الجمهورية الجزائرية الديمقراطية الشعبية
وزارة التعليم والبحث العلمي
جامعة قاصدي مرباح ورقلة
كلية العلوم الحديثة والاتصال

UNIVERSITY OF KASDI MERBAH OUARGLA

Faculty of New Technologies of Information and Communication

Master Degree

DOMAIN: **Science and technology**

SECTOR: **Electronics**

SPECIALTY: **Electronics of embedded systems**

THEME

Driver Drowsiness Detection Using Machine Learning

Presented by:

Abderraouf GHOCHI

Mohammed Oussama abouhafs BARKA

Supervised by:

Dr. Abdelemadjid YOUCEFA

Dr. BENSID khaled	President	UKM Ouargla
Dr. Samai Djamel	Examiner	UKM Ouargla

THE ACADEMIC YEAR 2022/2023

Dedication

With all respect and appreciation, I am pleased to present this gift to you and express my thanks to everyone who helped me in my academic career and the completion of the graduation project.

To my dear loved ones, my dear family, mom and dad, to my elder, The Teacher of reading, and all those who helped me in my career and achieve my dream of graduation, I extend my sincere thanks and deep appreciation to all of you. You have been a real support to me on this journey, and you have been a key factor in my achieving this that I have always dreamed of.

If it were not for your support and unlimited support, he would not have been able to order me, but thanks to you and believing in my abilities, I was able to overcome difficulties. Therefore, I wanted to express to you my sincere thanks and deep gratitude, for the support and encouragement you have always given me.

I thank you for enduring me during this crucial period of my life, and I promise to carry the support and encouragement you have given me to the future, and to carry the torch of success to future generations telling the story of a mother and father who were credited with all the successes in my life that I reached after all this giving and support.

Accept me with the utmost respect and reverence, and I give you all my love and appreciation.

Dedication

Enough praise be to God and prayers be upon the beloved and his family and companions. As for the following: Praise be to God who made us value this step in our academic career, with this note of ours, the fruit of effort and success, by the grace of God, dedicated to our honorable children, may God protect them, and their performance as a light for my path, and to all the generous family who supported me and are still brothers and sisters.

We also thank all my friends.

Special thanks to all teachers and sheikhs of the Qur'an.

All thanks and appreciation to everyone who stood with me.

Acknowledgments

We thank God that we succeeded in completing this scientific research, which provided us with health, well-being and determination،Thank God, thank you very much, with all warmth and reverence, I would like to extend my sincere thanks and appreciation to the supervisor of our work, Dr. YOUCEFA Abdelmadjid. You have been an essential support to us throughout this period, and spared no effort in guiding and guiding us, and providing assistance with all love and sincerity. Which was crowned with all excellence and success, and let's not forget to mention fellow professor Atallah Badreddine, for his contributions that we benefited from and from his valuable experiences and advice, and we learned from him the concepts and skills necessary to accomplish scientific research , we also extend our sincere thanks to the members of the distinguished discussion Committee, whom we hope to repay in our criticism and evaluation of our work. And do not forget to pay tribute to all the respected professors at the University of Raqqa and all those who studied at all our academic levels.

Prayers and peace be upon our Lord Muhammad and upon his God and companions and peace.

Abstract

Driver drowsiness is a critical problem that threatens the safety of drivers operating vehicles, especially drivers who do not take regular breaks when driving long distances, which carries a high risk of death. Among the solutions aimed at saving lives, artificial intelligence, in particular convolutional neural networks (CNN), applied in computer vision applications, have been adapted to predict the state of fatigue and drowsiness of drivers. In this work we used a model that classifies the driver's condition in real time. Based on the training database, **YAWDD** (YAWNING DETECTION DATASET), is divided into four categories with different postures: closed-eye, open-eye, yawning, and non-yawning to determine the driver's condition. Primacy in the bisection is given to the state of the eyes as an indicator of drowsiness. So, in turn, it is a good indicator to detect the drowsiness of the driver. He can also predict States of fatigue even before drowsiness occurs. We also used several algorithms such as **AlexNet, MobileNet, Model c** and Comparing their results with all scales, which gave promising results with accuracy (94%, 94% and 87%) respectively which proved that the MobileNet model is more efficient in detecting drowsiness and fatigue.

Keywords -Deep Learning, Transfer learning, Convolutional Neural Network (CNN), AlexNet, MobileNet.

Résumé

La somnolence des conducteurs est un problème critique qui menace la sécurité des conducteurs conduisant des véhicules, en particulier les conducteurs qui ne prennent pas de pauses régulières lorsqu'ils parcourent de longues distances, ce qui entraîne un risque élevé de décès. Parmi les solutions visant à sauver des vies, l'intelligence artificielle, en particulier les réseaux de neurones convolutifs (CNN), appliqués dans les applications de vision par ordinateur, ont été adaptés pour prédire l'état de fatigue et de somnolence des conducteurs. Dans ce travail, nous avons utilisé un modèle qui classe l'état du conducteur en temps réel. Basé sur la base de données de formation, **YAWDD** (JEU de DONNÉES DE DÉTECTION DU BÂILLEMENT), est divisé en quatre catégories avec différentes postures: yeux fermés, yeux ouverts, bâillements et non bâillements pour déterminer l'état du conducteur. La primauté dans la bissection est donnée à l'état des yeux comme indicateur de somnolence. Donc, à son tour, c'est un bon indicateur pour détecter la somnolence du conducteur. Il peut également prédire les états de fatigue avant même que la somnolence ne survienne. Nous avons également utilisé plusieurs algorithmes tels que **AlexNet, MobileNet, Model c** et Comparé leurs résultats avec toutes les échelles, ce qui a donné des résultats prometteurs avec précision (94%, 94% et 87%) respectivement qui ont prouvé que le modèle MobileNet est plus efficace pour détecter la somnolence et la fatigue.

Mot-clé : apprentissage profond, apprentissage par transfert, Réseau neuronal convolutif, AlexNet, MobileNet.

الملخص

يعد نعاس السائق مشكلة حرجة تهدد سلامة السائقين الذين يشغلون المركبات، وخاصة السائقين الذين لا يأخذون فترات راحة منتظمة عند القيادة لمسافات طويلة مما ينطوي على مخاطر عالية تأدي بدورها الى نتيجة وخيمة وهي الموت. من بين الحلول التي تهدف إلى إنقاذ الأرواح تم تكييف الذكاء الاصطناعي ولا سيما الشبكات العصبية التلافوفية (سي إن إن)، المطبقة في تطبيقات الرؤية الحاسوبية للتنبؤ بحالة الارهاق والنعاس عند السائقين. في هذا العمل قمنا باستخدام نموذج يصنف حالة السائق في الوقت الفعلي. بالاعتماد على قاعدة بيانات التدريب ياود مقسمة إلى أربع فئات بمختلف الوضعيات: تتمثل في العين المغلقة، العين المفتوحة، التثاؤب وعدم التثاؤب لتحديد حالة السائق. تعطى الاولية في التنصيف لحالة العينين كمؤشر على النعاس. إذا بدوره يعد مؤشرا جيدًا لاكتشاف نعاس السائق. ويمكنه أيضًا التنبؤ بحالات التعب حتى قبل حدوث النعاس. كما اننا استخدمنا العديد منا الخوارزميات مثل ألكس نت والموبايل نت ومقارنة نتائجها مع جميع المقاييس ، والتي أعطت نتائج واعدة بدقة (94٪ , 94 ٪ , 87٪) على التوالي والتي أثبتت أن نموذج موبايل نت أكثر كفاءة في الكشف عن النعاس والتعب.

كلمات مفتاحية: التعلم العميق، التعليم المتنقل، الشبكات العصبية التلافوفية، AlexNet، MobileNet

Contents

Abbreviations

Acc: Accuracy.

CNN: Convolutional Neural Network.

EEG: electroencephalogram.

FastICA: Fast Independent component analysis.

Fsc: F1-Score.

FP: False Positive.

FN: False Negative.

Psc: Precision.

ResNet: Residual Network.

RELU: Rectified Linear Activation Unit.

Rcl: Recall.

SENet: Squeezeand-Excitation block Network.

Spc: Specificity.

TP: True Positive.

TN: True Negative.

Spc: Specificity.

VGG: Visual Geometry Group.

List of figures

List of tables

Chapter I: General Introduction

I-1 Introduction:

Road accidents are a major cause of death in addition to wars and famines. Accidents occur under a range of reasons, the most important of which are excessive speed, then negligence through being busy with the phone or turning on the radio, putting alcoholic beverages and drugs that lead to a state of lack of concentration and drowsiness, which in turn reduces a person's ability to stay alert and make decisions quickly significantly during the sleepy phase. This situation represents a direct problem for drivers that leads them to serious consequences, namely death. The National Highway Traffic Safety Administration estimates that between 56,000 and 100,000 accidents are the direct results of drowsiness resulting in more than 1,500 deaths and 71,000 injuries per year [1]. In this thesis, we have also showed as official statistics provided by the General Directorate of national security the Wilayat of Ouargla and by the regional group of the National Gendarmerie the Wilayat of Ouargla in the state of Algeria. Material and human losses, with the mention of their causes, most of which focus on the human factor (all statistics have been published in this chapter) including fatigue and drowsiness.

I-2 Relevant literature:

There are currently a lot of searches regarding how to detect tiredness while driving([2], [3], [4]) the majority of them are focused on developing a system that can identify tiredness in all drivers while ignoring these disparities, This study suggests a real-time sleepiness detection method for drivers that takes into account their unique characteristics For example, deep learning based on traditional methods and Physiological methods, which include measuring some physiological responses, such as heart rate .

I-3 Problem:

- We concluded that previous studies have some limitations, which we believe have a direct impact on future studies and even current results. They are listed below:

- Lack of reliance on a database of huge capacity, which includes several scenarios in the real world, which leaves a weakness in the results of the model.
- The lack of comprehensiveness of previous studies on most of the existing algorithms, comparison between them, and even suggesting some developments for these processes.
- Failure to verify the compatibility of the results of the model with the course of reality, which calls into question the credibility of the ratios and results It must also be relied upon and given higher priorities in reality in order to bridge the loss gap as well as develop the capabilities of the model.

I-4 Contributions:

In this work, we want to achieve:

- Exceeding the accuracy of previous works.
- Transparency in the study, by clarifying the most important parts concerned with the study.

- Processing the database with different algorithms, comparing them and choosing the optimal algorithm for the study.
- Build a model CONVOLUTIONAL NEURAL NETWORKS (CNN) has an accuracy greater than or equal to 90.
- We used a more efficient way to address the phenomenon by combining the elements (eye / mouth) in order to detect the driver's drowsiness with high accuracy.

I-5 Solution:

In order to avoid the problem of poor accuracy caused by artificial feature extraction, a deep cascading convolutional neural network was constructed for face region detection CNN has been used to calculate face detection and identification since it is a very potent performer. With the help of computations based on the pixels in the photographs and how the image information (eye, mouth, and head movement) is arranged, CNN successfully isolates or describes a significant number of these individual aspects from the image, the performance of those CNN architecture varies depending on the designated computer vision problems. For instance, AlexNet performs well compared to other models in different contexts and environmental shifts, such as indoor and outdoor, day and night, VGG FaceNet is performed well for the extraction of facial features such as racial groups. On the other hand, for behavioral characteristics and head movements, Flow ImageNet, and for hand gestures, ResNet demonstarted better performance [2].

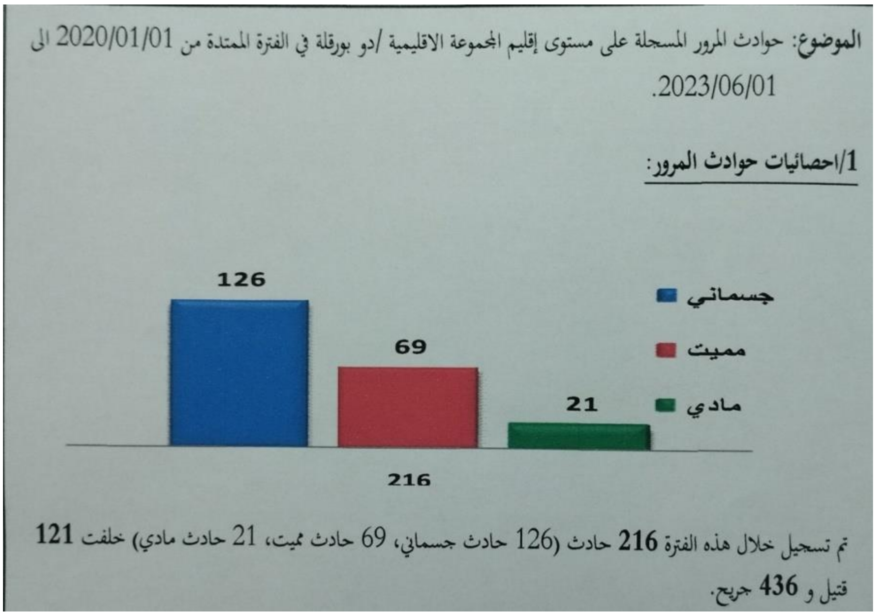
I-6 Official documents in traffic accident statics:

The official toll of traffic accidents and their causes from 2017 to 2023 by the General Directorate of National Security, Ouargla Province:

Table 1 :The official toll of traffic accidents and their causes.

The years	number of wounded	death toll
2017/2023	1292	60

The official toll of traffic accidents and their causes from the year 2020 to 2023 provided by the regional group of the National Gendarmerie, Ouargla Province:

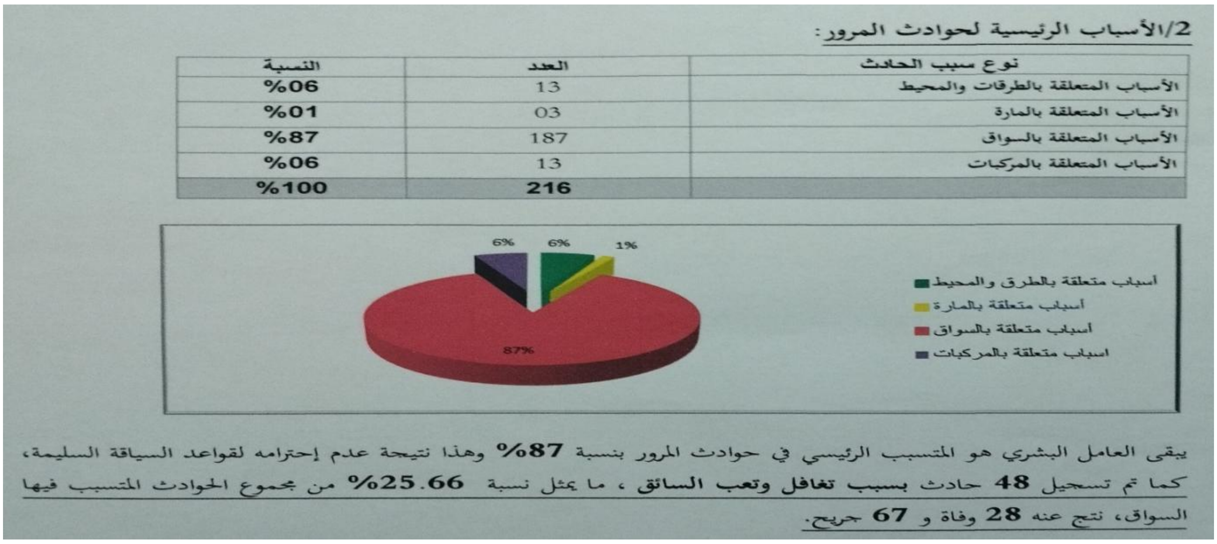


الموضوع: حوادث المرور المسجلة على مستوى إقليم المجموعة الاقليمية /دو بورقلة في الفترة الممتدة من 2020/01/01 الى 2023/06/01.

1/احصائيات حوادث المرور:

تم تسجيل خلال هذه الفترة 216 حادث (126 حادث جسماني، 69 حادث مميت، 21 حادث مادي) خلفت 121 قتيل و 436 جريح.

Figure 1 :Traffic accident statistics



2/الأسباب الرئيسية لحوادث المرور:

النسبة	العدد	نوع سبب الحادث
%06	13	الأسباب المتعلقة بالطرقات والمحيط
%01	03	الأسباب المتعلقة بالمارة
%87	187	الأسباب المتعلقة بالسواق
%06	13	الأسباب المتعلقة بالمركبات
%100	216	

يبقى العامل البشري هو المتسبب الرئيسي في حوادث المرور بنسبة 87% وهذا نتيجة عدم إحترامه لقواعد السياقة السليمة، كما تم تسجيل 48 حادث بسبب تغافل وتعب السائق ، ما يمثل نسبة 25.66% من مجموع الحوادث المتسبب فيها السواق، نتج عنه 28 وفاة و 67 جريح.

Figure 2 : The main causes of Traffic accidents

I-7 Thesis structure:

Thesis follow-up is organized as follows. The first chapter presents a general introduction to the phenomenon of sudden drowsiness and fatigue of drivers, clarifying the risks resulting from it, and proposing solutions to reduce this phenomenon. In the second chapter, we present the previous techniques used to detect driver drowsiness and their effectiveness. In Chapter Three, he presents a method for designing and developing an accurate, efficient, and robust drowsiness detection system for real driving conditions. In the fourth chapter, we will explain the technique used to extract features, present an algorithm to detect fatigue and sleepiness, compare them, and choose a model to be adopted in the study. Finally, we will draw some conclusions and future work.

I-8 Conclusion:

In this chapter some basic definitions on the topic have been presented as an introduction to the study. We mentioned the causes and hindering problems and cited official statistics on the number and causes of traffic accidents in Ouargla for recent years. Moreover, some conventional solutions have been presented to reduce this phenomenon, namely the driver drowsiness detection.

Chapter II: Related Work

II-1 Introduction:

Driver drowsiness detection is a vehicle safety technology that helps prevent drowsy accidents. In order to prevent accidents, the system must be able to detect low concentration early which may lead to drowsiness. Existing systems use various methods to detect drowsiness, the most prominent of which is the convolutional neural network.

II-2 Some of the techniques used in the previous work:

There are several different algorithms used to detect sleepiness among them [3]:

- Convolutional Neural networks (CNN).
- Images Processing Based Techniques.
- EEG Signal Features.

II-2.1 Images Processing Based Techniques:

Images Processing Based Techniques In image processing based techniques, drivers face images are used for processing so that one can find its states. From the face image one can see that driver is awake or sleeping. Using same images, they can define drowsiness of driver because in face image if driver is sleeping or dozing then his/her eyes are closed in image. And other symptoms of drowsiness can also be detected from the face image. We can classify these techniques in three sub-categories [3].

II-2.1.1 Template Matching Technique:

In this technique, one can use the states of eye i.e, if driver closes eye/s for some particular time then system will generate the alarm, because in this techniques system has both close and open eyes template of driver. This system can also be trained to get open and closed eye templates of driver [3].

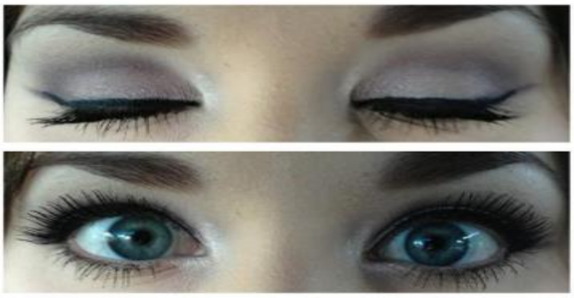


Figure 3 : Open and closed eyes template.

This method is simple and easy to implement because templates of both open and closed eye states shown in Figure 3 are available to system.

II-2.1.2 Eye Blinking based Technique:

In this eye blinking rate and eye closure duration is measured to detect driver's drowsiness. Because when driver felt sleepy at that time his/her eye blinking and gaze between eyelids are different from normal situations so they easily detect drowsiness. In this system the position of irises and eye states are monitored through time to estimate eye blinking frequency and eye close duration. And in this type of system uses a remotely placed camera to acquire video and computer vision methodes are then applied to sequentially localize face, eyes and eyelids positions to measure ratio of closure. Using these eyes closer and blinking ration one can detect drowsiness of driver [3].

II-2.1.3 Yawning Based Technique:

Yawn is one of the symptoms of fatigue. The yawn is assumed to be modeled with a large vertical mouth opening. Mouth is wide open is larger in yawning compared to speaking. Using face tracking and then mouth tracking one can detect yawn. In [6], they detect yawning based on opening rate of mouth and the amount changes in mouth contour area as shown in Figure 4 [3].

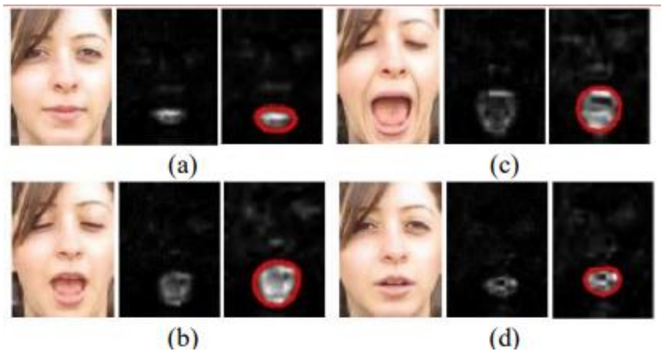


Figure 4:Yawning based on mouth opening rate and amount of changes in the perioral area.

II-2.2 - EEG Based Technique:

In this technique it is compulsory to wear electrode helmet by drivers while driving. This helmet has various electrode sensors which placed at correct place and get data from brain. Researchers have used the characteristic of EEG signal in drowsy driving. A method based on power spectrum analysis and FastICA algorithm was proposed to determining the fatigue degree. In a driving simulation system, the EEG signals of subjects were captured by instrument NT-9200 in two states, one state was sober, and the other was drowsy. The multi-channel signals were analyzed with FastICA algorithm, to remove ocular electric, my electric and power frequency interferences. photo 3 shows how EEG based systems get data for acquisition. Experimental results show that the method presented in this paper can be used to determine the drowsiness degree of EEG signal effectually [3].

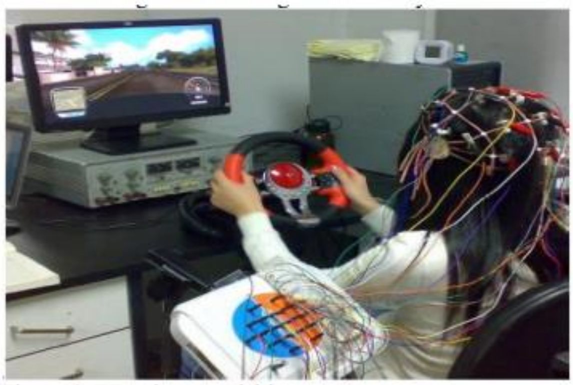


Figure 5 : EEG data acquisition system.

II-2.3 -convolutional neural network:

When you hear people referring to an area of machine learning called deep learning, they're likely talking about neural networks.

Neural networks are modeled after our brains. There are individual nodes that form the layers in the network, just like the neurons in our brains connect different areas.

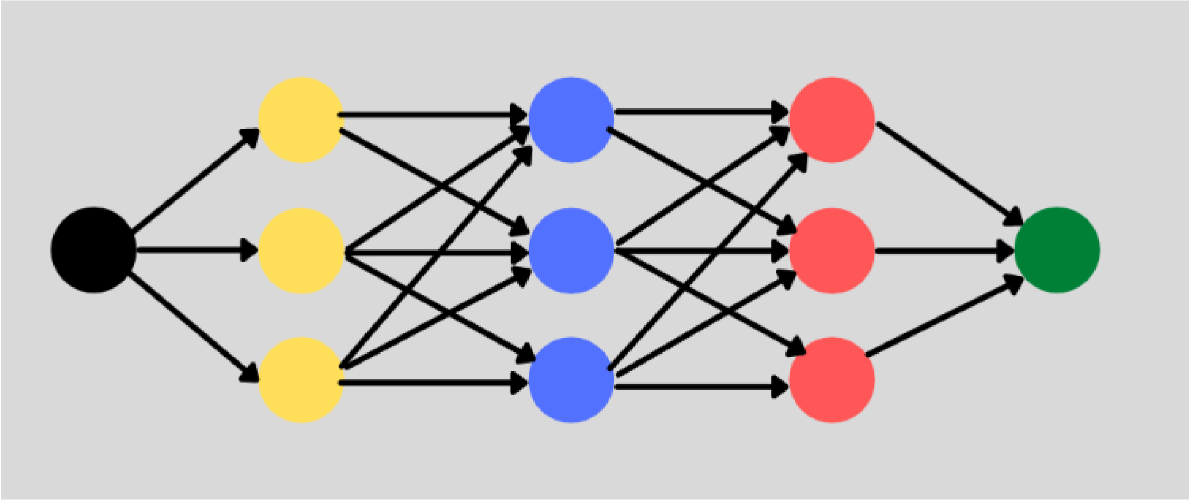


Figure 6: Neural network with multiple hidden layers, each layer has multiple nodes.

The inputs to nodes in a single layer will have a weight assigned to them that changes the effect that parameter has on the overall prediction result. Since the weights are assigned on the links between nodes, each node maybe influenced by multiple weights.

The neural network takes all of the training data in the input layer. Then it passes the data through the hidden layers, transforming the values based on the weights at each node. Finally, it returns a value in the output layer.

One of the great things about CNNs is the number of complex problems they can be applied to. From self-driving cars to detecting drowsiness, CNNs can process this type of data and provide accurate predictions. We mention some of them :

II-2.3.1 convolution neural network VGGNet (VGG):

The Visual Geometry Group (VGG), was the runner up of the 2014 ILSVRC. The main contribution of this work is that it shows that the depth of a network is a critical component to achieve better recognition or classification accuracy in CNNs. The VGG architecture consists of two convolutional layers both of which use the ReLU activation function. Following the activation function is a single max pooling layer and several fully connected layers also using a ReLU activation function. The final layer of the model is a Softmax layer for classification. In VGG the convolution filter size is changed to a 3x3 filter with a stride of 2. Three VGG models, VGG-11, VGG-16, and VGG-19; were proposed the models had 11,16, and 19 layers respectively [13].

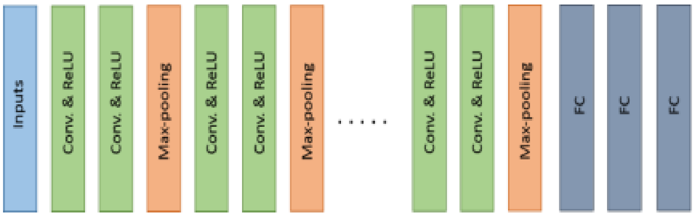


Figure 7 : Basic building block of VGG network: Convolution (Conv) and FC for fully connected layers.

All versions of the VGG-E models ended the same with three fully connected layers. However, the number of convolution layers varied VGG-11 contained 8 convolution layers, VGG-16 had 13 convolution layers, and VGG-19 had 16 convolution layers. VGG-19, the most computational expensive model, contained 138M weights and had 15.5M MACs [13].

II-2.3.2 convolution neural network GoogleNet:

GogLeNet, the winner of ILSVRC 2014, was a model proposed by Christian Szegedy [14]of Google with the objective of reducing computation complexity compared to the traditional CNN. The proposed method was to incorporate "Inception Layers" that had variable receptive fields, which were created by different kernel sizes. These receptive fields created operations that captured sparse correlation patterns in the new feature map stack [13].

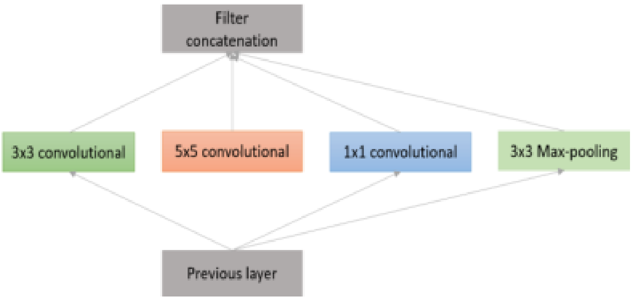


Figure 8 : Inception layer: naive version .

The initial concept of the Inception layer can be seen in Figure 18. GoogLeNet improved the state-of-the-art recognition accuracy using a stack of Inception layers seen in Figure 19. The difference between the naïve inception layer and final Inception Layer was the addition of 1x1 convolution kernels. These kernels allowed for dimensionality reduction before computationally expensive layers. GoogLeNet consisted of 22 layers in total, which was far greater than any network before it. However, the number of network parameters GoogLeNet used was much lower than its predecessor AlexNet or VGG. GoogLeNet had 7M network parameters when AlexNet had 60M and VGG-19 138M. The computations for GoogLeNet also were 1.53G MACs far lower than that of AlexNet or VGG [13].

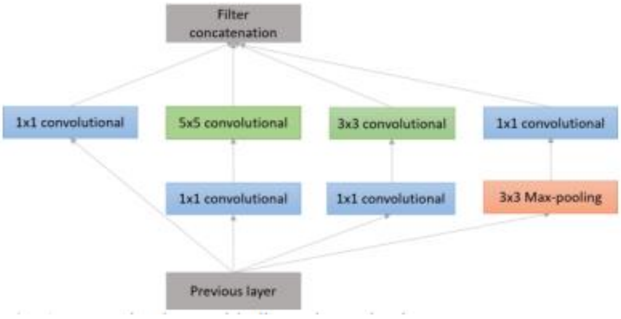


Figure 9 :Inception layer with dimension reduction.

II-2.3.3 convolution neural network Residual Network (ResNet):

The winner of ILSVRC 2015 was the Residual Network architecture, ResNet. Resnet was developed by Kaiming [15]with the intent of designing ultra-deep networks that did not suffer from the vanishing gradient problem that predecessors had. ResNet is developed with many different numbers of layers; 34, 50,101, 152, and even 1202. The popular ResNet50 contained

49 convolution layers and 1 fully connected layer at the end of the network. The total number of weights and MACs for the whole network are 25.5M and 3.9M respectively .

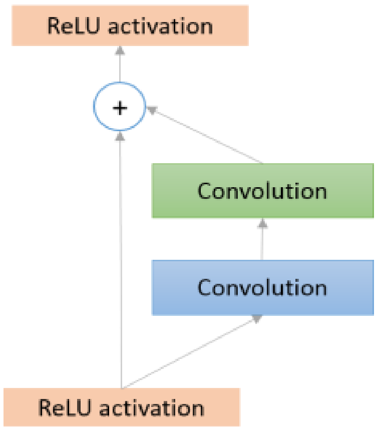


Figure 10:Basic diagram of Residual block.

The basic block diagram of the ResNet architecture is shown in Figure 20 ResNet is a traditional feed forward network with a residual connection. The output of a residual layer can be defined based on the outputs of [13].

II-2.3.4 SENet:

In previous architectures of CNN, data representation by learning spatial information has improved tremendously with increasing network depth. This hierarchy of features from each layer forms CNN models representation of the input dataset. Improving and introducing better methods in deep learning continues with new research. Jie Hu et al. (2017) introduced a new network structure called "Squeezeand-Excitation block" (SE) that incorporates channel-wise response and models their interdependency. The activation responses from channel-wise features form the base of SE blocks. SENet built from stacks of SE blocks won the ISLVRC 2017 challenge with an accuracy of 2.251 % top -5 error rate.

The main idea behind SE blocks is to model the relationship between the channels of the convoluted features. Thereby, allowing the network to learn more important features and suppress the trivial ones [10].

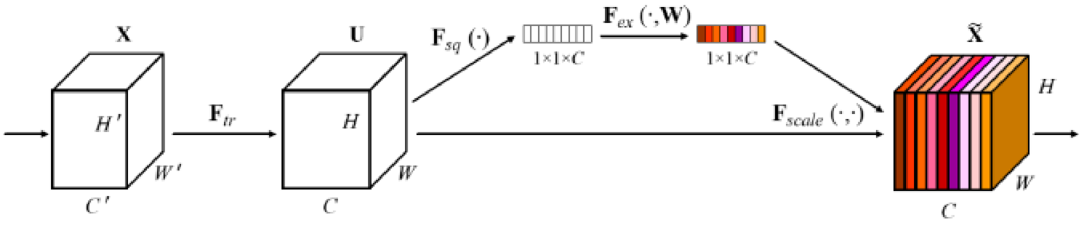


Figure 11: se block.

II-3 Various types of assistive techniques in determining the contours of the face:

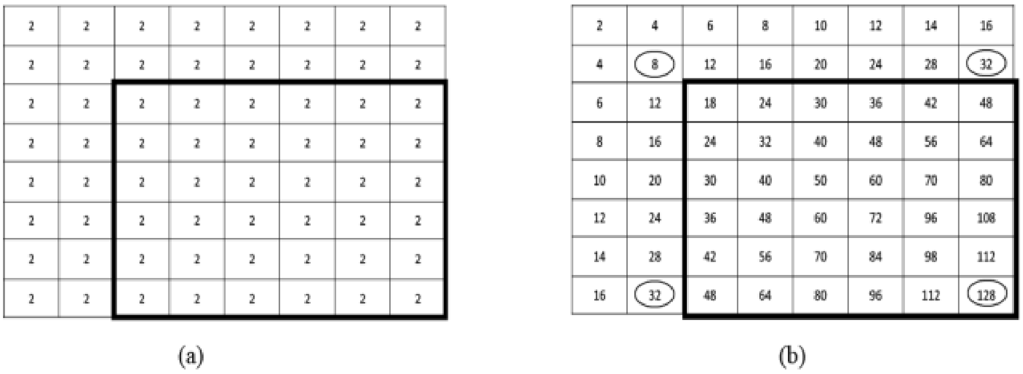
II-3.1 Human Face Detection Techniques:

Several ingeniously developed algorithms for face detection have been proposed, and several face detection algorithms are being used by researchers, such as Viola-Jones, Local Binary Pattern Histograms (LBPH), Eigenfaces, Fisher face, OpenCV, and DLIB, Among them.

II-3.1.1 Viola–Jones:

Viola and Jones came up with an object detection framework in 2001. The main purpose of the framework was to solve the problem of face detection which the algorithm achieved with faster, high detection accuracy, even though the algorithm could detect a diverse class of objects. The algorithm solved problems of real-time face detection, such as slowness, computational complexity, etc.

The Viola–Jones algorithm functions in two steps: training and detection. In the detection stage, the image is converted into grayscale. The algorithms then find the face on the grayscale image, using a box search throughout the image. After that, it finds the location in the colored image. For searching the face in a grayscale image, Haar-like features are used to search an image. All human faces consist of the same features, and Haar-like features explores this similarity by making three types of Haar features for the face, namely edges, line and four-sided features. With the help of these, a value for each feature of the face is calculated. An integral image is made out of them and compared very quickly, as shown in Figure 23. An integral image is what makes this model faster because it reduces the computation costs by reducing number of array references, as shown in Figure 24. In the second stage, a boosting algorithm named the Adaboost learning algorithm is employed to select a few numbers of prominent features out of large set to make the detection efficient. A simplified version was delineated by Viola and Jones in 2003. Finally, a cascaded classifier is assigned to quickly reject non-face images in which prominent facial features selected by boosting are absent, as shown in Figure 25 [18].



2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2

(a)

2	4	6	8	10	12	14	16
4	8	12	16	20	24	28	32
6	12	18	24	30	36	42	48
8	16	24	32	40	48	56	64
10	20	30	40	50	60	70	80
12	24	36	48	60	72	96	108
14	28	42	56	70	84	98	112
16	32	48	64	80	96	112	128

(b)

Figure 12: Example of integral image.

Example of integral image: (a) an 8 × 8 sized input images expressed with pixel values. Using conventional method, the 6 × 6 rectangle has a summation of 72 pixels, which uses all 36 array references; (b) integral image of the input image. Using this in input image. Using this integral image, the value is calculated as (4 + 1 − 2 − 3). Here, 1, 2, 3, 4 are the positions of the rectangles shown with circles. So, the sum of the pixels in 6 × 6 rectangle is 128 + 8 − 32 − 32 = 72, which is same as the real value, using only 4 array references instead of 36.

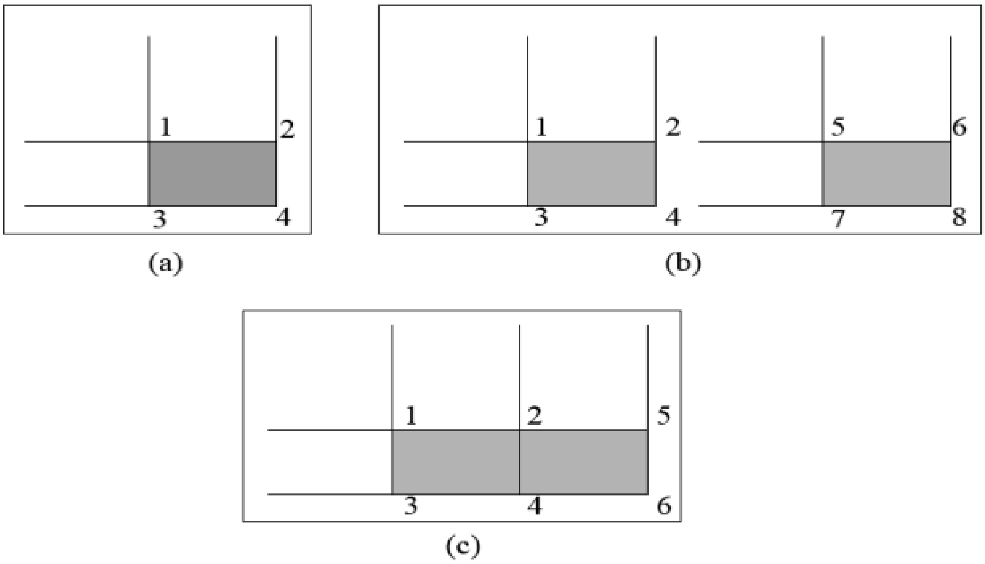


Figure 13: Integral image array reference

Integral image array reference: (a) the sum within the dark shaped location is computed as 4 + 1 − (2 + 3) (four array references); (b) the sum within the two dark shaped location is computed as 4 + 1 − (2 + 3) and 8 + 5 − (6 + 7), respectively, (eight array references); (c) the sum within the two adjacent dark shaped location is computed as 4 + 1 − (2 + 3) + 6 + 2 − (4 + 5) and hence, 6 + 1 − (3 + 5) (six array references) .

However, the algorithm can only detect the frontal side of the face. The algorithm possesses an intensely larger training time. Training with a limited number of classifiers can result in far less accurate [18].

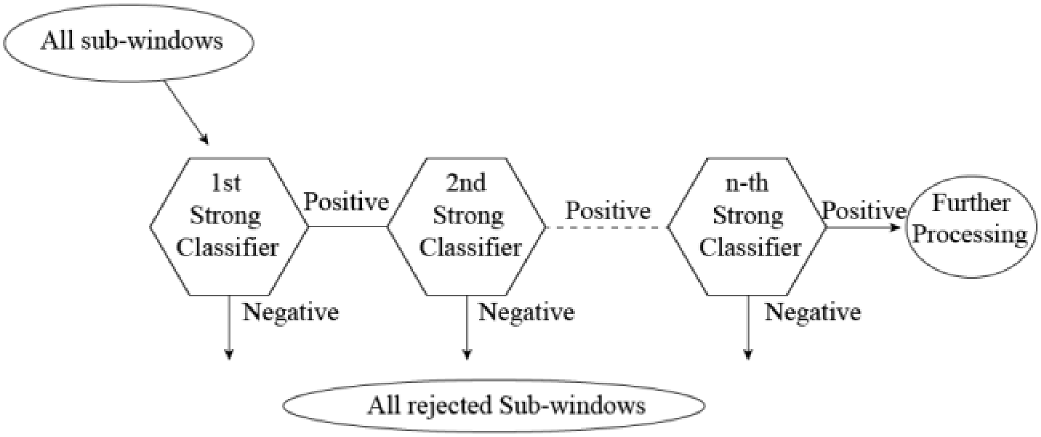


Figure 14: Schematic diagram of the detection cascade.

Schematic diagram of the detection cascade. Strong classifiers can be different facial features, such as the mouth, eyes, etc. An image without a human mouth or other strong classifiers is surely not a human face. Hence, the window is rejected, which makes the process faster. On the other hand, if all the strong classifiers are present in an image, it is classified as a face.

II-3.1.2 Local Binary Pattern (LBP):

LBP was mainly proposed for monochrome still images. LBP is based on the texture analysis on images. The texture analysis model was proposed first in 1990.

LBP looks for nine pixels at a time of an image—to be exact, a 3 × 3 matrix—and particularly puts interest in the central pixel. LBP compares the central pixel (cp) with its neighboring pixel (np) and assigns 0 for np < cp and 1 for np > cp in the corresponding neighbor. Then, it turns the eight binary np into one single byte which corresponds to an LBP-code or decimal number. This is done by multiplying the matrix component wise with an eight-bit number representative matrix as shown in Figure 26. This decimal number is used in the training process. We are basically interested in edges; in the image, the transition from 1 to 0 and vice versa presents a change in brightness of the image. These changes are the edge descriptors. When we look at a whole image, we look for comparisons or change in pixels or brightness, and the edges are received [18].

LBP is tolerant of monochromatic illumination changes because LBP just compares the neighboring pixels; a change in illumination would change the comparative values, which would not result in change in values in the comparison. LBP is mostly popular for its computation simplicity and fast performance. LBP can detect a moving object by subtracting the background, and has high discriminating power with a low false detection rate. The algorithm yields the same detection accuracy in offline images and in real-time operation. However, LBP is not invariant to rotations and high computation complexity. LBP uses only pixel difference while ignoring the magnitude information. It is not sensitive to minor adjustments in the image [18].

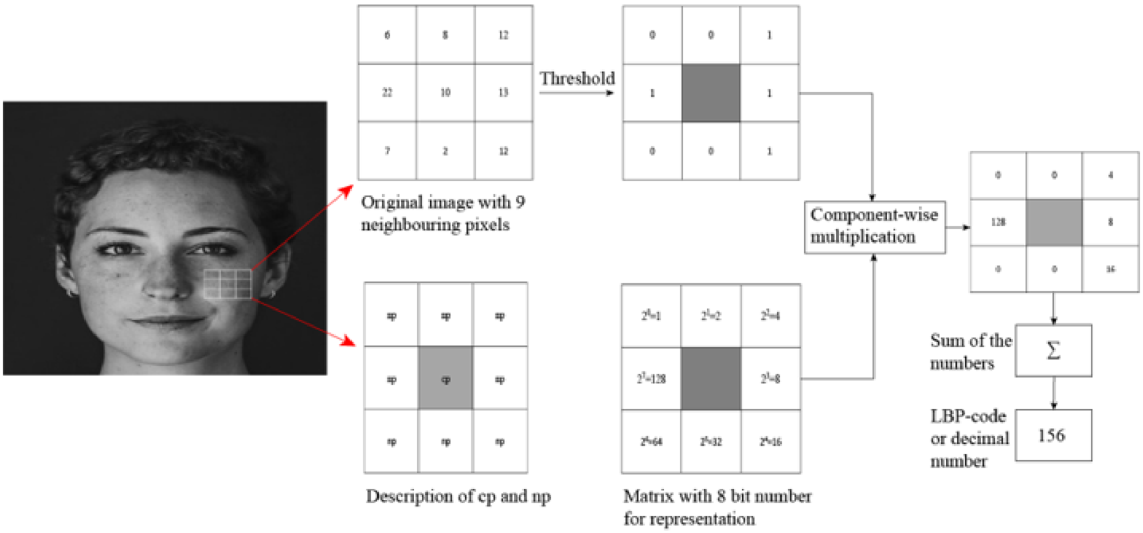


Figure 15: The process of calculating LBP code.

The process of calculating LBP code. Every neighboring 3 × 3 pixel is taken under np < cp and np > cp threshold to produce the binary comparison 3 × 3 matrix. The binary matrix is then multiplied component wise with eight bit representative 3 × 3 matrix and summed up to generate the LBP code or the decimal number representation.

II-3.2 Eye Detection Techniques :

II- 3.2.1Haar Classifiers :

Once the cascade classifiers detect and pass an image detected as face, the eyes are searched for. Using the Haar Classifiers, feature points are analyzed and pixel values are determined from the passed image. A sample of Haar feature types are passed and the filter gradient calculates the pixel values of the region of interest which are the eyes here. Some of the Haar types used in searching and locating the eyes are shown in Fig 27 [19] .

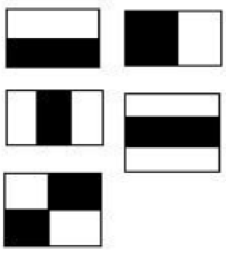


Figure 16: Haar Types used to detect eyes.

II-3.3 Detection of Other Components of the Face :

In some driver face monitoring systems, other component of driver faces such as mouth or nose were detected. However, there are a few researches in this field, the methods presented for detection of other components of driver face is reviewed in this section [20]:

II-3.3.1 Mouth Detection *:*

In some driver hypo-vigilance detection systems such as open or closed mouth was used as a measure of driver fatigue. Most of these systems detect the mouth based on red color features of lips. The most important disadvantage of these methods is that they can only work properly with color image in visible spectrum and suitable light conditions [20].

II-3.3.2 Nose Detection:

Nose tip location with respect to head and eyes is an appropriate criterion for determination of the direction of the head. Three-dimensional model of face can be estimated by only one camera and using location of nose tip with respect to eyes. After estimating 3D model of face, determination of head direction is easily possible. Bergasa et al. [21] extracted the nostrils for determination of head direction. In this method, camera was located slightly lower that driver face while nostrils are often visible. The most important property of nostrils is blackness with respect to face skin. Thus, by applying a threshold on nose area, nostrils are detected. This method is failed for nostrils detection of black people and people who have mustache [20].

II-3.3.3 Salient Points Detection :

In some recent researches such as, the salient points of face are detected after face detection. In these researches, the salient points are tracked over time. Thus, according to the relative position of salient points, hypo-vigilance symptoms are extracted. The most common method which is used for salient point detection is Shi-Tomasi method. This method detects salient corners of a gray-level image as salient points. Tracking and analysis of salient points of face make the system more robust against occlusion [20] .

II-4 the difference between the methods:

Table 2 :Face Detection Techniques.

Techniques	Advantage	Limitation	References
(using Viola-Jones method)	Good Robustness	Ineffective to detect titled faces and sensitive to lighting conditions	[8]
Local Binary Pattern (LBP)	High detection accuracy	Sensitive to lighting conditions and rotations	[18]
Feature-based (in HSV color space)	Average Robustness and the chromaticity are decoupled from the intensity	Non-removable singularities	[8]

Table 3:Eye Detection Techniques.

Techniques	Advantage	Limitation	Ref
Haar Classifier	Execution speed and detection accuracy are high	Complexity is definitely increasing, less robustness to different lighting conditions	[8]
Support Vector Machine (SVM)	Increase the overall robustness of the system and uses the kernel trick.	-The head position does not deviate a lot when fully awake. -Need long training time on large data sets.	[8]
Fuzzy expert system	Enhances decision quality, solve real-time problems efficiently	Difficult to build and maintain, require a large amount of time to train.	[8]

Table 4 :Mouth/Yawn Detection Techniques.

Techniques	Advantage	Limitation	Ref

Kalman-filter	Decreases the dimensionality error, robust and estimates the dynamic changes of a state	In case of nonlinear function need to use extended Kalman-filter	[8]
Haar-like Features	High execution speed and works even if the driver turns his face	Incredible complexity, accuracy depends on different lighting conditions	[8]
Improved Fuzzy C-Means clustering technique	Works robustly at night time because of the IR illuminator being used	Performance decreases during daytime especially in bright days, fails to detect when the head is rotated	[8]

II-5 Some previous work:

Table 5: Related works

Author	Title	METHODOLOGIE	CRITICISM
[4]	Detecting Driver Drowsiness in Real Time through Deep Learning based Object Detection.	The proposed methodology treats drowsiness detection as an object detection task, and from an incoming video stream of a driver, detects and localizes open and closed eyes. MobileNet CNN architecture with Single Shot Multibox Detector (SSD) is used for this task of object detection.	One major improvement that could be made in the future is to refine the Drowsy dataset and add low-light images (in near-infrared light) to enable the model to detect drowsiness in low-light conditions. Work needs to be done in incorporating the yawning information (which was labeled in the dataset but not used in this methodology) into an algorithm.
[5]	Deep CNN: A Machine Learning Approach for Driver Drowsiness Detection Based on Eye State.	(1) Viola-jones face detection algorithm is used to detect the face the images and given as input to Viola-jones eye detection algorithm (2) Once the face is detected, Viola-jones eye detection algorithm is used to extract the eye region from the facial images	You need to use transfer learning to improve system performance. The experiment was conducted using only one dataset rather than multiple datasets and among them at night.

		and given as input to CNN. (3) CNN with four convolutional layers are used to extract the deep features and those features are passed to fully connected layer. (4) Soft max layer in CNN classifies the images in to sleepy or non-sleepy images.	
[6]	A Review of EEG Signal Features and Their Application in Driver Drowsiness Detection Systems.	Review EEG signal features used in the literature for a variety of tasks, then review applications of EEG features and deep learning methods in detecting driver drowsiness.	Physiological based drowsiness detection systems such as EEG have the limitation that the driver has to wear electrodes on his body that could prove to be a hindrance and an annoyance to the driver.
[7]	Driver Drowsiness Detection Model Using Convolutional Neural Networks Techniques for Android Application.	The proposed CNN model was used to build a real-time driver drowsiness detection system for embedded systems and Android devices.	There are limitations to this technique, such as obstruction of seeing facial features by wearing sunglasses and poor lighting conditions .
[8]	A Brief Review on Different Driver's Drowsiness Detection Techniques.	Various technologies of the driver's face monitoring system, which are used to identify drowsiness, are discussed. Each of the popular methods is discussed based on advantages and limitations.	The most difficult problem with driver drowsiness detection systems is related to night light detection.
[9]	System and Method for Driver Drowsiness Detection Using Behavioral and Sensor-Based Physiological Measures.	The proposed hybrid model uses AI-based Multi-Task Cascaded Convolutional Neural Networks (MTCNN) as a behavioral measure to recognize the driver's facial features, and the Galvanic Skin Response (GSR) sensor.	The efficacy of the proposed model may be improved by integrating other sensors, such as the PPG, pulse rate sensor and IR-UWB radar.
[1]	DRIVER DROWSINESS DETECTION USINGENSEMBLE	Four different convolutional neural network (CNN) techniques have been applied to YawDD.	• The experiment was conducted using only one dataset rather than multiple datasets.

	CONVOLUTIONAL NEURAL NETWORKS ON YAWDD.		• Instead of comparing the performance of the proposed model with that of the known literature, four different techniques based on CNN are proposed.

II-6 Conclusion:

In this chapter, we have focused our attention on some of the techniques that cause an obstacle to obtaining better results, and we have demonstrated the different uses of algorithms for detecting drowsiness and drive fatigue. In the next chapter, we will explain the methodology used in our thesis.

Chapter III: Methodology

III-1 Introduction:

Various methods have been proposed to detect drowsiness and extract the face, eye, mouth and many other auxiliary features such as Viola Jones. This section introduces a method for designing and developing an accurate, effective and powerful drowsiness detection system for real driving conditions. In this chapter we provide details about the algorithms. Then we introduce the feature extraction part.

III-2 Machine learning :

Machine learning is an overlap between computer science and statistical methods. Through machine learning, computer actions modify or adapt to become more accurate and their accuracy depends on how well the changes reflect the correct results. In other words, machine learning uses algorithms on raw data to extract features in the form of a model and use this model to infer data that has not yet been tested. In short, we want machines to learn from data so that they become smart enough to learn from experiments like humans. Machine learning is constantly evolving as is the combination of pattern recognition and computer learning theory in artificial intelligence. There are two types of machine learning; supervised and unsupervised learning. These two types of learning are based on the theory that learning is based on past experience as well as on testing new information. In this new information, the data is tested on the similarities between the data points that help us in learning. ML is also classified based on the type of production produced. Here are the different types of machine learning [10].

III-2.1 Supervised Learning:

In supervised learning, learning data altogether with the corresponding labels are fed to the program. The aim is to devise a function that relates the data to its correct class. In this type of learning, a set of examples with known correct responses are used for training. The algorithm uses the information from the training data to generalize and respond correctly to all given inputs. The training set consists of a set of input data attached with the answers that the algorithms are envisaged to produce. The function model is as given in equation 1:

$$y_i = a + Bx_i$$

Where x_i is the input, y_i is the target, and◌̇ iterates from 1 to N, a is the y-intercept and B is the input feature. Within supervised learning, when unlabeled data are used with labelled data as part of the training, it is called a semi-supervised learning. Again, within the supervised learning, when data is fed to the algorithm from the dynamic environment as rewards and/or punishment, it is called reinforcement learning. For example, it could be flying a plane in a simulated environment or an autonomous car driving on a given path. Supervised learning is the most common learning technique. It is used in computer games such that computers learn and form experiences from previous games that were played. The more games are played, the more intelligent it becomes. K-nearest neighborhood is an example of supervised learning algorithm. Statistical methods such as regression models use supervised learning widely. Bayesian networks, support vector machines and Markov models are also some examples of supervised learning [10].

III-2.2 Unsupervised Learning:

In unsupervised learning, the algorithm depends mostly on hidden patterns and structures in the data and is not provided with labels. So, the algorithm tries to determine similarities and underlying relationships between the input data points. The outputs for any test data point are unknown and it is hard to build models such as regression. Therefore, similar inputs are clustered together. The aim also includes finding out the intrinsic patterns, dimensionality reduction, outlier detection and learning feature of the data.

Method k-means clustering is a classic example of unsupervised learning. In k-means clustering algorithm, k is defined first. The algorithm then creates k clusters and puts the test data randomly to different clusters. It then sets the data point values to the means of the clusters as a starting point and puts the values to clusters that is closest to the means. The algorithm keeps updating the cluster means. After clustering all the data points, the algorithm restarts and keeps clustering until the clusters mean values do not change anymore.

Unsupervised learning is also used in neural computing such as self-organizing maps (SOM) and adaptive resonance theory (ART) [10].

III-2.3 Deep Learning:

Machine learning has been a field of building models to extract patterns from data/inputs for decades. However, it required a thorough knowledge of statistical methods, computation and data which were not easily available. But, during the last two decades things have changed significantly. With statistical modelling and programming, powerful machine learning models that perform with high accuracy can be built. Deep learning is an automated way of extracting useful patterns from data using neural network and optimization. Currently, there are several libraries like TensorFlow and Pytorch, which make it possible to build powerful models in less time than ever before. These libraries are powered by the easy availability of data, massively parallel implementation of the neural network computation, often using GPU based parallel processing. In addition to computation hardware, there are also a lot of efficient initialization and computation tricks that help the learning. Similarly, with deep learning a lot of progress has been made with face recognition, image classification, speech recognition, text-to-speech generation, self-driving cars, recommendation systems, games and machine translation. In this chapter we will focus on the main architectures in deep learning that have empowered these developments.

Since the last two decades deep learning has become a widely used phenomena. Within deep learning, deep networks differ from neural networks if: Deep networks have more neurons and more deeper layers; The layers are connected in more complex ways and that means the number of parameters has risen from thousands to even millions; and Automatic feature extraction.

The architecture has evolved over the past two decades and new research in the field of deep networks has continued [10]. In this chapter, we focus on key architectures for deep networks including **convolutional neural networks**.

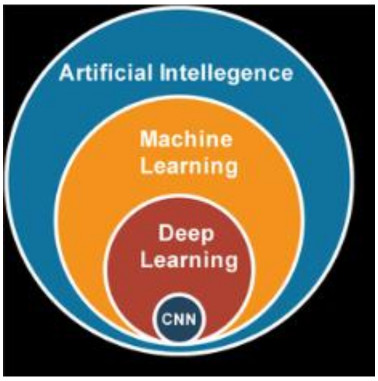


Figure 17 : The relationship between CNN and deep learning and machine learning.

III-2.4 Convolutional neural networks:

This is a summary of CNN in general. CNN is used to learn higher-order features in the data using convolutions. They perform exceptionally well with object recognition in images. Similarly, CNN can be used for text analysis with character recognition, and analyzing words as discreet textual units. Similarly, CNN works well with voice data. However, CNN is better known for image recognition. Today CNN is used in several applications including self-driving cars, robotics and drones. CNN tends to work well with data that has some structure and spatial correlation. Figure 5 shows CNNs used in computer vision [10].

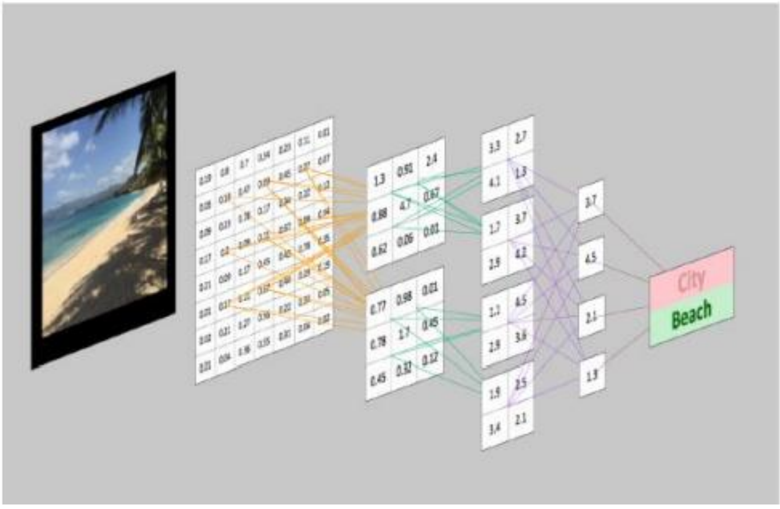


Figure 18 : CNNs and computer vision.

The intuition behind CNN comes from the fact that traditional neural networks do not scale well with image data. Image data allows to change the network architecture. That is why with CNNs, the

neurons can be aligned in a three-dimensional structure using length, height and depth and these attributes can be mapped with the image width pixel, height pixel and the RGB channels. To put it simple, CNNs transform the input image through several connected layers and output a set of class probabilities. All CNN architectures share some common layers as given in the Figure 8 [10].

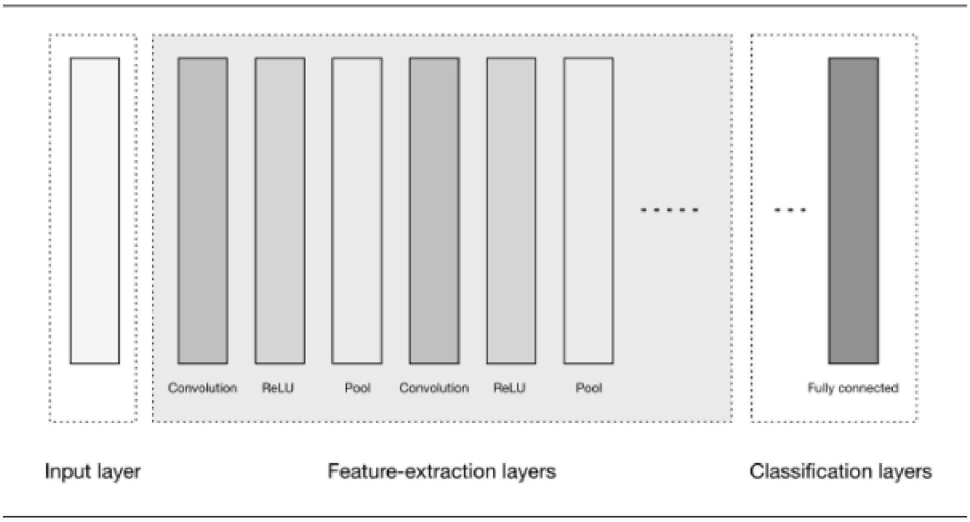


Figure 19: High-level CNN architecture.

As seen in figure 9, the CNN input data is loaded to the input layer for processing. The input layer takes three dimensions (width, height and RGB channel) of the image data [10].

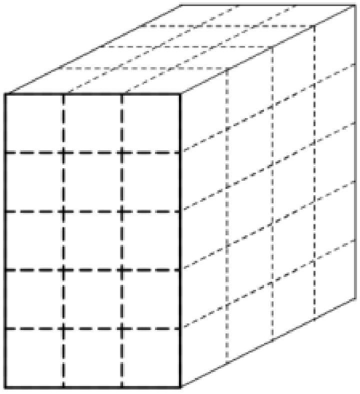


Figure 20: 3D data input.

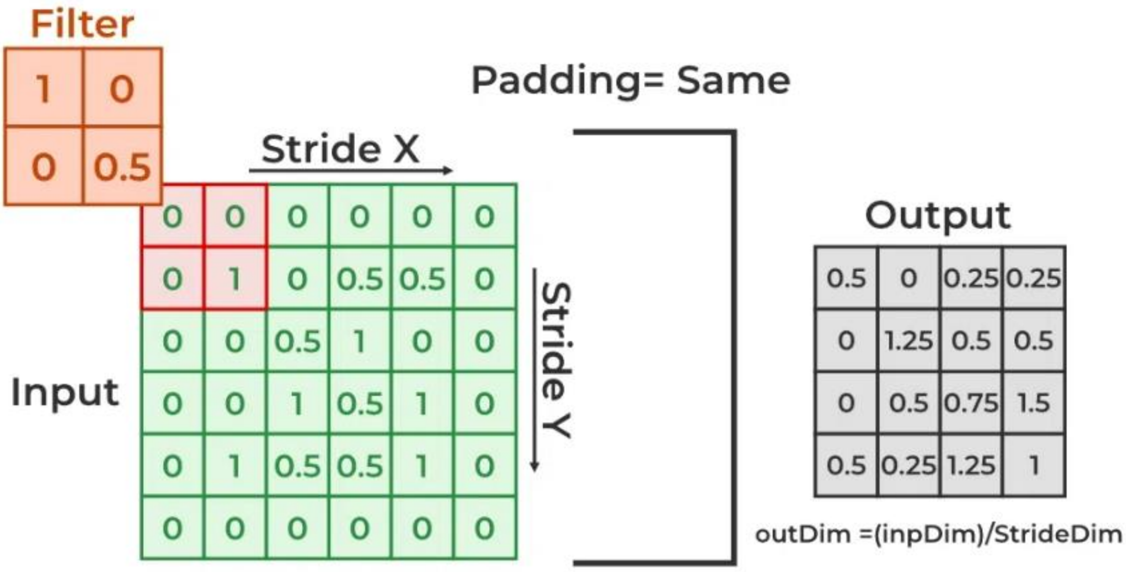
III-3 Parts of convolutional neural networks:

Feature Layer of CNN:

The featured layers of CNN include convolutional layers, ReLU layer, Pooling Layer, and Fully Connected Layer, Softmax, Padding. Following a detail discussion of the different layers.

Padding:

padding is a technique used to preserve the spatial dimensions of the input image after convolution operations on a feature map. Padding involves adding extra pixels around the border of the input feature map before convolution.



Filter

1	0
0	0.5

Input

0	0	0	0	0	0
0	1	0	0.5	0.5	0
0	0	0.5	1	0	0
0	0	1	0.5	1	0
0	1	0.5	0.5	1	0
0	0	0	0	0	0

Output

0.5	0	0.25	0.25
0	1.25	0.5	0.5
0	0.5	0.75	1.5
0.5	0.25	1.25	1

padding in convolutional network

*Figure 21 : there is another layer added to the 4 * 4 image and now it has been converted to a 5 * 5 image.*

Convolutional Layers:

Convolutional layers are the main components of CNN architectures. Convolution layers transform the input image by applying a filter or kernel. The layer performs dot product between the region of the neurons in the input layer and the filters to generate the feature map. Figure 9 shows Convolution layer with input and output volume [10].

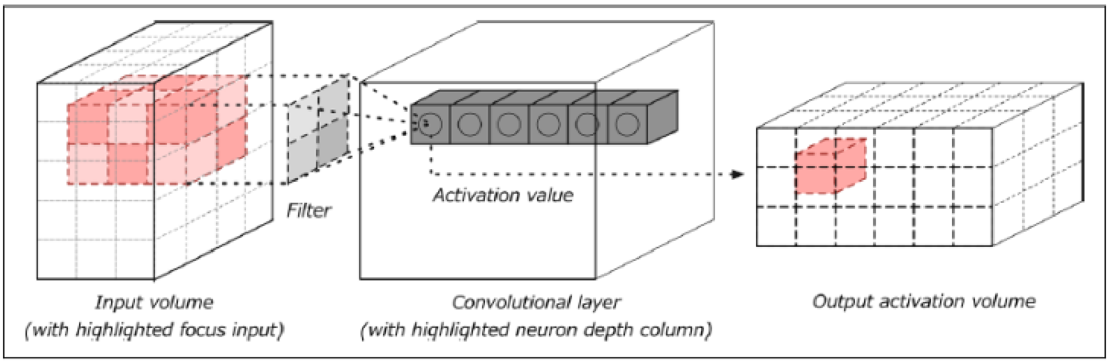
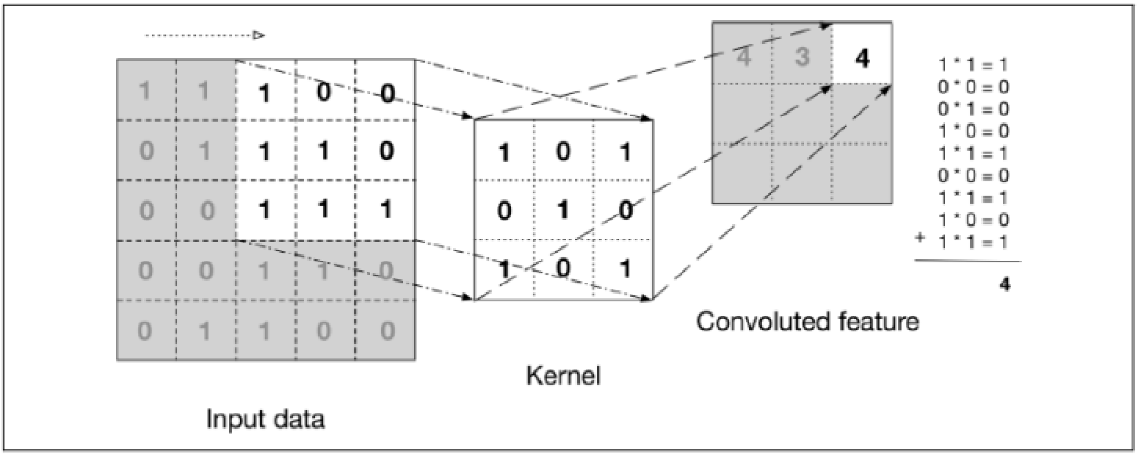


Figure 22: Convolution layer with input and output volume.

The output generated after the convolution has the same dimensions usually as the input as seen in figure 12.



1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

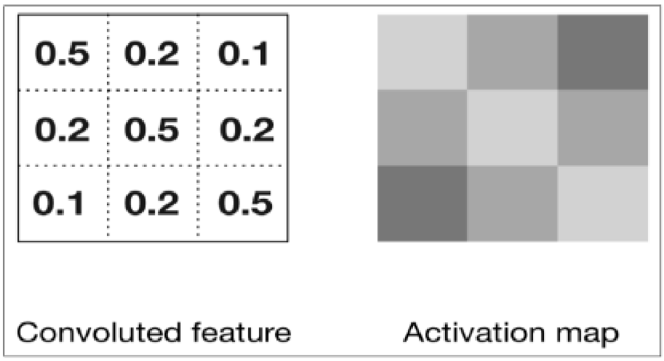
1	0	1
0	1	0
1	0	1

4	3	4

1 * 1 = 1
0 * 0 = 0
0 * 1 = 0
1 * 0 = 0
1 * 1 = 1
0 * 0 = 0
1 * 1 = 1
1 * 0 = 0
+ 1 * 1 = 1
4

Figure 23: The convolution operation.

As show in figure 11, the filter or kernel has a smaller size than the input size slid. It uses a given stride value on the input data to produce convolved feature. This process is also known as feature detector. The feature map or activation map (as shown in figure 13) for each filter is summed along the depth dimension to construct the 3D output. The learning of the feature detector is reflected from the activation value. Therefore, each filter learns to detect a certain feature. The sliding of filter on the input generates a two-dimensional activation map for that filter [10].



0.5	0.2	0.1
0.2	0.5	0.2
0.1	0.2	0.5

Figure 24: Convolution and activation maps.

The staked activation maps form output volume. The values in activation volume correspond to neurons outputs that cover a small area of the input volume.

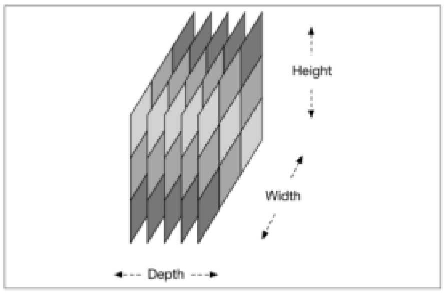


Figure 25: Activation volume output of convolutional layer.

Receptive field is the connection of neurons to the input layer. It is used to set the size of filter maps. For example, if the size of the filter is set to 5 x 5 x 3, the number of the weights for a neuron is 5 * 5 * 3 = 75. CNNs use parameter-sharing to limit parameter counts, thus save computation time during training data. Each filter learns one feature. As show in figure 12 once a feature for example a horizonal line is learnt by a filter in one space of the input, it does not need to learn it for another location in the image, which makes CNN position invariant [10].

ReLU Layer:

The network uses the Rectified Linear Unit nonlinear (ReLU) function as the output of the neuron for the input x, F(x)=max (0, x) . CNNs with ReLUs train in lesser time than their equivalents with tanh function. Not only this, the CIFAR-10 dataset reaches 25% training error in much lesser iterations. Another reason for ReLUs use is that it does not need normalized inputs. Inputs with positive values show learning happening in those neurons. Thus, local response normalization helps to reduce error rate [10].

Pooling Layer:

Pooling layers are added after successive convolutional layer. A pooling layer in CNNs aggregates the outputs of neighboring neurons in the same filter or map to reduce spatial size of the feature maps. Pooling layers reduce data representation and help control overfitting. Each pooling summarizes the area they cover. Max pooling uses the max () operation to resize input spatially. For example, with a 2 x 2 filter, max () takes the largest value out of the four values. The result is a compact feature map [10].

Fully Connected Layer:

This layer is used to calculate the probabilities of the output classes for the input data. The output is a vector with N numbers representing the probability of each of the N output classes [10].

Softmax:

Softmax nonlinearity is more specialized compared to the general nonlinearities listed above. It is defined a

$$\text{Softmax } (x)_i = \frac{\exp(xi)}{\sum_{j=1}^{n} \exp(xj)} \text{ , } \quad \text{x} \in R^n$$

and maps a vector x ∈ R^n to a probability vector of length n. The intuition be-hind softmax is as follows: map x → exp(x) gives an order-preserving bijection between the set of real numbers R and the set of strictly positive real numbers R>0, so that for any indexes i,j we have $xi < xj$ if and only if exp(xi) < exp(xj).

Subsequent division by $\sum_{j=1}^{n} \exp(xj)$ normalizes the result, giving probability vec-tor as the output. This nonlinearity is used, e.g., in classification tasks, after the final fully connected layer with n outputs in a n-class classification problem. It should be noted, however, that softmax outputs do not truly model prediction uncertainty in the scenario of noisy labels (such as noisy organ segmentations in medical imaging) [11] .

III-4 Different types of algorithms to find out about Sleepiness:

III-4.1 convolution neural network AlexNet:

In 2012, Alex Krizhevesky and others [12]proposed a deeper and wider CNN model compared to LeNet and won the most difficult ImageNet challenge for visual object recognition called the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2012. AlexNet achieved state-of-the-art recognition accuracy against all the traditional machine learning and computer vision approaches. It was a significant breakthrough in the field of machine learning and computer vision for visual recognition and classification tasks and is the point in history where interest in deep learning increased rapidly [13] .

The architecture of AlexNet is shown in Figure 16. The first convolutional layer performs convolution and max pooling with Local Response Normalization (LRN) where 96 different receptive filters are used that are 11×11 in size. The max pooling operations are performed with 3×3 filters with a stride size of 2. The same operations are performed in the second layer with 5×5 filters. 3×3 filters are used in the third, fourth, and fifth convolutional layers with 384, 384, and 296 feature maps respectively. Two fully connected (FC) layers are used with dropout followed by a Softmax layer at the end. Two networks with similar structure and the same number of feature maps are trained in parallel for this model. Two new concepts, Local Response Normalization (LRN) and dropout, are introduced in this network. LRN can be applied in two different ways: first applying on single channel or feature maps, where an N×N patch is selected from same feature map and normalized based one the neighborhood values. Second, LRN can be applied across the channels or feature maps (neighborhood along the third dimension but a single pixel or location) [13] .

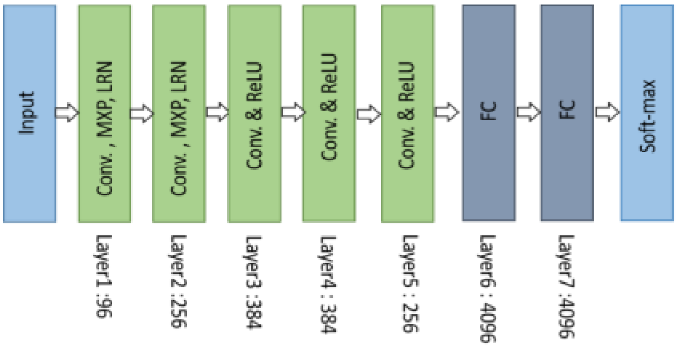
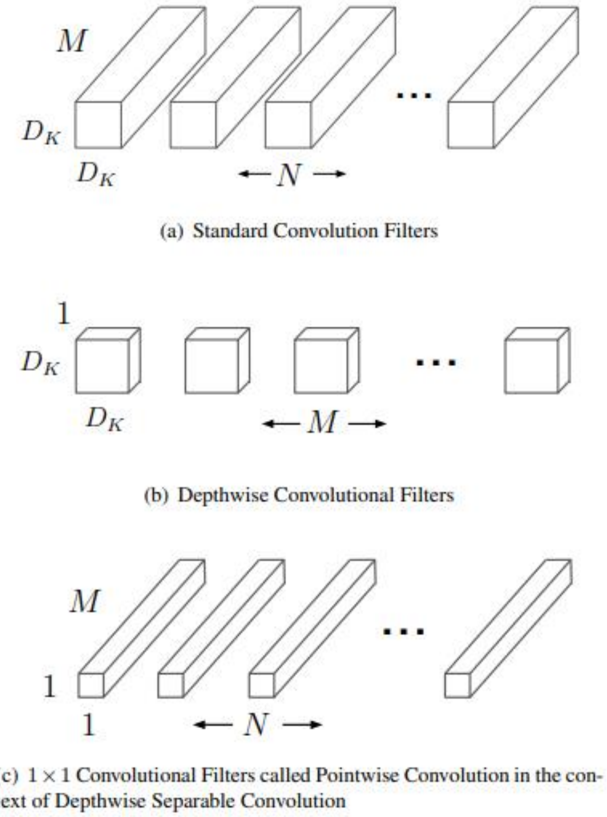


Figure 26:Architecture of AlexNet: Convolution, max-pooling, LRN and fully connected (FC) layer.

AlexNet has 3 convolution layers and 2 fully connected layers. When processing the ImageNet dataset, the total number of parameters for AlexNet can be calculated as follows for the first layer: input samples are 224×224×3, filters (kernels or masks) or a receptive field that has a size 11, the stride is 4, and the output of the first convolution layer is 55×55×96. According to the equations in section 3.1.4, we can calculate that this first layer has 290400 (55×55×96) neurons and 364 (11 ×11×3 = 363 + 1 bias) weights. The parameters for the first convolution layer are 290400×364 = 105,705,600. Table II shows the number of parameters for each layer in millions. The total number of weights and MACs for the whole network are 61M and 724M respectively [13] .

III-4.2 MobileNet:

The MobileNet model is based on depthwise separable convolutions which is a form of factorized convolutions which factorize a standard convolution into a depthwise convolution and a 1×1 convolution called a pointwise convolution. For MobileNets the depthwise convolution applies a single filter to each input channel. The pointwise convolution then applies a 1×1 convolution to combine the outputs the depthwise convolution. A standard convolution both filters and combines inputs into a new set of outputs in one step. The depthwise separable convolution splits this into two layers, a separate layer for filtering and a separate layer for combining. This factorization has the effect of drastically reducing computation and model size. Figure 22 shows how a standard convolution 2(a) is factorized into a depthwise convolution 2(b) and a 1 × 1 pointwise convolution 2(c) [16] .


M
D_K
D_K
$\leftarrow N \rightarrow$
1
D_K
D_K
$\leftarrow M \rightarrow$
M
1
1
$\leftarrow N \rightarrow$

(a) Standard Convolution Filters

(b) Depthwise Convolutional Filters

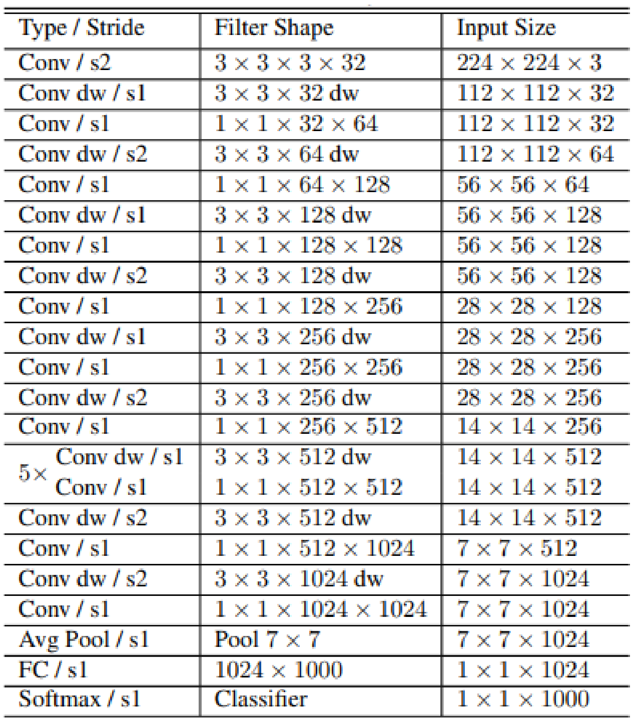
(c) 1 × 1 Convolutional Filters called Pointwise Convolution in the context of Depthwise Separable Convolution

.

Figure 27: The standard convolutional filters in (a) are replaced by two layers: depthwise convolution in (b) and pointwise convolution in (c) to build a depthwise separable filter.

The MobileNet structure is built on depthwise separable convolutions as mentioned in the previous section except for the first layer which is a full convolution. By defining the network in such simple terms, we are able to easily explore network topologies to find a good network. The MobileNet architecture is defined in Table 3. All layers are followed by a batchnorm and ReLU nonlinearity with the exception of the final fully connected layer which has no nonlinearity and feeds into a softmax layer for classification. Down sam-pling is handled with strided convolution in the depthwise convolutions as well as in the first layer. A final average pooling reduces the spatial resolution to 1 before the fully connected layer. Counting depthwise and pointwise convo-lutions as separate layers, MobileNet has 28 layers [16] .

Table 6 :MobileNet Body Architecture.



Type / Stride	Filter Shape	Input Size
Conv / s2	3 × 3 × 3 × 32	224 × 224 × 3
Conv dw / s1	3 × 3 × 32 dw	112 × 112 × 32
Conv / s1	1 × 1 × 32 × 64	112 × 112 × 32
Conv dw / s2	3 × 3 × 64 dw	112 × 112 × 64
Conv / s1	1 × 1 × 64 × 128	56 × 56 × 64
Conv dw / s1	3 × 3 × 128 dw	56 × 56 × 128
Conv / s1	1 × 1 × 128 × 128	56 × 56 × 128
Conv dw / s2	3 × 3 × 128 dw	56 × 56 × 128
Conv / s1	1 × 1 × 128 × 256	28 × 28 × 128
Conv dw / s1	3 × 3 × 256 dw	28 × 28 × 256
Conv / s1	1 × 1 × 256 × 256	28 × 28 × 256
Conv dw / s2	3 × 3 × 256 dw	28 × 28 × 256
Conv / s1	1 × 1 × 256 × 512	14 × 14 × 256
5× Conv dw / s1	3 × 3 × 512 dw	14 × 14 × 512
Conv / s1	1 × 1 × 512 × 512	14 × 14 × 512
Conv dw / s2	3 × 3 × 512 dw	14 × 14 × 512
Conv / s1	1 × 1 × 512 × 1024	7 × 7 × 512
Conv dw / s2	3 × 3 × 1024 dw	7 × 7 × 1024
Conv / s1	1 × 1 × 1024 × 1024	7 × 7 × 1024
Avg Pool / s1	Pool 7 × 7	7 × 7 × 1024
FC / s1	1024 × 1000	1 × 1 × 1024
Softmax / s1	Classifier	1 × 1 × 1000

III-4.3 Model C :

The Model C has been introduced with a drowsiness detection system [17] and consists of:

The first convolutional layer performs convolution and max pooling using local response normalization (LRN) in which 256 different receive filters of 3 x 3 size are used. Max pooling is performed using 2 x 2 filters. The same operations are performed in layer two, three, and four. Filters are used 3×3 in the 3rd, 4th, and 5th conv layers with 128, 64, and 32 with max pooling performed using 2×2 filters after each conv layer. Two fully connected (FC) layers are used with a dropout followed by a Softmax layer at the end. Model C is a simple and lightweight model on systems, which allows it to easily learn and implement in real time and is compatible with weak devices.

III-5 Human Face Detection Techniques :

There are several techniques used in the identification of facial features in this work were used:

OpenCV :

let’s examine a popular tool used to implement them. OpenCV is a computer vision library that supports programming languages like Python, C++, and Java.The package was initially created by Intel in 1999 and was later made open-source and released to the public.

OpenCV allows developers and non-mathematicians to build computer vision applications easily without having to code them from scratch. The library has over 2,500 algorithms that allow users to perform tasks like face recognition and object detection.

In this work, we will use OpenCV to perform face detection in Python [22].

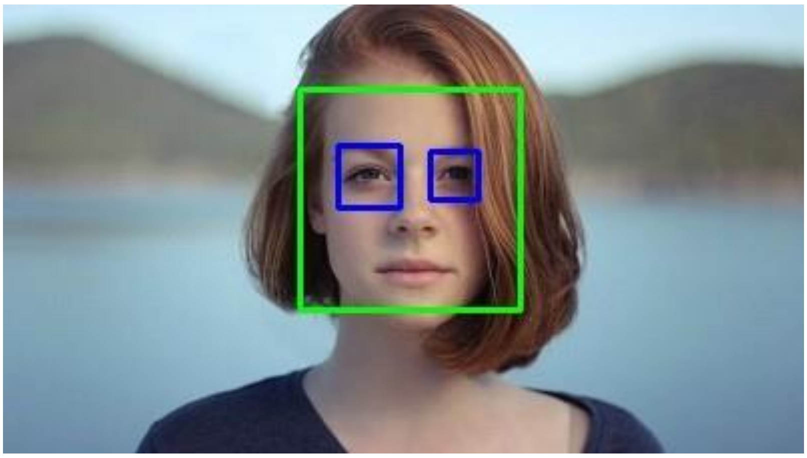


Figure 28 :An example of using opencv

III-6 Conclusion:

In this chapter various methods of detecting drowsiness, extracting the face, eyes, mouth and many other auxiliary features such as Viola Jones were proposed. Methods were presented for the design and development of an accurate, effective and powerful drowsiness detection system for real driving conditions. In the next chapter we will by implementing some cnn models using Yawdd data.

Chapter IV: Discussing and findings results

IV-1 Introduction:

In this chapter, we will review the results that we obtained from the application of various algorithms, and we will compare them and reach the best algorithm.

IV-2 Data used:

The driver drowsiness dataset was extracted from the faces and cropped drivers' eyes from the videos of the real-life drowsiness dataset. Frames were extracted from the video clips as images. Next, extract the region of interest from the captured images. The obtained dataset was used to train and test a CNN architecture for driver drowsiness detection in the paper “Detection and Prediction of Driver Inactivity for Road Accident Prevention Using Deep Neural Network Techniques” [22].

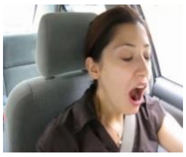
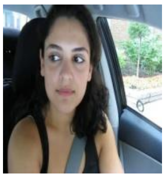
Table 7: How to extract the result.

classifications	case	Model output
closed eye	There is drowsiness	2
open eye	There is no drowsiness	3
yawn	There is fatigue	0
Don't yawn	There is no fatigue	1

The table above shows the possibilities and classifications of the study components.

Corresponding to the designations of the real issues and the expectations set by the model, as well as the outputs of the model represented by numbers as abbreviations to express the result reached.

Table 8 :to explain the data.

data	Yawn	Don't yawn	closed eye	open eye
Some samples of the data used in the study	 	  	 	
Number of photos	723	725	726	726

The above table shows the data divisions proposed by US and provides some random examples taken from the aggregated data used in the study for all the cases followed. In order to clarify the existing differences and increase transparency and credibility of the results.

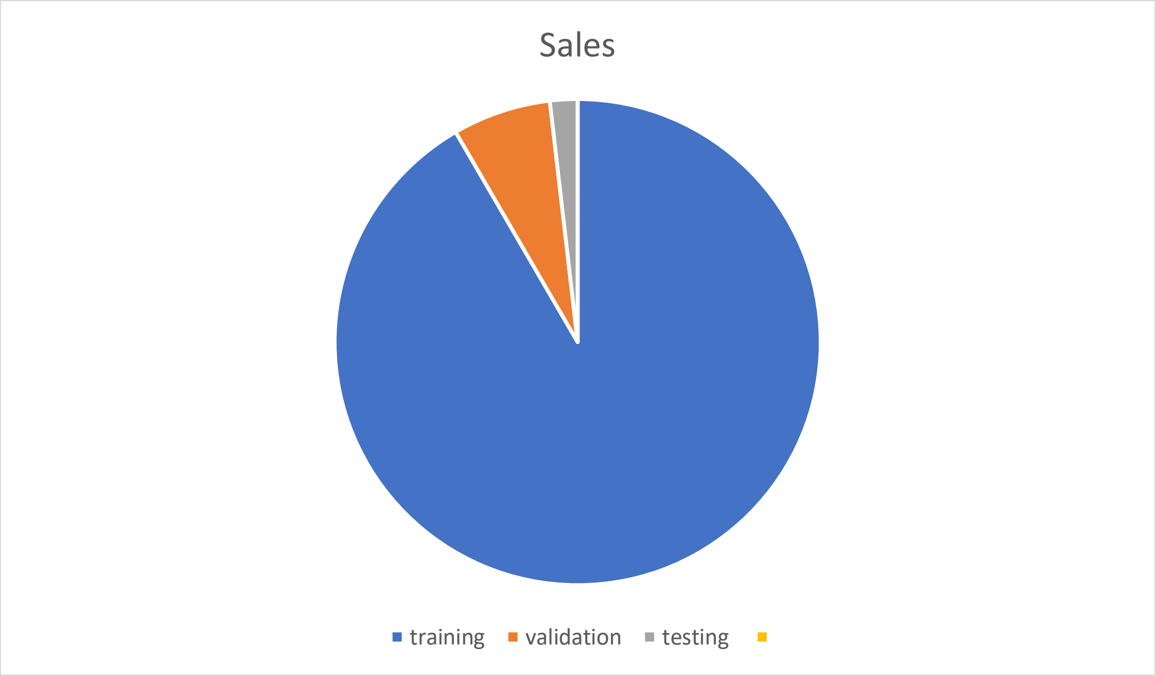


Figure 29:Percentage of the data.

The following drawing shows a proportional circle showing the division ratios of the studied data. It was divided into three parts:

- First, the training part is the most important part of deep learning, which in turn took the largest percentage of 70 in the division in order to teach the model and give it a huge amount of data and detailed information in various situations, making it easier for him to make the right decisions according to his future information

- validation, the 10% certification is considered a non-mandatory but important part, its importance lies in testing the models built during the training period and choosing the optimal model from among the group to be the approved model in the process

- testing, the 20% test is considered an important part of it in the model formation cycle, its importance lies in testing the credibility of the training, demonstrating its transparency and verifying its decisions. This results in an accuracy ratio, which in turn is the end result of a model and the approved reference that shows the efficiency of the model in classifying cases and its own description that distinguishes it from other models.

IV-2 Materiel:

Our drowsiness detection system is implemented at Kaggle and is powered by:

Used HP Pavilion 15 laptop contains Intel I5 11 Generation processor and Intel Iris screen cart, which comes with 8 GB RAM and 512 GB SSD internal memory, with Microsoft Windows 11 64-bit operating system.

IV-3 Performance evaluations:

Confusion matrices are used to objectively evaluate the performance of models used in solving classification problems. Confusion matrices have 4 different values obtained according to the classification results of the models. These are true positive (TP), true negative (TN), false positive (FP), and false negative (FN) values. A two-class confusion matrix is shown in Table 9 [23].

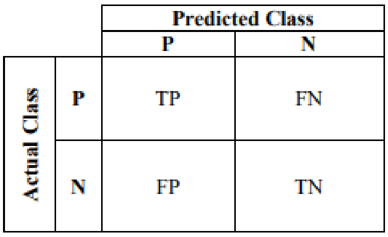
true positive (TP): TP is the number of times the intelligent system correctly identified the data. In other words, it is a measure of how many valid positive results the intelligent system has found.

true negative (TN): TN is the number of times the intelligent system correctly identified the data; in other words it is a measure of the amount of correct negative results found by the intelligent system.

false positive (FP): FP is that the intelligent system provides a false positive analysis, in other words it is a measure of the amount of false positive results that the intelligent system gives.

false negative (FN): FN is that the smart system fails to identify the data correctly, in other words measuring the impact of the wrong results given by the smart system.

Table 9 :Two-class confusion matrix.



		Predicted Class	
		P	N
Actual Class	P	TP	FN
	N	FP	TN

Evaluation of models is done with different metrics by using confusion matrix values. The metrics used in the study are Accuracy (ACC), F-1 score (FSC), Precision (PSC), Recall (RCL) and Specificity (SPC). The calculation of these metrics is shown in Table 10 [23].

Accuracy : The definition of model accuracy is the ability of the model to predict the truth or correctly predict a new sample. The higher the accuracy value, the more correctly the model can predict the truth or predict a new sample. Accordingly, improving the accuracy of the model depends on choosing and using appropriate tools to train and implement the model in line with the available data, technical and cultural requirements, and other influencing factors.

Precision (p) : Precision AI metric is the metric that identifies the proportion of items that are correctly classified as part of the classification of the given category, compared to the total number of items that are classified as part of the classification of the given category, when new data is sent to a machine learning model that has been trained to categorize items into different categories . The Precision metric is used to assess the quality and accuracy of the machine learning model in selecting the correct category.

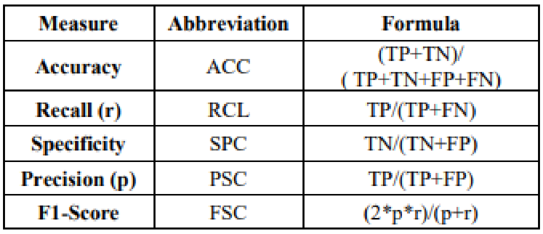
Recall (r) : The sensitivity measure in artificial intelligence is a measure of the ability of the artificial intelligence model to accurately detect certain cases, and it expresses the rate of the number of true positive cases that are detected (True Positives) divided by the number of correct positive cases found in the basic data.

If the desired value is that all positive cases are detected accurately, then the model should have a sensitivity value close to 100%.

specificity : It is the ability to recognize images/data of a particular category compared to others. specificity is calculated when we have a model that is trained to classify images/data into different categories, when a new image/data is sent to the model, its ability to accurately recognize which category the image/data belongs to is calculated, in other words, how many images that are already recognized as being part of the selected category compared to the total number of images submitted for analysis in the selected category. Specificity is used as a metric to assess the quality of a machine learning model and its ability to accurately identify objects.

F1-Score : F1 score is a common metric used to evaluate the performance of AI models in detecting True Positives among expected positives as well as extracting the number of False Positives and False Negatives. It is calculated as the ratio of average accuracy and recall.

Table 10 :Calculation of performance metrics.



Measure	Abbreviation	Formula
Accuracy	ACC	(TP+TN)/ (TP+TN+FP+FN)
Recall (r)	RCL	TP/(TP+FN)
Specificity	SPC	TN/(TN+FP)
Precision (p)	PSC	TP/(TP+FP)
F1-Score	FSC	(2*p*r)/(p+r)

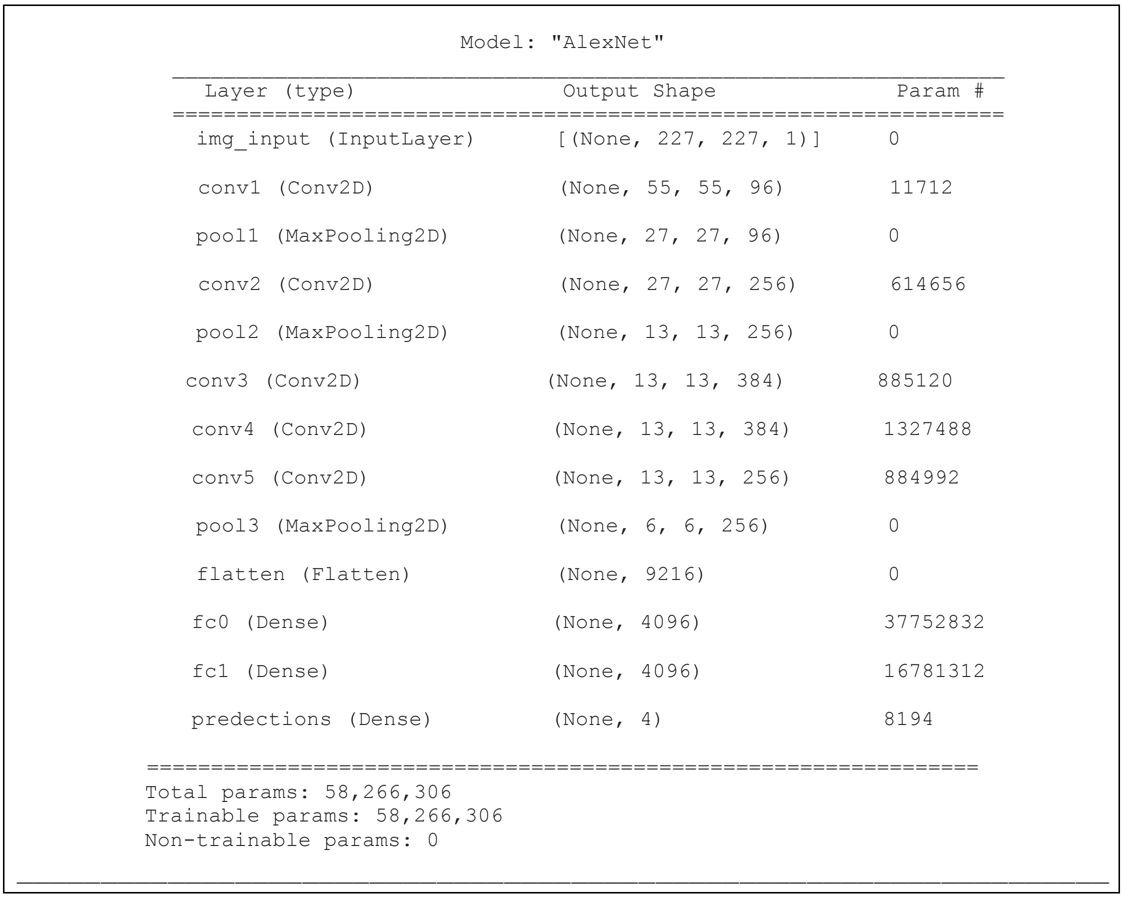
IV-4 Evaluation metrics:

Build the form and Implementation and Results:

Model number one (AlexNet):

The form shows the model layers display with the output matrix after applying and calculating settings:

Table 11 :layer and output and parm (AlexNet).



```
                         Model: "AlexNet"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 img_input (InputLayer)      [(None, 227, 227, 1)]     0

 conv1 (Conv2D)              (None, 55, 55, 96)        11712

 pool1 (MaxPooling2D)        (None, 27, 27, 96)        0

 conv2 (Conv2D)              (None, 27, 27, 256)       614656

 pool2 (MaxPooling2D)        (None, 13, 13, 256)       0

 conv3 (Conv2D)              (None, 13, 13, 384)       885120

 conv4 (Conv2D)              (None, 13, 13, 384)       1327488

 conv5 (Conv2D)              (None, 13, 13, 256)       884992

 pool3 (MaxPooling2D)        (None, 6, 6, 256)         0

 flatten (Flatten)           (None, 9216)              0

 fc0 (Dense)                 (None, 4096)              37752832

 fc1 (Dense)                 (None, 4096)              16781312

 predections (Dense)         (None, 4)                 8194

=================================================================
Total params: 58,266,306
Trainable params: 58,266,306
Non-trainable params: 0
_________________________________________________________________
```

Figure 30 shows the validation accuracy of the AlexNet convolutional neural network. Adam, an adaptive learning rate optimization technique, was used with an initial learning rate of 0.00001, batch size of 32, 50 points, 43iterations, and an accuracy of 94%. The elapsed time is 3 hours 46 minutes. To better visualize the overall prediction performance, confusion matrices were used.

Figure 31 shows the confusion matrix of the AlexNet that is used to calculate the model properties table 12.

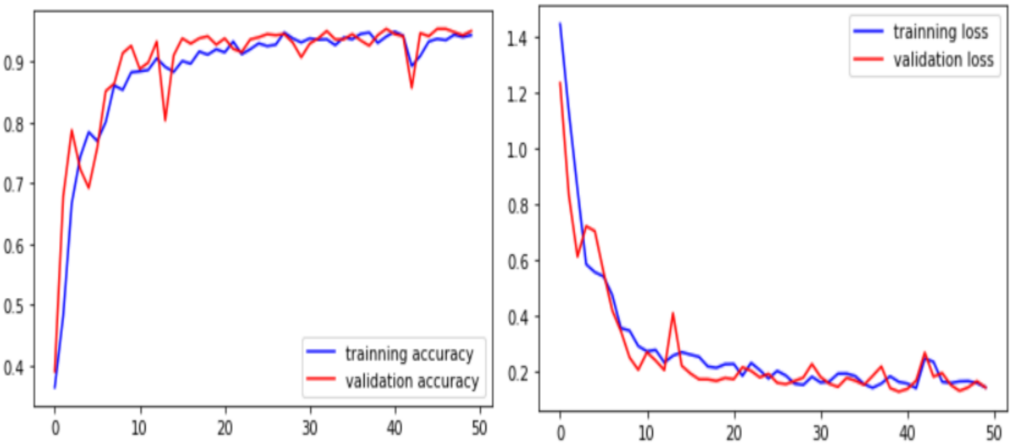


Figure 30: Validation and accuracy and loss for AlexNet.

We note through the graph the training of the model in improving the accuracy, which reached 97.6% and reducing the loss, which reached 2.4%.

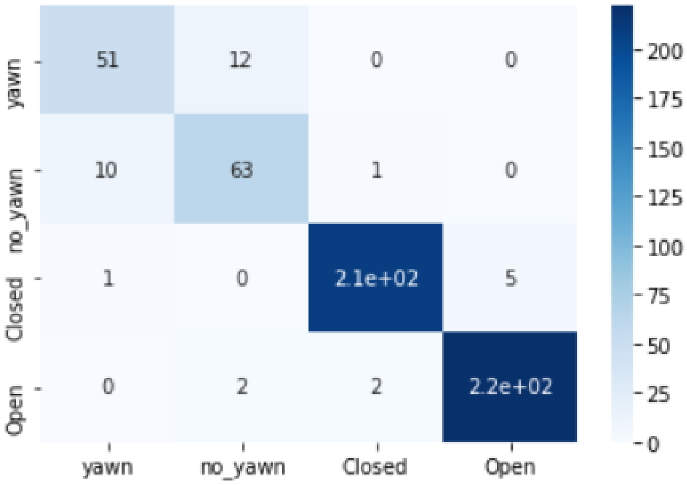
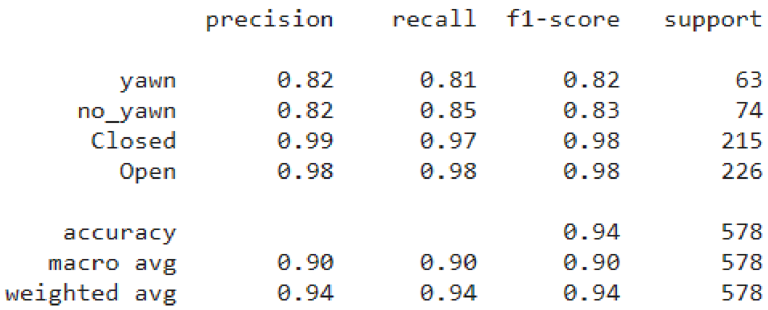


Figure 31: confusion matrices for AlexNet.

Through the statistical reading of the confusion matrix showing the total prediction probabilities and decisions made by the model, where we note that the model distributed the data as follows: 220 eyes open, 210 eyes closed, 51 yawns and 63 no yawns, as it gave greater priority to the cases of eyes (closed / open) in decision-making and forecasting, which in turn represents an accurate indicator in identifying cases of drowsiness, which was reflected in the high accuracy in the results of drowsiness, it also provided acceptable performance in the study of other cases (yawn/ no yawn) it is classified to identify cases of fatigue for a driver even before the occurrence of drowsiness, which gives him high efficiency in detecting and even predicting the occurrence of drowsiness .

Table 12 :Model properties extracted from the matrix.



```
              precision    recall  f1-score   support

        yawn       0.82      0.81      0.82        63
     no_yawn       0.82      0.85      0.83        74
      Closed       0.99      0.97      0.98       215
        Open       0.98      0.98      0.98       226

    accuracy                           0.94       578
   macro avg       0.90      0.90      0.90       578
weighted avg       0.94      0.94      0.94       578
```

The table represents the detailed and final results of a model, as it shows several of the reference measures to judge the credibility of the model.

First **precision** which achieved a result (82%/82%/99%/98%) for all cases

(Yawn/ don't yawn/ closed eye /open eye) in order at a rate of macro avg =90%

Secondly **recall** that yielded results (81%/85%/97%/98%) for all cases (yawn/ no yawn/ closed eye /open eye) respectively with a macro avg =90%

Third **F1/score** which achieved results (82%/83%/98%/98%) for all cases (yawn/ no yawn/ closed eye /open eye) respectively with a macro avg = 90 %

Finally, the overall accuracy of the model, which was an excellent result, reached **94.**

Model number two (MobileNet):

The form shows the model layers display with the output matrix after applying and calculating settings:

Table 13 :layer and output and parm (MobileNet).

```
                    Model: " MobileNet"
_________________________________________________________________
 Layer (type)              Output Shape              Param #
=================================================================
 conv2d_4 (Conv2D)          (None, 143, 143, 96)      2688
_________________________________________________________________
 max_pooling2d_4 (MaxPooling2 (None, 71, 71, 96)      0
_________________________________________________________________
conv2d_5 (Conv2D)          (None, 69, 69, 256)       221440
_________________________________________________________________
max_pooling2d_5 (MaxPooling2 (None, 34, 34, 256)      0
_________________________________________________________________
conv2d_6 (Conv2D)          (None, 32, 32, 384)       885120
_________________________________________________________________
max_pooling2d_6 (MaxPooling2 (None, 16, 16, 384)      0
_________________________________________________________________
conv2d_7 (Conv2D)          (None, 14, 14, 384)       1327488
_________________________________________________________________
 max_pooling2d_7 (MaxPooling2 (None, 7, 7, 384)       0
_________________________________________________________________
 conv2d_8 (Conv2D)          (None, 5, 5, 256)         884992
_________________________________________________________________
 max_pooling2d_8 (MaxPooling2 (None, 2, 2, 256)       0
_________________________________________________________________
      flatten_1 (Flatten)       (None, 1024)              0
_________________________________________________________________
  dense_2 (Dense)             (None, 4096)              4198400
_________________________________________________________________
      dropout_1 (Dropout)       (None, 4096)              0
_________________________________________________________________
 dense_3 (Dense)             (None, 4096)              16781312
_________________________________________________________________
      dropout_2 (Dropout)       (None, 4096)              0
_________________________________________________________________
      dense_4 (Dense)           (None, 4)                 16388
=================================================================
                 Total params: 24,317,828
               Trainable params: 24,317,828
                 Non-trainable params: 0
_________________________________________________________________
```

Figure 32 shows the validation accuracy of the MobileNet convolutional neural network. Adam, an adaptive learning rate optimization technique, was used with an initial learning rate of 0.00001, batch size of 32, 50 points, 43iterations, and an accuracy of 94%. The elapsed time is 3 hours 12 minutes. To better visualize the overall prediction performance, confusion matrices were used.

Figure 33 shows the confusion matrix of the VGG-16 that is used to calculate the model properties table 14.

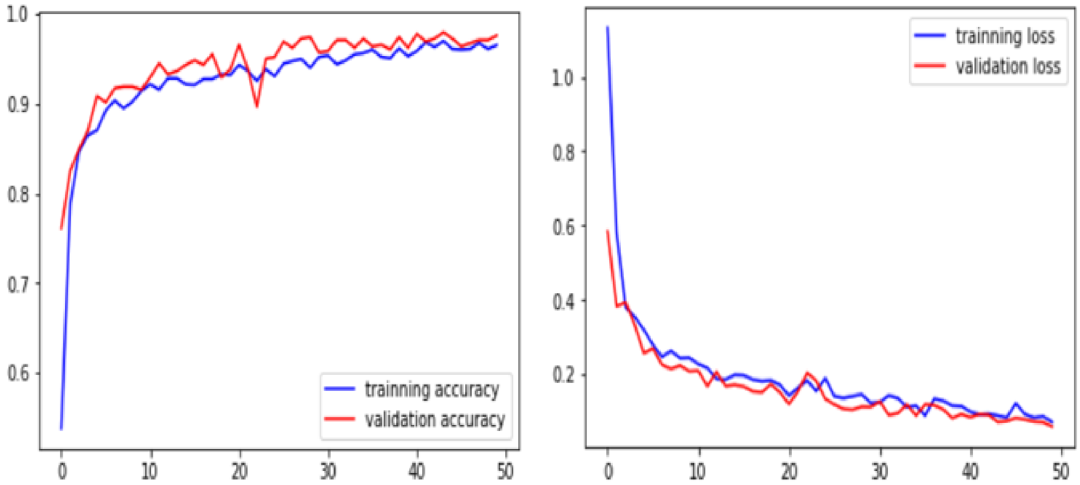


Figure 32 :Validation and accuracy and loss for MobileNet.

The graph demonstrates how training improved the model's accuracy, which increased to 94%, and decreased loss, which decreased to 6%.

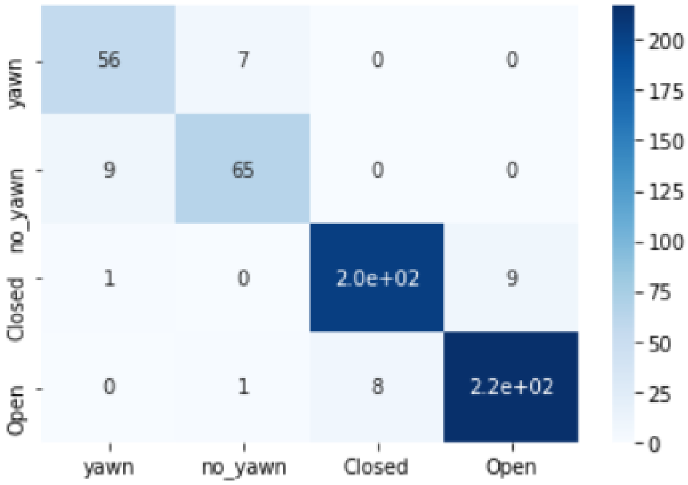
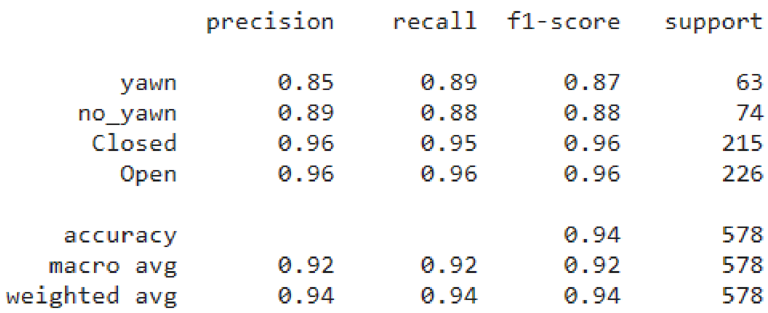


Figure 33: confusion matrices for MobileNet.

Through the statistical reading of the confusion matrix that shows the total prediction probabilities and decisions made by the model, where we note that the model distributed the data as follows: 220 eyes open, 200 eyes closed, 56 yawns and 65 do not yawn, as the second algorithm also gave more priority to the cases of eyes (closed / open) in decision-making and forecasting, which in turn represents an accurate indicator in identifying cases of drowsiness, which was reflected in the high accuracy in the results of drowsiness, and also provided acceptable performance in the study of other cases (yawning / non-yawning) is classified to identify cases of fatigue of the driver even before the occurrence of drowsiness, which gives him high efficiency in detecting and even predicting the occurrence of drowsiness .

Table 14:Model properties extracted from the matrix.



```
              precision    recall  f1-score   support

        yawn       0.85      0.89      0.87        63
     no_yawn       0.89      0.88      0.88        74
      Closed       0.96      0.95      0.96       215
        Open       0.96      0.96      0.96       226

    accuracy                           0.94       578
   macro avg       0.92      0.92      0.92       578
weighted avg       0.94      0.94      0.94       578
```

The table represents the detailed and final results of a model, as it shows several of the reference measures to judge the credibility of the model

First **precision** which achieved a result (85%/89%/96%/96%) for all cases (Yawn/ don't yawn/ closed eye /open eye) in order at a rate of macro avg =92%

Secondly **recall** that yielded results (89%/88%/95%/96%) for all cases (yawn/ no yawn/ closed eye /open eye) respectively with a macro avg =92%

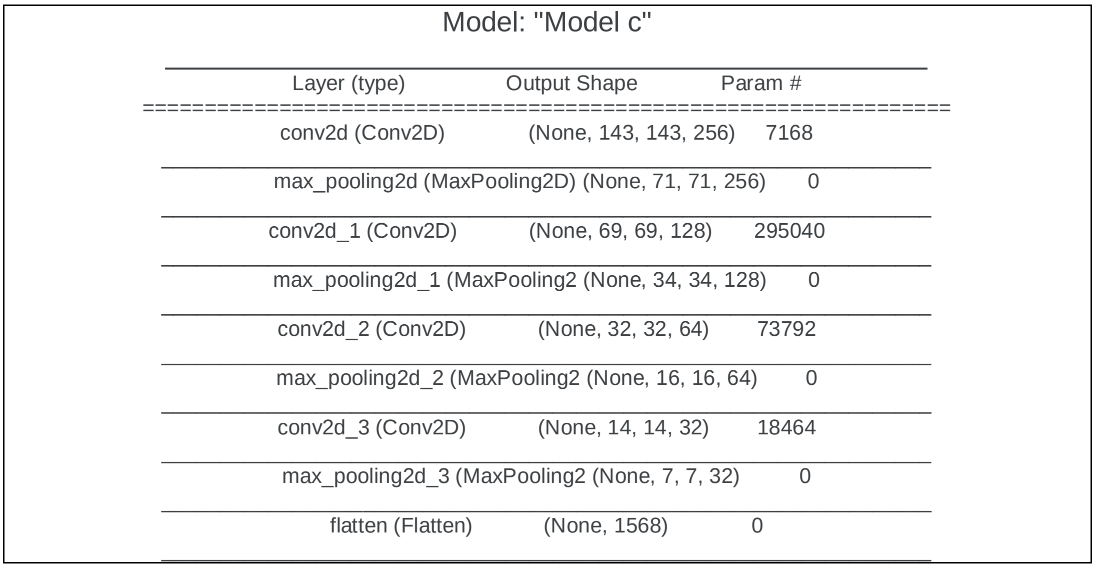
Third **F1/score** which achieved results (87%/88%/96%/96%) for all cases (yawn/ no yawn/ closed eye /open eye) respectively with a macro avg = 92 %

Finally, the overall accuracy of the model, which was an excellent result like the first model, reached **94%.**

Model number three (sequential) :

The form shows the model layers display with the output matrix after applying and calculating settings :

Table 15 :layer and output and parm (Model c).



```
Model: "Model c"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 143, 143, 256)     7168
_________________________________________________________________
 max_pooling2d (MaxPooling2D) (None, 71, 71, 256)      0
_________________________________________________________________
 conv2d_1 (Conv2D)           (None, 69, 69, 128)       295040
_________________________________________________________________
 max_pooling2d_1 (MaxPooling2 (None, 34, 34, 128)      0
_________________________________________________________________
 conv2d_2 (Conv2D)           (None, 32, 32, 64)        73792
_________________________________________________________________
 max_pooling2d_2 (MaxPooling2 (None, 16, 16, 64)       0
_________________________________________________________________
 conv2d_3 (Conv2D)           (None, 14, 14, 32)        18464
_________________________________________________________________
 max_pooling2d_3 (MaxPooling2 (None, 7, 7, 32)         0
_________________________________________________________________
 flatten (Flatten)           (None, 1568)              0
_________________________________________________________________
```

```
                         dropout (Dropout)          (None, 1568)            0
              ________________________________________________________________
                       dense (Dense)             (None, 64)              100416
              ________________________________________________________________
                         dense_1 (Dense)           (None, 4)               260
            ===================================================================
     Total params: 495,140
     Trainable params: 495,140
      Non-trainable params: 0
```

Figure 34 shows the validation accuracy of the model c convolutional neural network. Adam, an adaptive learning rate optimization technique, was used with an initial learning rate of 0.00001, batch size of 32, 50 points, 43iterations, and an accuracy of 94%. The elapsed time is 3 hours. To better visualize the overall prediction performance, confusion matrices were used.

Figure 35 shows the confusion matrix of the Model C1 that is used to calculate the model properties table 16.

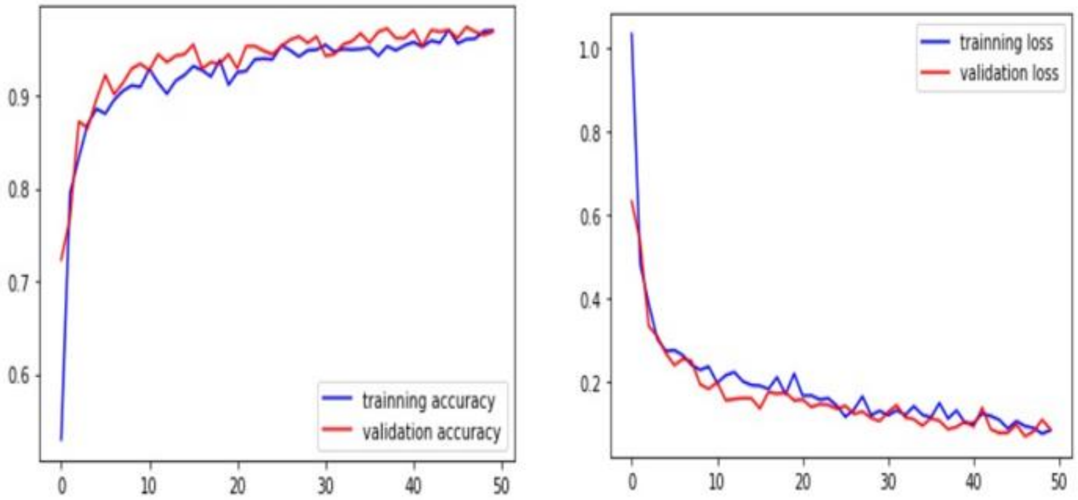


Figure 34: Validation and accuracy and loss for Model C .

The graph shows how the model's training increased accuracy, which reached 96%, and decreased loss, which reached 4%.

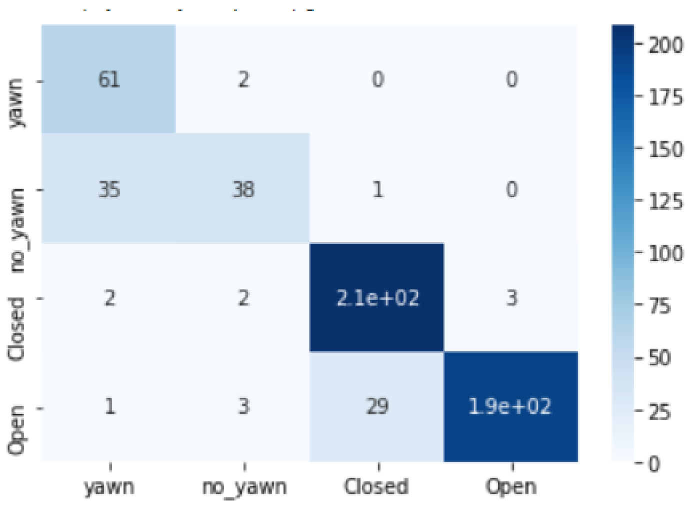
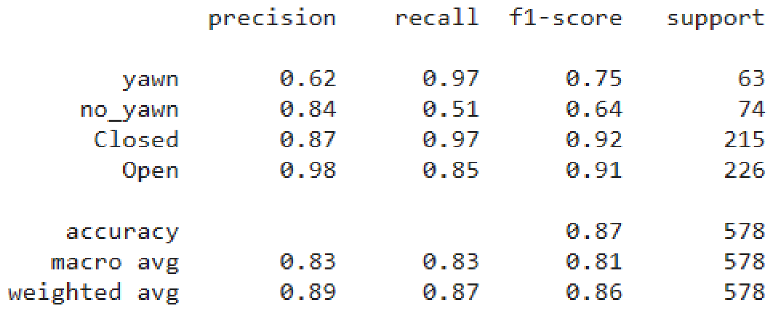


Figure 35: confusion matrices.

Through the statistical reading of the confusion matrix that shows the total prediction probabilities and decisions made by the model, where we note that the model distributed the data as follows: 190 eyes open, 210 eyes closed, 61 yawns and 38 do not yawn, as the second algorithm also gave more priority to the cases of eyes (closed / open) in decision-making and forecasting, which in turn represents an accurate indicator in identifying cases of drowsiness, which was reflected in the high accuracy in the results of drowsiness, and also provided acceptable performance in the study of other cases (yawning / non-yawning) is classified to identify cases of fatigue of the driver even before the occurrence of drowsiness, which gives him high efficiency in detecting and even predicting the occurrence of drowsiness .

Table 16: Model properties extracted from the matrix.



```
              precision    recall  f1-score   support

        yawn       0.62      0.97      0.75        63
     no_yawn       0.84      0.51      0.64        74
      Closed       0.87      0.97      0.92       215
        Open       0.98      0.85      0.91       226

    accuracy                           0.87       578
   macro avg       0.83      0.83      0.81       578
weighted avg       0.89      0.87      0.86       578
```

The table represents the detailed and final results of a model, as it shows several of the reference measures to judge the credibility of the model:

First **precision** which achieved a result (62%/84%/87%/98%) for all cases (Yawn/ don't yawn/ closed eye /open eye) in order at a rate of macro avg =83%

Secondly **recall** that yielded results (97%/51%/97%/85%) for all cases (yawn/ no yawn/ closed eye /open eye) respectively with a macro avg =83%

Third **F1/score** which achieved results (75%/64%/92%/91%) for all cases (yawn/ no yawn/ closed eye /open eye) respectively with a macro avg = 81 %

Finally, the overall accuracy of the model, which was a result less than the rest of the algorithms, reached **87%.**

IV-5 Discussion:

The results below represent the overall description of a model, as it depends on the classification of the driver's real-time condition and determining his condition within one of the four classifications (sleepy/not sleepy/tired /not tired) , without relying on tribal inputs represented in an image divided into four groups under classifications (closed eye/open eye/ yawn/non-yawn) .

It also gives a higher priority in determining the condition of the eyes, as it in turn represents an effective indicator in determining the state of drowsiness. An optimal choice for our own study is the detection of driver drowsiness. It can also distinguish the immediate state of fatigue of a driver after the absence of a proven state of drowsiness to give us excellent results in predicting cases of drowsiness or fatigue in the studied position.

The following table shows a summary of the final results of the set of algorithms that make up the model with independent cases and used in the study. There are several reference measures for evaluating these algorithms. Without dependence on a macro avg scale. For all cases.

Table 17: General description of the model weights.

Cnn\Performance	accuracy	Precision	Recall	F1-score
AlexNet	94%	90%	90%	90%
MobileNet	94%	92%	92%	92%
Model C	87%	83%	83%	81%

Preference and choice of the final model:

The table below represents a key to reading the results of the last table 19 summary of our study, where it gives the indication of the ranking by giving more weight to each measure of the highest efficiency in order to choose the optimal final model:

Table 18: The key to reading the results of the last table.

rank	1	2	3
Weights(w)	3	2	1

Table 19:Trade-off table.

Performance \Cnn	AlexNet	MobileNet	Model C
accuracy	3	3	2
precision	2	3	1
recall	2	3	1
F1-score	2	3	1
k	0.913	0.925	0.835

The table above shows the final discussion and model selection.

The optimal is Mosher's proposal to separate the algorithms under the abbreviation (k) based on several scales attached with weights that give the indication of the order among them.

The method of calculating the scale (k) is to multiply the number of weights by the ratio of the accuracy of the scale and divide the result obtained by the sum of the weights of each algorithm. This is in order to obtain accurate results and an indicator for the adoption of a final model summarizing the result of our study.

The results of the algorithms according to Indicator K are as follows,

K=$\frac{acc*w_0+psc*w_1+rcl*w_2+fsc*w_3}{w_0+w_1+w_2+w_3}$

. $k_0(MobileNet) = 0.913$

. $k_1(AlexNet)$ =0.925

.$k_2(Model\ c) = 0.835$

From the results shown, we can choose the highest percentage model It is (mobile net), which in turn represents the most efficient model of the other models. And the last choice as a result of the whole field of our study. And the model relies in detecting the drowsiness of the driver in real life.

Chapter v: general conclusion

V-1 conclusion general :

Driver drowsiness means feeling overwhelmed while driving. Which affects the attention and ability of the driver to respond to various situations on the road and can lead to serious traffic accidents. In this study, we used artificial intelligence techniques based on the CNN algorithm by analyzing different eye and mouth positions. To identify cases of fatigue and drowsiness in drivers. The importance of the topic lies in reducing traffic accidents caused by drowsiness and fatigue, preserving lives and the safety of individuals, which in turn is of paramount importance and purpose in all scientific research journals.

V-1.1 Contributions:

We have also reached many results in the study, most notably:

Analyze the studied phenomenon and use algorithms adapted to it.

- Obtaining official reports and statistics on the number and causes of traffic accidents in Algiers by the security and the National Gendarmerie.

- The database was processed with different algorithms, compared with each other and selected the optimal algorithm for the study, relying on several measures to classify the most important accuracy.

- A model of convolutional neural networks has been built that is capable of real-time classification with accuracy (91%).

- Commitment to the time of studying and processing the topic submitted by the supervisor and the concerned party and delivering it within the required deadlines.

- One of the most important contributions is not to rely on the method common in previous studies of separating the specific elements of the study (eyes / mouth) in their different cases. And compare them with several algorithms. But in turn, we were able to find an effective solution by combining these elements consisting of four groups by finding a relationship between them. Using it as an input in a single model is able to determine the drowsiness of a driver with high accuracy and higher sensitivity to the detection of the studied phenomenon.

V-1.2 Future works:

- Improved model accuracy

- The design of the model is actually used directly to classify the cases to strengthen the cars of medium performance and cost

- Generalization of the study, depending on other characteristics of the face, such as the position of the eyebrows. Total facial expressions. Etc.

- The formation of a model supported by electronic elements to provide additional information in order to increase the accuracy of decisions.

- Using other deep learning algorithms such as RNN (Recurrent Neural Network) and try to combine them to get high-resolution results.

RNN recurrent neural networks differ from CNN serial artificial neural networks mostly in their use for processing cyclic data such as audio, video and text, whereas CNN networks rely on spatial data such as images and graphs.

CNN is better at handling three-dimensional digital images and signals, finding spatial relationships in images, and it also relies on detailed levels to extract important landmarks and features from the input data.

In contrast, RNNs rely on memory oriented to memorize and use previous information in current processing and deal better with structural and sequential data such as speaking, translation, audio signals and also text data. Which can give us better results, especially in the studied phenomenon of detecting driver drowsiness. It analyzes the entire situation of a driver and gives us an accurate prediction and decision of the actual situation.

Taken together, both models differ in their applications, but they can be used in combination to take advantage of the advantages of each in different areas of artificial intelligence, such as voice and image recognition in robots used for autonomous driving.

References

[1] A. G. Correa, L. Orosco and E. Laciar, "Automatic detection of drowsiness in EEG records based on multimodal analysis.," *Medical engineering & physics ,* pp. 244-249, 28 july 2013.

[2] R. . M. Salman, M. Rashid, R. Roy, M. . M. Ahsan and Z. Manjurul, "DRIVER DROWSINESS DETECTION USING ENSEMBLE CONVOLUTIONAL NEURAL NETWORKS ON YAWDD," *arXiv preprint arXiv,* pp. 2112-10298, 21 December 2021.

[3] D. Bosamiya and j. D. Fuletra, "A Survey on Driver's Drowsiness Detection Techniques," *International Journal on Recent and Innovation Trends in Computing and Communication ,* pp. 816-819, November 2013.

[4] R. S. Tavhare, "REAL TIME SLEEP/DROWSINESS DETECTION," pp. 2-26, 22 4 2019.

[5] V. R. R. Chirra, S. R. Uyyala and V. K. K. Kolli, "Deep CNN: A Machine Learning Approach for Driver Drowsiness Detection Based on Eye State," *d'Intelligence Artif,* pp. 461-466, 28 11 2019.

[6] I. Stancin, M. Cifrek and A. Jovic, "A Review of EEG Signal Features and Their Application in Driver Drowsiness Detection Systems," *Sensors,* pp. 1-29, 30 May 2021.

[7] R. Jabbar, M. Shinoy, M. Kharbeche, K. Al-Khalifa, M. Krichen and K. Barkaoui, "Driver drowsiness detection model using convolutional neural networks techniques for android application," *In 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT) ,* pp. 237-242, 17 Jan 2020.

[8] A. R. Niloy, A. I. Chowdhury and N. Sharmin, "A brief review on different Driver's drowsiness detection techniques," *International Journal of Image, Graphics and Signal Processing,* pp. 41-50, 08 June 2020.

[9] J. S. Bajaj , N. Kumar, R. K. Kaushal, H. L. Gururaj, F. Flammini and R. Natarajan, ""System and method for driver drowsiness detection using behavioral and sensor-based physiological measures.," *Sensors,* 23 January 2023.

[10] I. Aziz, "Deep learning: an overview of Convolutional Neural Network (CNN)," April 2020.

[11] J. Teuwen and N. Moriakov, "Chapter 20 - Convolutional neural networks," pp. 481-500, 2020.

[12] A. Krizhevsky, I. Sutskever and G. E. Hinton, "Imagenet classification with deep convolutional neural networks.," *Communications of the ACM,* pp. 84-90, JUNE 2017 .

[13] C. Yakopcic, S. Westberg, M. Z. Alom, T. M. Taha, P. Sidike, M. S. Nasrin, B. C. Van Esesn, A. A. S. Awwal and V. K. Asari, "The history began from alexnet: A comprehensive survey on deep learning approaches," *arXiv preprint arXiv:1803.01164 ,* 12 Sep Sep 2018.

[14] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich, "Going deeper with convolutions.," *Proceedings of the IEEE conference on computer vision and pattern recognition,* 2015.

[15] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition.," *Proceedings of the IEEE conference on computer vision and pattern recognition.,* pp. 770-778, 2016.

[16] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications.," *arXiv preprint arXiv : 1704.04861,* 17 April 2017.

[17] "kaggle kernels pull saurabhprajapat/driver-drowsiness-using-keras," 2022. [Online].

[18] M. K. Hasan , M. S. Ahsan, A. Al-Mamun, S. H. Shah Newaz and G. M. Lee, "Human Face Detection Techniques: A Comprehensive Review and Future Research Directions," *Electronics 2021,* 26 September 2021.

[19] C. MURUKESH and P. PADMANABHAN, "Drowsiness detection for drivers using computer vision," *WSEAS Transactions on Information Science and Applications,,* pp. 43-50, 2015.

[20] M. H. Sigari , M. R. Pourshahabi, M. Soryani and M. Fathy, "A review on driver face monitoring systems for fatigue and distraction detection," *International Journal of Advanced Science and Technology ,* pp. 73-100, 2014.

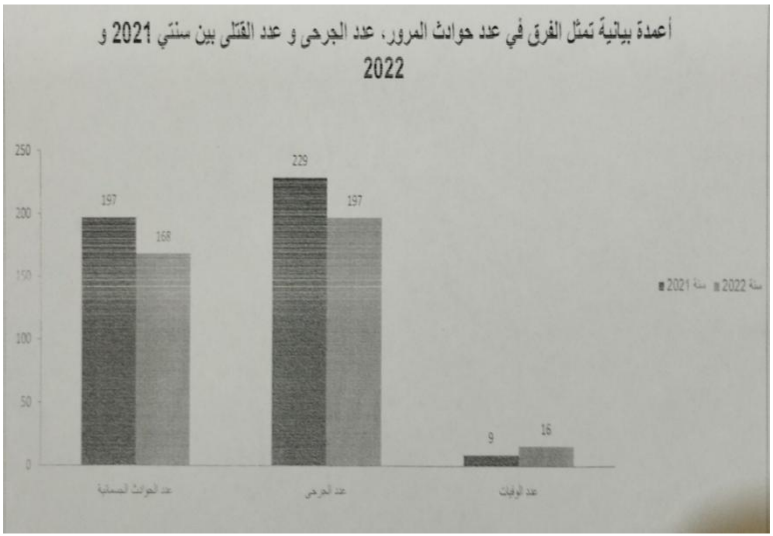
[21] L. M. Bergasa, J. Nuevo, M. A. Sotelo, R. Barea and M. E. Lopez, "Real-Time System for Monitoring Driver," *IEEE Transactions on Intelligent Transportation Systems,* 2016.

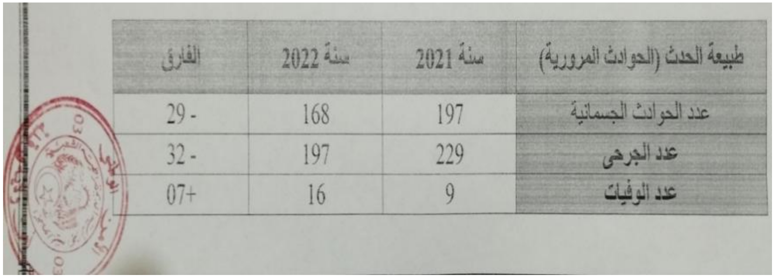
[22] S. Abtahi, M. Omidyeganeh, S. Shirmohammadi and B. Hariri, "YawDD: Yawning Detection Dataset," *IEEE Dataport,* 2020.

[23] S. . K. S. Al-Doori , Y. . S. Taspinar and M. Koklu, "Distracted Driving Detection with Machine Learning Methods by CNN Based Feature Extraction," *International Journal of Applied Mathematics Electronics and Computers ,* pp. 116-121, 22 December 2021.

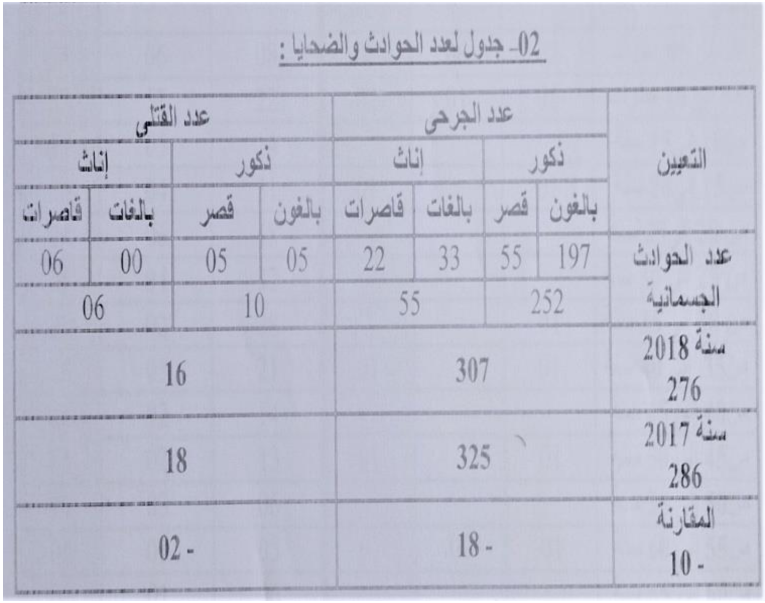
Appendices :

Appendix 01 :


أعمدة بيانية تمثل الفرق في عدد حوادث المرور، عدد الجرحى و عدد القتلى بين سنتي 2021 و 2022

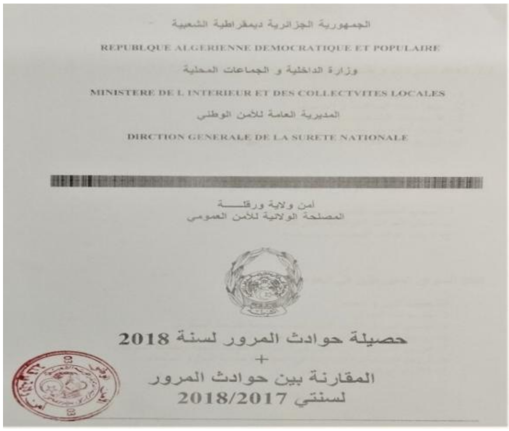


طبيعة الحدث (الحوادث المرورية)	سنة 2021	سنة 2022	الفارق
عدد الحوادث الجسمانية	197	168	29 -
عدد الجرحى	229	197	32 -
عدد الوفيات	9	16	07+



02- جدول لعدد الحوادث والضحايا :

التعيين	عدد الجرحى				عدد القتلى			
	ذكور		إناث		ذكور		إناث	
	بالغون	قصر	بالغات	قاصرات	بالغون	قصر	بالغات	قاصرات
عدد الحوادث الجسمانية	197	55	33	22	05	05	00	06
	252		55		10		06	
سنة 2018 276	307				16			
سنة 2017 286	325				18			
المقارنة 10 -	18 -				02 -			



الجمهورية الجزائرية ديمقراطية الشعبية

REPUBLQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة الداخلية و الجماعات المحلية

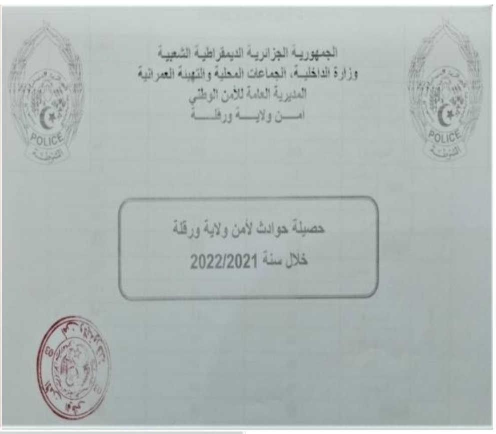
MINISTERE DE L INTERIEUR ET DES COLLECTVITES LOCALES

المديرية العامة للأمن الوطني

DIRCTION GENERALE DE LA SURETE NATIONALE

أمن ولاية ورقلــــــة
المصلحة الولائية للأمن العمومي

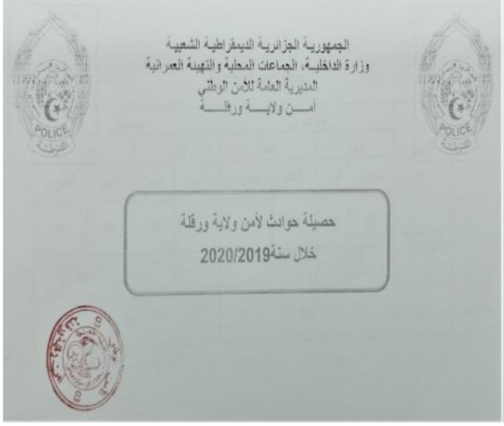
حصيلة حوادث المرور لسنة 2018
+
المقارنة بين حوادث المرور
لسنتي 2018/2017



الجمهوريـة الجزائريـة الديمقراطيـة الشعبيـة
وزارة الداخليـة، الجماعات المحلية والتهيئة العمرانية
المديرية العامة للأمن الوطني
أمــــن ولايــــة ورقلــــة

POLICE

حصيلة حوادث لأمن ولاية ورقلة
خلال سنة 2022/2021



الجمهوريـة الجزائريـة الديمقراطيـة الشعبيـة
وزارة الداخليـة، الجماعات المحلية والتهيئة العمرانية
المديرية العامة للأمن الوطني
أمــــن ولايــــة ورقلــــة

POLICE

حصيلة حوادث لأمن ولاية ورقلة
خلال سنة2020/2019