**ABSTRACT**

In the realm of road safety, the "Driver Drowsiness Detection System Using Image Processing" emerges as a pioneering technological advancement. This project harnesses the power of python , employing a strategic blend of image processing techniques and machine learning algorithms - K-Nearest Neighbors (KNN) and Random Forest - to achieve an impressive accuracy rate of 97%. The proposed system offers a comprehensive solution, encompassing image acquisition, pre-processing, segmentation, feature extraction, and classification stages.

In the initial phase of image acquisition, data is collected from the vehicle's interior environment, providing crucial input for subsequent analysis. The subsequent pre-processing stage plays a pivotal role, commencing with grayscale conversion to enhance computational efficiency. Moreover, the system employs advanced techniques for eye detection, ensuring precise localization of the driver's eyes within the captured images.

The following phase of the system is dedicated to segmentation, takes center stage by executing intricate IRIS segmentation and extraction processes. This critical step serves as the foundation for accurate feature extraction in the subsequent module. The next phase employs Discrete Cosine Transform (DCT) and Speeded-Up Robust Features (SURF) to extract discriminative features from the segmented iris images, facilitating robust identification of drowsiness-related patterns.

In the final phase, the system enters the classification phase, utilizing both KNN and Random Forest classifiers. These machine learning models have been fine-tuned to deliver exceptional accuracy in distinguishing between alert and drowsy states. Additionally, the project introduces an innovative concept - the Fusion score, calculated as the weighted average of KNN and Random Forest scores, utilizing the formula KNN+RF/2. This fusion mechanism enhances the system's reliability by leveraging the strengths of both classifiers.

The culmination of these efforts results in the system's ability to make a decisive determination: whether the driver's eyes are open and alert or exhibiting signs of drowsiness. The seamless integration of image processing, machine learning, and fusion techniques makes this project an indispensable tool in enhancing road safety by preventing accidents caused by driver drowsiness.

In essence, the "Driver Drowsiness Detection System Using Image Processing" showcases the tremendous potential of MATLAB, KNN, and Random Forest in the realm of road safety. With an exceptional accuracy rate of 97%, it stands as a testament to innovation and engineering prowess, offering a lifeline to countless lives on the road.

**NTRODUCTION**

ROAD traffic accidents claim more than one million people every year and 90% are caused by driver distraction [1]. Driving distraction can occur at any time while the vehicle is operating due to many factors [2]. Drowsiness is a common cause of distracted driving. Therefore, it is necessary to build driver assistance systems that can avoid accidents. A driver eye status surveillance system is an application built to alert drivers who are falling asleep. Drowsiness comes when the driver experiences a long journey, uses alcoholic drinks, such as alcohol and beer or has a medical condition. From those observations, the researchers mainly focused on analyzing driver behavior, vehicle behavior, and driver physiology [3]. Driver behavior can be recognized by extracting features of the driver body, such as the features of eyes, mouth, hands, head posture, and body posture.

Vehicle behavior is a method of surveillance for unusual vehicle movements when going out of a lane, swerving on the road, or interacting with other vehicles in the vicinity. Driver physiology is monitored through electrical pulses from the heart rate, blood pressure, and changes in body temperature. The development of devices to monitor vehicle behavior and driver physiology require complex, high-cost devices, and synchronous infrastructure. Moreover, wearable devices can cause discomfort to the driver while operating and several natural factors in the driver body can interfere with the received electrical signal. Realizing the importance of eye status in the early stage [3] of sleep and the abovementioned analysis, this article proposes a driver eye status surveillance system.

This system is based on lightweight CNNs with a new version of attention mechanisms named the triplet attention module. The use of lightweight CNN architectures to optimize network parameters while it is still ensuring the feature extraction process. On the other hand, the attention mechanism helps the system to focus on processing information in the eye area and ignores useless or background information. The proposed system aims to detect the status of the driver eye with the following three stages: face detection, eye detection, and eye classification. The system is built on the mechanism of driver eye status surveillance through the front-mounted camera. It does not influence the operation or does not cause unpleasant effects on the driver’s body, as the abovementioned methods. The main contributions of this article are shown as follows. 1) This work proposes two lightweight CNNs for eye detection and eye classification tasks. These networks are aggregated with a real-time face detector (nano YOLO5Face) to build a three-stage driver eye status surveillance system.

The CNNs in this article are designed for use in low-computing devices such as a CPU and an Nvidia Jetson Nano. With network parameter optimization based on compact architectural designs, the proposed networks solve the problem of computational and deployment costs. In addition, it is not invasive to the driver during use 2) It also provides the datasets for locating the eye area in images under different situations and follows the PASCAL VOC dataset format. These are basic datasets for machine learning developers who use features from the driver eye. Along with the proposed eye detection and eye classification networks, they can use in other fields, such as eye-tracking, medical, and biomedical. 3) On the application side: the proposed system is tested in real-time on a CPU-based personal computer and a Jetson Nano device without high latency while ensuring accuracy. The rest of this article is organized as follows. Section II introduces the related works in eye detection and classification including the advantages and disadvantages of each technical group. Section III explains in detail the proposed CNN architectures and real-time processing system. Section IV describes and analyzes the experiments and results. Finally, Section V concludes this article.

**EXISTING SYSTEM**

* Before the advent of the proposed "Driver Drowsiness Detection System Using Image Processing" leveraging MATLAB, KNN, and Random Forest, an existing system had been developed utilizing Convolutional Neural Networks (CNN). This preceding system, which represented a significant step in the evolution of drowsiness detection technology, harnessed the power of deep learning to address the critical issue of driver alertness.
* The existing system's core architecture revolved around CNN, a state-of-the-art neural network model renowned for its ability to automatically learn and extract intricate features from images. The CNN-based approach was chosen due to its proficiency in capturing complex patterns and nuances within image data, making it ideal for the task of driver drowsiness detection.
* In this existing system, the image acquisition process played a crucial role, capturing real-time data from the vehicle's interior environment, including the driver's face and eyes. These images were then fed into the CNN model, which underwent extensive training on a diverse dataset encompassing both alert and drowsy states. This training allowed the CNN to autonomously learn and adapt to the subtle visual cues indicative of driver drowsiness.
* The CNN architecture consisted of multiple convolutional layers, pooling layers, and fully connected layers, meticulously designed to hierarchically process the input images. These layers systematically extracted relevant features and progressively refined them, culminating in a high-level representation that facilitated robust drowsiness detection.
* The existing system excelled in its ability to detect drowsiness in real-time, providing instantaneous feedback to the driver or triggering alert mechanisms when signs of drowsiness were detected. It was lauded for its accuracy and efficiency in identifying even slight variations in the driver's alertness level.
* In summary, the previous drowsiness detection system, founded upon Convolutional Neural Networks (CNN), represented a significant milestone in the evolution of driver safety technology. Its utilization of deep learning and image analysis techniques enabled it to provide valuable insights into driver alertness, ultimately contributing to safer roads and reduced accidents caused by drowsy driving.

**DISADVANTAGES OF EXISTING SYSTEM:**

* High Computational Demands: CNN-based systems often require significant computational resources, including powerful GPUs, to operate efficiently. This can limit their deployment in resource-constrained environments and in cost-sensitive applications.
* Training Data Requirements: CNNs are data-hungry models, meaning they require large and diverse datasets for effective training. Collecting and annotating such datasets can be time-consuming and expensive.
* Limited Generalization: The CNN model may struggle to generalize well to diverse driving conditions, lighting conditions, and driver appearances that were not well-represented in the training data. This can result in reduced accuracy in real-world scenarios.
* Inability to Detect Early Signs: CNN-based systems may not always detect very early signs of drowsiness, such as subtle changes in eye behavior or facial expressions, which could potentially lead to accidents if not identified promptly.
* Dependency on Camera Quality: The accuracy of the system can be highly dependent on the quality of the camera and the clarity of captured images. Poor-quality cameras or environmental factors like glare or low light can affect performance.
* False Positives/Negatives: Like all machine learning models, CNN-based systems are susceptible to false positives (incorrectly identifying a driver as drowsy when they are not) and false negatives (failing to detect drowsiness when it is present), which can impact the system's reliability.
* Limited Explainability: CNNs are often considered "black-box" models, meaning they provide limited insights into why a particular decision was made. This lack of explainability can be a drawback in safety-critical applications where understanding the model's reasoning is important.
* Maintenance and Updates: Keeping the CNN-based system up-to-date with evolving driver behaviors, vehicle interiors, and safety regulations can be a logistical challenge, as it may require retraining the model and adapting the system's architecture.
* Privacy Concerns: Continuous image capture and analysis raise privacy concerns, as the system must record and process images of the driver's face and eyes. Ensuring the security and privacy of this data is essential.
* Cost of Implementation: Implementing a CNN-based drowsiness detection system can be expensive, both in terms of hardware and software development. This cost can be a barrier to adoption, especially for smaller vehicles or fleet operators.

**PROPOSED SYSTEM**

* The proposed "Driver Drowsiness Detection System Using Image Processing" marks a significant stride in the field of road safety technology. This innovative system capitalizes on the strengths of MATLAB, K-Nearest Neighbors (KNN), and Random Forest to introduce a comprehensive and highly accurate solution for detecting driver drowsiness.
* The proposed system encompasses multiple modules and employs a combination of cutting-edge techniques to ensure precise and reliable drowsiness detection. The system initiates with real-time image acquisition within the vehicle's interior environment, focusing on the driver's face and eyes. In the next module, images are converted to grayscale to simplify subsequent analysis while preserving essential facial and eye details.
* In the next module of the proposed system does the Eye Detection. Advanced algorithms are applied to accurately detect and locate the driver's eyes within the acquired images. Then comes the module: Iris Segmentation & Extraction. The system meticulously segments the iris region, a critical step in isolating the areas crucial for drowsiness detection.
* In the next module the Feature Extraction is done using DCT and SURF. DCT (Discrete Cosine Transform): Employed to extract discriminative features from the segmented iris region. SURF (Speeded-Up Robust Features): Utilized to capture additional intricate patterns and details, enhancing feature diversity.
* In the final module the Classification is made using KNN Classifier and Random Forest Classifier.
* KNN Classifier: Leveraging the K-Nearest Neighbors algorithm, the system effectively categorizes the extracted features, providing insights into the driver's state of alertness.
* Random Forest Classifier: This ensemble learning method complements KNN, further improving classification accuracy by considering a multitude of decision trees.
* Later the Fusion Score is calculated. A novel concept is introduced - the Fusion Score. It is calculated as the weighted average of the KNN and Random Forest classifier scores, achieved using the formula (KNN + RF) / 2. This innovative fusion mechanism harnesses the collective intelligence of both classifiers to enhance detection robustness. Then the Final Determination is made. The culmination of the system's efforts results in a definitive determination of whether the driver's eyes are open and alert or if signs of drowsiness are evident.
* The proposed system boasts an impressive accuracy rate of 97%, making it a pivotal tool in enhancing road safety by preventing accidents attributable to drowsy driving. Its integration of image processing, machine learning, and fusion techniques sets a new standard for drowsiness detection systems, offering a lifeline to countless lives on the road.
* In conclusion, the "Driver Drowsiness Detection System Using Image Processing" represents an innovative and vital advancement in road safety technology. Its exceptional accuracy, integration of state-of-the-art algorithms, and introduction of the Fusion Score concept firmly establish it as a game-changer in the quest for safer roads.

**ADVANTAGES OF PROPOSED SYSTEM**

* High Accuracy: The system achieves an impressive accuracy rate of 97%, ensuring reliable detection of driver drowsiness. This high level of accuracy minimizes false alarms and false negatives, enhancing road safety.
* Real-time Detection: The system operates in real-time, continuously monitoring the driver's alertness. It can provide timely alerts, allowing the driver to take corrective actions and prevent potential accidents due to drowsy driving.
* Comprehensive Analysis: By combining various image processing techniques and machine learning algorithms, the system conducts a comprehensive analysis of the driver's facial and eye features. This multi-stage approach increases the system's ability to detect drowsiness accurately.
* Adaptive Learning: Machine learning components, such as KNN and Random Forest, have the capacity to adapt and improve their accuracy over time. As more data becomes available, the system can fine-tune its models, enhancing its performance.
* Incorporation of Fusion Score: The introduction of the Fusion Score, calculated as the weighted average of KNN and Random Forest scores, enhances the system's reliability. This innovative approach combines the strengths of both classifiers, reducing the likelihood of misclassification.
* Reduced False Alarms: The Fusion Score concept helps mitigate false alarms by requiring agreement between KNN and Random Forest classifiers before classifying a driver as drowsy. This reduces the chances of unnecessary alerts to the driver.
* Non-intrusive: The system operates solely based on image data, making it a non-intrusive solution for drowsiness detection. It does not require additional sensors or equipment that may be uncomfortable for the driver.
* Scalability: The system can be scaled and adapted for use in various vehicle types, from personal cars to commercial trucks and public transportation. It can contribute to improving road safety across a wide range of applications.
* Privacy-Preserving: Unlike some biometric-based drowsiness detection systems, this image processing approach does not capture or store sensitive biometric data. It focuses on eye and facial features while respecting driver privacy.
* Cost-Effective: In comparison to some complex sensor-based solutions, this system can be more cost-effective to implement, making it accessible to a broader range of vehicle owners and fleet operators.
* Safety Enhancement: Ultimately, the system's primary advantage is its potential to significantly enhance road safety by reducing accidents caused by drowsy driving. It serves as an additional safety layer, especially during long journeys or night-time driving.
* In summary, the proposed drowsiness detection system offers a host of advantages, including high accuracy, real-time monitoring, adaptability, privacy preservation, and cost-effectiveness. Its potential to improve road safety and save lives makes it a valuable technological innovation.

**LITERATURE REVIEW**

**Characteristics, likelihood and challenges of road traffic injuries in China before COVID-19 and in the postpandemic era**

Through a review of previous studies, this paper analysed the epidemiological characteristics and attempts to determine the various trends of road traffic injuries (RTIs) in China before and after the coronavirus disease 2019 (COVID-19). This paper proposed effective measures and suggestions for responding to RTIs in China. Moreover, this paper aimed to provide some references for studies on RTIs in the future. According to a reference review, 50 articles related to RTIs were published and viewed in the China National Knowledge Infrastructure (CNKI), Wanfang database, Weipu (VIP) database and PubMed/MEDLINE database. Articles were selected according to the exclusion and inclusion criteria and then classified and summarized. Regarding cases, RTIs in China were highest in summer, autumn, and in rural areas and lowest in February. Men, elderly individuals and people living in rural areas were more susceptible to RTIs. In addition, thanks to effective and proactive policies and measures, the number of RTIs and casualties in China has substantially decreased, while there has been a growing number of traffic accidents along with the increase in nonmotor vehicles.

However, it is worth noting that the number of RTIs obviously fell during the COVID-19 pandemic due to traffic lockdown orders and home quarantine policies. Nevertheless, accidents related to electric bicycles increased unsteadily because of the reduction in public transportation use at the same time. The factors that cause RTIs in China can be divided into four aspects: human behaviours, road conditions, vehicles and the environment. As a result, measures responding to RTIs should be accordingly proposed. Moreover, the road traffic safety situation in developing countries was more severe than that in developed countries. RTIs in China showed a downward trend attributed to road safety laws and various policies, and the downward trend was more significant during the COVID-19 pandemic owing to traffic lockdowns and home quarantine measures. It is urgent and necessary to promote road traffic safety, reduce injuries, and minimize the burden of injuries in developing countries.

Road traffic injuries (RTIs) refer to fatal or nonfatal injuries caused by the collision of at least one moving vehicle on a public road. At present, RTIs are the eighth leading cause of death in the world and the fifth leading cause of reduced life expectancy and have been a persistent problem that the Chinese government is committed to dealing with.

As the World Health Organization’s (WHO) Global status report on road safety noted, the number of RTIs worldwide continues to rise, with an increase of ~100,000 in just three years. In 2018, a study indicated that there were ~1.35 million deaths attributed to RTIs every year that seriously threaten the safety of human life and property, and almost 90% of the deaths occurred in low- and middle-income countries (WHO, [2018](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9808728/#CR38)). The WHO predicted that RTIs may become the fifth leading cause of death worldwide in 2030. Therefore, the United Nations (UN) proposed a Decade of Action for Road Safety, which aimed to halve the number of casualties caused by RTIs by 2020 (Pérez-Núñez et al., [2021](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9808728/#CR25)).

Over the past decade, members of the UN have pledged to undertake actions to promote road safety and enhance road safety management by such actions as developing and enforcing legislation regarding key RTI risk factors (such as limiting speed, reducing drunk driving, and increasing the use of seat belts, child restraints and motorcycle helmets); improving road and vehicle safety standards; and promoting road safety and enhancing road safety management (Peden, [2010](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9808728/#CR24)). The actions in the theoretical framework are shown in Fig. [​Fig.1.1](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9808728/figure/Fig1/). Considering the seriousness of RTIs worldwide, the WHO and the UN have called on all countries to address road traffic safety and take effective measures to prevent RTIs. China is the largest developing country in the world that has the longest high-speed roads. In China, RTIs have become the primary cause of injury-related casualties (Zhang et al., [2011](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9808728/#CR44)). According to Global Burden of Disease (GBD) study data, China’s road traffic deaths accounted for 21% of global road traffic deaths in 2017, and the number was ~262,000 (Wang et al., [2019](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9808728/#CR33)). The main cause of RTIs is a large number of motor vehicles. RTIs can also give rise to serious economic losses, and the total economic cost of RTIs in China was calculated as 490.1 billion RMB in 2017, which was equivalent to 0.60% of the GDP (gross domestic product, GDP) (Tan et al., [2020](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9808728/#CR29)).

[[An external file that holds a picture, illustration, etc.
Object name is 41599_2022_1482_Fig1_HTML.jpg](https://www.ncbi.nlm.nih.gov/core/lw/2.0/html/tileshop_pmc/tileshop_pmc_inline.html?title=Click%20on%20image%20to%20zoom&p=PMC3&id=9808728_41599_2022_1482_Fig1_HTML.jpg)](https://www.ncbi.nlm.nih.gov/core/lw/2.0/html/tileshop_pmc/tileshop_pmc_inline.html?title=Click%20on%20image%20to%20zoom&p=PMC3&id=9808728_41599_2022_1482_Fig1_HTML.jpg" \t "tileshopwindow)

[Fig. 1](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9808728/figure/Fig1/)

Five pillars of national activites encouraged in the Decade of Road Safety.

Although some studies have reported that RTIs in China have shown a downward trend, the mortality of road traffic accidents is still much higher than that in developed countries. Additionally, traffic control during the COVID-19 epidemic period substantially improved China’s road traffic safety, but we cannot reduce RTIs by restricting road traffic. In the post-COVID-19 pandemic era, the implementation of road traffic safety is also a major challenge that China needs to tackle as a result of the rapid development of China’s economy and society.

The main purpose of this review was to clarify the trend of RTIs before and after the COVID-19 pandemic and the main reasons for the change in RTIs in China, particularly exploring the impact of China’s road traffic safety law, traffic lockdowns during the COVID-19 and home quarantine policies on RTIs. Moreover, it is essential to summarize the primary factors of RTIs in China and develop suggestions to formulate preventive measures to reduce the burden of RTIs in China and other similar countries.

# Real-Time Machine Learning-Based Driver Drowsiness Detection Using Visual Features

# Drowsiness-related car accidents continue to have a significant effect on road safety. Many of these accidents can be eliminated by alerting the drivers once they start feeling drowsy. This work presents a non-invasive system for real-time driver drowsiness detection using visual features. These features are extracted from videos obtained from a camera installed on the dashboard. The proposed system uses facial landmarks and face mesh detectors to locate the regions of interest where mouth aspect ratio, eye aspect ratio, and head pose features are extracted and fed to three different classifiers: random forest, sequential neural network, and linear support vector machine classifiers. Evaluations of the proposed system over the National Tsing Hua University driver drowsiness detection dataset showed that it can successfully detect and alarm drowsy drivers with an accuracy up to 99%.

Drowsiness is a major concern with respect to road safety. Drivers’ unconsciousness due to microsleep can frequently lead to destructive accidents. Falling asleep at the wheel is usually related to lack of sleep, exhaustion, or mental health problems. In the UAE, the ministry of interior recorded 2931 car crashes in 2020. The number increased in 2021 to 3488 records. The majority of these traffic accidents were caused by distracted driving due to drowsiness, sudden swerving, or failure to maintain a safe distance between vehicles [[1](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#B1-jimaging-09-00091)]. In this situation, it is crucial to exploit new technologies to plan and design systems that can track drivers and estimate their level of attention while driving. As multiple countries are concerned regarding this issue, researchers worldwide worked on building Driver Drowsiness Detection (DDD) systems that are capable of detecting drivers’ drowsiness signs in the early stages.

According to the literature, drowsiness detection systems can be grouped into three categories based on the measures that are used to detect the drowsiness signs [[2](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#B2-jimaging-09-00091),[3](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#B3-jimaging-09-00091),[4](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#B4-jimaging-09-00091),[5](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#B5-jimaging-09-00091)]: biological-based, vehicle-based, and image-based systems. In the first category, biological-based measures rely on monitoring the body’s physiological signals including, ElectroEncephaloGraphy (EEG), ElectroCardioGraphy (ECG), ElectroMyoGraphy (EMG), Electro-OculoGraphy (EOG) signals, and blood pressure [[6](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#B6-jimaging-09-00091),[7](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#B7-jimaging-09-00091),[8](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#B8-jimaging-09-00091),[9](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#B9-jimaging-09-00091)]. In this type of system, drowsiness is determined by detecting the signal’s deviation from the standard state’s characteristics and analyzing if the new signal indicates drowsiness. In the second category, vehicle-based measures depend on monitoring variations in the car’s movement patterns through different sensors’ installed to measure various vehicle and street parameters. To infer the drowsiness level, vehicle-based systems analyze the changes or abnormal behavior of the car, including, for example, the steering wheel angle, speed, or deviation from the lane [[10](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#B10-jimaging-09-00091),[11](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#B11-jimaging-09-00091)]. The third category is the image-based measures which depend mainly on the drowsiness signs that appear on the driver’s face and head. These systems detect drowsiness by monitoring the drivers’ head movements and facial parameters such as the eyes, mouth facial expressions, eyebrows, or respiration [[12](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#B12-jimaging-09-00091),[13](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#B13-jimaging-09-00091),[14](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#B14-jimaging-09-00091)].

All three categories have some limitations [[2](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#B2-jimaging-09-00091),[15](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#B15-jimaging-09-00091)]. Biological-based systems can detect drowsiness in the initial stages due to their ability to compare the continuous changes in the physiological signals, but, in most biological-based systems, it is demanded that electrodes be connected to the driver’s body. This setup is usually inconvenient and uncomfortable for the driver. It also involves noise that affects the signal quality, leading to decreased accuracy. Vehicle-based systems depend generally on vehicle types, and can greatly be affected by multiple factors, including road characteristics, climate conditions, and the driver’s experience, habits, and ability to drive. Limitations of the image-based systems are strictly related to the quality of the camera used and its adaptability to different lighting conditions. The existence of objects covering parts of the face, such as glasses, sunglasses, masks, etc., can also affect the accuracy of image-based DDD systems. However, among these three systems, image-based systems are considered to be fully non-invasive, low cost, and minimally affected by road conditions. Therefore, image-based measures are widely deployed to develop versatile, affordable, real-time and, fully portable DDD devices [[2](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#B2-jimaging-09-00091),[12](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#B12-jimaging-09-00091),[13](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#B13-jimaging-09-00091),[14](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#B14-jimaging-09-00091),[16](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#B16-jimaging-09-00091),[17](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#B17-jimaging-09-00091)].

In this work, we present a new image-based DDD system. It uses a unique combination of features derived from the driver’s facial parameters to train and test three classifiers, namely Random Forest (RF), sequential Neural Networks (NN), and linear Support Vector Machine (SVM). The features used in this system are Eye Aspect Ratio (EAR), Mouth Aspect Ratio (MAR), and head pose estimation. The proposed system is convenient for the driver in the sense that it does not require any sensors or equipment to be attached to the driver’s body. It is adaptable to be used in different vehicles, including buses, cars, motorcycles, and others. Evaluations of the proposed system on the National Tsing Hua University DDD (NTHUDDD) video dataset show that it can achieve accuracy up to 99%, indicating that it is an effective solution.

The rest of the paper is organized as follows: [Section 2](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#sec2-jimaging-09-00091) summarizes the recent studies relating to the features used in this work. The methodology is presented in [Section 3](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#sec3-jimaging-09-00091). [Section 4](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10219078/#sec4-jimaging-09-00091) presents and discusses the results. The last section states the conclusions and the future directions.

# Real Time Eye Detector with Cascaded Convolutional Neural Networks

# An accurate and efficient eye detector is essential for many computer vision applications. In this paper, we present an efficient method to evaluate the eye location from facial images. First, a group of candidate regions with regional extreme points is quickly proposed; then, a set of convolution neural networks (CNNs) is adopted to determine the most likely eye region and classify the region as left or right eye; finally, the center of the eye is located with other CNNs. In the experiments using GI4E, BioID, and our datasets, our method attained a detection accuracy which is comparable to existing state-of-the-art methods; meanwhile, our method was faster and adaptable to variations of the images, including external light changes, facial occlusion, and changes in image modality.

In recent years, eye detection has become an important research topic in computer vision and pattern recognition [1, 2], because the human eyes locations are essential information for many applications, including psychological analysis, facial expression recognition, auxiliary driving, and medical diagnosis [3]. However, eye detection is quite challenging in many practical applications. The cameras are sensitive to light variations and the shooting distance, which makes the human eyes very eccentric in a facial image. Sometimes the face is partially occluded and we are not able to obtain a complete facial image. For example, half of the face was covered in a cover test for detecting squint eyes [4]. In this case, some existing eye detection methods do not work, because they rely on a facial model detection to locate the eyes. An eye detector is also expected to work well in various image modalities, that is, infrared and visible images. Moreover, the eye detection algorithm should be fast because it is supposed to be online in many practical cases. Although many methods have been proposed to detect the eyes from facial images, it is difficult to find one method that performs well in terms of accuracy, robustness, and efficiency. Therefore, we are attempting to develop an efficient and robust eye detection algorithm to fulfill the requirements of the applications as much as possible.

The rest of this paper is organized as follows. In the Related Work, a review of related works is presented. In the Method, we proposed an efficient method for eye detection, which consists of candidate region generation, eye region determination and classification, and eye center positioning. Then, the training scheme, evaluation results, and discussions were presented in the Training Scheme, Evaluation, and Further Discussion, respectively. Finally, conclusion remarks were drawn in the last section.

# An Intelligent Time Series Model Based on Hybrid Methodology for Forecasting Concentrations of Significant Air Pollutants

Rapid industrialization and urban development are the main causes of air pollution, leading to daily air quality and health problems. To find significant pollutants and forecast their concentrations, in this study, we used a hybrid methodology, including integrated variable selection, autoregressive distributed lag, and deleted multiple collinear variables to reduce variables, and then applied six intelligent time series models to forecast the concentrations of the top three pollution sources. We collected two air quality datasets from traffic and industrial monitoring stations and weather data to analyze and compare their results. The results show that a random forest based on selected key variables has better classification metrics (accuracy, AUC, recall, precision, and F1). After deleting the collinearity of the independent variables and adding the lag periods using the autoregressive distributed lag model, the intelligent time-series support vector regression was found to have better forecasting performance (RMSE and MAE). Finally, the research results could be used as a reference by all relevant stakeholders and help respond to poor air quality.

Pollution caused by rapid industrialization is the main reason for the deterioration of air quality. According to the global energy and carbon dioxide (CO2) status report [1], global energy-related CO2 emissions increased to a historic high of 33.1 gigatons in 2018. Air quality monitoring stations were set up in densely populated areas of Taiwan where high pollution was likely to occur or reflect more significant air quality problems. The air quality data of the monitoring stations across Taiwan [2] show that the air quality index (AQI) of western Taiwan is greater than 100 days in a year (unhealthy level). Northern Taiwan has a better AQI, ranging from 23 to 64 days. The central region comes next, ranging from 69 to 84 days. Southern Taiwan is the worst, with 97 to 160 days. As stated in [3], PM2.5 is the main source of pollution in Taiwan, among which traffic pollution accounts for 36%, Chinese imports 27%, industrial pollution 25%, and natural diffusion 12%. The negative effects of long-term exposure to air pollution on human health have been extensively researched [4,5].

According to the World Health Organization (WHO), air pollution causes 7 million deaths every year, and 4.2 million people died from environmental or outdoor pollution in 2016 alone [6]. Air pollution caused by traffic and industrial emissions is a significant problem in many cities, and the pollutants that are detrimental to human health include suspended particulate matter (PM), ozone (O3), sulfur dioxide (SO2), carbon monoxide (CO), volatile organic compounds (VOCs), and nitrogen oxides (NOx). Many studies have shown the unexpected health effects caused by air pollution, and it has been shown that long-term exposure to air pollution increases the risk of cardiovascular and respiratory diseases, type 2 diabetes, cancer, and premature death [7,8]. In terms of the global burden of disease [9], the main factors contributing to deteriorating health because of air pollution are mainly PM2.5 and O3.

WHO also warned that 90% of the global population is currently affected by toxic air and that damage to children is particularly serious [10]. Therefore, all countries have begun to pay attention to the problems of health and national economic development affected by air pollution, and related policies have been introduced. Air pollution research needs to consider natural environmental factors and the related knowledge of causality, and most previous studies analyzed it from a statistical perspective [11]. Air pollution data require a huge amount of climate information, and statistical analysis of air pollution data cannot effectively catch the interactions of environmental and air quality factors. Algorithms based on artificial intelligence (a broad definition that includes machine learning and deep learning) have big data classification and prediction capabilities, which can be applied to study air quality [12,13]. Data mining can extract and discover the truth that is hidden in large amounts of data. The use of classification rules is a data mining technique to extract frequent patterns embedded between classes and observations in a specific dataset. Variable selection is a method that can reduce input variables to a manageable size for processing and analysis, which reduces the number of variables used and predetermines the cut-off point for the number of variables considered when building a model [14]. As there are many environmental factors involved in building a model, this study used variable selection to effectively filter out the important variables that affect air quality.

The AQI can provide decision-makers with the information needed to implement pollution mitigation measures and make air quality management decisions; therefore, accurate forecasting is essential for early control of air pollution and the protection of public health. Based on this, in the present study, we used an artificial intelligence algorithm and variable selection to build a forecasting model for generating rules that meet air pollution conditions to predict air quality levels. Next, we made numerical forecasts for the top three pollution sources to understand the influence of environmental conditions on the concentrations of air pollutants. In summary, in this study, we carried out the following: (1) Applied five variable selection methods to filter out the important variables for the collected datasets and used the integrated variable selection method (IVSM) to find the key variables. (2) Used four rule-based classifiers to classify air quality and generate classification rules, in which we found the top three pollutants (PM2.5, PM10, and O3) from the generated rules. (3) Deleted collinear variables and added lag periods of variables by the autoregressive distributed lag (ARDL) test.

(4) Forecast concentrations of PM2.5, PM10, and O3 by using four intelligent time-series forecast methods based on IVSM-selected variables, ARDL-selected variables, and full variables. (5) Gave appropriate explanations to provide the results to stakeholders for reference and enact countermeasures for dynamic environmental factors. The remaining sections of the paper are arranged as follows. Section 2 reviews the literature on air pollution, variable selection, and machine learning techniques. The study’s concept, proposed research procedure, and computation steps are introduced in Section 3. Section 4 introduces the experimental environment, datasets, and experimental results. Finally, Section 5 summarizes the conclusions and provides recommendations.

# When I Look into Your Eyes: A Survey on Computer Vision Contributions for Human Gaze Estimation and Tracking

The automatic detection of eye positions, their temporal consistency, and their mapping into a line of sight in the real world (to find where a person is looking at) is reported in the scientific literature as gaze tracking. This has become a very hot topic in the field of computer vision during the last decades, with a surprising and continuously growing number of application fields. A very long journey has been made from the first pioneering works, and this continuous search for more accurate solutions process has been further boosted in the last decade when deep neural networks have revolutionized the whole machine learning area, and gaze tracking as well. In this arena, it is being increasingly useful to find guidance through survey/review articles collecting most relevant works and putting clear pros and cons of existing techniques, also by introducing a precise taxonomy.

This kind of manuscripts allows researchers and technicians to choose the better way to move towards their application or scientific goals. In the literature, there exist holistic and specifically technological survey documents (even if not updated), but, unfortunately, there is not an overview discussing how the great advancements in computer vision have impacted gaze tracking. Thus, this work represents an attempt to fill this gap, also introducing a wider point of view that brings to a new taxonomy (extending the consolidated ones) by considering gaze tracking as a more exhaustive task that aims at estimating gaze target from different perspectives: from the eye of the beholder (first-person view), from an external camera framing the beholder’s, from a third-person view looking at the scene where the beholder is placed in, and from an external view independent from the beholder.

Gaze is a fundamental communication mean, since it can express emotions, feelings and intentions [[**1**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B1-sensors-20-03739)]. Evidence of its importance is in the noticeable quantity of applications that have been presented in the last few decades, spacing from Human-Computer and Human-Robot Interaction (HCI/HRI) [[**2**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B2-sensors-20-03739),[**3**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B3-sensors-20-03739)] to assistive devices [[**4**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B4-sensors-20-03739),[**5**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B5-sensors-20-03739),[**6**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B6-sensors-20-03739)], healthcare/clinical assessment and diagnosis [[**7**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B7-sensors-20-03739)], driver vigilance monitoring [[**8**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B8-sensors-20-03739),[**9**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B9-sensors-20-03739)], analysis of consumer market [[**10**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B10-sensors-20-03739)], analysis of behavioral patterns in disabilities or diseases [[**11**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B11-sensors-20-03739),[**12**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B12-sensors-20-03739)], gaming design [[**13**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B13-sensors-20-03739),[**14**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B14-sensors-20-03739)], and so on. Gaze tracking consists of the procedure of obtaining the direction or the point of regard (PoR) of the gaze with continuity, on a digital screen or in the physical environment, and it is achieved through mechanical, electronic, optical, and/or other methods.

Although researchers often use the terms of eye tracking and gaze tracking as synonyms, there is a slight difference between them. In particular, eye tracking is the measurement of eye movement/activity, while gaze tracking is the analysis of eye tracking data with regards to the head or the visual scene (a known three-dimensional (3D) environment, a screen, a surface, etc.). In other words, eye tracking consists in detecting the existence and position of the eyes into an image and to track them over time (in the following images), and then in measuring activities as eye blinks, fixations, saccadic movements [[**15**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B15-sensors-20-03739)], pupil dilation [[**16**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B16-sensors-20-03739)], etc. The term gaze tracking instead refers to the estimation (and the temporal tracking) of where a person is looking at, in terms of interaction with the scene, and it is obtained by determining the 3D line of sight between the user and the target. A common practice is to consider the gaze tracker as the last block of a system that starts by localizing and tracking the eyes [[**17**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B17-sensors-20-03739)]. On the other hand, some works define gaze trackers as a subset of the eye tracker families [[**18**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B18-sensors-20-03739)].

In this arena, it is being increasingly useful to find guidance trough exhaustive survey/review articles collecting most relevant works and putting clear the pros and cons of existing techniques, also by introducing a precise taxonomy. This kind of manuscripts allows for researchers and technicians to choose the better way to move towards their application or scientific goals. Valuable attempts to give a review of the existing literature have been provided in [[**19**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B19-sensors-20-03739),[**20**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B20-sensors-20-03739)]. These manuscripts crystallized the state of the art in 2009 and 2013, respectively, and they gave a holistic view of hardware, user-interaction, eye detection, and gaze mapping techniques. Afterward, a plethora of new works have appeared in the literature, and then it has become more useful and fruitful to focus on every single aspect of the process rather than on the entire algorithmic pipeline. This led authors in [[**21**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B21-sensors-20-03739)] to review methods for eye region and pupil detection and localization, whereas, in [[**22**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B22-sensors-20-03739)], the gaze mapping functions based on interpolation or approximation that determine, starting from the pupil glint, the coordinates on a screen, are summarized. Works dealing with full-face appearance-based gaze estimation have been discussed in [[**23**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B23-sensors-20-03739)] and an overview of key technologies of gaze tracking has been proposed in [[**24**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B24-sensors-20-03739)] instead. Insight into the issues related to algorithms, system configurations, user conditions, and performance has been finally provided in [[**25**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B25-sensors-20-03739)]. However, the recent advancements reached in computer vision, and more generally in artificial intelligence, boosted gaze tracking technology than any other scientific field. Unfortunately, how these great advancements in computer vision and pattern recognition have impacted gaze estimation and tracking has not been analyzed yet. This paper attempts to fill this gap and it also tries to give a broader analysis about gaze tracking that brings to a new taxonomy (extending the consolidated ones), established by considering gaze tracking as a more exhaustive task that has to estimate gaze targets from different points of view:

* from the eye of the beholder (first-person view);
* from an external camera framing the beholder’s;
* from a third-person view looking at the scene where the beholder is placed in; and,
* from an external view independent from the beholder.

Active researches and the most relevant advancements are introduced and discussed, while taking the aforementioned innovative points of view on this largely debated scientific topic into account. It should be clarified that, in this work, the systems that use active illumination techniques will not be reviewed, unless they provide some innovation in terms of the computer vision technique, or in the case of more complete architectures where infrared only represents a fraction of the contribution. Indeed, this is a common and well-established technology, already on the market and with many patented approaches, although they still present their own specific challenges to be addressed and some research lines are still open, especially for improving electronic lighting components [[**26**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B26-sensors-20-03739)] Systems that use active illumination techniques have not received a dramatic improvement from the computer vision perspective, indeed, in our opinion, the summarization in [[**20**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B20-sensors-20-03739),[**25**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#B25-sensors-20-03739)] can be considered to be still valid. However, it is worth noting that this dyadic view (systems using active lighting vs systems not using active lighting) is relatively recent with respect to the time in which the early gaze tracking solutions appeared. This is a consequence of the progress of computer vision and its successful exploitation in many assistive fields, as opposed to older solutions based on active lighting that anyway continue to be the reference point in terms of precision. Thus, in [**Section 3**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#sec3-sensors-20-03739), both categories of solutions will be considered, highlighting the moment and how the evolution of gaze estimation based on computer vision, the main subject of this manuscript, has progressively led to the search for solutions that are based on the absence of active/near-infrared illuminators.

Summing up, the main contributions of this document are:

* an update of existing literature on methods based on computer vision, aiming at inferring the gaze;
* a critical review of the impact of deep learning on this research area; and,
* a new broader analysis of gaze tracking approaches from different perspectives.

In the rest of the manuscript, at first, [**Section 2**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#sec2-sensors-20-03739) gives a quick overview of the terms used in the literature and it clarifies some possible ambiguities. Subsequently, [**Section 3**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#sec3-sensors-20-03739) shortly introduces the reader to the history of eye/gaze tracking technology, whereas, [**Section 4**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#sec4-sensors-20-03739) describes the methods followed to select papers/links. The subsequent [**Section 5**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#sec5-sensors-20-03739) provides a scheme to classify gaze tracking techniques and it introduces an innovative taxonomy. Each branch of the proposed scheme is analyzed, and related works reviewed and discussed, in [**Section 6**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#sec6-sensors-20-03739) and [**Section 7**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#sec7-sensors-20-03739). Afterwards, [**Section 8**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#sec8-sensors-20-03739) overviews available related datasets. Metrics and ways to evaluate the different solutions from both functional and non-functional perspectives are introduced in [**Section 9**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#sec9-sensors-20-03739). Afterwards, the new challenges and a glimpse of future research directions are discussed in [**Section 10**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#sec10-sensors-20-03739). [**Section 11**](https://www.mdpi.com/1424-8220/20/13/3739?type=check_update&version=1#sec11-sensors-20-03739) concludes the document.

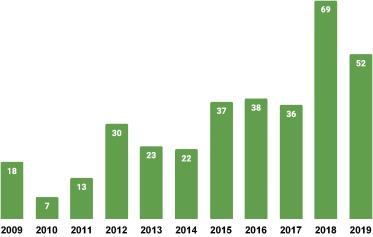
# Pupillary light reflex as a diagnostic aid from computational viewpoint: A systematic literature review

This work presents a detailed and complete review of publications on pupillary light reflex (PLR) used to aid diagnoses. These are computational techniques used in the evaluation of pupillometry, as well as their application in computer-aided diagnoses (CAD) of pathologies or physiological conditions that can be studied by observing the movements of miosis and mydriasis of the human pupil. A careful survey was carried out of all studies published over the last 10 years which investigated, electronic devices, recording protocols, image treatment, computational algorithms and the pathologies related to PLR. We present the frontier of existing knowledge regarding methods and techniques used in this field of knowledge, which has been expanding due to the possibility of performing diagnoses with high precision, at a low cost and with a non-invasive method.

Over the last decades, [pupillometry](https://www.sciencedirect.com/topics/medicine-and-dentistry/pupillometry) has been useful to health professionals. Initially, the method was limited to the use of a flashlight aimed at the pupil to assess impairment of consciousness [[1]](https://www.sciencedirect.com/science/article/pii/S1532046421000861#b0005), [[2]](https://www.sciencedirect.com/science/article/pii/S1532046421000861#b0010). Advances in technology and the development of devices capable of accurately measuring the diameter of the pupil have enabled more sophisticated and accurate assessments. Currently, studies show that pupillary reaction to light can help specialists perform different types of diagnoses.

Diagnosis is the analytical process of examining clinical images to reach a conclusion. This analysis is often supported by exams. These can be images or even the evaluation of metadata generated from laboratory evaluations. As examples, we can mention blood counts, electrocardiograms, retinal exam images and several others. Based on these tests, specialists perform the diagnosis and determine the patient’s health or physiological condition.

A computer-aided detection and diagnosis system (Computer-Aided Detection and Diagnosis - CAD) is a class of computer systems whose objective is to help find and diagnose diseases and provide a “second opinion” in the interpretation of exams [[3]](https://www.sciencedirect.com/science/article/pii/S1532046421000861#b0015). The goal of [CAD](https://www.sciencedirect.com/topics/medicine-and-dentistry/computer-aided-diagnosis) systems is to improve the hit rates of experts and reduce the time required for image interpretation [[4]](https://www.sciencedirect.com/science/article/pii/S1532046421000861#b0020). In this sense, automated pupillometry is a CAD system used for pupil images that can help diagnose pathologies, physiological conditions, and cognitive or emotional status. It also allows the assessment of interest, effort in decision-making, tiredness, fatigue, drug use, and functions of the autonomic system. The practicality and assertiveness of pupillometry has roused the interest of the research community, and thus, a considerable amount of research has been published in recent years to aid diagnostics (see [Fig. 1](https://www.sciencedirect.com/science/article/pii/S1532046421000861#f0005)).



# Pupil Localization Algorithm Based on Improved U-Net Network

Accurately localizing the pupil is an essential requirement of some new human–computer interaction methods. In the past, a lot of work has been done to solve the pupil localization problem based on the appearance characteristics of the eye, but these methods are often specific to the scenario. In this paper, we propose an improved U-net network to solve the pupil location problem. This network uses the attention mechanism to automatically select the contribution of coded and uncoded features in the model during the skip connection stage of the U-net network in the channel and spatial axis. It can make full use of the two features of the model in the decoding stage, which is beneficial for improving the performance of the model. By comparing the sequential channel attention module and spatial attention module, average pooling and maximum pooling operations, and different attention mechanisms, the model was finally determined and validated on two public data sets, which proves the validity of the proposed model.

As computers become increasingly prevalent in modern society, the focus of people’s interaction technology is shifting from the computer as the center to the human center, and cross-domain man–machine barrier technology has become a new research hotspot. Eye movement tracking technology is a novel type of human–computer interaction technology, whose principal interaction mode is staring interaction, though it has also developed many interaction modes. Currently, gaze interaction has been successfully applied in various fields, such as human–machine interface, virtual reality, medical care, and so on. Gaze interaction is divided into two phases, pupil localization, and fixation locus description. Accurate pupil localization is one of the most critical and fundamental requirements of eye-tracking technology, and it is an essential component of the human–computer interaction task.

Pupil localization is subject to many factors, such as the shape of the eye and light conditions. Early eye movement detection products use infrared cameras for detection. In these products, the corneal reflex is used to estimate the pupil center, resulting in high accuracy of pupil position. However, these products have several limitations, such as expensive devices and, most importantly, they are cumbersome to wear and cause eye irritation. In recent years, computer vision has gradually entered the public field of vision and has been widely used in various image tasks. Unlike professional equipment, computer vision technology can directly determine the location of the pupil, which has therefore attracted the attention of researchers. Although detecting pupils using a non-wearable camera is easy in ordinary scenes, obstacles such as image brightness, angle of view, and resolution are still obstacles to improving the accuracy of pupil detection. To address this, a Fully Convolution Network (FCN) has been successfully applied to the semantic segmentation task, which is similar to pupil location. During training, a face picture and a heat map of pupil positions are used as the input to the FCN, with the predicted heat map converted into pupil position during testing. The main framework used in this paper is the U-net network, which is improved through the attention mechanism. This architecture can make full use of coded features and uncoded features in the skip connection stage of the U-net network by using the attention mechanism, rather than simply connecting the two features as in the original paper [[**1**](https://www.mdpi.com/2079-9292/12/12/2591#B1-electronics-12-02591)]. For each stage, the contributions of coded features and uncoded features to the model are different, but the architecture provided in this paper automatically selected them on the channel and spatial axis at each skip connection stage of the U-net network.

Our contributions are as follows:

1. Use of the attention mechanism to connect coded features and uncoded features in the skip connection stage of the U-net network.
2. Our module can not only use the attention mechanism to learn the noteworthy information from both coded features and uncoded features on the channel and spatial axis but also automatically adjust the contribution of coded features and uncoded features to the model by the Softmax function. The model was finally determined by comparing the sequential channel attention module (CAM), spatial attention module (SAM), and average pooling operations and maximum pooling operations, and selecting different attention modules on model performance.
3. The effectiveness of the model has been verified in BioID [[**2**](https://www.mdpi.com/2079-9292/12/12/2591#B2-electronics-12-02591)] and GI4E [[**3**](https://www.mdpi.com/2079-9292/12/12/2591#B3-electronics-12-02591)] data sets and achieved competitive performance.

The rest of this paper is organized as follows. In [**Section 2**](https://www.mdpi.com/2079-9292/12/12/2591#sec2-electronics-12-02591), we review the related work on pupil location, encompassing traditional methods, deep learning, and attention mechanisms. In [**Section 3**](https://www.mdpi.com/2079-9292/12/12/2591#sec3-electronics-12-02591), the structure is presented in detail. In [**Section 4**](https://www.mdpi.com/2079-9292/12/12/2591#sec4-electronics-12-02591), we discuss the sequential of CAM and SAM, the influence of average pooling and maximum pooling, and different attention mechanisms on the model accuracy. We also compared our proposal on two datasets. Finally, in [**Section 5**](https://www.mdpi.com/2079-9292/12/12/2591#sec5-electronics-12-02591), we conclude the manuscript with some final remarks.

# Eye detection using discriminatory Haar features and a new efficient SVM

This paper presents an accurate and efficient eye detection method using the discriminatory Haar features (DHFs) and a new efficient [support vector machine](https://www.sciencedirect.com/topics/computer-science/support-vector-machine) (eSVM). The DHFs are extracted by applying a discriminating feature extraction (DFE) method to the 2D Haar [wavelet transform](https://www.sciencedirect.com/topics/computer-science/wavelet-transforms). The DFE method is capable of extracting multiple discriminatory features for two-class problems based on two novel measure vectors and a new criterion in the whitened principal component analysis (PCA) space. The eSVM significantly improves the computational efficiency upon the conventional SVM for eye detection without sacrificing the [generalization performance](https://www.sciencedirect.com/topics/computer-science/generalization-performance). Experiments on the Face Recognition Grand Challenge (FRGC) database and the BioID face database show that (i) the DHFs exhibit promising classification capability for eye detection problem; (ii) the eSVM runs much faster than the conventional SVM; and (iii) the proposed eye detection method achieves near real-time eye detection speed and better eye detection performance than some state-of-the-art eye detection methods.

Accurate and efficient eye detection is important for building a fully automatic face recognition system [1], [2], [3], [4], and the challenges for finding a robust solution to this problem have attracted much attention in the pattern recognition community [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19]. Example challenges in accurate and efficient eye detection include large variations in image illumination, skin color (white, yellow, and black), facial expression (eyes open, partially open, or closed), as well as scale and orientation. Additional challenges include eye occlusion caused by eye glasses or long hair, and the red eye effect due to the photographic effect. All these challenge factors increase the difficulty of accurate and efficient eye detection.

We present in this paper an accurate and efficient eye detection method using the discriminatory Haar features (DHFs) and a new efficient support vector machine (eSVM). The DHFs are extracted by applying a discriminating feature extraction (DFE) method to the 2D Haar wavelet transform [20], [21], [22]. The DFE method improves upon the popular principal component analysis (PCA) and Fisher linear discriminant (FLD) methods. The PCA method is capable of extracting the optimal features for signal or image representation in the sense of mean square error [23]. However, it does not extract discriminatory features for classification [24]. An alternative to the PCA method is the FLD method, which extracts the optimal features for classification by optimizing a criterion on scatter matrices [23]. However, a significant disadvantage of the FLD method is that the maximum number of features it can derive does not exceed the number of classes minus one [23], [25]. For a two-class pattern classification problem, the FLD method thus can derive at most one feature, which is usually inadequate for achieving satisfactory classification performance, especially when the problem becomes complex, such as the eye detection problem. The DFE method, based on two novel measure vectors and a new criterion, is capable of extracting multiple discriminatory features for eye detection.

The eSVM method is proposed to address the inefficiency problem of the conventional support vector machine (SVM) for eye detection. Since it was introduced, SVM has become a popular tool for two-class classification problems [26], [27], [28], [29], [30], [31]. When the classification problem becomes complex, the conventional SVM tends to generate a large number of support vectors, which subsequently leads to the increase of the model complexity. As a result, SVM becomes less efficient due to the expensive computational cost of the decision function, which involves an inner product of all the support vectors for the linear SVM and a kernel computation of all the support vectors for the kernel SVM. The eSVM, by contrast, significantly reduces the number of support vectors by applying only a single slack variable. In addition, a Θ set is introduced into the definition of the eSVM, which consists of the training samples on the wrong side of their margin derived from the conventional SVM. The Θ set plays an important role in controlling the generalization performance. The eSVM therefore improves the computational efficiency upon the conventional SVM without sacrificing the generalization performance.

Fig. 1 shows the architecture of our eye detection method. First, we apply the Bayesian Discriminating Features (BDF) method [32] to detect a face from an image and normalize the detected face to a predefined size. Second, we use some geometric constraints to extract an eye strip from the upper portion of the detected face. Illumination variations are then attenuated by means of an illumination normalization procedure that consists of Gamma correction, difference of Gaussian filtering, and contrast equalization as applied in [33], [34]. Third, the DFE method applies the 2D Haar basis functions to derive the discriminatory Haar features (DHFs). Finally, the eSVM classifier applies the DHFs features for eye detection. Usually there are multiple detections around the pupil center. The average of these multiple detections is eventually chosen as the eye location.

We evaluate our proposed eye detection method on the Face Recognition Grand Challenge (FRGC) database [35], [36] and the BioID face database [10]. Experimental results show that (i) the DHFs exhibit promising classification capability for the eye detection problem; (ii) the eSVM runs much faster than the conventional SVM; and (iii) the proposed eye detection method achieves near real-time eye detection speed and better eye detection performance than some state-of-the-art eye detection methods.

# Etracker: A Mobile Gaze-Tracking System with Near-Eye Display Based on a Combined Gaze-Tracking Algorithm

Eye tracking technology has become increasingly important for psychological analysis, medical diagnosis, driver assistance systems, and many other applications. Various gaze-tracking models have been established by previous researchers. However, there is currently no near-eye display system with accurate gaze-tracking performance and a convenient user experience. In this paper, we constructed a complete prototype of the mobile gaze-tracking system ‘*Etracker*’ with a near-eye viewing device for human gaze tracking. We proposed a combined gaze-tracking algorithm. In this algorithm, the convolutional neural network is used to remove blinking images and predict coarse gaze position, and then a geometric model is defined for accurate human gaze tracking. Moreover, we proposed using the mean value of gazes to resolve pupil center changes caused by nystagmus in calibration algorithms, so that an individual user only needs to calibrate it the first time, which makes our system more convenient. The experiments on gaze data from 26 participants show that the eye center detection accuracy is 98% and *Etracker* can provide an average gaze accuracy of 0.53° at a rate of 30–60 Hz.

In recent years, eye tracking has become an important research topic in computer vision and pattern recognition, because the human gaze positions are essential information for many applications including human–computer interaction (HCI) [[**1**](https://www.mdpi.com/1424-8220/18/5/1626#B1-sensors-18-01626),[**2**](https://www.mdpi.com/1424-8220/18/5/1626#B2-sensors-18-01626)], driver assistance [[**3**](https://www.mdpi.com/1424-8220/18/5/1626#B3-sensors-18-01626)], optometry, market data analysis, and medical diagnosis. There have been studies using human eye movements for strabismus examination [[**4**](https://www.mdpi.com/1424-8220/18/5/1626#B4-sensors-18-01626)]. Parkinson’s disease can also be detected on the basis of human eye blinking. In our previous work [[**5**](https://www.mdpi.com/1424-8220/18/5/1626#B5-sensors-18-01626)], we proposed diagnosing developmental coordination disorders (DCD) by detecting changes in patients’ gaze positions and body motion.

The fundamental issues of gaze-tracking technology include tracking system, tracking algorithms, and user experiences. These three issues are closely related to each other. A tracking system may consist of eye cameras, display devices or front-facing cameras, and processing units. Tracking algorithms may include eye region detection, gaze position detection, gaze mapping algorithms, and calibration algorithms, depending on the tracking system. User experiences are highly dependent on the tracking systems and algorithms. Most of the current gaze-tracking systems are either table-mounted or mobile systems. A table-mounted system usually works with an external display screen, which makes human–computer interaction convenient, but it is not robust to head movement. A mobile system is robust to significant head movements, but it is not easy to conduct human–computer interaction with its front-facing camera. A near-eye viewing system combines the advantages of both. To the best of our knowledge, there is no complete prototype for such a system.

The fundamental problem of tracking algorithms is to track human eye regions and gaze positions accurately. The cameras are sensitive to light variations and shooting distance, which makes the human eyes very eccentric in the recorded images. In addition, illumination changes, blinking, eyelids, and eyelashes make accurate gaze tracking very challenging. A robust gaze-tracking algorithm is supposed to have stable performance in different environments and efficiently meet the needs of various applications. From the perspective of user experience, the current gaze-tracking systems still have some problems, such as being inconvenient to use, occluding participants’ field of view, a complicated operator interface, and the need to calibrate it before every use. Moreover, most of the current commercial systems are quite expensive. These factors limit the applications of gaze-tracking technology in various research topics.

Regarding the above issues, we propose a mobile gaze-tracking system with a near-eye viewing device, reliable tracking algorithms, and a convenient user experience. The main contributions of our research compared to previous works are as follows.

* We create a complete prototype of an efficient, easy-to-use, and inexpensive mobile gaze-tracking system, *Etracker*. Compared to existing gaze-tracking systems [[**6**](https://www.mdpi.com/1424-8220/18/5/1626#B6-sensors-18-01626),[**7**](https://www.mdpi.com/1424-8220/18/5/1626#B7-sensors-18-01626),[**8**](https://www.mdpi.com/1424-8220/18/5/1626#B8-sensors-18-01626),[**9**](https://www.mdpi.com/1424-8220/18/5/1626#B9-sensors-18-01626)], *Etracker* is small, lightweight, unobtrusive, and user-friendly. It records eye movements and computes gaze positions in real time.
* We use a novel near-eye viewing device in the gaze-tracking system, which replaces the traditional large display devices, e.g., computer monitors, TVs, and projectors. The near-eye viewing device has a millimeter-sized display chip with a resolution of 1024×7201024×720 pixels and displays a size of 35 × 2535 × 25 cm2 virtual image at a distance of 0.5 m from the human eyes.
* We propose a combined gaze estimation method based on CNNs (ResNet-101) and a geometric model. The CNN is used to remove the blinking eye images and locate the coarse gaze position, and the accurate gaze positions are detected by a geometric model. The gaze accuracy can reach 0.53°.
* We propose using the mean value of pupil centers to smooth the changes caused by nystagmus in calibration algorithms. Therefore, an individual user only needs to calibrate it the first time.

The rest of this paper is organized as follows. In [**Section 2**](https://www.mdpi.com/1424-8220/18/5/1626#sec2-sensors-18-01626), a review of previous works on gaze-tracking systems and eye detection algorithms are presented. The proposed gaze-tracking device and algorithms are described in detail in [**Section 3**](https://www.mdpi.com/1424-8220/18/5/1626#sec3-sensors-18-01626). In [**Section 4**](https://www.mdpi.com/1424-8220/18/5/1626#sec4-sensors-18-01626), the analysis and explanations of the results are presented. Finally, some discussions and conclusions are drawn in [**Section 5**](https://www.mdpi.com/1424-8220/18/5/1626#sec5-sensors-18-01626).

# When I Look into Your Eyes: A Survey on Computer Vision Contributions for Human Gaze Estimation and Tracking

The automatic detection of eye positions, their temporal consistency, and their mapping into a line of sight in the real world (to find where a person is looking at) is reported in the scientific literature as gaze tracking. This has become a very hot topic in the field of computer vision during the last decades, with a surprising and continuously growing number of application fields. A very long journey has been made from the first pioneering works, and this continuous search for more accurate solutions process has been further boosted in the last decade when deep neural networks have revolutionized the whole machine learning area, and gaze tracking as well. In this arena, it is being increasingly useful to find guidance through survey/review articles collecting most relevant works and putting clear pros and cons of existing techniques, also by introducing a precise taxonomy. This kind of manuscripts allows researchers and technicians to choose the better way to move towards their application or scientific goals. In the literature, there exist holistic and specifically technological survey documents (even if not updated), but, unfortunately, there is not an overview discussing how the great advancements in computer vision have impacted gaze tracking. Thus, this work represents an attempt to fill this gap, also introducing a wider point of view that brings to a new taxonomy (extending the consolidated ones) by considering gaze tracking as a more exhaustive task that aims at estimating gaze target from different perspectives: from the eye of the beholder (first-person view), from an external camera framing the beholder’s, from a third-person view looking at the scene where the beholder is placed in, and from an external view independent from the beholder.

Gaze is a fundamental communication mean, since it can express emotions, feelings and intentions [[**1**](https://www.mdpi.com/1424-8220/20/13/3739#B1-sensors-20-03739)]. Evidence of its importance is in the noticeable quantity of applications that have been presented in the last few decades, spacing from Human-Computer and Human-Robot Interaction (HCI/HRI) [[**2**](https://www.mdpi.com/1424-8220/20/13/3739#B2-sensors-20-03739),[**3**](https://www.mdpi.com/1424-8220/20/13/3739#B3-sensors-20-03739)] to assistive devices [[**4**](https://www.mdpi.com/1424-8220/20/13/3739#B4-sensors-20-03739),[**5**](https://www.mdpi.com/1424-8220/20/13/3739#B5-sensors-20-03739),[**6**](https://www.mdpi.com/1424-8220/20/13/3739#B6-sensors-20-03739)], healthcare/clinical assessment and diagnosis [[**7**](https://www.mdpi.com/1424-8220/20/13/3739#B7-sensors-20-03739)], driver vigilance monitoring [[**8**](https://www.mdpi.com/1424-8220/20/13/3739#B8-sensors-20-03739),[**9**](https://www.mdpi.com/1424-8220/20/13/3739#B9-sensors-20-03739)], analysis of consumer market [[**10**](https://www.mdpi.com/1424-8220/20/13/3739#B10-sensors-20-03739)], analysis of behavioral patterns in disabilities or diseases [[**11**](https://www.mdpi.com/1424-8220/20/13/3739#B11-sensors-20-03739),[**12**](https://www.mdpi.com/1424-8220/20/13/3739#B12-sensors-20-03739)], gaming design [[**13**](https://www.mdpi.com/1424-8220/20/13/3739#B13-sensors-20-03739),[**14**](https://www.mdpi.com/1424-8220/20/13/3739#B14-sensors-20-03739)], and so on. Gaze tracking consists of the procedure of obtaining the direction or the point of regard (PoR) of the gaze with continuity, on a digital screen or in the physical environment, and it is achieved through mechanical, electronic, optical, and/or other methods.

Although researchers often use the terms of eye tracking and gaze tracking as synonyms, there is a slight difference between them. In particular, eye tracking is the measurement of eye movement/activity, while gaze tracking is the analysis of eye tracking data with regards to the head or the visual scene (a known three-dimensional (3D) environment, a screen, a surface, etc.). In other words, eye tracking consists in detecting the existence and position of the eyes into an image and to track them over time (in the following images), and then in measuring activities as eye blinks, fixations, saccadic movements [[**15**](https://www.mdpi.com/1424-8220/20/13/3739#B15-sensors-20-03739)], pupil dilation [[**16**](https://www.mdpi.com/1424-8220/20/13/3739#B16-sensors-20-03739)], etc. The term gaze tracking instead refers to the estimation (and the temporal tracking) of where a person is looking at, in terms of interaction with the scene, and it is obtained by determining the 3D line of sight between the user and the target. A common practice is to consider the gaze tracker as the last block of a system that starts by localizing and tracking the eyes [[**17**](https://www.mdpi.com/1424-8220/20/13/3739#B17-sensors-20-03739)]. On the other hand, some works define gaze trackers as a subset of the eye tracker families [[**18**](https://www.mdpi.com/1424-8220/20/13/3739#B18-sensors-20-03739)].

In this arena, it is being increasingly useful to find guidance trough exhaustive survey/review articles collecting most relevant works and putting clear the pros and cons of existing techniques, also by introducing a precise taxonomy. This kind of manuscripts allows for researchers and technicians to choose the better way to move towards their application or scientific goals. Valuable attempts to give a review of the existing literature have been provided in [[**19**](https://www.mdpi.com/1424-8220/20/13/3739#B19-sensors-20-03739),[**20**](https://www.mdpi.com/1424-8220/20/13/3739#B20-sensors-20-03739)]. These manuscripts crystallized the state of the art in 2009 and 2013, respectively, and they gave a holistic view of hardware, user-interaction, eye detection, and gaze mapping techniques. Afterward, a plethora of new works have appeared in the literature, and then it has become more useful and fruitful to focus on every single aspect of the process rather than on the entire algorithmic pipeline. This led authors in [[**21**](https://www.mdpi.com/1424-8220/20/13/3739#B21-sensors-20-03739)] to review methods for eye region and pupil detection and localization, whereas, in [[**22**](https://www.mdpi.com/1424-8220/20/13/3739#B22-sensors-20-03739)], the gaze mapping functions based on interpolation or approximation that determine, starting from the pupil glint, the coordinates on a screen, are summarized. Works dealing with full-face appearance-based gaze estimation have been discussed in [[**23**](https://www.mdpi.com/1424-8220/20/13/3739#B23-sensors-20-03739)] and an overview of key technologies of gaze tracking has been proposed in [[**24**](https://www.mdpi.com/1424-8220/20/13/3739#B24-sensors-20-03739)] instead. Insight into the issues related to algorithms, system configurations, user conditions, and performance has been finally provided in [[**25**](https://www.mdpi.com/1424-8220/20/13/3739#B25-sensors-20-03739)]. However, the recent advancements reached in computer vision, and more generally in artificial intelligence, boosted gaze tracking technology than any other scientific field. Unfortunately, how these great advancements in computer vision and pattern recognition have impacted gaze estimation and tracking has not been analyzed yet. This paper attempts to fill this gap and it also tries to give a broader analysis about gaze tracking that brings to a new taxonomy (extending the consolidated ones), established by considering gaze tracking as a more exhaustive task that has to estimate gaze targets from different points of view:

* from the eye of the beholder (first-person view);
* from an external camera framing the beholder’s;
* from a third-person view looking at the scene where the beholder is placed in; and,
* from an external view independent from the beholder.

Active researches and the most relevant advancements are introduced and discussed, while taking the aforementioned innovative points of view on this largely debated scientific topic into account. It should be clarified that, in this work, the systems that use active illumination techniques will not be reviewed, unless they provide some innovation in terms of the computer vision technique, or in the case of more complete architectures where infrared only represents a fraction of the contribution. Indeed, this is a common and well-established technology, already on the market and with many patented approaches, although they still present their own specific challenges to be addressed and some research lines are still open, especially for improving electronic lighting components [[**26**](https://www.mdpi.com/1424-8220/20/13/3739#B26-sensors-20-03739)] Systems that use active illumination techniques have not received a dramatic improvement from the computer vision perspective, indeed, in our opinion, the summarization in [[**20**](https://www.mdpi.com/1424-8220/20/13/3739#B20-sensors-20-03739),[**25**](https://www.mdpi.com/1424-8220/20/13/3739#B25-sensors-20-03739)] can be considered to be still valid. However, it is worth noting that this dyadic view (systems using active lighting vs systems not using active lighting) is relatively recent with respect to the time in which the early gaze tracking solutions appeared. This is a consequence of the progress of computer vision and its successful exploitation in many assistive fields, as opposed to older solutions based on active lighting that anyway continue to be the reference point in terms of precision. Thus, in [**Section 3**](https://www.mdpi.com/1424-8220/20/13/3739#sec3-sensors-20-03739), both categories of solutions will be considered, highlighting the moment and how the evolution of gaze estimation based on computer vision, the main subject of this manuscript, has progressively led to the search for solutions that are based on the absence of active/near-infrared illuminators.

Summing up, the main contributions of this document are:

* an update of existing literature on methods based on computer vision, aiming at inferring the gaze;
* a critical review of the impact of deep learning on this research area; and,
* a new broader analysis of gaze tracking approaches from different perspectives.

In the rest of the manuscript, at first, [**Section 2**](https://www.mdpi.com/1424-8220/20/13/3739#sec2-sensors-20-03739) gives a quick overview of the terms used in the literature and it clarifies some possible ambiguities. Subsequently, [**Section 3**](https://www.mdpi.com/1424-8220/20/13/3739#sec3-sensors-20-03739) shortly introduces the reader to the history of eye/gaze tracking technology, whereas, [**Section 4**](https://www.mdpi.com/1424-8220/20/13/3739#sec4-sensors-20-03739) describes the methods followed to select papers/links. The subsequent [**Section 5**](https://www.mdpi.com/1424-8220/20/13/3739#sec5-sensors-20-03739) provides a scheme to classify gaze tracking techniques and it introduces an innovative taxonomy. Each branch of the proposed scheme is analyzed, and related works reviewed and discussed, in [**Section 6**](https://www.mdpi.com/1424-8220/20/13/3739#sec6-sensors-20-03739) and [**Section 7**](https://www.mdpi.com/1424-8220/20/13/3739#sec7-sensors-20-03739). Afterwards, [**Section 8**](https://www.mdpi.com/1424-8220/20/13/3739#sec8-sensors-20-03739) overviews available related datasets. Metrics and ways to evaluate the different solutions from both functional and non-functional perspectives are introduced in [**Section 9**](https://www.mdpi.com/1424-8220/20/13/3739#sec9-sensors-20-03739). Afterwards, the new challenges and a glimpse of future research directions are discussed in [**Section 10**](https://www.mdpi.com/1424-8220/20/13/3739#sec10-sensors-20-03739). [**Section 11**](https://www.mdpi.com/1424-8220/20/13/3739#sec11-sensors-20-03739) concludes the document.

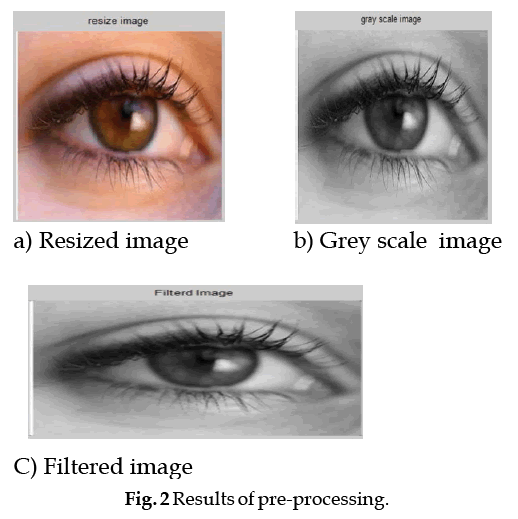
# The Motor Action Analysis Based on Deep Learning

For the slow speed and low accuracy of slow motor action recognition methods, this study proposes a motor action analysis method based on the CNN network and the softmax classification model. First, in order to obtain motor action feature information, by using static spatial features of BN-inception based on CNN network extracted actions and high-dimensional features of 3D ConvNet, then based on softmax classifier structure and realizing taxonomic recognition of the motor actions. Finally, through the decision-layer fusion and time semantic continuity optimization strategy, the motion action recognition accuracy is further improved and the more efficient motion action classification recognition is realized. The results show that the proposed method can complete the motor action analysis and achieve the classification recognition accuracy to 83.11%, which has certain practical value.

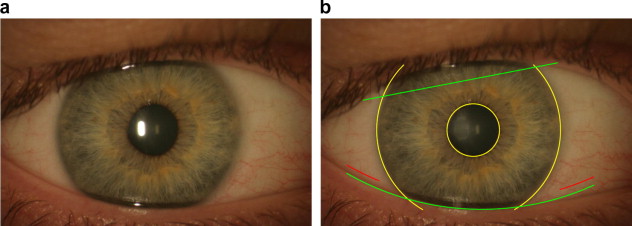
Movement action analysis is an important branch of computer vision, which also involves data mining, image processing, and other content, and is widely used in sports, music playing, and many other scenes. Due to the complex patterns of movement action and the big differences in movement rules of different individuals, the movement action recognition analysis is somewhat challenging and has attracted the keen attention of relevant researchers. At present, motion action analysis mainly focuses on motion detection and recognition and has achieved remarkable research results. For example, Hua-xin Zhang et al. realized the estimation of human posture by capturing 3D motion [1]. In addition, Xiaoqiang Li et al. applied the convolutional neural network to action recognition. The results show that the action recognition results of a convolutional neural network with the dual-attention mechanism are comparable to the recognition results of the latest algorithm [2]. Haohua Zhao et al. extracted intraframe feature vectors by deep network training to form a multimode feature matrix. The matrix is input into CNN to achieve feature classification. The results show that the proposed method has better performance than the existing LSTM in video action recognition [3]. Ran Cui et al. analyzed the motion by constructing skeletal joints and static and dynamic features. The prediction of motion is realized through motion recognition [4]. Manikandaprabu et al. detected the ROI of the human body using the combination of background subtraction and frame subtraction [5]. Then the CAMShift algorithm is adopted for recognition. The results show that this method has good precision and has great advantages compared with the most advanced algorithms. It can be seen from the above studies that convolutional neural networks are widely used in action recognition, among which the CNN attracts more attention due to its unique characteristics.

Despite the great progress in motor motion analysis, its overall performance still needs to be improved, mainly due to the blurred boundary of motor motion, which increases the difficulty of the study. For the difficulties, this study applies powerful deep learning capabilities, based on the CNN network and the softmax classifier, and proposes a deep learning-based motion action analysis method.

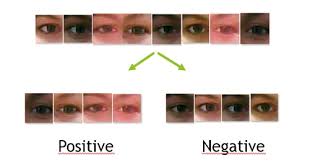
**SYSTEM ARCHITECTURE**











**MODULES:**

* Image Acquisition
* Pre-processing
* Segmentation
* Feature extraction
* Classification

**MODULES DESCSRIPTION**

**Image Acquisition**

* The "Image Acquisition" module serves as the foundational element of the proposed "Driver Drowsiness Detection System Using Image Processing." In this pivotal phase, the system initiates the process of collecting real-time visual data from the interior environment of the vehicle, with a specific focus on capturing the driver's face and eyes.
* Within the realm of driver drowsiness detection, the significance of image acquisition cannot be overstated. The quality, consistency, and timeliness of the data acquired at this stage form the bedrock upon which all subsequent analysis and decision-making are built. To achieve accurate drowsiness detection, it is imperative to capture detailed and representative images of the driver's facial features.
* Ultimately, the image acquisition module lays the foundation for the entire drowsiness detection system. Its ability to capture high-quality, real-time visual data is instrumental in facilitating subsequent stages of image processing, segmentation, feature extraction, and classification. This initial phase, therefore, plays a critical role in enhancing road safety by providing the system with the essential information needed to make accurate determinations regarding the driver's state of alertness.

**Pre-processing**

* The "Pre-processing" module represents the critical second phase within the proposed "Driver Drowsiness Detection System Using Image Processing." In this integral step, the system takes the raw image data acquired from the vehicle's interior environment and prepares it for subsequent analysis. This phase is paramount in enhancing the efficiency and effectiveness of the drowsiness detection process.
* The initial task undertaken in the pre-processing module is the conversion of the acquired images into grayscale. Grayscale conversion serves a dual purpose: it simplifies the computational complexity of the subsequent stages, making the analysis more efficient, and it retains essential facial and eye details. By converting images to grayscale, the system removes the color dimension, focusing solely on luminance, which is instrumental in detecting features such as eye movements and facial expressions.
* Simultaneously, the pre-processing module encompasses advanced algorithms for eye detection. The precision and accuracy of eye detection are pivotal in the system's ability to isolate the region of interest—the driver's eyes—within the images. The algorithms employed are designed to identify the eyes within the grayscale images with a high degree of accuracy. This stage forms the basis for subsequent segmentation and feature extraction processes.
* Additionally, this module plays a crucial role in handling variations in lighting conditions and potential image artifacts. It employs techniques to enhance image quality, reduce noise, and ensure consistency in the captured data. This robustness is essential for the system to perform reliably in a variety of real-world scenarios, including day and night driving, varying weather conditions, and different vehicle interiors.
* In summary, the pre-processing module serves as the gateway between raw image data and the subsequent stages of analysis. Its tasks of grayscale conversion and eye detection set the stage for precise and efficient drowsiness detection. By simplifying the data while preserving essential facial and eye features, this module ensures that the system can make accurate determinations regarding the driver's state of alertness in real-time.

**Segmentation**

* The "Segmentation" module represents a pivotal phase within the proposed "Driver Drowsiness Detection System Using Image Processing." In this module, the system takes the pre-processed grayscale images and delves into the intricate process of isolating the specific regions of interest within the images. The primary objective here is to segment and extract the driver's iris, a key area for further analysis.
* Iris segmentation serves as the bridge between the preliminary image processing steps and the subsequent stages of feature extraction and classification. Accurate iris segmentation is vital, as it lays the foundation for the system's ability to detect subtle changes and patterns within the driver's eyes, which are indicative of drowsiness.
* The segmentation process is designed to identify and delineate the boundaries of the iris within each eye. This is achieved through a combination of advanced image processing techniques and algorithms. These techniques analyze the variations in pixel intensity, texture, and edge information to accurately locate the iris. It's worth noting that the iris is chosen as a region of interest due to its unique and stable characteristics, making it suitable for biometric analysis.
* Furthermore, the segmentation module ensures that the extracted iris regions are of uniform size and orientation, facilitating consistent and reliable feature extraction in the subsequent phase. Any distortions or irregularities in the segmented iris regions are corrected to minimize variability and enhance the system's overall accuracy.
* Beyond isolating the iris, this module also handles issues related to occlusions, eyelid movements, and partial iris visibility. Robust algorithms are employed to account for these factors, ensuring that even under real-world conditions, the system can effectively segment and extract the iris regions.
* In essence, the segmentation module serves as the critical intermediary step in the drowsiness detection process. Its ability to precisely identify and extract the iris from the pre-processed images enables subsequent stages to focus exclusively on this key area for feature extraction and classification. The accuracy and reliability of iris segmentation are essential in ensuring that the system can detect even subtle signs of drowsiness with precision, ultimately contributing to enhanced road safety.

**Feature extraction**

* The "Feature Extraction" module constitutes a pivotal phase within the proposed "Driver Drowsiness Detection System Using Image Processing." In this module, the system delves into the intricate process of analyzing the segmented iris regions, extracting discriminative features that are essential for determining the driver's state of alertness.
* Feature extraction is a critical step in the drowsiness detection pipeline, as it transforms raw iris data into a concise and informative representation. The extracted features serve as the basis for subsequent classification, enabling the system to differentiate between alert and drowsy states accurately.
* One of the primary techniques employed in feature extraction within this module is the Discrete Cosine Transform (DCT). DCT is a powerful tool for capturing the frequency characteristics of the iris texture. It decomposes the segmented iris region into a set of coefficients, highlighting the dominant spatial frequencies present in the texture. These coefficients, often referred to as DCT coefficients, are used as feature vectors that encapsulate unique patterns within the iris.
* In addition to DCT, the module leverages the Speeded-Up Robust Features (SURF) algorithm. SURF is adept at identifying distinctive points and patterns within images. By applying SURF to the iris region, the system extracts additional features that complement those obtained through DCT. This combination of DCT and SURF allows for a rich and diverse feature set, increasing the system's sensitivity to subtle variations in the iris texture.
* In summary, the feature extraction module transforms segmented iris data into a compact yet informative feature representation. By utilizing DCT, SURF, and other techniques, it captures the subtle patterns and characteristics of the iris, enabling subsequent classification modules to make precise determinations regarding the driver's state of alertness. The efficiency and effectiveness of feature extraction contribute significantly to the system's overall accuracy and reliability in drowsiness detection.

**Classification**

* The "Classification" module represents a pivotal phase within the proposed "Driver Drowsiness Detection System Using Image Processing." In this module, the system leverages machine learning algorithms, specifically the K-Nearest Neighbors (KNN) and Random Forest classifiers, to make informed decisions regarding the driver's state of alertness based on the extracted iris features.
* Classification is the stage at which the system's intelligence is brought into play, as it employs trained models to categorize the feature vectors extracted from the segmented iris regions. The primary objective is to differentiate between two key states: an alert driver with open eyes and a drowsy driver with potentially drooping or closed eyes.
* K-Nearest Neighbors (KNN) is a machine learning algorithm used within this module to classify feature vectors. It operates on the principle of proximity, where each feature vector is compared to its k nearest neighbors in the feature space. By assessing the class labels of these neighbors, KNN assigns a class label to the feature vector in question. KNN is known for its simplicity and effectiveness, making it a valuable component of the system's classification strategy.
* In parallel, the system employs the Random Forest classifier, a powerful ensemble learning method. Random Forest operates by constructing multiple decision trees during the training phase and combining their outputs during classification. This ensemble approach enhances the system's robustness, as it can capture complex relationships and patterns within the data. Random Forest is particularly adept at handling high-dimensional feature spaces, which is beneficial in the context of iris feature vectors.
* Furthermore, this module introduces an innovative concept: the Fusion Score. The Fusion Score is calculated as the weighted average of the KNN and Random Forest classifier scores, using the formula (KNN + RF) / 2. This fusion mechanism harnesses the collective intelligence of both classifiers, reducing the likelihood of misclassification. The Fusion Score is employed to make the final determination regarding the driver's state of alertness.
* In summary, the classification module is the brain of the drowsiness detection system, using machine learning algorithms like KNN and Random Forest to make informed decisions based on extracted iris features. Its robustness, adaptability, and the introduction of the Fusion Score contribute significantly to the system's ability to accurately distinguish between alert and drowsy states, ultimately enhancing road safety.

**SYSTEM REQUIREMENTS**

**HARDWARE REQUIREMENTS**

* System : Pentium i3 Processor.
* Hard Disk : 500 GB.
* Monitor : 15’’ LED.
* Input Devices : Keyboard, Mouse.
* Ram : 8 GB.

**SOFTWARE REQUIREMENTS**

* Operating system : Windows 10 Pro.
* Coding Language : MATLAB
* Tool : MATLAB R2023b

**BASE PAPER COMPLETE MATTER**

**RELATED WORKS**

1. **Traditional Techniques**

Traditional techniques relied heavily on the geometrical structure of the eye to detect the center eye position in an image or video. The work in [4] applied the Isophotes Curvature Estimation technique that determined the center of the eye and then built a voting system for that point. The authors in [5] introduced a method using an ensemble of randomized regression trees for locating the pupil of the eye. The method in [6] localized the center of the eye based on the image gradient and applied a squared dot product between the center candidates and the image gradient. The research in [7] used the Haar-like feature combined with a K-means cluster to navigate the pupil and fit an ellipse into the pupil using the Random Sample Consensus technique. The approach in [8] proposed an inner product detector based on correlation filters for detecting pupil landmarks. Traditional approaches applied computationally complex techniques and were very sensitive to environmental lighting conditions and facial occlusion, although they provided useful information.

**B. Machine Learning Techniques**

The eye is an important organ of the driver body and its status can reflect the first stage of drowsiness. From that observation, many machine learning studies have achieved high efficiency by focusing on exploiting and surveillance eye location. Several basic machine learning methods have been applied, such as the Haar wavelet and support vector machine (SVM) [9], histogram of oriented gradient-based SVM [10], self-similarity information combined with shape analysis [11], and Viola-Jones [12]. These techniques can overcome the disadvantages of traditional methods and achieve higher accuracy. However, they also need to be used in conjunction with other modern methods for implementation in real-time systems. Nowadays, the impressive advances of machine learning have made the development of computer vision applications based on CNNs more and more popular. In this trend, eye and pupil detection also used deep and complicated CNNs for improving extraction features and learning them. These CNNs were combined with multiple tasks to fully exploit the ability to learn important information from feature maps. Khan et al. [13] achieved 95% accuracy using the eyelid curvature angle as the basis for determining whether the eye is closed or open. The researchers in [14] considered drowsiness detection as an object detection task by identifying and detecting closed or open eyes. This work used a combination of MobileNet with single shot multibox detector (SSD) and achieved an average precision of 84%. The authors in [15] used the facial landmark technique to calculate the eye aspect ratio and eye closure ratio for drowsiness detection with 84% of accuracy. The study in [16] used an ensemble of two lightweight convolutional neural networks to extract and classify the eye patch with high accuracy. Another work [17] proposed a 4-D model for eye categorization by improving the VGG architecture family. In summary, machine learning methods exhibited outstanding advantages in feature extraction and eye location detection. In general, these methods focused on exploiting the features of the eyes and facial organs related to the sleepy state, such as the eyelids and mouth. Meanwhile, these organs were quite small in the image and lacked specialized datasets for detection tasks. It required complex techniques, high computational costs to accurately detect those locations, and high labor costs to annotate the datasets. Therefore, to simplify eye position detection and reduce the computational cost of the system, this article proposed an effective method of eye location detection and eye classification based on lightweight CNN architectures. Besides, this work also provides the datasets for eye position detection, which is popularly used in the object detection field.

**PROPOSED METHODOLOGY**

**A.Eye Detection Network**

The detailed architecture description of this network is shown in Fig.

**It is built with four following modules:**

Shrinking, inception, triplet attention, and detection modules.

**1)Shrinking Module:**

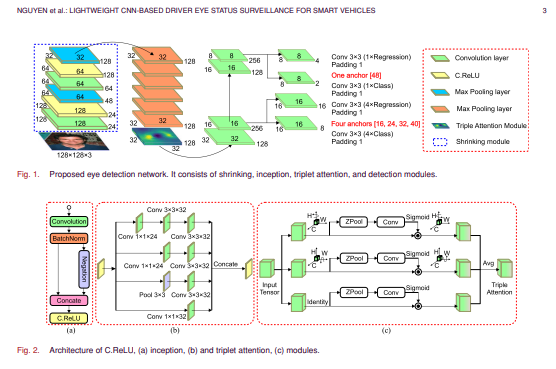
This module mainly uses 3 × 3 convolution layers and strides of 1, 2, 1, 2 in the convolution and max pooling layers, respectively. It helps to extract the basic features of the eye and reduce the input image from 128 × 128 to 32 × 32. In addition, the concatenated rectified linear unit (C.ReLu) module [18] is also applied to increase the efficiency of feature extraction. C.ReLu is described, as shown in Fig. 2(a).

**2) Inception Module:**

The inception module is made up of six inception layers [19]. Each inception layer contains four convolution branches. Each branch uses one to three 1 × 1 or 3 × 3 convolution layers. Following each convolution layer is the batch normalization (BN) and the rectified linear unit (ReLU) activation function. In addition, the second branch adds one more max pooling layer to extract feature that is different from the remaining branches. With the idea of expanding the network width using multiple scales, this module increases the receptive field for the network. It maintains the scaling of the input feature map dimension 32 × 32 and provides output informative feature maps. Fig. 2(b) shows the structure of the inception layer.

**3)Triplet Attention Module:**

The triplet attention module [20] consists of three branches that are shown in Fig. 2(c). It assumes that the input tensor is F ∈ RC×H×W and passes it

****

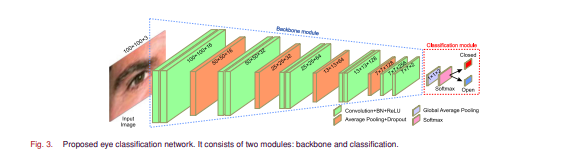
through each branch in this module. The first branch considers the interaction between the width dimension and the channel dimension of the feature map by applying a 90◦ anticlockwise tensor rotation along the H-axis and the rotated tensor annotated by F1 1 ∈ RW×H×C . After that, F1 1 go through the ZPool layer and the dimension of F1 1 is reduced to F2 1 ∈ R2×H×C . Then, F2 1 is passed to a 1 × 1 convolution layer followed by the BN layer, which generates the output tensor of (1 × H × C). This output goes through the sigmoid function σ to obtain the attention weights and applies them to the feature mapF1 1 using the channel element-wise multiplication operation. Then, the attention feature map rotates 90◦ clockwise along the H-axis to get the output feature map, which is the same shape as the input feature map F. Similarly, the second branch has the same architecture as the first branch except that it focuses on the relationship between the height dimension and the channel dimension. Hence, it applies a 90◦ tensor rotation to the W-axis. The last branch also uses the architecture of the first two branches but does not use tensor rotation. The results from the three branches were aggregated by simple averaging. Therefore, the final output of this module is an attention feature map of size (C × H × W). The process of the attention module can be presented as

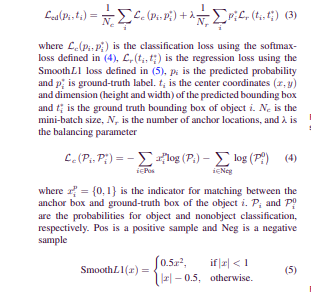
****

where σ is sigmoid activation function; φ1, φ2, and φ3 are 1 × 1 convolution layers followed by the BN layer. is 90◦ clockwise rotation to retain original input shape. The ZPool layer is presented as follows: ZPool(F)=[M0d(F), A0d(F)] (2) where 0d is 0th-dimension in which the max pooling (M) and average pooling (A) layers apply in the concatenation operation [·]. The ZPool layer takes an input tensor of shape (C × H × W) and generates an output tensor of shape (2 × H × W). As can be seen, the triple attention module considers the input feature map tensor on all three dimensions, increasing the network’s ability to focus on the prominent features of the human eye image. 4) Detection Module: First, this module uses four 1 × 1 and 3 × 3 convolution layers to further reduce the dimension of the feature maps and produce four feature maps of 32 × 32 × 128, 16 × 16 × 256, 16 × 16 × 128, and 8 × 8 × 256. From these output feature maps, the network only chooses two feature maps with dimensions of 16 × 16 × 256 and 8 × 8 × 256 as inputs for the detection module to detect eyes at two different levels. Detectors use two sibling convolution layers for classification and bounding box regression. To predict a bounding box, these detectors use different predefined square anchors of 16, 24, 32, 40 for small and medium eyes, and 48 for large eyes. The regression head generates a four-dimensional vector(x, y, w, h) as the offset of the bounding box location. The classification head generates a two-dimensional (eye or not-eye) vector as the label classification.

**1)Loss Function:**

The loss function consists of classification loss and regression loss. The entire loss function is defined

****

****

# K-NEAREST NEIGHBOR(KNN) ALGORITHM FOR MACHINE LEARNING

* K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique.
* K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.
* K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.
* K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.
* K-NN is a **non-parametric algorithm**, which means it does not make any assumption on underlying data.
* It is also called a **lazy learner algorithm** because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.
* KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.
* **Example:** Suppose, we have an image of a creature that looks similar to cat and dog, but we want to know either it is a cat or dog. So for this identification, we can use the KNN algorithm, as it works on a similarity measure. Our KNN model will find the similar features of the new data set to the cats and dogs images and based on the most similar features it will put it in either cat or dog category.



## Why do we need a K-NN Algorithm?

Suppose there are two categories, i.e., Category A and Category B, and we have a new data point x1, so this data point will lie in which of these categories. To solve this type of problem, we need a K-NN algorithm. With the help of K-NN, we can easily identify the category or class of a particular dataset. Consider the below diagram:



## How does K-NN work?

The K-NN working can be explained on the basis of the below algorithm:

* **Step-1:** Select the number K of the neighbors
* **Step-2:** Calculate the Euclidean distance of **K number of neighbors**
* **Step-3:** Take the K nearest neighbors as per the calculated Euclidean distance.
* **Step-4:** Among these k neighbors, count the number of the data points in each category.
* **Step-5:** Assign the new data points to that category for which the number of the neighbor is maximum.
* **Step-6:** Our model is ready.

Suppose we have a new data point and we need to put it in the required category. Consider the below image:



* Firstly, we will choose the number of neighbors, so we will choose the k=5.
* Next, we will calculate the **Euclidean distance** between the data points. The Euclidean distance is the distance between two points, which we have already studied in geometry. It can be calculated as:



* By calculating the Euclidean distance we got the nearest neighbors, as three nearest neighbors in category A and two nearest neighbors in category B. Consider the below image:



* As we can see the 3 nearest neighbors are from category A, hence this new data point must belong to category A.

## How to select the value of K in the K-NN Algorithm?

Below are some points to remember while selecting the value of K in the K-NN algorithm:

* There is no particular way to determine the best value for "K", so we need to try some values to find the best out of them. The most preferred value for K is 5.
* A very low value for K such as K=1 or K=2, can be noisy and lead to the effects of outliers in the model.
* Large values for K are good, but it may find some difficulties.

## Advantages of KNN Algorithm:

* It is simple to implement.
* It is robust to the noisy training data
* It can be more effective if the training data is large.

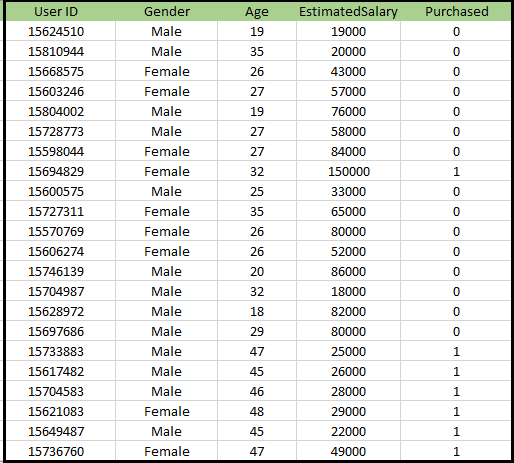
## Disadvantages of KNN Algorithm:

* Always needs to determine the value of K which may be complex some time.
* The computation cost is high because of calculating the distance between the data points for all the training samples.

## Python implementation of the KNN algorithm

To do the Python implementation of the K-NN algorithm, we will use the same problem and dataset which we have used in Logistic Regression. But here we will improve the performance of the model. Below is the problem description:

**Problem for K-NN Algorithm:** There is a Car manufacturer company that has manufactured a new SUV car. The company wants to give the ads to the users who are interested in buying that SUV. So for this problem, we have a dataset that contains multiple user's information through the social network. The dataset contains lots of information but the **Estimated Salary** and **Age** we will consider for the independent variable and the **Purchased variable** is for the dependent variable. Below is the dataset:



**Steps to implement the K-NN algorithm:**

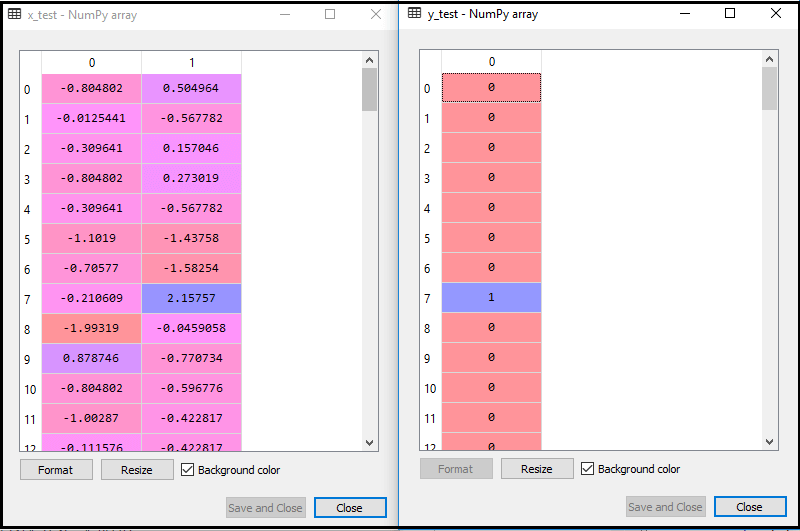
* Data Pre-processing step
* Fitting the K-NN algorithm to the Training set
* Predicting the test result
* Test accuracy of the result(Creation of Confusion matrix)
* Visualizing the test set result.

**Data Pre-Processing Step:**

The Data Pre-processing step will remain exactly the same as Logistic Regression. Below is the code for it:

1. # importing libraries
2. **import** numpy as nm
3. **import** matplotlib.pyplot as mtp
4. **import** pandas as pd
6. #importing datasets
7. data\_set= pd.read\_csv('user\_data.csv')
9. #Extracting Independent and dependent Variable
10. x= data\_set.iloc[:, [2,3]].values
11. y= data\_set.iloc[:, 4].values
13. # Splitting the dataset into training and test set.
14. from sklearn.model\_selection **import** train\_test\_split
15. x\_train, x\_test, y\_train, y\_test= train\_test\_split(x, y, test\_size= 0.25, random\_state=0)
17. #feature Scaling
18. from sklearn.preprocessing **import** StandardScaler
19. st\_x= StandardScaler()
20. x\_train= st\_x.fit\_transform(x\_train)
21. x\_test= st\_x.transform(x\_test)

By executing the above code, our dataset is imported to our program and well pre-processed. After feature scaling our test dataset will look like:



From the above output image, we can see that our data is successfully scaled.

* **Fitting K-NN classifier to the Training data:**  
  Now we will fit the K-NN classifier to the training data. To do this we will import the **KNeighborsClassifier** class of **Sklearn Neighbors** library. After importing the class, we will create the **Classifier** object of the class. The Parameter of this class will be
  + **n\_neighbors:** To define the required neighbors of the algorithm. Usually, it takes 5.
  + **metric='minkowski':** This is the default parameter and it decides the distance between the points.
  + **p=2:** It is equivalent to the standard Euclidean metric.

And then we will fit the classifier to the training data. Below is the code for it:

1. #Fitting K-NN classifier to the training set
2. from sklearn.neighbors **import** KNeighborsClassifier
3. classifier= KNeighborsClassifier(n\_neighbors=5, metric='minkowski', p=2 )
4. classifier.fit(x\_train, y\_train)

**Output: By executing the above code, we will get the output as:**

Out[10]:

KNeighborsClassifier(algorithm='auto', leaf\_size=30, metric='minkowski',

metric\_params=None, n\_jobs=None, n\_neighbors=5, p=2,

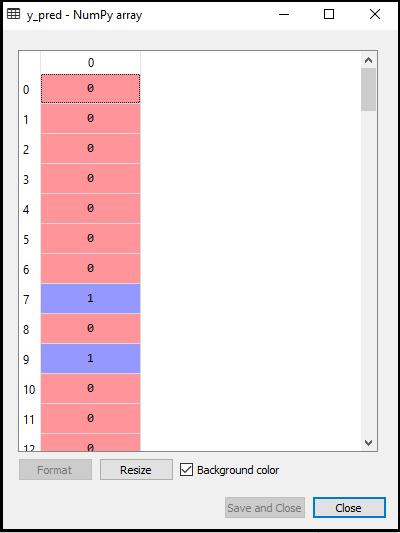
weights='uniform')

* **Predicting the Test Result:** To predict the test set result, we will create a **y\_pred** vector as we did in Logistic Regression. Below is the code for it:

1. #Predicting the test set result
2. y\_pred= classifier.predict(x\_test)

**Output:**

The output for the above code will be:

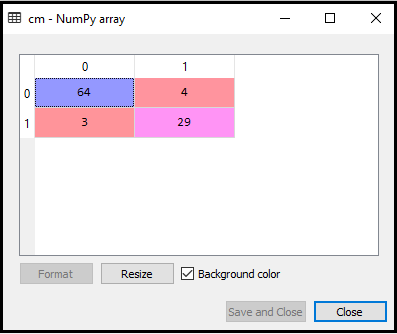


* **Creating the Confusion Matrix:**  
  Now we will create the Confusion Matrix for our K-NN model to see the accuracy of the classifier. Below is the code for it:

1. #Creating the Confusion matrix
2. from sklearn.metrics **import** confusion\_matrix
3. cm= confusion\_matrix(y\_test, y\_pred)

In above code, we have imported the confusion\_matrix function and called it using the variable cm.

**Output:** By executing the above code, we will get the matrix as below:



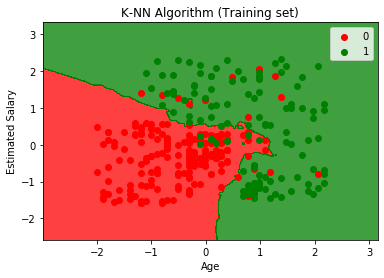
In the above image, we can see there are 64+29= 93 correct predictions and 3+4= 7 incorrect predictions, whereas, in Logistic Regression, there were 11 incorrect predictions. So we can say that the performance of the model is improved by using the K-NN algorithm.

* **Visualizing the Training set result:**  
  Now, we will visualize the training set result for K-NN model. The code will remain same as we did in Logistic Regression, except the name of the graph. Below is the code for it:

1. #Visulaizing the trianing set result
2. from matplotlib.colors **import** ListedColormap
3. x\_set, y\_set = x\_train, y\_train
4. x1, x2 = nm.meshgrid(nm.arange(start = x\_set[:, 0].min() - 1, stop = x\_set[:, 0].max() + 1, step  =0.01),
5. nm.arange(start = x\_set[:, 1].min() - 1, stop = x\_set[:, 1].max() + 1, step = 0.01))
6. mtp.contourf(x1, x2, classifier.predict(nm.array([x1.ravel(), x2.ravel()]).T).reshape(x1.shape),
7. alpha = 0.75, cmap = ListedColormap(('red','green' )))
8. mtp.xlim(x1.min(), x1.max())
9. mtp.ylim(x2.min(), x2.max())
10. **for** i, j in enumerate(nm.unique(y\_set)):
11. mtp.scatter(x\_set[y\_set == j, 0], x\_set[y\_set == j, 1],
12. c = ListedColormap(('red', 'green'))(i), label = j)
13. mtp.title('K-NN Algorithm (Training set)')
14. mtp.xlabel('Age')
15. mtp.ylabel('Estimated Salary')
16. mtp.legend()
17. mtp.show()

**Output:**

**By executing the above code, we will get the below graph:**

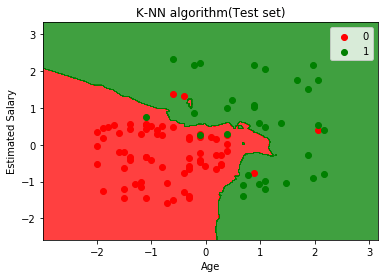


The output graph is different from the graph which we have occurred in Logistic Regression. It can be understood in the below points:

* + As we can see the graph is showing the red point and green points. The green points are for Purchased(1) and Red Points for not Purchased(0) variable.
  + The graph is showing an irregular boundary instead of showing any straight line or any curve because it is a K-NN algorithm, i.e., finding the nearest neighbor.
  + The graph has classified users in the correct categories as most of the users who didn't buy the SUV are in the red region and users who bought the SUV are in the green region.
  + The graph is showing good result but still, there are some green points in the red region and red points in the green region. But this is no big issue as by doing this model is prevented from overfitting issues.
  + Hence our model is well trained.
* **Visualizing the Test set result:**  
  After the training of the model, we will now test the result by putting a new dataset, i.e., Test dataset. Code remains the same except some minor changes: such as **x\_train and y\_train** will be replaced by **x\_test and y\_test**.  
  Below is the code for it:

1. #Visualizing the test set result
2. from matplotlib.colors **import** ListedColormap
3. x\_set, y\_set = x\_test, y\_test
4. x1, x2 = nm.meshgrid(nm.arange(start = x\_set[:, 0].min() - 1, stop = x\_set[:, 0].max() + 1, step  =0.01),
5. nm.arange(start = x\_set[:, 1].min() - 1, stop = x\_set[:, 1].max() + 1, step = 0.01))
6. mtp.contourf(x1, x2, classifier.predict(nm.array([x1.ravel(), x2.ravel()]).T).reshape(x1.shape),
7. alpha = 0.75, cmap = ListedColormap(('red','green' )))
8. mtp.xlim(x1.min(), x1.max())
9. mtp.ylim(x2.min(), x2.max())
10. **for** i, j in enumerate(nm.unique(y\_set)):
11. mtp.scatter(x\_set[y\_set == j, 0], x\_set[y\_set == j, 1],
12. c = ListedColormap(('red', 'green'))(i), label = j)
13. mtp.title('K-NN algorithm(Test set)')
14. mtp.xlabel('Age')
15. mtp.ylabel('Estimated Salary')
16. mtp.legend()
17. mtp.show()

**Output:**



The above graph is showing the output for the test data set. As we can see in the graph, the predicted output is well good as most of the red points are in the red region and most of the green points are in the green region.

However, there are few green points in the red region and a few red points in the green region. So these are the incorrect observations that we have observed in the confusion matrix(7 Incorrect output).

**CONCLUSION**

This article proposes a three-stage driver eye status surveillance system that includes face detection, eye detection, and eye classification stages. The research builds a complete driver eye surveillance system that achieves 33.12 FPS on VGA resolution. In addition, this work also provides the eye detection dataset in various scenarios. They serve as a foundation for drowsiness warning applications in smart vehicles. In the future, a two-stage driver eye status surveillance system will be developed focusing on the eye detection network modification to directly detect and classify the eye status without the face detection phase. The new system is applied in the night driving environment with the infrared camera.

**REFERENCE**

[1] “Road traffic injuries,” 2023. Accessed: Jan. 19, 2023. [Online]. Available: <https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries>

[2] S. Ansari, F. Naghdy, and H. Du, “Human-machine shared driving: Challenges and future directions,” IEEE Trans. Intell. Veh., vol. 7, no. 3, pp. 499–519, Sep. 2022.

[3] 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.

[4] R. Valenti and T. Gevers, “Accurate eye center location through invariant isocentric patterns,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 34, no. 9, pp. 1785–1798, Sep. 2012.

[5] N. Markuš, M. Frljak, I. S. Pandži´c, J. Ahlberg, and R. Forchheimer, “Eye pupil localization with an ensemble of randomized trees,” Pattern Recognit., vol. 47, no. 2, pp. 578–587, 2014.

[6] F. Timm and E. Barth, “Accurate eye centre localisation by means of gradients,” in Proc. Int. Conf. Comput. Vis. Theory Appl., 2011, pp. 125– 130.

[7] L. Swirski, A. Bulling, and N. Dodgson, “Robust real-time pupil tracking ´ in highly off-axis images,” in Proc. Symp. Eye Tracking Res. Appl., 2012, pp. 173–176.

[8] G. M. Araujo, F. M. Ribeiro, E. A. Silva, and S. K. Goldenstein, “Fast eye localization without a face model using inner product detectors,” in Proc. IEEE Int. Conf. Image Process., 2014, pp. 1366–1370.

[9] S. Chen and C. Liu, “Eye detection using discriminatory Haar features and a new efficient SVM,” Image Vis. Comput., vol. 33, pp. 68–77, 2015.

[10] R. Sharma and A. Savakis, “Lean histogram of oriented gradients features for effective eye detection,” J. Electron. Imag., vol. 24, no. 6, 2015, Art. no. 063007.

[11] M. Leo, D. Cazzato, T. De Marco, and C. Distante, “Unsupervised approach for the accurate localization of the pupils in near-frontal facial images,” J. Electron. Imag., vol. 22, no. 3, 2013, Art. no. 033033.

[12] V. Chirra, U. S. Reddy, and V. KishoreKolli, “Deep CNN: A machine learning approach for driver drowsiness detection based on eye state,” Revue d’Intelligence Artificielle, vol. 33, pp. 461–466, 2019.

[13] T. Khan et al., “Smart real-time video surveillance platform for drowsiness detection based on eyelid closure,” Wireless Commun. Mobile Comput., vol. 2019, pp. 1–9, 2019.

[14] M. F. Shakeel, N. A. Bajwa, A. M. Anwaar, A. Sohail, A. Khan, and H.-urRashid, “Detecting driver drowsiness in real time through deep learning based object detection,” in Advances in Computational Intelligence, I. Rojas, G. Joya, and A. Catala, Eds. Cham, Switzerland: Springer, 2019, pp. 283–296.

[15] S. Sathasivam, A. K. Mahamad, S. Saon, A. Sidek, M. M. Som, and H. A. Ameen, “Drowsiness detection system using eye aspect ratio technique,” in Proc. IEEE Student Conf. Res. Develop., 2020, pp. 448–452.

[16] S. Saurav, P. Gidde, R. Saini, and S. Singh, “Real-time eye state recognition using dual convolutional neural network ensemble,” J. Real-Time Image Process., vol. 19, no. 3, pp. 607–622, 2022.

[17] I. Jahan et al., “4D: A real-time driver drowsiness detector using deep learning,” Electronics, vol. 12, no. 1, 2023, Art. no. e235. [Online]. Available: <http://dx.doi.org/10.3390/electronics12010235>

[18] W. Shang, K. Sohn, D. Almeida, and H. Lee, “Understanding and improving convolutional neural networks via concatenated rectified linear units,” in Proc. 33rd Int. Conf. Int. Conf. Mach. Learn., 2016, pp. 2217–2225.

[19] C. Szegedy et al., “Going deeper with convolutions,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2015, pp. 1–9.

[20] D. Misra, T. Nalamada, A. U. Arasanipalai, and Q. Hou, “Rotate to attend: Convolutional triplet attention module,” in Proc. IEEE Winter Conf. Appl. Comput. Vis., 2021, pp. 3138–3147.

[21] D. Qi, W. Tan, Q. Yao, and J. Liu, “Yolo5Face: Why reinventing a face detector,” in Proc. Eur. Conf. Comput. Vis., 2022, pp. 228–244.

[22] “The BioID face database,” 2020. Accessed: Oct. 23, 2020. [Online]. Available: <https://www.bioid.com/facedb>

[23] F. Song, X. Tan, X. Liu, and S. Chen, “Eyes closeness detection from still images with multi-scale histograms of principal oriented gradients,” Pattern Recognit., vol. 47, pp. 2825–2838, 2014.

[24] “Gi4e - Gaze interaction for everybody,” 2020. Accessed: Oct. 23, 2020. [Online]. Available: http://www.unavarra.es/gi4e/databases?languageId= 1

[25] C. Thomaz and G. Giraldi, “A new ranking method for principal components analysis and its application to face image analysis,” Image Vis. Comput., vol. 28, pp. 902–913, 2010.

[26] A. Georghiades, P. Belhumeur, and D. Kriegman, “From few to many: Illumination cone models for face recognition under variable lighting and pose,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 23, no. 6, pp. 643–660, Jun. 2001.

[27] R. Fusek, “Pupil localization using geodesic distance,” in Proc. Int. Symp. Vis. Comput., 2018, pp. 433–444.

[28] S. Zhang, X. Zhu, Z. Lei, H. Shi, X. Wang, and S. Z. Li, “FaceBoxes: A CPU real-time face detector with high accuracy,” in Proc. IEEE Int. Joint Conf. Biometrics, 2017, pp. 1–9.

[29] W. Liu et al., “SSD: Single shot multibox detector,” in Proc. Eur. Conf. Comput. Vis., 2016, pp. 21–37.

[30] G. J. et al., “ultralytics/yolov5: V3.1 - Bug fixes and performance improvements,” Oct. 2020. [Online]. Available: https://doi.org/10.5281/zenodo. 4154370.