# Low-cost Driver Monitoring System Using Deep Learning

Marco Fernández-Pérez, Miguel Antunes-García, Santiago Montiel-Marín,
Franck Fierro and Luis M. Bergasa

University of Alcalá, Spain,
marco.fernandez@edu.uah.es, miguel.antunes@uah.es,
santiago.montiel@uah.es, franck.yaguargos@uah.es, luism.bergasa@uah.es,
Robesafe Research Group
Electronics Department

**Abstract.** Driver drowsiness is a major contributor to traffic accidents, making its early and reliable detection a key challenge in intelligent transportation systems. Despite advances in computer vision, accurately assessing a driver's state in real time remains difficult due to individual variability and environmental factors. In this context, we present a Deep Learning-based neural network architecture for driver drowsiness detection, specifically designed for low-cost devices. Using temporal sequences of images, the driver's facial region is extracted via an initial detection model. These sequences are then processed with a model that analyses spatial features and temporal patterns of the face to generate a binary classification (alert/drowsy). The proposal is implemented as a modular pipeline in PyTorch, optimised for inference on limited hardware. A comprehensive benchmark of the trained models on the same database validates the approach and identifies the optimal architecture for the task.

**Keywords:** Deep Learning, Computer Vision, Driver Monitoring, Drowsiness Detection, Temporal Processing.

## 1 Introduction

Road safety remains a critical global challenge, with driver drowsiness being a common and dangerous risk factor. Alarmingly, 15–30% of car accidents are directly or indirectly related to drowsiness. This phenomenon can occur at any time of day, whether due to night-time fatigue, morning impairment or post-meal sluggishness. This urgency is now being addressed through strict regulatory action: European Safety Regulation 2019/2144 [1], enacted in July 2022, requires that all newly homologated vehicles to integrate Advanced Driver Assistance Systems (ADAS), including Drowsiness Detection (DDR), by 2026.

The regulation imposes critical design constraints. DDR systems must minimise error rates under real driving conditions and operate without collecting biometric data (e.g., facial recognition). Additionally, they must adhere strictly to EU data protection laws by avoiding continuous storage of driver data. Using computer vision and deep learning, this research develops a real-time regulatory-compliant solution.

In this paper, we propose a non-intrusive single-camera system that is mounted on the vehicle dashboard and processes image sequences in order to detect drowsiness via a neural network architecture. Upon identification of impairment, the system triggers multi-modal alerts (visual/auditory) to help the driver recover and stop the vehicle safely. By aligning technical innovation with European legislative requirements, prioritizing privacy, accuracy, and real-world applicability, this work aims to mitigate the significant human cost of fatigue-related accidents.

## 2 Related Work

This section explores solution approaches centred on a modular architecture with two key components: a face detection and cropping module, and a sequence classification module for drowsiness detection. Each module is trained and evaluated in different datasets focused on each task.

### 2.1 2D face detection

The goal of 2D face detection is to accurately locate and isolate the driver's face within vehicle cabin images, especially those taken by wide-angle cameras, which capture surrounding areas. This task is critical for driver monitoring systems, where isolating the facial region from complex backgrounds is essential. State-of-the-Art (SOTA) object detectors are leveraged to achieve robust real-time performance. Among prevalent models, YOLO [2] (You Only Look Once), Faster R-CNN [3] (Region-based Convolutional Neural Network), and RT-DETR [4] (Real-Time DEtection TRansformer) represent the most widely adopted approaches, each offering distinct trade-offs in speed, accuracy, and computational efficiency.

The YOLO family models prioritise inference speed and resource efficiency through their single-stage architecture, making them ideal for embedded systems with stringent latency requirements. In contrast, Faster R-CNN employs a two-stage framework that first generates region proposals before classification, yielding higher accuracy at the cost of significant computational overhead. RT-DETR introduces an end-to-end transformer-based design that eliminates heuristic components like Non-Maximum Suppression (NMS), achieving a competitive speed-accuracy balance. Comparative evaluations highlight YOLO's superiority in latency (3–9 ms) and efficiency for real-time applications, whereas Faster R-CNN incurs higher latency (50–120 ms) despite superior precision. RT-DETR (10–15 ms latency) offers architectural innovation but demands greater computational resources than optimised YOLO variants.

In driver monitoring systems, where low latency, minimal resource consumption, and deployability on edge hardware are critical requirements, lightweight single-stage detectors such as YOLO are commonly preferred. The subsequent model selection process prioritises variants that offer an optimal trade-off between precision and inference speed, guided by empirical latency-accuracy benchmarks. In our case, we selected the YOLOv11-nano variant due to its favourable balance between detection accuracy and computational efficiency.

### 2.2 Drowsiness classifiers

The aim of drowsiness classification is to detect states of fatigue or alertness in individuals by analysing video sequences over time. This task can be approached in multiple ways: using end-to-end video understanding models to directly extract spatio-temporal features, or using hybrid frameworks that combine an image classifier with a temporal component to model sequential dynamics.

The paper evaluates these two approaches using the following architectures for each case:

- **End-to-end video models**: Video ResNet [5], Video SwinTransformer [6], Video S3D [7]
- **Hybrid sequential models**: ResNet [8] + LSTM, EfficientNet [9] + LSTM

Performance was assessed in terms of computational efficiency (parameter count) and inference speed, with focus on real-time applicability. Among all architectures, Video S3D and Video SwinTransformer demonstrated superior operational efficiency. Video S3D's factorized convolutions minimise computational overhead while maintaining temporal awareness, and the hierarchical attention mechanism of Video SwinTransformer achieves rapid processing through optimised token grouping.

These two models, which prioritise low latency and balanced resource demands, are identified as the optimal solutions for real-time drowsiness detection systems. The Experiments section provides a comparison of classification end-to-end models for this particular task.

## 3 Architecture

The architectural design is intended for real-time driver drowsiness detection, utilising a dashboard camera as the primary surveillance device. The system utilises a two-stage modular pipeline, as illustrated in Figure 1, for the processing of video frames at a fixed rate. The combination of facial localisation with temporal behaviour analysis enables the detection of fatigue signs (e.g. prolonged eye closure, yawning).

First, the system initiates when a front-facing dashboard camera captures a raw image of the driver. This image is immediately processed by Module 1 (Facial Region Detector and Extractor) to detect the driver's face within a bounding box and extract a facial crop. The cropped facial image is then fed into a sliding temporal buffer that maintains the last 10 consecutive facial crops using a FIFO

(First-In-First-Out) mechanism, where each new frame displaces the oldest in the sequence.

Once the buffer accumulates 10 frames, representing 1 second of temporal data, the sequence is used as input to Module 2 (Spatio-Temporal Feature Extractor and Classifier). This block processes the facial sequence using either an End-to-End video architecture or a hybrid image classifier coupled with a temporal aggregator. It extracts spatial features (eye/blink dynamics, yawning) and temporal patterns (sustained eye closure, head nodding), outputting a drowsiness probability.

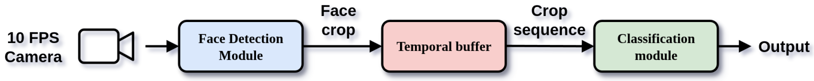


Fig. 1: Architecture of the real-time driver drowsiness detection system.

If drowsiness is detected with probability greater than a threshold, the system triggers multi-modal alerts: a visual warning on the dashboard and an audible alarm.

### 3.1 Face Detection

The face detection module is provided with an image of an arbitrary resolution. The model infers the image and extracts a 1:1 aspect ratio with 224 x 224 pixel resolution face crop. This crop is then inserted into the sliding temporal buffer for subsequent utilisation in the classification process. As previously indicated in the Related Work section, we elected the YOLOv11-nano variant for this module.

### 3.2 Drowsiness Classification

The drowsiness classification stage employs 10 sequential face crops, which are stored in the sliding temporal buffer, as input. The images, which possess an identical aspect ratio and resolution, are processed to identify drowsiness, producing a probability value ranging from 0 to 1.

## 4 Experiments

In this section, the results of experiments conducted using different datasets for each module are presented. The performance of each module is evaluated, and the joined application is tested on different scenarios. Furthermore, some ablation studies are presented to achieve the optimal number of images for classification and output threshold.

### 4.1 Datasets

**Person Faces Dataset** [10] is a dataset for face detection. The database contains 3,011 images with a resolution of 640 x 640, along with bounding boxes indicating the presence of people in the photographs. The images depict a range of facial cases, incorporating diverse individuals, genders, ethnicities, facial expressions, glasses and angled faces. The dataset has built-in data augmentation capabilities, including a selection of three augmentations per sample. Among these options are horizontal flip, random zoom crop, rotation, grayscale, hue shift, saturation adjustment, brightness adjustment, exposure adjustment, blur, and random occlusions. The dataset features a single class, face, for all bounding box annotations. This dataset is used to train and evaluate the face detector stage of our architecture.

**Driver Drowsiness Detection** [11] is a dataset focused on video classification tasks. The dataset contains 41,732 images with a resolution of 227 x 227, extracted from 54 real videos featuring 27 different individuals. The videos have been filmed from a variety of angles and under different lighting conditions. In order to adapt the dataset to the 10-frame format, a manual split into groups of 10 sequential frames was performed.

In addition, we conduct a qualitative assessment of the model using sequences from our own recorded dataset.

### 4.2 Metrics

We use the following metrics to evaluate the different stages:

- Accuracy: Measures the proportion of correct predictions relative to the total number of samples.
- Confusion Matrix: A table that explicitly shows the model's correct and incorrect predictions, broken down by actual and predicted classes.
- Inference Time: The time it takes for the trained model to generate a prediction for a new sample once deployed. For a real time application, minimal inference time is required.
- Number of parameters: Defines the total number of adjustable parameters the neural network model acquires during training. This value determines hardware requirements in production environments.
- Precision: The ratio of what is truly positive to what was predicted as positive.
- mAP50: A key metric in 2D object detection. It simultaneously evaluates classification precision and localization accuracy. It calculates the average precision for all classes using an IoU (Intersection over Union) threshold.

### 4.3 Face Detector

To perform the initial face detection step within the proposed architecture, we trained the YOLOv11-nano model. The training process was conducted under two configurations: using transfer learning from the COCO dataset for 15 epochs, and training from scratch for 30 epochs to evaluate the model's ability to adapt to the specific task. All experiments were executed on a desktop computer equipped with an AMD RADEON RX 9070 GPU, allowing for consistent benchmarking of training performance and inference speed.

Table 1: Comparison of YOLOv11-nano with and without transfer learning

| | **With Transfer Learning** | | **Without Transfer Learning** | |
|---|---|---|---|---|
| | **Positive Pred.** | **Negative Pred.** | **Positive Pred.** | **Negative Pred.** |
| **Positive Label** | **87.16**% | 8.56% | 74.81% | 18.84% |
| **Negative Label** | 4.28% | | 6.35% | |

Table 1 presents the confusion matrices for the YOLOv11-nano model, comparing training with and without transfer learning, using an IoU threshold of 0.5. The results clearly demonstrate that the transfer learning approach significantly accelerates the training process and improves performance. Specifically, training from scratch for 30 epochs yields inferior results compared to just 15 epochs with transfer learning. The model trained with transfer learning achieves a mAP@0.5 of 0.957 on the validation set, with a higher proportion of true positives, indicating superior adaptation to the face detection task. Furthermore, the measured inference time of 9 milliseconds confirms its suitability for real-time driver monitoring applications. Figure 2 shows real detection results in our custom recorded dataset.

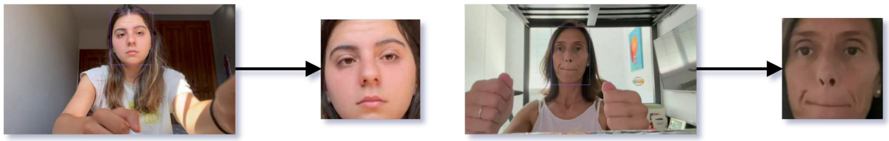


Fig. 2: Face detection and cropping inference example

## 4.4 Drowsiness Classifier

This final classification module leverages sequences of cropped facial images to detect driver drowsiness. To address the classification task, we evaluated a range of deep learning models, including both end-to-end and hybrid architectures. For hybrid models, we adapted them to the sequence-based nature of the problem by removing their final classification layers and extracting feature vectors instead. These vectors were then processed by a Long Short-Term Memory (LSTM) network, which captures temporal dependencies across the input sequence. The output from the final LSTM time step, summarising the temporal information, was fed into a binary classifier to produce the final alert/drowsy prediction. In contrast, end-to-end architectures directly generate classification probabilities without requiring additional temporal processing modules.

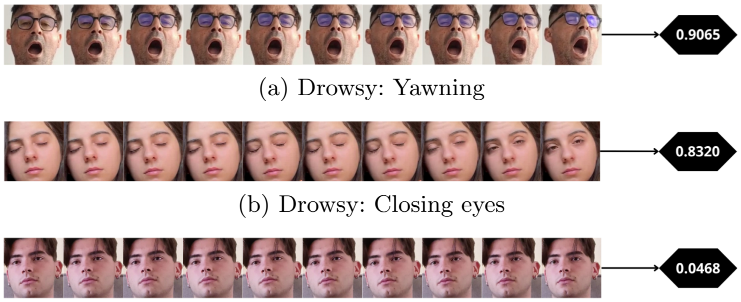
The Driver Drowsiness Detection dataset exhibits a class imbalance, with approximately 33% of samples labelled as ”Drowsy” and 66% as ”Non-Drowsy.” To address this issue during training, class weighting was applied to compensate for the uneven distribution, ensuring that both classes contribute proportionally to the loss function and improving the model's ability to detect minority class instances. All models use a batch size of 16 and a dataset split of 80% training (3326 sequences), 15% validation (623 sequences), and 5% testing (209 sequences). Transfer learning was applied to enhance performance and training speed.

Table 2: Metrics for drowsiness classification models

| **Model** | **Val Acc** | **Test Acc** | **TI (ms)** | **Params** | **Epochs** |
|---|---|---|---|---|---|
| Video ResNet | 84.91% | 78.68% | 32.1 | 33 M | 10 |
| Video S3D | 87.56% | 84.21% | 9.4 | 7 M | 15 |
| Video SwinTransformer | 85.09% | 84.69% | **9.2** | 27 M | 15 |
| ResNet18 + LSTM | 86.68% | 89.00% | 24.0 | 11 M | 15 |
| ResNet50 + LSTM | 86.84% | 90.43% | 56.7 | 23 M | 15 |
| EfficientNet + LSTM | **88.44%** | **93.30%** | 73.0 | **4 M** | 15 |

Table 2 shows different classification metrics, uses a standard 0.5 threshold, and inference times measured on an AMD Radeon 9070 GPU. While ResNet50 + LSTM and EfficientNet + LSTM achieve high accuracy (90.43% and 93.30%), their inference times (56.7 and 73 ms) exceed the constraints required for real-time applications. Additionally, the gap between validation and test accuracy suggests overfitting, which is further confirmed during inference on our recorded videos, where these models demonstrate reduced generalisation capability. In contrast, architectures such as Video S3D and ResNet18+LSTM offer acceptable latency (9.4 and 24 ms), but still exhibit validation-test accuracy gaps of up to +5.53%, likely due to limited dataset diversity (27 subjects), which hinders generalization.

In this context, the Video Swin Transformer emerges as the optimal solution, meeting key requirements for real-time driver drowsiness detection. It demonstrates strong predictive consistency, with a minimal gap between validation and test accuracy (85.09% vs. 84.69%), and offers the fastest inference time among evaluated models (9.2 ms). This balance between competitive accuracy and sub-10 ms latency makes it particularly suitable for safety-critical applications. Additionally, it achieves the lowest rate of false negatives, minimising the risk of misclassifying drowsy sequences as non-drowsy and thereby enhancing reliability in real-world scenarios. Figure 3 shows inference examples in diverse sequences.



(a) Drowsy: Yawning

(b) Drowsy: Closing eyes

(c) Alert

Fig. 3: Drowsiness classification inference example

### 4.5 Model Pipeline

Based on the evaluation of all relevant performance metrics, YOLOv11-nano was selected for facial detection due to its optimal balance between accuracy and inference speed. For drowsiness classification, the Video Swin Transformer was chosen for its superior generalisation and real-time capabilities. The complete implementation follows the workflow illustrated in Figure 4, integrating both modules into a unified pipeline for efficient and accurate driver monitoring.

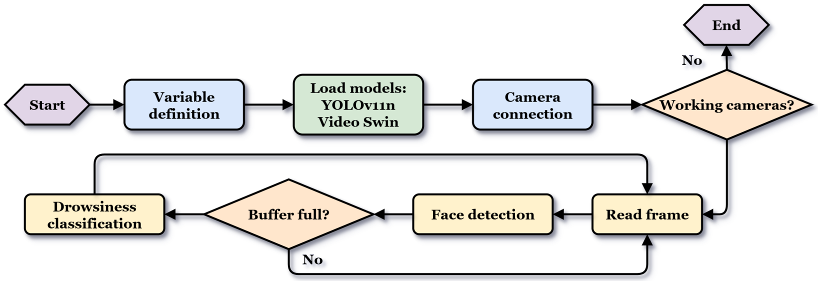


Fig. 4: Deployment workflow

In environments with similar visual characteristics, physical cameras may be replaced by pre-recorded video sources. A key distinction lies in the data acquisition method: while cameras provide the most recent frame in real time, recorded sources deliver frames sequentially. To simulate real-time conditions during evaluation, recorded frames are sampled at 0.1-second intervals, corresponding to a frame rate of 10 FPS.

We evaluated the pipeline in two distinct environments to assess its performance under different hardware constraints. The first setup involved a desktop PC equipped with an AMD RADEON RX 9070 GPU, which was also used for the previously described experiments. The second setup employed an NVIDIA Jetson AGX Xavier embedded device to measure performance in low-cost, resource-constrained scenarios. Performance measurements reveal notable differences between the two hardware environments. On the desktop setup, the GPU alone consumes approximately 350W, with the YOLOv11-nano detector running at 9 ms and the Video Swin Transformer at 9.2 ms, resulting in a total pipeline latency of 27.7 ms. In contrast, the embedded NVIDIA Jetson AGX Xavier device operates at just 50W and executes the full pipeline in 57.41 ms, representing a 3.15× slowdown. Despite this, the system maintains a processing rate above 10 Hz, satisfying the minimum requirement for real-time monitoring.

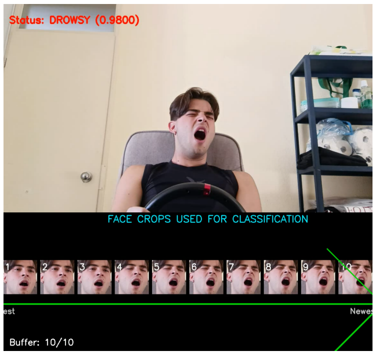


Fig. 5: End-to-end pipeline execution

Figure 5 shows an example of the output of the whole architecture. During qualitative evaluation, occasional issues emerged, such as failures in face detection when subjects covered their faces with their hands, and classification inaccuracies for individuals wearing glasses.

### 4.6 Ablation Studies

Experimental ablation studies were conducted to optimise key parameters of the drowsiness classification system. Two critical aspects were investigated: the optimal number of images per sequence to balance accuracy and real-time latency, and the most effective classification threshold to improve decision reliability.

To minimise latency while preserving classification accuracy, sequences of 5, 10, and 20 images were evaluated. The Video Swin Transformer model was trained for 15 epochs using the same dataset employed in the sequence classification task, with a default decision threshold of 0.5. Inference times were measured on the same desktop PC used in previous experiments. The results of this ablation study are summarised in Table 3.

While the 20-image sequence achieved the highest (90.29%) validation accuracy, it resulted in the lowest test accuracy (78.85%) and the highest latency

Table 3: Ablation study of image sequence length

| **Images** | **Val Acc (%)** | **Test Acc (%)** | **Inf Time (ms)** |
|---|---|---|---|
| 5 | 86.63 | **89.4** | **22.47** |
| 10 | 85.09 | 84.69 | 27.7 |
| 20 | **90.29** | 78.85 | 43.8 |

(43.8 ms), exceeding acceptable limits for low-cost devices. In contrast, the 5-image sequence offered the best test accuracy (89.4%) and the lowest latency (22.47 ms). However, real-world testing revealed critical limitations with shorter sequences: normal blinks, typically spanning only 2–3 frames, were frequently misclassified as drowsiness due to insufficient temporal context. As a result, a 10-image sequence was selected as the optimal configuration. It provides a robust test accuracy of 84.69%, maintains a manageable latency of 27.7 ms, and crucially, demonstrates sufficient resilience to avoid misclassifying typical blinks as drowsiness in real-world scenarios.

Table 4: Ablation study of the output classification threshold

| **Threshold** | **Test Acc (%)** |
|---|---|
| 0.2 | 59.81 |
| 0.3 | 77.51 |
| 0.4 | 83.73 |
| 0.5 | 84.69 |
| 0.6 | 87.56 |
| 0.7 | 90.43 |
| **0.8** | **90.91** |
| 0.9 | 86.60 |

To evaluate the robustness of the architecture, the default drowsiness threshold of 0.5 was compared against alternative values ranging from 0.2 to 0.9. This analysis was conducted using the final YOLOv11-nano and Video Swin Transformer pipeline on the Driver Drowsiness Detection dataset. The results of this threshold ablation study are presented in Table 4.

A classification threshold of 0.8 achieved the highest test accuracy (90.91%), significantly outperforming the default threshold of 0.5 (84.69%). Validation on drowsiness videos reveals that the 0.5 threshold produces false positives during rapid blinks, where brief score spikes are misclassified as drowsiness due to insufficient temporal filtering. In contrast, the 0.8 threshold reliably identifies true drowsiness episodes while effectively suppressing these transient artefacts. This behaviour is illustrated in Figure 6, where 6a shows the output probability over time using a threshold of 0.5, and 6b shows the same sequence with a threshold of 0.8. In both plots, the blue line represents the output probability, and the red dotted line indicates the classification threshold.

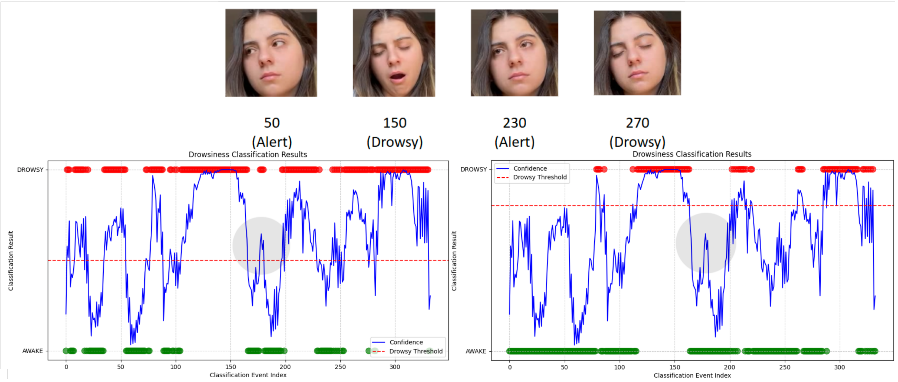


Fig. 6: Confidence values in a sequence. a) Threshold = 0.5, b) Threshold = 0.8.

The default threshold of 0.5 can misclassify brief increases in the output score, which are often caused by normal blinking, as indicators of drowsiness. These transient fluctuations lack sufficient temporal context and should not trigger alerts in a safety-critical system. Figure 7 shows a sequence that is correctly classified but includes a momentary increase in the score that would trigger a false alert if the threshold were set at 0.5. In contrast, a higher threshold of 0.8 demonstrates greater robustness by effectively filtering out these noisy activations and focusing on sustained patterns that better reflect genuine drowsiness.

## 5 Conclusions and Future Work

The proposed driver monitoring system offers a viable, modular, deep-learning architecture that is optimised for low-cost devices and achieves real-time inference suitable for vehicular applications. Its modular design enables individual components to be updated without the need for full model retraining. The so-

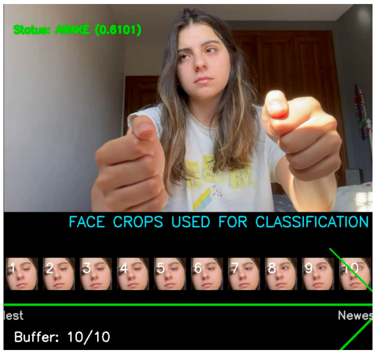


Fig. 7: Peak classification

lution effectively balances inference speed with minimising false negatives. Furthermore, its compliance with EU Regulation 2019/2144, which stipulates that biometric data must not be stored and that the system must operate in real time, further supports its potential for widespread adoption across the automotive industry.

Future optimisation efforts will focus on reducing latency by minimising memory operations, adjusting numerical precision, and leveraging TensorRT for model compilation. To enhance generalisation, face detection performance could be improved through the use of carefully curated datasets that incorporate fixed camera angles and a wide range of head poses. Additionally, drowsiness classification systems will benefit from the development of balanced datasets that reflect greater diversity in ethnicity, gender, and facial features.

## References

1. Regulation (EU) 2019/2144 of the European Parliament and of the Council of 27 November 2019, Official Journal of the European Union, L 325, 16/12/2019, Available at: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32019R2144
2. Jocher, G., Qiu, J., and Chaurasia, ”YOLO by Ultralytics”, Jul. 2025 [Online]. https://github.com/ultralytics/ultralytics
3. S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks”, 2016.
4. Yian Zhao, Wenyu Lv, Shangliang Xu, Jinman Wei, Guanzhong Wang, Qingqing Dang, Yi Liu, and Jie Chen, ”RT-DETRv2: Improved Baseline with Bag-of-Freebies for Real-Time Detection Transformer”, 2024.
5. Tran, D., Wang, H., Torresani, L., Ray, J., LeCun, Y., and Paluri, M., “A closer look at spatiotemporal convolutions for action recognition,” 2018.
6. Liu, Z., Ning, J., Cao, Y., Wei, Y., Zhang, Z., Lin, S., and Hu, H., “Video Swin Transformer,” 2022.
7. Xie, S., Sun, C., Huang, J., Tu, Z., and Murphy, K., “Rethinking Spatiotemporal Feature Learning: Speed-Accuracy Trade-offs in Video Classification,” 2018.
8. He, K., Zhang, X., Ren, S., and Sun, J., “Deep residual learning for image recognition,” 2016.
9. Tan, M., and Le, Q.V., “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks,” 2019.
10. Open Source F. Detection and R. Dataset, ”Person Faces Dataset” 2023 [Online]. https://universe.roboflow.com/face-detection-and-recognition-dataset/person-faces
11. Ismail Narsi, Mohammed Karrouchi, Hajar Snoussi, Kamal Kassmi and Abdelhafid Messaoudi ”Driver Drowsiness Dataset” [Online]. https://www.kaggle.com/datasets/ismailnasri20/driver-drowsiness-dataset-ddd