histogram of Oriented Gradients

HOG is widely used as a feature descriptor in applications where object shape is important. Specifically, it is utilized in computer vision for object detection by counting occurrences of gradient orientations in localized regions of an image. To achieve this, the input image is divided into small sections known as cells, and the histogram provides a bar chart of gradient directions [28]. For this method, standardization is crucial for improving performance and stability in varying illumination conditions among different images.

b) Local Binary Pattern

LBP is a texture-based feature extraction technique that computes local patterns using the values of the surrounding pixels. Due to its computational efficiency and stability while the image's gray levels are monotonous, LBP is widely deployed as a feature descriptor. The basic LBP operator evaluates a 3 × 3 neighborhood by applying a threshold and determines the center element's value by converting the adjacent values into decimals [29].

c) Scale-Invariant Feature Transform

SIFT, introduced in 2004, is a commonly used feature detection and description algorithm [30]. This method employs a Difference of Gaussian (DoG) operator as an approximation of the Laplacian of Gaussian (LoG), while the maxima within the local DoG are utilized to detect feature points in subject images at different scales. Feature description is performed by analyzing a 16 × 16 region around each identified feature, divided into smaller blocks, resulting in 128 bin values. Despite SIFT's resistance to image rotations, scale changes, and minor affine variations, its main drawback is its high computational cost.

d) Gabor Filters

Gabor filters are typically deployed in computer vision for feature extraction, particularly in texture analysis [31]. They act as linear filters that examine specific frequency and orientation information within a small area surrounding the analyzed area of the image. This operation can be described in:

$$G(x,y) = \frac{1}{2\pi s_x s_y} exp\left[-\frac{1}{2}\left(\frac{x'^2}{s_x^2} + \frac{y'^2}{s_y^2}\right)\right] exp\left[j\frac{2\pi \bar{x}}{\lambda}\right] \quad (1)$$

$$x' = cos(\theta)\cdot x + sin(\theta)\cdot y$$

$$y' = -sin(\theta)\cdot x + cos(\theta)\cdot y \quad (2)$$

Equations (1, 2) describe the spatial and frequency characteristics of the filter, where (x, y) represents the pixel location, θ defines the orientation, and s_x and s_y denote the standard deviations along the x and y axes.

2) Deep Feature Extraction

Along with the traditional feature extraction methods, various modern CNN models are utilized in this research as feature extractors. These included pretrained models, including MobileNet, VGG16, and Extreme Inception (Xception), available in the Keras library, trained on an ImageNet dataset comprising around 1.28 million images from 1,000 categories. For each extractor, the pretrained weights were used without further training.

a) MobileNet Feature Extraction

MobileNet is based on depthwise separable convolutions, which reduce computational cost while maintaining accuracy [32], using lightweight convolutions to process intermediate feature representations efficiently [33].

b) VGG16 Feature Extraction

VGG16, developed by the VGG for the ImageNet Challenge 2014, is composed of 13 convolutional layers, 5 pooling layers, and 3 fully connected layers followed by a SoftMax layer [34]. It is widely employed due to its strong performance in classification tasks.

c) Xception Feature Extraction

Xception operates by replacing standard inception modules with depthwise separable convolutions [35]. Specifically, it combines depthwise convolution with pointwise convolution (1 × 1) to efficiently map feature representations across channels.

E. Classification Techniques

Driver fatigue detection in this study is performed using both machine learning and deep learning classifiers in combination with different feature extraction methods.

1) Machine Learning Classification Algorithms

The machine learning classifiers employed were SVM, RF, and NB.

a) Support Vector Machine Classification

SVM, developed by CORINNA Cortes and VAPNIK, identifies the optimal hyperplane for classification by maximizing the separation between classes, with greater separation improving model generalization and reducing classification error. By mapping nonlinear data into a higher-dimensional space, SVM effectively handles nonlinear classification problems [36] and has been utilized for driver drowsiness detection with positive outcomes [7, 9, 19].

b) Random Forest Classification

The RF model is the most frequently used algorithm among ensemble methods. RF operates by constructing multiple decision trees using independently sampled subsets of the data, and each tree contributes to the final prediction through voting. Its main advantage is that, with large datasets, increased feature diversity results in fewer errors and lower overfitting. RF has also been extensively used for driver drowsiness detection [8, 37, 38].

c) Naïve Bayes Classification

NB is a probabilistic classification algorithm based on Bayes' theorem. It assigns a class label by calculating the probability of each category given the observed data and selecting the highest probability. Although typically employed in text analysis, opinion mining, and medical diagnosis, it is also applied in machine learning classification tasks. In recent years, it has been used for driver drowsiness detection [6, 39].

2) Deep Learning Classification Algorithms

Deep learning classification in this study is performed using CNN, LeNet, AlexNet, and VGG16. To address overfitting, several techniques were applied, including kernel regularizer = l2(0.01) in all Conv2D layers, BatchNormalization after each Conv2D layer, and Dropout after MaxPooling layers. These measures were applied only to individual datasets (sub-datasets: M-YAWDD, EYE1, Yawn and Eye Close Dataset, M-NTHU, and E-NTHU).

a) Convolutional Neural Network Classification

The CNN model includes three convolutional layers with 6, 16, and 64 filters, followed by an output layer with 4 nodes for datasets with 4 classes and 2 nodes for datasets with 2 classes. Each convolutional layer is followed by a max pooling layer with a 2 × 2 window to reduce feature map size and input dimensionality. The network is converted into a fully connected neural network through a flatten layer. All layers use Rectified Linear Unit (ReLU) activation, except the final layer, which utilizes SoftMax activation. Dropout rates between 0.4 and 0.6 were applied when using deep feature extractors across all datasets. With traditional feature extraction techniques, dropout was set to 0.5 only for the eye and mouth datasets, without batch normalization and with HOG.

b) LeNet Classification

The LeNet architecture, proposed in [40], consists of two convolutional layers with 20 and 50 5 × 5 filters, both using the same padding and having equal spacing. These layers are followed by ReLU activation and 2 × 2 max pooling. The resulting feature maps are flattened and passed to a fully connected layer with 500 units employing ReLU activation, followed by a SoftMax output layer. Dropout rates between 0.5 and 0.7 were applied depending on the dataset and feature extraction method. Network parameters were also adjusted when deep feature extractors were used to improve performance.

c) AlexNet Classification

The AlexNet-based model includes three layers of convolutions with 6, 16, and 64 filters, plus an output layer [41]. Each convolutional layer is followed by a max pooling layer with a 2 × 2 window and a stride to reduce the number of feature maps and decrease input dimensionality, while the network is converted into a fully connected neural network through a flatten layer. All layers use ReLU activation, except the final SoftMax layer. Dropout was applied at varying rates depending on the dataset and feature extraction method, with MaxPooling parameters adjusted when combined with deep feature extractors to improve generalization and reduce overfitting.

d) VGG16 Classification

The VGG16 architecture consists of five convolutional blocks followed by three fully connected layers. Each convolution is followed by a ReLU activation, and each block ends with a max pooling operation to reduce spatial dimensions. A key limitation of VGG16 is its high computational cost and memory requirements, with approximately 138 million parameters. Dropout was applied at different rates depending on the dataset and feature extraction method. Additional regularization techniques, including L2 regularization and batch normalization, were also used in some experiments to improve generalization and reduce overfitting. The detailed hyperparameter configurations for both feature extraction and classification algorithms are provided in Table II.

TABLE II. PARAMETER SETTINGS FOR THE PROPOSED METHODOLOGIES

Methodology	Hyperparameters and configurations
Feature extraction (traditional)	
HOG	Orientations: 9; pixels/cell: (8, 8); block: (2, 2); resized: (128, 64)
LBP	Kernel/neighborhood size = 3×3
SIFT	OpenCV defaults; min-max norm (0-255)
Gabor Filters	Kernel :11×11; σ= 1.5; γ= 1.2; λ= 3; 6 angles
Feature extraction (deep learning)	
MobileNet	Weights: ImageNet; α: 0.50; layer: [-5] output
VGG16	Weights: ImageNet; layer: [-5] output
Xception	Weights: ImageNet; Pooling: GlobalAvg2D; batch size: 32
Machine learning classifiers	
NB	var_smoothing: [1e-9, 1e-8, 1e-7]
SVM	C: [0.1, 1, 10]; kernel: [linear, rbf]; gamma: [scale, auto]
RF	Trees: [100, 200, 300]; depth: [none, 5, 10]; splits: [2, 5, 10]
Deep learning classifiers	
VGG16	Opt: Adam; LR: 1e-5; loss: Categorical Cross entropy; L2 Reg: 0.01; dropout: Up to 0.7
AlexNet	Opt: Adam; LR: 1e-5; loss: Categorical Cross entropy; L2 Reg: 0.01; dropout: Up to 0.7
LeNet	Opt: Adam; LR: 1e-5; loss: Categorical Cross entropy; L2 Reg: 0.01; dropout: Up to 0.7
CNN	Opt: Adam; LR: 1e-5; loss: Categorical Cross entropy; L2 Reg: 0.01; dropout: Up to 0.6

III. RESULTS AND DISCUSSION

A. Experimental Results

The present study conducted empirical evaluations of various combinations of feature extractors and classification algorithms for driver drowsiness detection to assess their performance using the accuracy metric [42, 43]. Three main experimental categories were evaluated:

- Deep feature extraction with traditional machine learning classifiers.
- Deep feature extraction with deep learning classifiers.
- Traditional feature extraction with deep learning classifiers.

1) Results of Combining Deep Feature Extraction Techniques with Machine Learning Classifiers

In the first case, deep CNNs (MobileNet, VGG16, and Xception) were used solely as feature extractors, while SVM, RF, and NB served as classifiers. The performance heatmap in Figure 3 shows that SVM consistently achieved the highest accuracy, ranging from 0.87 to 0.99 across most datasets, whereas RF showed slightly lower performance, ranging from 0.86 to 0.99, and NB exhibited even lower performance, ranging from 0.66 to 0.96. Meanwhile, despite the combination of Xception with RF on the NTHU-DDD dataset not being feasible due to memory and session limitations in Google Colab Pro, Xception was successfully combined with SVM and NB for all subsets, and RF for the remaining subsets.

The strong performance of SVM reflects the linear separability of the features generated by deep feature extraction techniques, allowing effective classification through margin maximization. RF achieved competitive results, while NB proved unsuitable for the classification problem using deep extracted features. The reason for the low performance of NB is its assumption of feature independence, which contradicts the fact that for image-based tasks, the pixels and the feature maps tend to be highly correlated.

Another important observation is that the combinations used on the NTHU-DDD datasets exhibited lower accuracies, due to being tested on realistic conditions (illumination changes, head rotations, and occlusions). In contrast, when combinations were tested on datasets combining eye and mouth features, they displayed improved performance, with many configurations achieving near-perfect accuracy (~0.99), highlighting the benefit of multi-feature analysis.

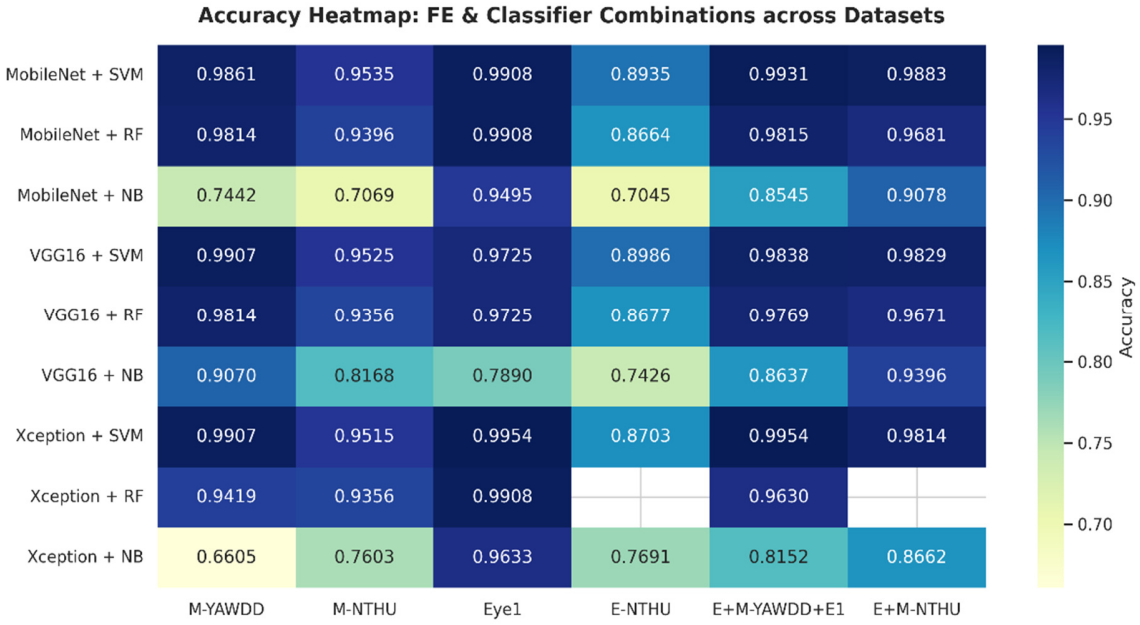

Fig. 3. Accuracy heatmap of combining deep feature extraction techniques with machine learning classifiers.

2) Deep Feature Extraction with Deep Learning Classifiers

In the second case, pretrained convolutional networks (MobileNet, VGG16, and Xception) were utilized as deep feature extractors, followed by four deep learning classifiers (CNN, VGG16, LeNet, and AlexNet). The accuracy heatmap in Figure 4 demonstrates that the VGG16 + LeNet combination achieved the highest performance with accuracies ranging between 0.90 and 0.99 across all datasets. The high performance of this combination can be attributed to the fact that VGG16 provides a dense and general feature representation, while LeNet prevents overfitting by focusing on decision separation rather than further feature learning. In contrast, combinations using MobileNet and Xception achieved slightly lower accuracies (0.79-0.99); meanwhile, the baseline models, trained without feature extraction or augmentation, yielded substantially lower accuracy (0.65-0.99), confirming the importance of transfer learning.

Moreover, performance variations among datasets were also observed. Controlled datasets, such as YAWDD and Eye Dataset 1, achieved near-perfect accuracy, greater than 0.99, whereas the NTHU-DDD dataset showed lower accuracy, typically between 0.61 and 0.96. This observation constitutes another instance where combining eye and mouth features consistently improved performance across all models.

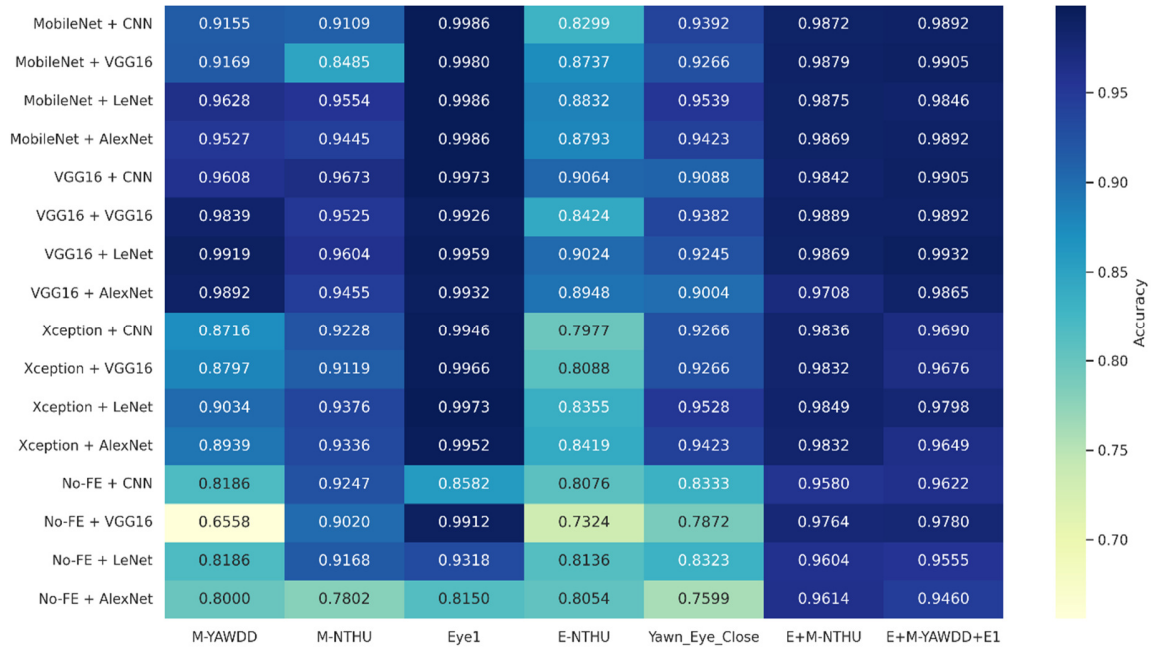

Fig. 4. Accuracy heatmap of combining deep feature extraction techniques with deep learning classifiers.

3) Results of Combining Traditional Feature Techniques with Deep Learning Classifiers

In the last case, classical computer-vision techniques, including Gabor filters, HOG, SIFT, and LBP, were combined with deep learning classifiers, including CNN, VGG16, LeNet, and AlexNet. The performance heatmap in Figure 5 shows that the best-performing combinations across datasets were LBP with CNN and SIFT with CNN, with accuracies ranging from 0.80 to 0.99, while combinations employing Gabor or HOG features showed slightly lower performance, with accuracy ranging from 0.79 to 0.98. Furthermore, deep learning models trained directly on raw inputs ("Alone"), such as CNN or VGG16, also achieved high accuracies (~0.99).

B. Discussion

A comprehensive comparison across all three experimental cases reveals several key findings/that:

- **Effect of combining features**: Combining eye and mouth features improves detection accuracy and stability under varying conditions. Despite the application of more aggressive regularization techniques to prevent overfitting, such as the use of L2 regularization, batch normalization, high dropout rates, and aggressive data augmentation, these measures inevitably constrained the models' ability to learn sufficiently discriminative representations from isolated features. In contrast, the combination of eye and mouth information offered complementary information to better and more stably represent the characteristics of driver drowsiness, even with less aggressive regularization techniques. Figure 6 presents a comparison between combined and individual feature usage.
- **Dataset characteristics:** Controlled datasets, such as YAWDD and Eye Dataset 1, consistently yielded near-perfect accuracy, whereas the NTHU-DDD dataset exhibited reduced accuracy, typically between 0.85 and 0.90, due to variations in lighting, head pose, and occlusion.
- **Model complexity:** The results indicate that complex deep learning architectures are not necessary when a pretrained deep feature extractor is used. Lightweight classifiers, such as LeNet and SVM, achieve comparable or better performance at lower computational cost.
- **Best-performing model:** The highest accuracy values, exceeding 99%, were achieved by hybrid approaches, particularly VGG16 combined with SVM. This is attributed to the ability of SVM to effectively separate the feature space generated by VGG16. The results also confirm that combining eye and mouth features significantly improves reliability compared to single-feature approaches, particularly in cases of partial occlusion. Statistical analysis further confirms that SVM outperforms deep classifiers ($p < 0.05$).

In order to further test the robustness of the proposed hybrid model, VGG16 + SVM, and ensure that the model does not possess class bias, a confusion matrix was created on the NTHU-DDD dataset, which consists of 1,988 images. As shown in Figure 7, the model correctly classified 1,954 images, with 34 misclassifications. Specifically, there were 975 True Positive (TP) cases, 979 True Negative (TN) cases, 15 False Negative (FN) cases, and 19 False Positive (FP) cases. An additional in-depth statistical analysis was conducted to validate the observed performance differences between algorithms, as depicted in Figure 8. Among feature extractors, VGG16 achieved the highest mean accuracy (0.9430), followed by MobileNet (0.9333) and Xception (0.9213). The paired t-test between VGG16 and MobileNet resulted in $p = 0.164$, indicating no statistically significant difference, although VGG16 showed slightly better stability.

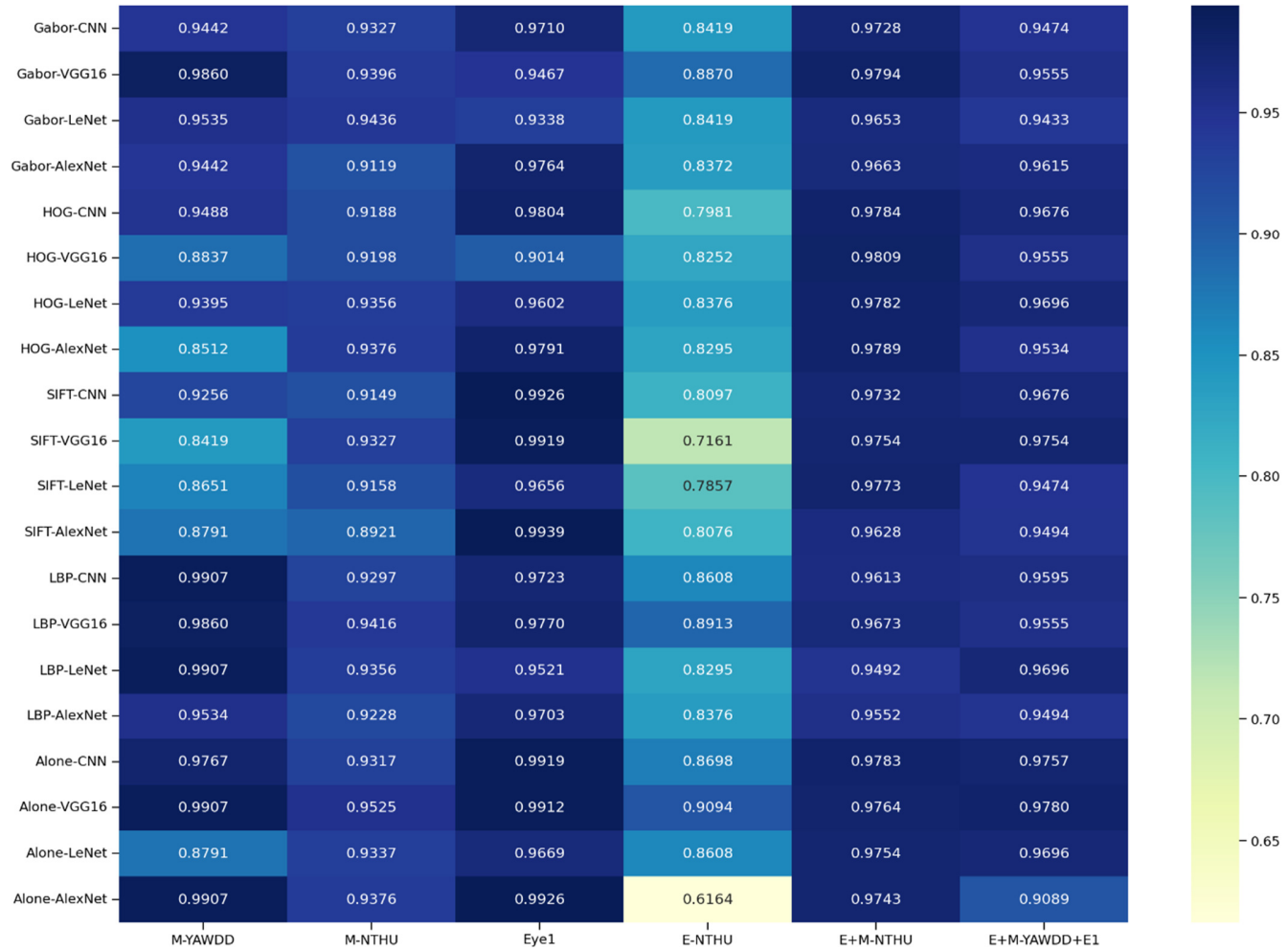

Fig. 5. Accuracy heatmap of combining deep feature extraction techniques with deep learning classifiers.

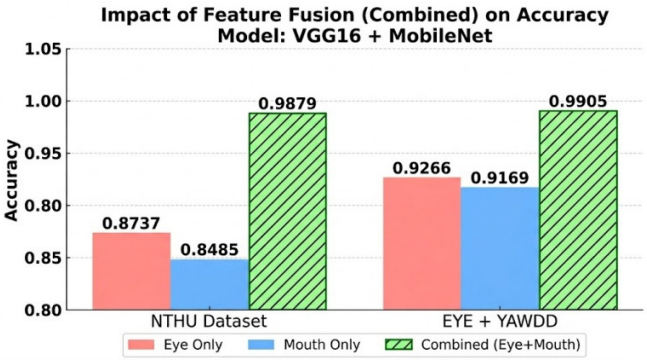

Fig. 6. The impact of merging features versus using a single feature.

In addition, traditional extractors, such as LBP and Gabor filters, demonstrated comparable performance ($p = 0.146$), with LBP achieving a slightly higher mean accuracy of 0.9420. This agrees with the finding that texture-based descriptors are still quite useful, especially in representing micro-patterns in the eye and mouth regions.

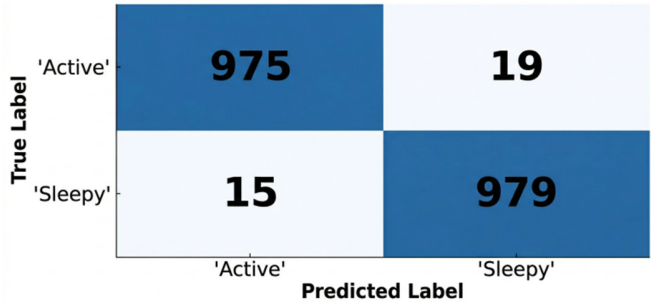

Fig. 7. Confusion matrix for the proposed VGG16 + SVM classifier on the combined NTHU-DDD dataset.

Among classifiers, SVM achieved the highest mean accuracy (0.9651) and significantly outperformed deep classifiers such as LeNet ($p = 0.0109$). Although RF achieved slightly lower performance (0.9538), no statistically significant difference from SVM ($p = 0.403$) was observed. Finally, among deep classifiers, LeNet and CNN achieved similar performance (0.9340 and 0.9327, respectively; $p = 0.833$), while LeNet significantly outperformed AlexNet ($p = 0.038$). Overall, these results indicate that hybrid approaches

combining deep feature extractors, particularly VGG16, with machine learning classifiers, such as SVM or RF, provide the most effective and reliable framework for driver drowsiness detection.

Model Comparison (A vs. B)	*p*-value	Statistical Significance ($\alpha = 0.05$)
VGG16 vs. MobileNet	0.164	Not Significant
LBP vs. Gabor Filter	0.146	Not Significant
SVM vs. LeNet	0.0109	Significant
SVM vs. Random Forest (RF)	0.403	Not Significant
LeNet vs. CNN	0.833	Not Significant
LeNet vs. AlexNet	0.038	Significant

Fig. 8. Results of the paired t-test, determining the statistical significance (p-values) of performance differences between the selected models.

C. Qualitative Analysis of Failure Cases

Although the hybrid models, such as VGG16 + LeNet and MobileNet + SVM, achieved high overall accuracy, the performance of these models had a slight degradation on the NTHU-DDD dataset. A qualitative analysis of misclassified frames (Figure 9) shows that this degradation is mainly due to challenging real-world conditions. For example, the extremely low illumination environment and the eyeglasses illustrated in Figure 9(a) completely obscure the micro-textures of the eyes. Additionally, severe head rotations, such as in the cases presented in Figure 9(b), where the driver interacts with a passenger, make the primary facial landmarks completely occluded from the camera's view. Finally, the combination of wearing dark sunglasses and the downward pose of the head, portrayed in Figure 9(c), prevents visual extractors from detecting the state of the driver's eyes.

(a) (b) (c)

Fig. 9. NTHU-DDD failure cases: (a) low illumination, (b) head rotation, and (c) eye occlusion.

D. Computational Complexity Analysis

To evaluate the feasibility for real-time applications, the number of parameters for all 23 model combinations is compared in Figure 10.

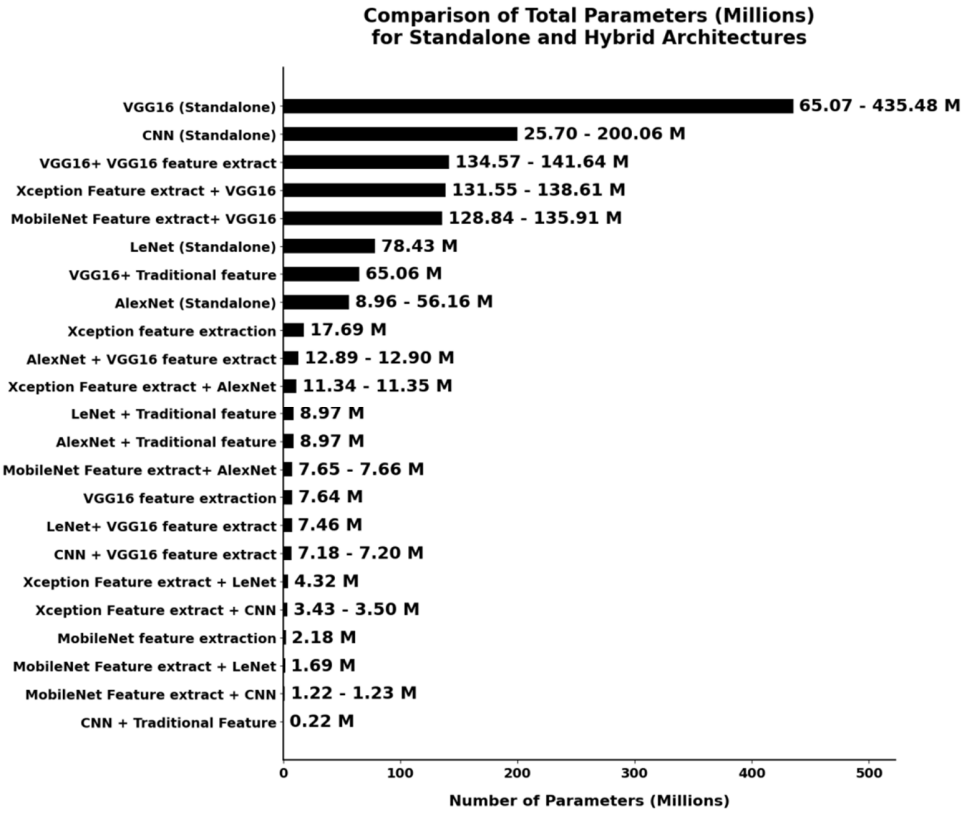

Fig. 10. Total parameters (M) for evaluated architectures.

Model complexity ranges from approximately 0.22M parameters for the Traditional + CNN model to 435.48M for the standalone VGG16 model. Moreover, hybrid models, particularly those using MobileNet and Xception as feature extractors, maintained relatively low complexity, ranging from 1.22M to 17.69M parameters. These results demonstrate that the proposed hybrid approaches achieved high accuracy while maintaining low computational cost, making them suitable for real-time deployment.

IV. CONCLUSION

Addressing the risks of drowsy driving requires intelligent systems capable of finding an optimal trade-off between accuracy, speed of processing, and reliability in the face of varying environmental conditions. This study presents a comprehensive comparative analysis of feature extraction and classification methods to identify the optimal architecture for this type of safety-critical application.

A systematic evaluation of traditional and deep learning technologies provided the following key conclusions: i) hybrid approaches performed better than standalone approaches, ii) models that combine pretrained feature extractors, such as Visual Geometry Group-16 (VGG16) and Mobile Networks (MobileNet), with lightweight classifiers, such as Support Vector Machine (SVM) or LeNet, outperform fully trained deep learning models. This performance is a result of the capability of transfer learning techniques to generate a linearly separable feature space that can be processed by traditional classifiers with high efficiency and low computation costs.

Among all configurations, the VGG16 + SVM model achieved the highest performance, with accuracy exceeding 99% in most test scenarios. Moreover, the results highlight the importance of feature combination. In particular, while eye movement is still the main fatigue indicator, the inclusion of additional mouth data serves as a very important stabilizing cofactor that reduces false alarms and False Positives (FPs) in cases where eye data are insufficient.

DECLARATION OF COMPETING INTERESTS

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

ACKNOWLEDGMENT

Not applicable to this work.

DATA AVAILABILITY

The datasets used in the current study are publicly available at [22-25].

REFERENCES

[1] R. Kolb, "National Highway Traffic Safety Administration (NHTSA)," in Encyclopedia of Business Ethics and Society, 2455 Teller Road, Thousand Oaks, California 91320 United States: SAGE Publications, Inc., 2008.

[2] Y. Liang, "Detecting driver distraction," Doctor of Philosophy, University of Iowa, Iowa City, IA, United States, 2009.

[3] National Highway Traffic Safety Administration. "Drowsy Driving." NHTSA. [Online]. Available: https://www.nhtsa.gov/risky-driving/drowsy-driving.

[4] M. Ramzan, H. U. Khan, S. M. Awan, A. Ismail, M. Ilyas, and A. Mahmood, "A Survey on State-of-the-Art Drowsiness Detection Techniques," *IEEE Access*, vol. 7, pp. 61904–61919, 2019, https://doi.org/10.1109/ACCESS.2019.2914373.

[5] S. Panda and M. Kolhekar, "Feature Selection for Driver Drowsiness Detection," in *Proceedings of International Conference on Computational Intelligence and Data Engineering*, 2019, vol. 28, pp. 127–140, https://doi.org/10.1007/978-981-13-6459-4_14.

[6] S. Bakheet and A. Al-Hamadi, "A Framework for Instantaneous Driver Drowsiness Detection Based on Improved HOG Features and Naïve Bayesian Classification," *Brain Sciences*, vol. 11, no. 2, Feb. 2021, Art. no. 240, https://doi.org/10.3390/brainsci11020240.

[7] R. Chinthalachervu, I. Teja, M. Ajay Kumar, N. Sai Harshith, and T. Santosh Kumar, "Driver Drowsiness Detection and Monitoring System using Machine Learning," Journal of Physics: Conference Series, vol. 2325, no. 1, Aug. 2022, Art. no. 012057, https://doi.org/10.1088/1742-6596/2325/1/012057.

[8] Y. Albadawi, A. AlRedhaei, and M. Takruri, "Real-Time Machine Learning-Based Driver Drowsiness Detection Using Visual Features," *Journal of Imaging*, vol. 9, no. 5, Apr. 2023, Art. no. 91, https://doi.org/10.3390/jimaging9050091.

[9] G. Setyo Utomo, E. Rachmawati, and F. Sthevanie, "Fatigue Detection Through Car Driver's Face Using Boosting Local Binary Patterns," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 7, no. 5, pp. 988–995, Aug. 2023, https://doi.org/10.29207/resti.v7i5.4798.

[10] D. A. Navastara, W. Y. M. Putra, and C. Fatichah, "Drowsiness Detection Based on Facial Landmark and Uniform Local Binary Pattern," *Journal of Physics: Conference Series*, vol. 1529, no. 5, May 2020, Art. no. 052015, https://doi.org/10.1088/1742-6596/1529/5/052015.

[11] B. K. Padamata and J. R. Singothu, "A Machine Learning Approach for Driver Drowsiness Detection," *International Journal of Engineering Research and Applications*, vol. 10, no. 11, pp. 58–65, Nov. 2020.

[12] S. Chen, Z. Wang, and W. Chen, "Driver Drowsiness Estimation Based on Factorized Bilinear Feature Fusion and a Long-Short-Term Recurrent Convolutional Network," *Information*, vol. 12, no. 1, Dec. 2020, Art. no. 3, https://doi.org/10.3390/info12010003.

[13] M. Dua, Shakshi, R. Singla, S. Raj, and A. Jangra, "Deep CNN models-based ensemble approach to driver drowsiness detection," *Neural Computing and Applications*, vol. 33, no. 8, pp. 3155–3168, Apr. 2021, https://doi.org/10.1007/s00521-020-05209-7.

[14] I. Jahan *et al.*, "4D: A Real-Time Driver Drowsiness Detector Using Deep Learning," *Electronics*, vol. 12, no. 1, Jan. 2023, Art. no. 235, https://doi.org/10.3390/electronics12010235.

[15] B. Akrout and S. Fakhfakh, "How to Prevent Drivers before Their Sleepiness Using Deep Learning-Based Approach," *Electronics*, vol. 12, no. 4, Feb. 2023, Art. no. 965, https://doi.org/10.3390/electronics12040965.

[16] R. K. Shukla, A. K. Tiwari, and A. K. Jha, "An Efficient Approach of Face Detection and Prediction of Drowsiness Using SVM," *Mathematical Problems in Engineering*, vol. 2023, no. 1, Jan. 2023, Art. no. 2168361, https://doi.org/10.1155/2023/2168361.

[17] O. Nasir *et al.*, "SAFE-DRIVE-AI: A CNN–LSTM–Attention Framework for Drowsiness Detection," *Engineering, Technology & Applied Science Research*, vol. 15, no. 5, pp. 27594–27600, Oct. 2025, https://doi.org/10.48084/etasr.12725.

[18] S. Anber, W. Alsaggaf, and W. Shalash, "A Hybrid Driver Fatigue and Distraction Detection Model Using AlexNet Based on Facial Features," *Electronics*, vol. 11, no. 2, Jan. 2022, Art. no. 285, https://doi.org/10.3390/electronics11020285.

[19] S. E. Bekhouche, Y. Ruichek, and F. Dornaika, "Driver drowsiness detection in video sequences using hybrid selection of deep features," *Knowledge-Based Systems*, vol. 252, Sept. 2022, Art. no. 109436, https://doi.org/10.1016/j.knosys.2022.109436.

[20] F. You, X. Li, Y. Gong, H. Wang, and H. Li, "A Real-time Driving Drowsiness Detection Algorithm With Individual Differences Consideration," *IEEE Access*, vol. 7, pp. 179396–179408, 2019, https://doi.org/10.1109/ACCESS.2019.2958667.

[21] V. Vijayan and K. P. Pushpalatha, "Fused Architecture with Enhanced Bag of Visual Words for Efficient Drowsiness Detection," *Journal of Advanced Computational Intelligence and Intelligent Informatics*, vol. 27, no. 2, pp. 182–189, Mar. 2023, https://doi.org/10.20965/jaciii.2023.p0182.

[22] *Drowsiness_dataset*. (2018), D. Perumandla. [Online]. Available: https://www.kaggle.com/datasets/dheerajperumandla/drowsiness-dataset.

[23] S. Abtahi, M. Omidyeganeh, S. Shirmohammadi, and B. Hariri, "YawDD: a yawning detection dataset," in *Proceedings of the 5th ACM Multimedia Systems Conference*, Singapore, Singapore, Mar. 2014, pp. 24–28, https://doi.org/10.1145/2557642.2563678.

[24] C.-H. Weng, Y.-H. Lai, and S.-H. Lai, "Driver Drowsiness Detection via a Hierarchical Temporal Deep Belief Network," in *Computer Vision – ACCV 2016 Workshops*, vol. 10118, C.-S. Chen, J. Lu, and K.-K. Ma, Eds. Cham: Springer International Publishing, 2017, pp. 117–133.

[25] *yawn and eye close Computer Vision Model*. (2023), lab502. [Online]. Available: https://universe.roboflow.com/lab502/yawn-and-eye-close-5kie4.

[26] J. Jo, S. J. Lee, K. R. Park, I.-J. Kim, and J. Kim, "Detecting driver drowsiness using feature-level fusion and user-specific classification," *Expert Systems with Applications*, vol. 41, no. 4, pp. 1139–1152, Mar. 2014, https://doi.org/10.1016/j.eswa.2013.07.108.

[27] R. C. Gonzalez and R. E. Woods, *Digital image processing*, Fourth, Global edition. New York, New York: Pearson Education, 2018.

[28] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2005, vol. 1, pp. 886–893, https://doi.org/10.1109/CVPR.2005.177.

[29] T. Ahonen, A. Hadid, and M. Pietikainen, "Face Description with Local Binary Patterns: Application to Face Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2037–2041, Dec. 2006, https://doi.org/10.1109/TPAMI.2006.244.

[30] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, Nov. 2004, https://doi.org/10.1023/B:VISI.0000029664.99615.94.

[31] N. Mishra and A. Bhatt, "Feature Extraction Techniques in Facial Expression Recognition," in *2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS)*, May 2021, pp. 1247–1251, https://doi.org/10.1109/ICICCS51141.2021.9432192.

[32] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2018, pp. 4510–4520, https://doi.org/10.1109/CVPR.2018.00474.

[33] X. Ran, T. Yan, and T. Cai, "MobileNet for Differential Constellation Trace Figure," in *2021 13th International Conference on Communication Software and Networks (ICCSN)*, June 2021, pp. 168–172, https://doi.org/10.1109/ICCSN52437.2021.9463655.

[34] J. Liu, "VGG, MobileNet and AlexNet on Recognizing Skin Cancer Symptoms," in *2022 3rd International Conference on Electronic Communication and Artificial Intelligence (IWECAI)*, Jan. 2022, pp. 525–528, https://doi.org/10.1109/IWECAI55315.2022.00107.

[35] F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 1800–1807, https://doi.org/10.1109/CVPR.2017.195.

[36] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, Sept. 1995, https://doi.org/10.1007/BF00994018.

[37] J. Gwak, A. Hirao, and M. Shino, "An Investigation of Early Detection of Driver Drowsiness Using Ensemble Machine Learning Based on Hybrid Sensing," *Applied Sciences*, vol. 10, no. 8, Apr. 2020, Art. no. 2890, https://doi.org/10.3390/app10082890.

[38] C. B. S. Maior, M. J. D. C. Moura, J. M. M. Santana, and I. D. Lins, "Real-time classification for autonomous drowsiness detection using eye aspect ratio," *Expert Systems with Applications*, vol. 158, Nov. 2020, Art. no. 113505, https://doi.org/10.1016/j.eswa.2020.113505.

[39] F. A. Jokhio and A. Jokhio, "Image Classification using AlexNet with SVM Classifier and Transfer Learning," *Journal of Information Communication Technologies and Robotic Applications*, vol. 10, no.1, pp. 44–51, June 2019.

[40] L. Chen, G. Xin, Y. Liu, and J. Huang, "Driver Fatigue Detection Based on Facial Key Points and LSTM," *Security and Communication Networks*, vol. 2021, pp. 1–9, June 2021, https://doi.org/10.1155/2021/5383573.

[41] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition." arXiv, Apr. 2015, https://doi.org/10.48550/arXiv.1409.1556.

[42] A. Bhanja, D. Parhi, D. Gajendra, K. Sinha, and A. K. Sahoo, "Driver drowsiness shield (DDSH): a real-time driver drowsiness detection system," *ROBOMECH Journal*, vol. 12, no. 1, May 2025, Art. no. 18, https://doi.org/10.1186/s40648-025-00307-4.

[43] S. Essahraui *et al.*, "Real-Time Driver Drowsiness Detection Using Facial Analysis and Machine Learning Techniques," *Sensors*, vol. 25, no. 3, Art. no. 812, Jan. 2025, https://doi.org/10.3390/s25030812.