

République Algérienne Démocratique et Populaire  
Ministère de l'enseignement supérieur et de la recherche  
scientifique Université 8Mai 1945 – Guelma  
Faculté des sciences et de la Technologie  
Département d'Électronique et Télécommunications



**Mémoire de fin d'étude**  
**Pour l'obtention du diplôme de Master Académique**

Domaine : **Sciences et Technologie**  
Filière : **Électronique**  
Spécialité : **Instrumentation**

---

---

**Détection de la fatigue du conducteur avec Raspberry Pi**

---

---

Présenté par :

-----  
**BERRAMDANE Abderraouf**  
-----

Sous la direction de :  
**Dr. TABA Mohamed Tahar**

Jun 2022

## **Remerciement**

*Tout d'abord je remercie Dieu le tout puissant et miséricordieux, qui m'a donné la force et la patience d'accomplir ce travail.*

*Je remercie aussi mon directeur de mémoire, Monsieur Mohamed Tahar TABA Docteur à l'université 08 Mai 1945 à Guelma. Je le remercie de m'avoir assuré le suivi de cette mémoire. Son expérience et son aide scientifique m'ont été essentielles.*

*Je remercie beaucoup Monsieur Samir HALLACI Docteur à l'université 08 Mai 1945 -Département de l'Informatique- à son aide importante dans la réalisation de ce projet.*

*J'adresse mes sincères remerciements à tous les professeurs, intervenants et toutes les personnes qui par leurs paroles, leurs écrits, leurs conseils et leurs critiques ont guidé mes réflexions et ont accepté de me rencontrer et de répondre à mes questions durant mes études.*

*Je remercie mes très chers parents, qui ont toujours été là pour moi.*

*Enfin, je remercie mes amis qui ont toujours été là pour moi. Leur soutien inconditionnel et leurs encouragements ont été d'une grande aide. Et spéciale remerciement à mon collègue Mouadh HAOUALA.*

# *Sommaire :*

Résumé .....	1
Introduction générale.....	2
Abréviations .....	4
Chapitre 1 : La bibliothèque OpenCV .....	6
1.1. Introduction :.....	7
1.2. Principes de base des images .....	7
1.2.1. Les Pixels :.....	7
1.2.1.1. Couleur du pixel : .....	7
1.2.1.1.1. Niveaux de gris :.....	7
1.2.1.1.2. Couleur : .....	8
1.3. Opérations de traitement d'images :.....	8
1.3.1. Étapes fondamentales du traitement de l'image :.....	8
1.3.1.1. Acquisition d'images.....	8
1.3.1.2. Amélioration de l'image.....	9
1.3.1.3. Restauration d'images .....	9
1.3.1.4. Traitement des images en couleur .....	9
1.3.1.5. Ondelettes et traitement multi résolution.....	9
1.3.1.6. Compression .....	9
1.3.1.7. Traitement morphologique .....	9
1.3.1.8. Segmentation .....	9
1.3.1.9. Représentation et description.....	9
1.3.1.11. Avantages du traitement d'images .....	10
1.3.2. Applications du traitement d'images :.....	10
1.3.2.1. Recherche d'images médicales .....	10
1.3.2.2. Technologies de détection du trafic .....	10
1.3.2.3. Reconstruction d'images .....	11
1.3.2.4. Détection de visages .....	11
1.4. Comprendre le système visuel humain : .....	12
1.5. La compréhension de l'image par l'humain : .....	13
1.6. Comprendre le contenu d'une image par une machine : .....	14
1.7. L'Apprentissage automatique : .....	15

1.8. OpenCV :	16
1.9. Construction de l'interface graphique (GUI) :	16
1.10. Analyse vidéo :	17
1.11. Reconstruction 3D :	17
1.12. Extraction de caractéristiques :	18
1.13. Détection d'objets :	19
1.14. La Reconnaissance des formes :	20
1.15. La différence entre reconnaissance et détection des formes :	21
1.16. Correspondance des modèles :	21
1.17. L'apprentissage profond :	23
1.17.1. Introduction aux réseaux neuronaux artificiels :	24
1.18. Conclusion :	26
Chapitre 2 : Les ordinateurs monocartes (SBC) :	27
2.1. Introduction :	28
2.2. Ordinateurs monocartes :	28
2.3. Les différentes familles SBC :	29
2.3.1. La famille Beagleboard :	29
2.3.2. ASUS Tinkerboard :	30
2.3.3. NVIDIA Jetson:	30
2.4. Raspberry Pi :	31
2.4.1. Les modèles de Raspberry Pi :	31
2.4.1.1. Raspberry Pi modèle 4B :	32
2.4.1.1.1. Introduction :	32
2.4.1.1.2. Caractéristiques:	32
2.4.1.1.2.1. Hardware:	32
2.4.1.1.2.2. Interfaces:	33
2.4.1.1.2.3. Software :	33
2.4.1.1.2.4. Spécifications mécaniques :	35
2.4.1.1.2.5. Spécifications électriques :	35
2.4.1.1.2.6. Alimentation électrique :	37
2.4.1.1.3. Périphériques :	37
2.4.1.1.3.1. GPIO Interface:	37
2.4.1.1.3.2. Interface parallèle d'affichage (DPI) :	38
2.4.1.1.3.3. Interface SD/SDIO :	38
2.4.1.1.3.4. Interfaces caméra et écran :	38

2.4.1.1.3.5.USB:.....	39
2.4.1.1.3.6. HDMI:.....	39
2.4.1.1.3.7. Audio et composite (sortie TV): .....	39
2.4.1.1.3.8. Plage de températures et thermales :.....	39
2.4.1.2. Raspberry pi Zero w: .....	40
2.4.1.2.1. Description: .....	40
2.4.1.2.2. L’histoire :.....	40
2.4.1.2.3. Caractéristiques :.....	41
2.4.1.2.4. Assignation des broches GPIO :.....	42
2.4.1.3. Raspbian Os :.....	42
2.5. Le meilleur ordinateur monocarte :.....	43
2.6. Conclusion : .....	43
Chapitre 3 : Application et Résultats. ....	44
3.1. Introduction :.....	45
3.2. Liste de matériel :.....	45
3.3. Les paquets nécessaires :.....	46
3.4. L’organigramme de surveillance des yeux pour détecter la fatigue :.....	47
3.4.1. Algorithme :.....	48
3.4.2. Méthodologie :.....	48
3.5. Résultats de la simulation et discussion :.....	49
3.6. Résultats d’expériences et d’analyses :.....	50
3.6.1. Détection des visages et des yeux :.....	50
3.6.2. Détection de la fatigue :.....	52
3.7. Conclusion : .....	53
Conclusion Générale : .....	55
References: .....	56

## *Liste des figures :*

Figure 1.1 : A gauche - image de circulation normale   A droite - une image VIPS avec des zones de détection. ....	11
Figure 1.2 : Reconstruction d'images endommagées à l'aide du traitement d'images. ....	11
Figure 1.3 : œil absorbe des rayons.....	12
Figure 1.4 : un humain et les angles sous lesquels nous voyons différentes choses.....	13
Figure 1.5 : L'application de GUI sur un bateau. ....	17
Figure 1.6 : Prendre une figure de plusieurs positions.....	18
Figure 1.7: Extraction des caractéristiques. ....	19
Figure 1.8: Détection des chaises.....	20
Figure 1.9 : Détection d'objet en temps réel et étiquetage. ....	20
Figure 1.10 : Détection d'objets à l'aide de la correspondance. ....	22
Figure 1.11 : points caractéristiques de l'image. ....	23
Figure 1.12: Un diagramme de Venn décrivant l'apprentissage profond comme un sous domaine de l'apprentissage automatique qui est à son tour un sous-domaine de l'intelligence artificielle. ...	23
Figure 1.13 : L'œil humain est un dispositif de détection qui fournit des informations au cerveau qui stocke les modèles.....	24
Figure 1.14 : Traitement de l'information dans les neurones humains. ....	25
Figure 1.15 : Dispositif de détection artificielle (une caméra) fournissant des images aux ordinateurs.....	26
Figure 1.16: Un neurone artificiel.....	26
Figure 2.1 : La carte BeagleBone AI .....	30
Figure 2.2: tinker board.....	30
Figure 2.3 : NVIDIA Jetson nano kit.....	31
Figure 2.4 : Raspberry Pi 4B vue de dessus.....	34
Figure 2.5: La vue de dessus du Raspberry Pi 4B. ....	34
Figure 2.6 : Raspberry Pi 4B sous un angle.....	35
Figure 2.7 : Dimensions mécaniques. ....	35
Figure 2.8 : Caractéristiques des entrées-sorties numériques. ....	37
Figure 2.9: USB Type-C pin. ....	37
Figure 2.10 : Brochage du connecteur GPIO. ....	38
Figure 2.11: Câble HDMI. ....	39
Figure 2.12 : Ports HDMI, mini-HDMI, et micro-HDMI.....	39
Figure 2.13: Raspberry pi zero w.....	41
Figure 2.14: Assignation des broches GPIO de Raspberry pi zero w.....	42
Figure 2.15: Raspberry pi os.....	43

Figure 3.1 : L'emplacement de la caméra dans la voiture. ....	46
Figure 3.2 : Algorithme de surveillance des yeux pour détecter la fatigue. ....	47
Figure 3.3 : Image à échelle de gris pour la détection des caractéristiques des bords et des lignes. .....	48
Figure 3.4 : Six marqueurs faciaux associés à l'œil. ....	49
Figure 3.5 : Visualisation des repères des yeux lorsqu'ils sont fermés et ouverts. ....	50
Figure 3.6 : Visualisation des 68 coordonnées des points de repère du visage à partir du jeu de données iBUG 300-W. ....	51
Figure 3.7 : Détection du visage et des yeux. ....	52
Figure 3.8 : les yeux dans l'état active. ....	53
Figure 3.9 : les yeux à l'état fatigue. ....	53

## *Liste des tableaux :*

Tableau 2.1 : Valeurs limites absolues. ....	36
Tableau 2.2 : Caractéristiques du courant continu.....	36
Tableau 2.3 : Caractéristiques des broches d'E/S numériques.....	36
Tableau 3.1 : le matériel utilisé dans l'application. ....	45

## *Résumé*

Ces dernières années, l'un des facteurs les plus importants dans les accidents de la route est la somnolence des conducteurs et la distraction au volant. Dans cette mémoire, nous décrivons un système qui surveille la détection de la fatigue ou de la somnolence.

Les solutions proposées suivent le regard du conducteur, et si le système identifie les yeux fermés, il déclenche un signal d'alarme destiné à prévenir la perte de contrôle de la voiture et les accidents de la route.

Le suivi du regard est le processus qui mesure la position et le mouvement des yeux.

La méthode proposée est structurée en trois phases.

Dans la première phase, les images des yeux sont capturées à intervalles de temps constants et converties en images en niveaux de gris. Dans la deuxième phase, ces images sont soumises à l'algorithme de HOG pour identifier les yeux du conducteur.

Dans la troisième phase, sur la base de la phase précédente, le système peut maintenant prendre des mesures pour continuer la surveillance ou déclencher une alarme pour avertir le conducteur si la somnolence a été détectée.

**Mots clés :** suivi des yeux, regard du conducteur, attention du conducteur, fatigue du conducteur.

## *Introduction générale*

Les véhicules modernes font non seulement partie intégrante de la vie des gens, mais sont également l'un des types de véhicules les plus dangereux. Selon des sources officielles, chaque année dans le monde, à la suite d'un accident, environ 1,2 million de personnes meurent, soit 3287 personnes par jour.

L'une des causes les plus fréquentes des décès des accidents de la route est la somnolence du conducteur au volant. Et c'est la cause d'un accident de la circulation sur cinq. Par conséquent, la condition la plus dangereuse pour une personne est le surmenage, qui entraîne une perte de concentration et une incapacité à conduire un véhicule.

À l'heure actuelle, les essais de fatigue comportent trois directions principales. Premièrement, la détection de la fatigue basée sur la méthode de détection de l'état du véhicule, principalement par l'angle de braquage, la vitesse de conduite du véhicule pour détecter si le conducteur est fatigué. Cette méthode est sujette à des interférences externes, la précision de la détection a un impact plus important.

Deuxièmement, la détection de la fatigue du conducteur, principalement en détectant le rythme cardiaque du conducteur, son pouls et d'autres signaux physiologiques pour déterminer si le conducteur est dans un état de fatigue. Nécessite que le conducteur porte un grand nombre d'équipements de test, ce qui est très encombrant, et le conducteur a une grande interférence.

Troisièmement, la méthode de détection de la fatigue basée sur la vision par ordinateur. Cette méthode est non intrusive, les caractéristiques du visage peuvent être calculées en analysant les changements d'expression du visage, tels que la durée de fermeture des yeux.

Le but de ce projet est de développer un prototype de détection de la fatigue du conducteur en utilisant la méthode de détection basée sur la vision par ordinateur, par la détection du degré d'ouverture des yeux. Puis le système informe le conducteur par une alarme et un avertissement qui s'affiche dans un écran situé en face du conducteur.

Ce projet cible de façon plus générale les conducteurs des camions et de bus qui se déplacent beaucoup et leurs accidents sont généralement trop meurtrières.

Dans cette mémoire nous avons trois chapitres :

-Chapitre 1 : OpenCv : dans ce chapitre on va discuter sur la bibliothèque (open source computer vision) et le principe d'image et son traitement.

-Chapitre 2 : Ordinateurs monocartes : dans ce chapitre on va voir quelque famille des ordinateurs monocartes puis on passera au Raspberry pi par prendre deux versions de Raspberry pi.

-Chapitre 3 : L'application : maintenant en passant à notre projet, on voit la liste des matériel et les étapes de traitement des données par un organigramme, Après on aura quelques résultats.

A la fin nous finirons avec avec une conclusion générale avec un perspective sur le projet.

## *Abréviations*

- OpenCv** : open-source computer vision.
- RVB (RGB)** : Rouge Vert Bleu
- 2D** : Deux dimensions
- VIPS**: Vessel identification and positioning system
- Highgui**: High level graphical user interface
- Videostab** : Video stabilisation
- Features2d** : Features detection on 2 dimensions
- Bioinspired** : Inspiré de la nature
- IA** : Intelligence artificielle
- ANN** : Artificiel neural networks
- IRM**: Magnetic resonance imaging
- SBC**: Single board computer
- PCB**: Printed circuit board
- USB**: universal serial bus
- MMD-1**: Mini micro designer 1
- RAM**: Random access memory
- CPU**: Central processing unit
- GPU**: Graphics processing unit
- GPIO**: Genaral purpose input/output
- IoT**: Internet of things
- SPI**: Serial peripheral interface
- I2C**: Inter integrated circuit
- SMBus**: System management bus
- PWM**: Pulse width modulation
- LPDDR4**: Low power double data rate
- SDRAM**: Synchronous dynamic random-access memory
- HDMI**: High-definition multimedia interface
- BLE**: Bluetooth low energy
- PoE**: Power over ethernet
- MIPI CSI**: Mobile industry processor interface Camera serial interface
- MIPI DSI**: Mobile industry processor interface Display serial interface
- UART**: Universal asynchronous receiver transmitter
- SDIO**: Secure digital input output
- DPI**: Dots per inch
- PCM**: Pulse code modulation
- GPCLK**: General purpose clock

- APMv8:** Advanced RISC machines version 8
- API:** Application programming interface
- TBD:** To be down
- BCM2711:** Body control module
- EEPROM:** Electrically erasable programmable read-only memory
- CD:** Compact disc
- TRS:** Tip, Ring and Sleeve
- MagPi:** Mid-Atlantic GigaPop in Philadelphia for Intranet2
- CMOS:** Complementary metal-oxide semiconductor
- GB:** Giga byte
- MB/S:** Megabyte per seconde
- HOG:** Histogram of oriented gradients
- EAR:** Eye aspect ratio
- TTC:** Total Target Compensation

# Chapitre 1 : La bibliothèque OpenCV



# Chapitre 1 : La bibliothèque OpenCv

## 1.1. Introduction :

Les applications de vision artificielle sont intéressantes et utiles, mais les algorithmes sous-jacents sont gourmands en ressources informatiques.

Avec l'avènement du cloud computing, nous disposons de plus de puissance de traitement pour travailler. La bibliothèque OpenCV nous permet d'exécuter efficacement les algorithmes de vision par ordinateur en temps réel. Elle existe depuis de nombreuses années et elle est devenue la bibliothèque standard dans ce domaine. L'un des principaux avantages d'OpenCV est qu'elle est hautement optimisée et disponible sur presque toutes les plateformes.

## 1.2. Principes de base des images

Une image numérique est une représentation électronique d'un objet/scène ou d'un document numérisé.

La numérisation d'une image consiste à la convertir en une série de chiffres et à stocker ces chiffres dans un système de stockage informatique.

### 1.2.1. Les Pixels :

Imaginez une série de points disposés en rangées et en colonnes, et ces points ont différentes couleurs. C'est à peu près comme cela que se forme une image.

Les points qui forment une image sont appelés pixels. Ces pixels sont représentés par des nombres, et les valeurs des nombres déterminent la couleur d'un pixel. Pensez à une image comme à une grille de cellules carrées dont chaque cellule est constituée d'un pixel d'une couleur particulière. Par exemple, une image de 300×400 pixels signifie que l'image est organisée en une grille de 300 lignes et 400 colonnes. Cela signifie que notre image a  $300 \times 400 = 120\,000$  pixels.

#### 1.2.1.1. Couleur du pixel :

Un pixel est représenté de deux façons : en niveaux de gris et en couleur.

##### 1.2.1.1.1. Niveaux de gris :

Dans une image en niveaux de gris, chaque pixel prend une valeur comprise entre 0 et 255. La valeur 0 représente noir, et 255 représente le blanc. Les valeurs intermédiaires sont des nuances de gris. Le site

Les valeurs proches de 0 sont des nuances de gris plus sombres, et les valeurs proches de 255 sont des nuances de gris plus claires.

## Chapitre 1 : La bibliothèque OpenCv

### 1.2.1.1.2. Couleur :

Le modèle de couleur RVB (RGB) (rouge, bleu et vert) est l'une des représentations les plus populaires d'un pixel.). Il existe d'autres modèles de couleurs.

Dans le modèle RVB, chaque pixel est représenté comme un tuple de trois valeurs, généralement représentées comme suit : (valeur pour la composante rouge, valeur pour la composante verte, valeur pour la composante bleue). Chacune des trois couleurs est représentée par des entiers allant de 0 à 255.

Voici quelques exemples :

(0,0,0) est une couleur noire.

(255,0,0) est une couleur rouge pure.

(0,255,0) est une couleur verte pure.

## 1.3. Opérations de traitement d'images :

Le traitement d'images consiste à transformer une image sous une forme numérique et à effectuer certaines opérations pour en tirer des informations utiles. Le système de traitement d'images traite généralement toutes les images comme des signaux 2D lorsqu'il applique certaines méthodes prédéterminées de traitement du signal.

Il existe cinq grands types de traitement d'images :

-Visualisation - Trouver des objets qui ne sont pas visibles dans l'image.

-Reconnaissance - Distinguer ou détecter des objets dans l'image

-Accentuation et restauration - Créer une image améliorée à partir de l'image originale.

-Reconnaissance des formes - Mesurer les différents motifs autour des objets dans l'image

-Récupération - Parcourir et rechercher des images dans une vaste base de données d'images numériques similaires à l'image originale.

### 1.3.1. Étapes fondamentales du traitement de l'image :

#### 1.3.1.1. Acquisition d'images

L'acquisition d'images est la première étape du traitement d'images. Cette étape est également connue sous le nom de prétraitement dans le traitement d'images. Elle consiste à récupérer l'image à partir d'une source, généralement une source matérielle.

## **Chapitre 1 : La bibliothèque OpenCv**

### **1.3.1.2. Amélioration de l'image**

L'amélioration d'une image consiste à faire ressortir et à mettre en évidence certaines caractéristiques intéressantes d'une image qui a été masquée. Il peut s'agir de modifier la luminosité, le contraste, etc.

### **1.3.1.3. Restauration d'images**

La restauration d'image est le processus d'amélioration de l'apparence d'une image. Cependant, contrairement au rehaussement d'image, la restauration d'image est effectuée à l'aide de certains modèles mathématiques ou probabilistes.

### **1.3.1.4. Traitement des images en couleur**

Le traitement des images en couleur comprend un certain nombre de techniques de modélisation des couleurs dans un domaine numérique. Cette étape a pris de l'importance en raison de l'utilisation importante d'images numériques sur Internet.

### **1.3.1.5. Ondelettes et traitement multi résolution**

Les ondelettes sont utilisées pour représenter des images à différents degrés de résolution. Les images sont subdivisées en ondelettes ou en régions plus petites pour la compression des données et pour la représentation pyramidale.

### **1.3.1.6. Compression**

La compression est un processus utilisé pour réduire le stockage nécessaire à l'enregistrement d'une image ou la bande passante requise pour la transmettre. Cela se fait notamment lorsque l'image est destinée à être utilisée sur Internet.

### **1.3.1.7. Traitement morphologique**

Le traitement morphologique est un ensemble d'opérations de traitement permettant de transformer des images en fonction de leurs formes.

### **1.3.1.8. Segmentation**

La segmentation est l'une des étapes les plus difficiles du traitement d'images. Elle consiste à partitionner une image en ses parties constitutives ou objets.

### **1.3.1.9. Représentation et description**

Après qu'une image a été divisée en régions au cours du processus de segmentation, chaque région est représentée et décrite sous une forme adaptée à un traitement informatique ultérieur. La représentation concerne les caractéristiques de l'image et

## **Chapitre 1 : La bibliothèque OpenCv**

les propriétés régionales. La description consiste à extraire des informations quantitatives qui permettent de différencier une classe d'objets d'une autre.

### **1.3.1.10. Reconnaissance**

La reconnaissance attribue une étiquette à un objet sur la base de sa description.

### **1.3.1.11. Avantages du traitement d'images**

La mise en œuvre de techniques de traitement d'images a eu un impact massif sur de nombreuses organisations technologiques. Voici quelques-uns des avantages les plus utiles du traitement d'images, quel que soit le domaine d'activité :

-L'image numérique peut être mise à disposition dans le format souhaité (image améliorée, radiographie, négatif photo, etc.).

Il permet d'améliorer les images pour l'interprétation humaine

-L'information peut être traitée et extraite des images pour l'interprétation par la machine.

-Les pixels de l'image peuvent être manipulés pour obtenir la densité et le contraste souhaités.

-Les images peuvent être stockées et récupérées facilement

-Il permet une transmission électronique facile des images à des fournisseurs tiers.

## **1.3.2. Applications du traitement d'images :**

### **1.3.2.1. Recherche d'images médicales**

Le traitement d'images a été largement utilisé dans la recherche médicale et a permis d'établir des plans de traitement plus efficaces et plus précis. Par exemple, il peut être utilisé pour la détection précoce du cancer du sein à l'aide d'un algorithme sophistiqué de détection des nodules dans les scanners mammaires. Étant donné que l'usage médical exige des processeurs d'images hautement qualifiés, ces applications nécessitent une mise en œuvre et une évaluation importante avant d'être acceptées.

### **1.3.2.2. Technologies de détection du trafic**

Dans le cas des capteurs de trafic, nous utilisons un système de traitement d'images vidéo ou VIPS. Celui-ci se compose) d'un système de capture d'images, b) d'un système de télécommunication et c) d'un système de traitement d'images. Lors de la capture vidéo, un VIPS comporte plusieurs zones de détection qui émettent un signal "on" lorsqu'un véhicule entre dans la zone, puis un signal "off" lorsque le véhicule sort de la zone de détection. Ces zones de détection peuvent être configurées pour plusieurs voies et peuvent être utilisées pour détecter le trafic dans une station particulière.

## Chapitre 1 : La bibliothèque OpenCv



Figure 1.1 : A gauche - image de circulation normale / A droite - une image VIPS avec des zones de détection.

En outre, il peut enregistrer automatiquement la plaque d'immatriculation du véhicule, distinguer le type de véhicule, surveiller la vitesse du conducteur sur l'autoroute et bien d'autres choses encore.

### 1.3.2.3. Reconstruction d'images

Le traitement d'images peut être utilisé pour récupérer et compléter les parties manquantes ou corrompues d'une image. Cela implique l'utilisation de systèmes de traitement d'images qui ont été entraînés de manière intensive avec des ensembles de données photographiques existants pour créer des versions plus récentes de photos anciennes et endommagées.

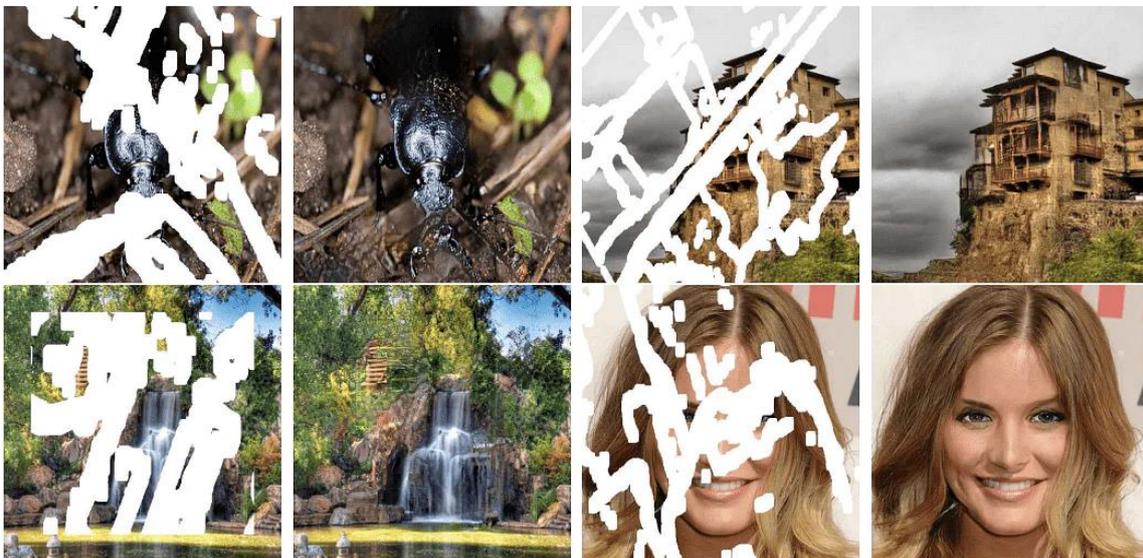


Figure 1.2 : Reconstruction d'images endommagées à l'aide du traitement d'images.

### 1.3.2.4. Détection de visages

L'une des applications les plus courantes du traitement d'images que nous utilisons aujourd'hui est la détection des visages. Elle suit des algorithmes d'apprentissage profond où la machine est d'abord entraînée avec les caractéristiques spécifiques des visages humains, comme la forme du visage, la distance entre les yeux, etc. Après avoir appris à la machine ces caractéristiques des visages humains, elle commencera

## Chapitre 1 : La bibliothèque OpenCv

à accepter tous les objets dans une image qui ressemblent à un visage humain. La détection des visages est un outil essentiel utilisé en matière de sécurité, de biométrie et même de filtres disponibles sur la plupart des applications de médias sociaux de nos jours.

### 1.4. Comprendre le système visuel humain :

Avant de sauter dans les fonctionnalités d'OpenCV, nous devons comprendre pourquoi ces fonctions ont été construites en premier lieu. Il est important de comprendre comment le système visuel humain afin de pouvoir développer les bons algorithmes. Le but des algorithmes de vision par ordinateur est de comprendre le contenu des images et des vidéos.

Les humains semblent y parvenir sans effort ! Alors, comment faire pour que les machines le fassent avec la même précision ?

Considérons la figure suivante :

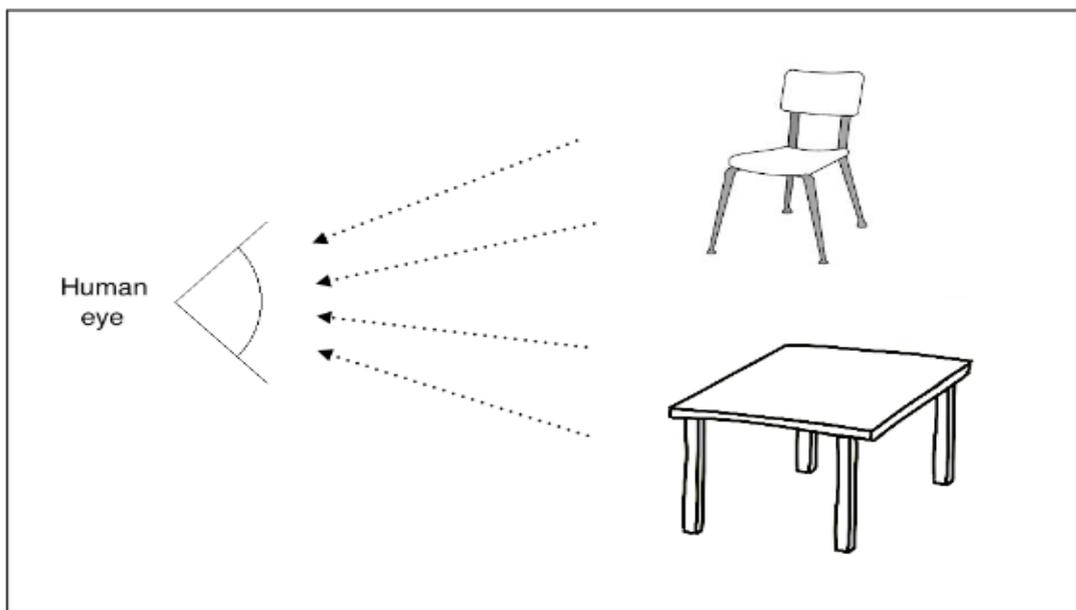


Figure 1.3 : œil absorbe des rayons.

L'œil humain capte toutes les informations qui lui parviennent telles que la couleur, les formes, la luminosité, et ainsi de suite. Dans l'image précédente, l'œil humain capte toutes les informations sur les deux objets principaux et les stocke d'une certaine manière. Une fois que nous comprenons comment notre système fonctionne, nous pouvons en tirer parti pour obtenir ce que nous voulons.

Par exemple, voici quelques éléments que nous devons connaître :

-Notre système visuel est plus sensible aux basses fréquences qu'aux hautes fréquences que les contenus à haute fréquence. Le contenu à basse fréquence fait

## Chapitre 1 : La bibliothèque OpenCv

référence à des régions planes où les valeurs des pixels ne changent pas rapidement et le contenu à haute fréquence fait référence aux régions avec des coins et des bords, où les valeurs de pixel fluctuent beaucoup. Nous aurons remarqué que nous pouvons facilement voir s'il y a des taches sur une surface plane, mais il est difficile de repérer ce genre de choses sur une surface très texturée.

- L'œil humain est plus sensible aux changements de luminosité qu'aux changements de couleur.

- Notre système visuel est sensible au mouvement. Nous pouvons rapidement reconnaître si quelque chose bouge dans notre champ de vision, même si nous ne le regardons pas directement.

- Nous avons tendance à noter mentalement les points saillants de notre champ de vision. Prenons une table blanche avec quatre pieds noirs et un point rouge à l'un des coins de la surface de la table. Lorsque vous regardez cette table, vous remarquerez immédiatement que la surface et les pieds ont des couleurs opposées et qu'il y a un point rouge sur l'un des coins. Notre cerveau est vraiment intelligent de cette façon ! Nous faisons cela automatiquement afin de pouvoir le reconnaître immédiatement si nous le rencontrons à nouveau.

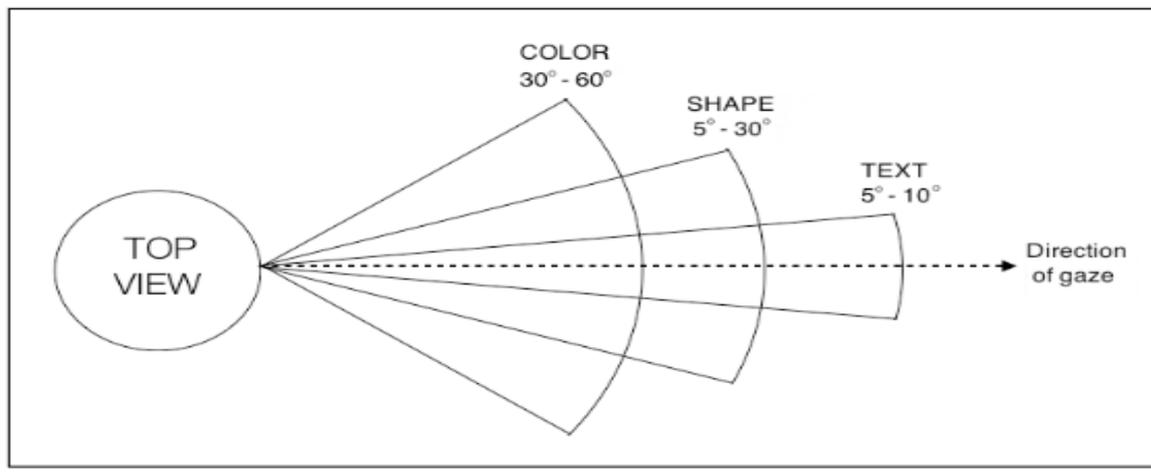


Figure 1.4 : un humain et les angles sous lesquels nous voyons différentes choses.

### 1.5. La compréhension de l'image par l'humain :

Si nous regardons autour de nous, nous verrons beaucoup d'objets. Nous pouvons rencontrer beaucoup d'objets différents chaque jour, et nous les reconnaissons presque instantanément sans aucun effort. Lorsque nous voyons une chaise, nous n'attendons pas quelques minutes avant de nous rendre compte qu'il s'agit bien d'une chaise !

## Chapitre 1 : La bibliothèque OpenCv

D'un autre côté, les ordinateurs ont beaucoup de mal à accomplir cette tâche. Les chercheurs ont travaillé pendant de nombreuses années pour comprendre pourquoi les ordinateurs ne sont pas aussi bons que nous dans ce domaine.

Pour obtenir une réponse à cette question, nous devons comprendre comment les humains s'y prennent. Le site traitement des données visuelles se fait dans le flux visuel ventral. Ce flux visuel ventral fait référence à la voie de notre système visuel associée à la reconnaissance des objets. Il s'agit essentiellement d'une hiérarchie de zones dans notre cerveau qui nous aide à reconnaître les objets. Les humains peuvent reconnaître différents objets sans effort, et nous pouvons regrouper des objets similaires. Nous pouvons le faire parce que nous avons développé une sorte d'invariance envers les objets de la même classe. Lorsque nous regardons un objet, notre cerveau extrait les points saillants de telle sorte que des facteurs tels que l'orientation, la taille, la perspective, et l'illumination n'ont pas d'importance.

Une chaise qui fait le double de la taille normale et qui est tournée de 45 degrés reste une chaise. Nous pouvons facilement la reconnaître grâce à la façon dont nous la traitons. Les machines ne peuvent pas le faire aussi facilement. Les humains ont tendance à se souvenir d'un objet en se basant sur sa forme et ses caractéristiques. Indépendamment de la façon dont l'objet est placé, nous pouvons toujours le reconnaître. Dans notre système visuel, nous construisons ces invariances hiérarchiques par rapport à la position, l'échelle et le point de vue qui nous aident à être très robustes.

Si nous regardons plus profondément dans notre système, nous verrons que les humains ont des cellules dans leur cortex visuel qui peuvent répondre à des formes telles que les courbes et les lignes.

Au fur et à mesure que nous nous déplaçons le long de notre flux ventral, nous verrons des cellules plus complexes qui sont entraînées à répondre à des objets plus complexes, comme des arbres, des portes, etc. Les neurones le long de notre ventral tendent à montrer une augmentation de la taille du champ réceptif. Ce phénomène est associé au fait que la complexité de leurs stimuli préférés augmente également.

### **1.6. Comprendre le contenu d'une image par une machine :**

Le problème est que nous ne comprenons toujours pas complètement comment notre cerveau reconnaît et organise ces données visuelles. Nous nous contentons d'extraire certaines caractéristiques des images et demandons à l'ordinateur

## Chapitre 1 : La bibliothèque OpenCv

d'apprendre à partir de celles-ci en utilisant des algorithmes d'apprentissage automatique.

Nous avons toujours ces variations telles que la forme, la taille, la perspective, l'angle, l'illumination, l'occlusion, et ainsi de suite. Par exemple, la même chaise semble très différente pour une machine lorsque nous la regardons de côté. Les humains peuvent facilement reconnaître qu'il s'agit d'une chaise quelle que soit la façon dont elle nous est présentée. Alors, comment expliquer cela à nos machines ?

Une façon de le faire serait de stocker toutes les différentes variations d'un objet, y compris les tailles, les angles, les perspectives, et ainsi de suite. Mais ce processus est long et fastidieux ! De plus, il n'est pas possible de collecter des données qui puissent englober chaque variation. Les machines consommeront une énorme quantité de mémoire et beaucoup de temps pour construire un modèle capable de reconnaître ces objets. Même avec tout cela, si un objet est partiellement occulté, les ordinateurs ne seront toujours pas en mesure de le reconnaître. Cela est dû parce qu'ils pensent qu'il s'agit d'un nouvel objet. Donc, lorsque nous construisons une bibliothèque de vision par ordinateur nous devons construire les blocs fonctionnels sous-jacents qui peuvent être combinés de différentes manières pour formuler des algorithmes complexes. OpenCV fournit beaucoup de ces fonctions et elles sont hautement optimisées. Donc, une fois que nous comprenons ce qu'OpenCV fournit, nous pouvons l'utiliser efficacement pour construire des applications intéressantes.

### 1.7. L'Apprentissage automatique :

La vision par ordinateur utilise divers algorithmes d'apprentissage automatique pour réaliser différentes choses.

OpenCV fournit un module appelé ml qui contient de nombreux algorithmes d'apprentissage automatique.

Certains de ces algorithmes incluent le classificateur de Bayes, K-Nearest Neighbors, Support Vector Machines, Decision Trees, Neural Networks, réseaux neuronaux, etc. Il possède également un module appelé flann qui contient des algorithmes pour la recherche rapide des plus proches voisins dans les grands ensembles de données. Les algorithmes d'apprentissage automatique sont largement utilisés pour construire des systèmes de reconnaissance d'objets, de classification d'images, la détection des visages, les recherches visuelles, etc.

## Chapitre 1 : La bibliothèque OpenCv

### 1.8. OpenCV :

OpenCV (Open Source Computer Vision Library) est une bibliothèque logicielle open source de vision par ordinateur et d'apprentissage automatique. OpenCV a été construit pour fournir une infrastructure commune pour les applications de vision par ordinateur et pour accélérer l'utilisation de la perception artificielle dans les produits commerciaux.

### 1.9. Construction de l'interface graphique (GUI) :

OpenCV fournit un module appelé **highgui** qui gère toutes les opérations de l'interface utilisateur. Disons que vous travaillez sur un problème et que vous souhaitez vérifier à quoi ressemble l'image avant de passer à l'étape suivante. Ce module possède des fonctions qui peuvent être utilisées pour créer des fenêtres afin d'afficher des images et/ou des vidéos.

Il existe également une fonction d'attente qui attendra que nous appuyions sur une touche du clavier pour passer à l'étape suivante.

Avant de passer à l'étape suivante. Il existe également une fonction qui peut détecter les événements de la souris.

Cette fonction est très utile pour développer des applications interactives. En utilisant cette fonction, nous pouvons dessiner des rectangles sur ces fenêtres de saisie et ensuite procéder en fonction de la région sélectionnée.

## Chapitre 1 : La bibliothèque OpenCv

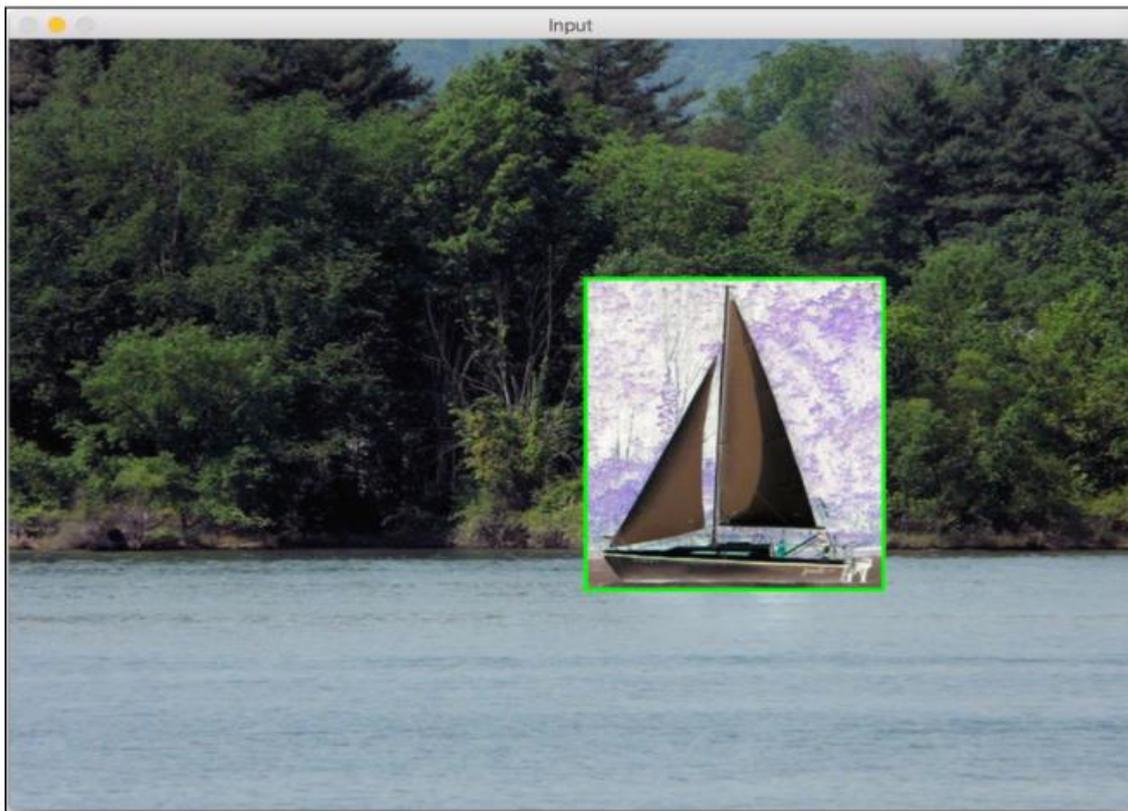


Figure 1.5 : L'application de GUI sur un bateau.

Comme on peut le voir, nous avons dessiné un rectangle vert sur l'image et appliqué un effet de film négatif à cette région. Une fois que nous avons les coordonnées de ce rectangle, nous pouvons opérer uniquement sur cette région.

### 1.10. Analyse vidéo :

L'analyse vidéo comprend des tâches telles que l'analyse du mouvement entre des images successives, le suivi de différents objets dans une vidéo, la création de modèles pour la vidéosurveillance, et ainsi de suite.

OpenCV fournit un module appelé **vidéo** qui peut gérer tout cela. Il existe un module appelé **videostab** qui traite de la stabilisation de la vidéo. La stabilisation de la vidéo est une partie importante des caméras vidéo. Quand vous capturez des vidéos en tenant la caméra dans nos mains, il est difficile de garder nos mains parfaitement stables. Si nous regardons la vidéo telle qu'elle est, elle aura l'air mauvaise et instable. Tous les appareils modernes utilisent des techniques de stabilisation vidéo pour traiter les vidéos avant de les présenter à l'utilisateur final.

### 1.11. Reconstruction 3D :

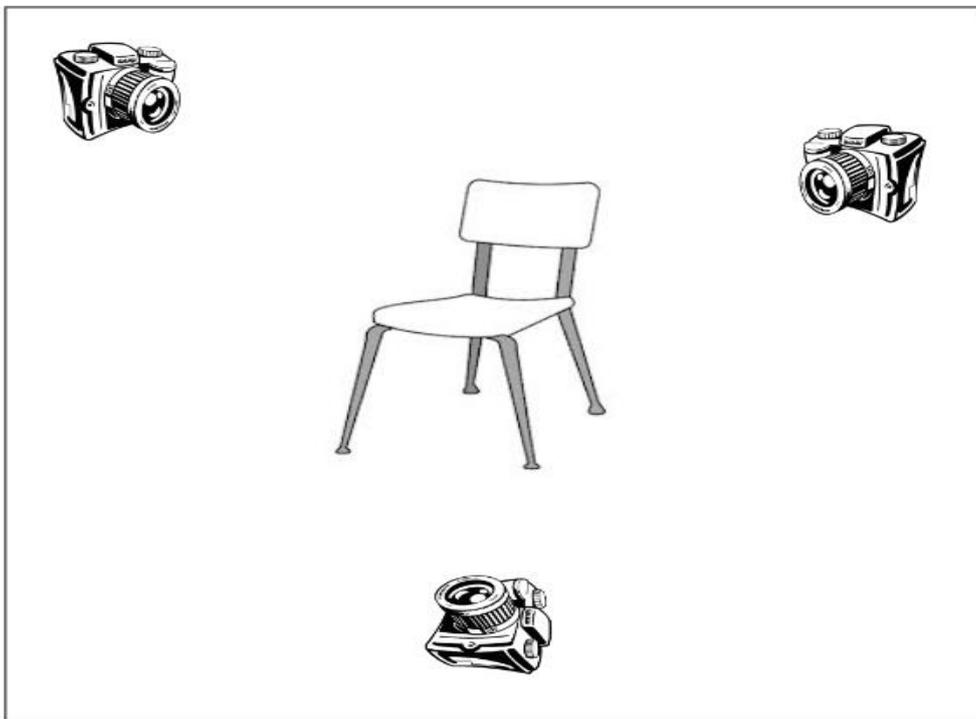
La reconstruction 3D est un sujet important en vision par ordinateur. Étant donné un ensemble d'images 2D nous pouvons reconstruire la scène 3D en utilisant les

## Chapitre 1 : La bibliothèque OpenCv

algorithmes appropriés. OpenCV fournit des algorithmes qui peuvent trouver la relation entre divers objets dans ces images 2D pour calculer leurs positions 3D.

Nous disposons d'un module appelé `calib3d` qui peut gérer tout cela. Ce module peut également gérer la calibration de la caméra, qui est essentielle pour estimer les paramètres de la caméra. Ces paramètres sont en fait les paramètres internes d'une caméra donnée qui les utilise pour transformer la scène capturée en une image.

Nous devons connaître ces paramètres pour concevoir des algorithmes, sinon nous pourrions obtenir des résultats inattendus. Considérons la figure suivante :



*Figure 1.6 : Prendre une figure de plusieurs positions.*

Comme le montre l'image précédente, le même objet est capturé sous plusieurs poses. Notre travail consiste à reconstruire l'objet original en utilisant ces images 2D.

### 1.12. Extraction de caractéristiques :

Le système visuel humain a tendance à extraire les caractéristiques saillantes d'une scène donnée afin de pouvoir les retrouver plus tard.

Pour imiter cela, on a commencé à concevoir divers extracteurs de caractéristiques qui peuvent extraire ces points saillants d'une image donnée. Parmi les algorithmes les plus populaires, citons SIFT (Scale Invariant Feature Transform), SURF (Speed Invariant Feature Transform),

## Chapitre 1 : La bibliothèque OpenCv

FAST (Features from Accelerated Segment Test), et ainsi de suite. Il existe un module appelé **features2d** qui fournit des fonctions pour détecter et extraire toutes ces caractéristiques. Il existe un autre module appelé **xfeatures2d** qui fournit quelques extracteurs de caractéristiques supplémentaires, dont certains sont encore en phase expérimentale.

Nous pouvons jouer avec eux si nous en avons l'occasion.

Il existe également un module appelé **bioinspired** qui fournit des algorithmes pour les modèles de vision par ordinateur d'inspiration biologique.

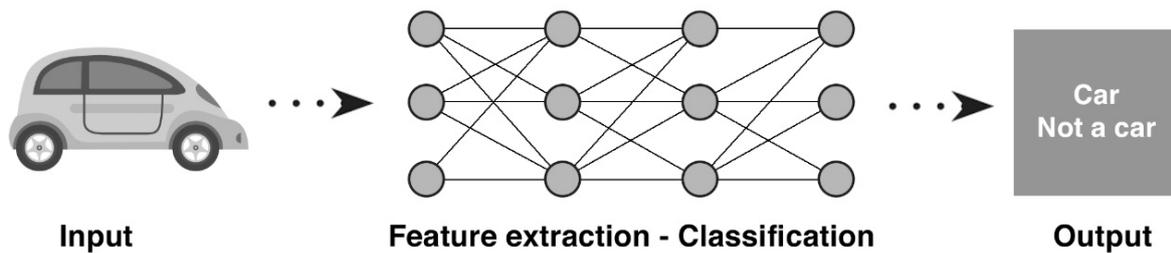


Figure 1.7: Extraction des caractéristiques.

### 1.13. Détection d'objets :

La détection d'objets consiste à détecter l'emplacement d'un objet dans une image donnée. Ce processus ne s'intéresse pas au type d'objet. Si vous concevez un détecteur de chaise, il vous dira simplement l'emplacement de la chaise dans une image donnée. Il ne vous dira pas s'il s'agit d'une chaise rouge avec un dossier haut ou d'une chaise bleue avec un dossier bas. Détecter l'emplacement des objets est une étape très importante dans de nombreux systèmes de vision par ordinateur. Considérons l'image suivante :

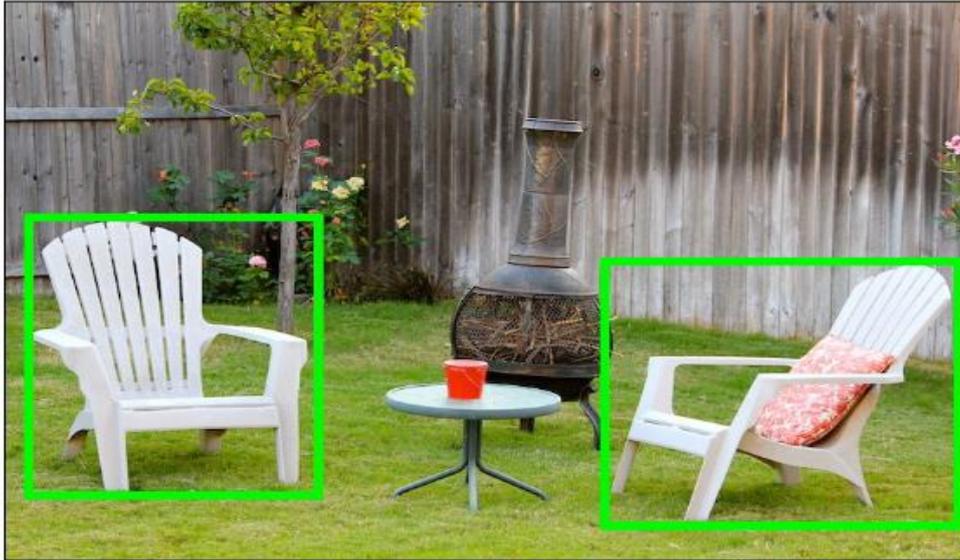


Figure 1.8: Détection des chaises.

### 1.14. La Reconnaissance des formes :

La détection des objets dans une image est une aptitude cruciale d'une application de vision par ordinateur.

La reconnaissance des formes est utilisée dans l'étiquetage des scènes, navigation robotique, voitures auto-conduite, visage et corps, détection des maladies et cancer, objets dans les images satellitaires, reconnaissance d'écriture, et beaucoup plus.

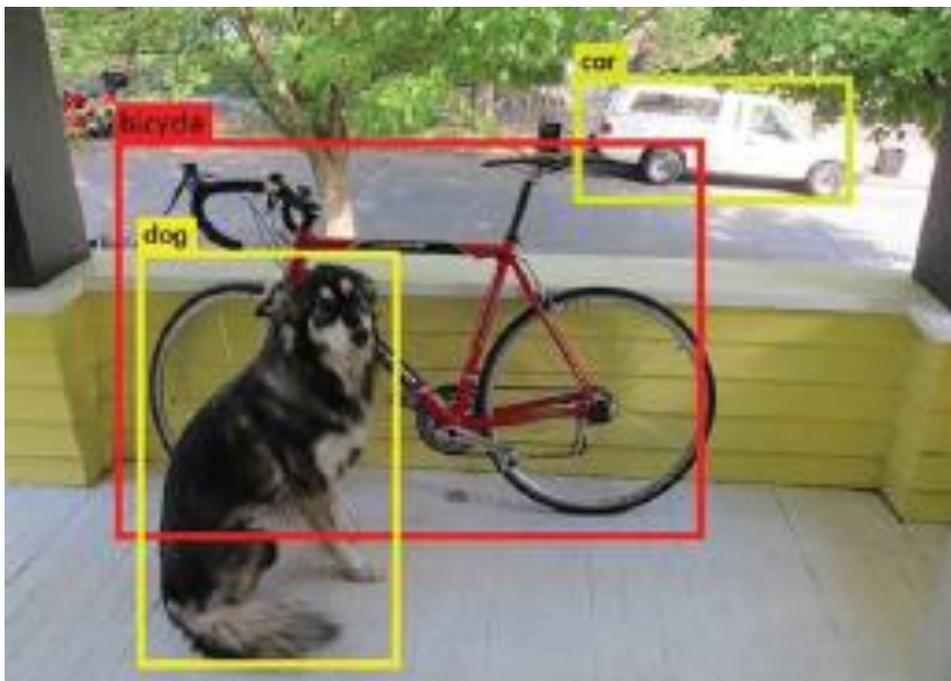


Figure 1.9 : Détection d'objet en temps réel et étiquetage.

## Chapitre 1 : La bibliothèque OpenCv

### 1.15. La différence entre reconnaissance et détection des formes :

Dans la figure 1.9, nous avons détecté s'il y avait un camion, un chien ou un vélo. Nous n'avons pas reconnu de qualités spécifiques telles que le model ou la couleur de la voiture, ou la couleur ou la race du chien par-ce-que l'objective était seulement d'identifier quels objets sont dans l'image. La reconnaissance des formes est la méthode d'identifier un objet ou une forme dans une image. Dans le cas de la reconnaissance des objets, nous détectons d'abord la voiture et sur la voiture recadrée, on applique des reconnaissants à reconnaître les caractéristiques de la voiture. Ceci est similaire avec les visages et les yeux aussi.

Tandis que les humains peuvent identifier une variété d'objets sans effort, pour ordinateurs c'est un problème complexe à résoudre avec précision. Il a échappé chercheurs de vision par ordinateur depuis des décennies maintenant et est devenu la sainte gail de vision par ordinateur.

En fonction de la position et de l'angle de l'objet, la tâche de détection est difficile. Définir une boîte à bornes de chaque objet est important.

### 1.16. Correspondance des modèles :

Dans le cadre de la détection et de la reconnaissance des objets, nous devons procéder à l'analyse des formes et l'analyse des caractéristiques. Pour ce faire, il existe une technique robuste appelée comparaison de modèles. Il s'agit techniquement d'un algorithme de force brute ou un mécanisme simple permettant d'extraire un objet sur la base d'un modèle précédemment acquis.

OpenCv dispose d'une fonction **match Template ()** pour effectuer un modèle.

Cette fonction prend une « fenêtre glissante » de l'image recherchée sue le fait glisser sur l'image recherchée pour déterminer sa présence. Elle fait cela un pixel à la fois. Ensuite, poue chacun de ces emplacements, un coefficient de corrélation est calculé s'il existe une correspondance. Les régions avec une forte corrélation élevée sont les régions qui correspondent.

La figure 1.10 montre une détection d'objet typique utilisant la mise en correspondance. Cette méthode utilise un modèle pour détecter un objet après segmentation. Si l'objet segmenté est similaire au modèle, le processus de détection d'objet est terminé ; dans le cas contraire un autre modèle est choisi pour une vérification de la similarité.

## Chapitre 1 : La bibliothèque OpenCv

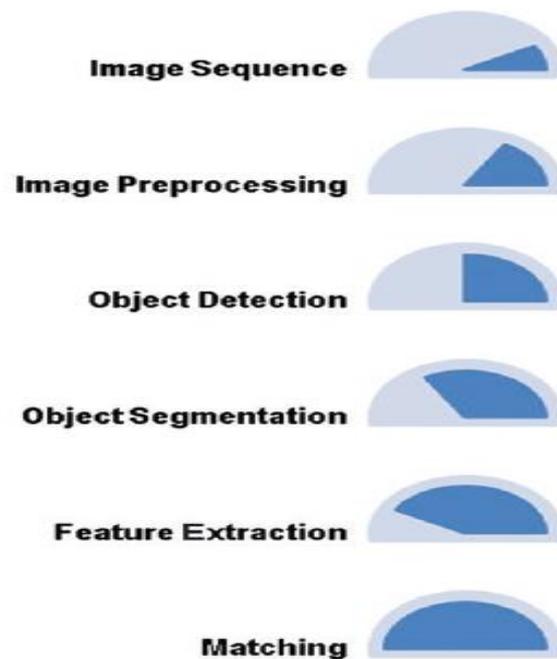


Figure 1.10 : Détection d'objets à l'aide de la correspondance.

Donc la correspondance des objets soit utile dans la détection et la reconnaissance d'objets dans une image, cette méthode présente plusieurs défis méthodologie, par exemple si l'image est pivotée, mise à l'échelle, modifié en termes de couleurs ou de la luminosité, ou transformée, il est difficile de faire correspondre ou de détecter un objet d'entrée dans l'image.

Dans une image, il peut y avoir des points intéressants et des inintéressants. Les points intéressants dans une image sont ceux qui peuvent donner le plus d'informations sur l'objet de l'image, tandis que les points inintéressants ne donnent aucune information sur l'image ou son objet.

## Chapitre 1 : La bibliothèque OpenCv

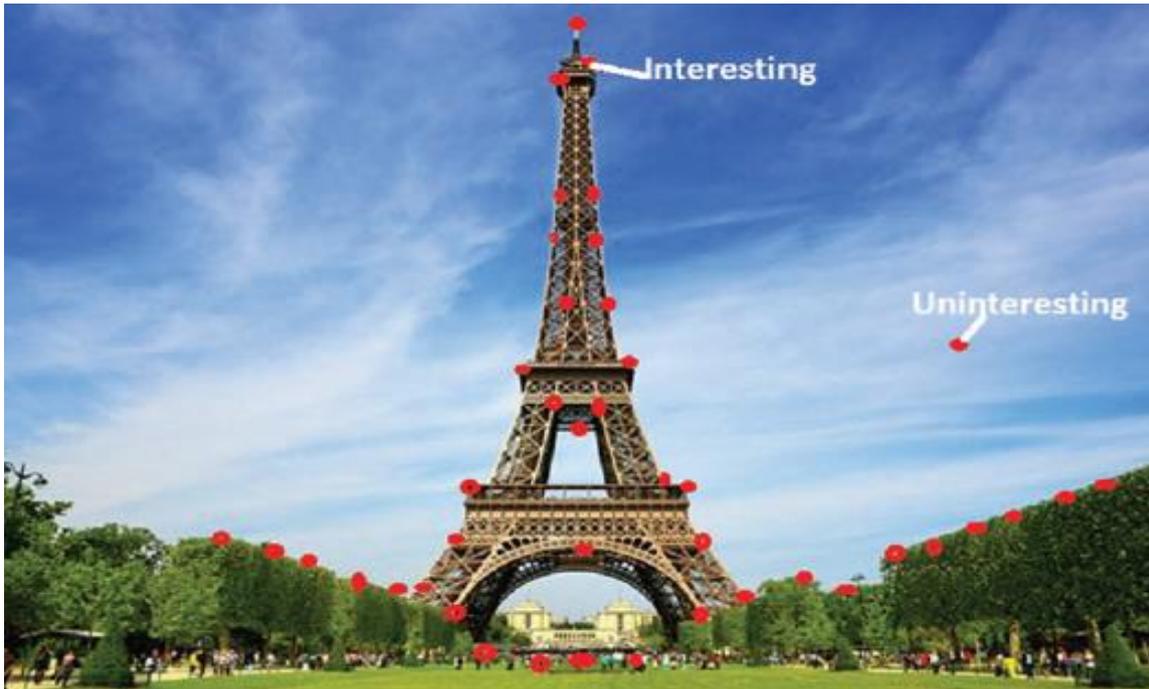


Figure 1.11 : points caractéristiques de l'image.

### 1.17. L'apprentissage profond :

L'apprentissage profond est un sous-domaine de l'apprentissage automatique, qui est à son tour un sous-domaine de l'intelligence artificielle (IA).

Pour une représentation graphique de cette relation, veillons-nous reporter à la figure 1.12.

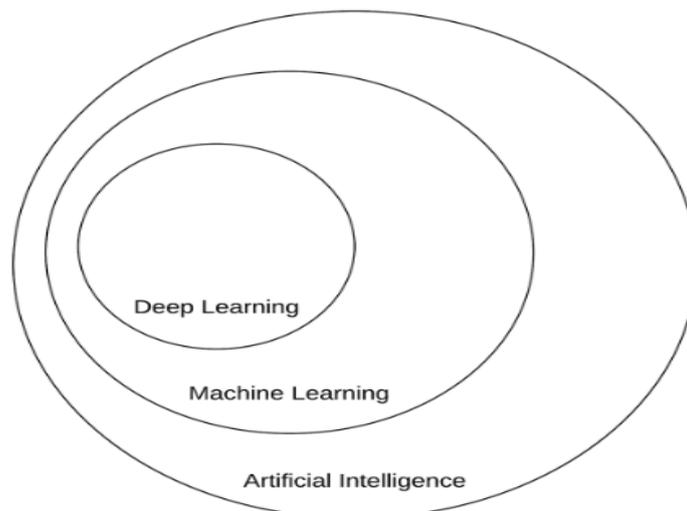


Figure 1.12: Un diagramme de Venn décrivant l'apprentissage profond comme un sous domaine de l'apprentissage automatique qui est à son tour un sous-domaine de l'intelligence artificielle.

## Chapitre 1 : La bibliothèque OpenCv

### 1.17.1. Introduction aux réseaux neuronaux artificiels :

Un réseau neuronal artificiel (ANN) est un système informatique conçu pour fonctionner à la manière du cerveau humain. Comprenons cela à l'aide d'un exemple. Supposons que vous voyez un objet que vous n'avez jamais vu auparavant. Quelqu'un nous dit que c'est une voiture. Puis nous voyons de nombreux autres objets et apprenons à les reconnaître.

Ensuite, nous voyons un autre objet et nous essayons de deviner ce que c'est. Nous pouvons dire quelque chose comme : "Je crois que j'ai déjà vu ça avant." Ou nous pouvons dire : "Je crois que c'est une voiture." Cela signifie que nous ne sommes pas sûr à 100 % certain d'identifier l'objet.

Maintenant, supposons que nous voyons de nombreuses voitures de différentes formes, tailles, orientations et couleurs différentes. Nous sommes parfaitement entraînés à identifier l'objet "voiture".

Très probablement, nous ne dirons pas "Je suppose", mais nous dirons "C'est une voiture". Cela signifie que notre confiance dans l'identification d'une voiture augmente au fur et à mesure que nous nous sommes entraînés à observer un grand nombre de voitures.

Ce qui se passe ici, c'est que lorsque nous voyons une voiture une seule fois ou quelques fois, nous apprenons à la reconnaître si elle est présentée dans un autre contexte. Ou de la même manière ou de manière similaire à celle que nous avons vue auparavant.

Mais lorsque vous voyez un grand nombre d'échantillons de différentes manières, nous apprenons à reconnaître l'objet à 100 % (ou presque).

Examinons les diagrammes de la figure 1.13 pour voir comment les informations sont traitées dans notre cerveau (une version simplifiée du fonctionnement du cerveau humain). Exemple simpliste.

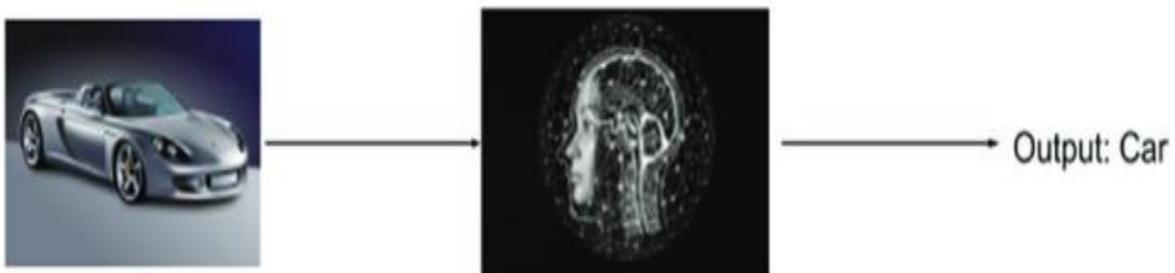


Figure 1.13 : L'œil humain est un dispositif de détection qui fournit des informations au cerveau qui stocke les modèles.

Notre œil fonctionne comme un dispositif de détection. Lorsque nous voyons un objet, nos yeux capturent une image de cet objet qui est transmise au cerveau comme un signal d'entrée. Les neurones de notre cerveau effectuent le calcul des signaux d'entrée et génèrent des sorties.

## Chapitre 1 : La bibliothèque OpenCv

Comme le montre la figure 1.14, les dendrites reçoivent les signaux d'entrée (X). Le neurone combine ces signaux d'entrée et effectue des calculs à l'aide d'une fonction. La sortie est transmise aux terminaisons des axones.

Le corps humain compte des milliards de neurones et des trillions d'interconnexions entre eux.

Ces neurones interconnectés sont appelés un réseau neuronal.

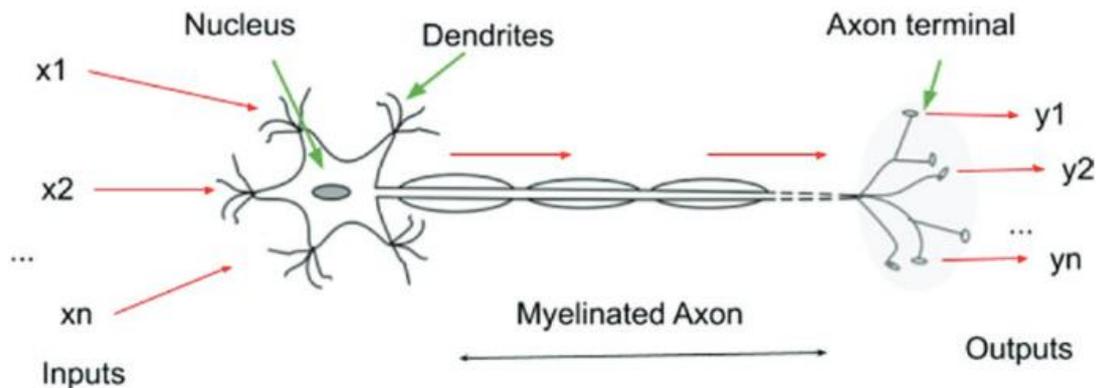


Figure 1.14 : Traitement de l'information dans les neurones humains.

Les informaticiens ont été inspirés par le système de vision humain et ont essayé d'imiter les réseaux neuronaux en créant un système informatique qui apprend et fonctionne de la même manière que notre cerveau. Ce système d'apprentissage s'appelle un réseau neuronal artificiel (ANN).

La figure 1.15 est analogue à la figure 1.13. Une caméra fonctionne comme un dispositif de détection comme nos yeux qui capturent des images d'objets. Les images sont transmises à un système d'interprétation, tel qu'un ordinateur, où elles sont traitées de la même façon qu'un neurone traite les signaux d'entrée.

Parmi les autres dispositifs de détection, on peut citer les appareils à rayons X, les scanners, et IRM, les systèmes d'imagerie par satellite et les scanners de documents. Les dispositifs d'interprétation, tels que les ordinateurs, assurent le traitement des données acquises par la caméra.

La plupart des calculs liés à la vision par ordinateur, tels que l'extraction de caractéristiques et la détermination de motifs, sont effectués dans l'ordinateur.

## Chapitre 1 : La bibliothèque OpenCv



Figure 1.15 : Dispositif de détection artificielle (une caméra) fournissant des images aux ordinateurs.

La figure 1.16 est analogue au neurone humain présenté à la figure 1.14. Les variables  $x_1, x_2, \dots, x_n$  sont les signaux d'entrée (par exemple, les caractéristiques de l'image) avec certains poids  $w_1, w_2, \dots, w_n$  sont associés à chaque signal d'entrée. Ces signaux d'entrée sont traités à l'aide de certaines fonctions mathématiques pour générer des sorties. L'unité de traitement qui combine ces signaux d'entrée est appelée un neurone, du nom du neurone humain. La fonction mathématique qui calcule la sortie du neurone s'appelle une fonction d'activation. Sur le site Figure 1.16, le cercle marqué du symbole de la fonction  $f(x)$  est le neurone. La sortie  $y$  est générée par le neurone.

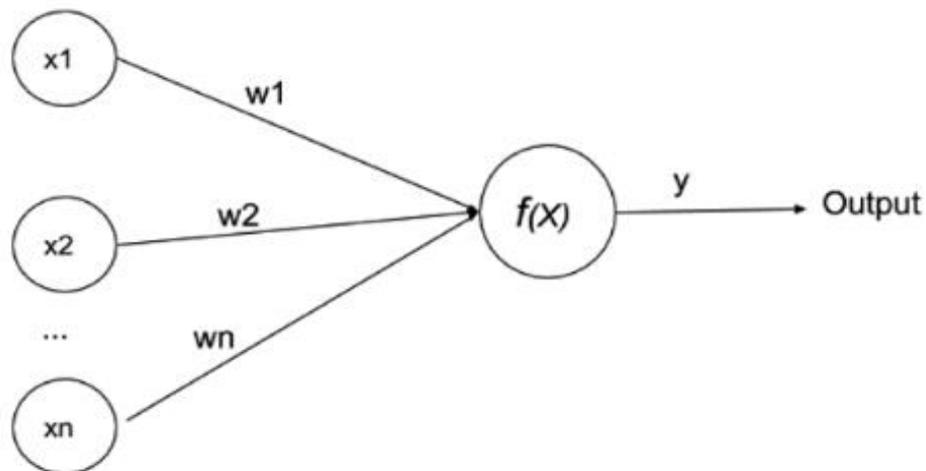


Figure 1.16: Un neurone artificiel.

### 1.18. Conclusion :

Dans ce chapitre, nous avons appris le système visuel d'humain et les difficultés qui sont des obstacles pour la machine de détecter ou visualiser un objet. Aussi on a appris la différence entre la détection d'objets et la reconnaissance. Nous avons appris ce que sont les caractéristiques d'une image et comment elles sont importantes pour la détection d'objets et le suivi des caractéristiques.

En outre, le chapitre couvre l'apprentissage profond et les réseaux de neurones artificiels (ANN).

# Chapitre 1 : La bibliothèque OpenCv

## Chapitre 2 : Les ordinateurs monocartes (SBC)

---



## **Chapitre 2 : Les ordinateurs monocartes (SBC)**

### **2.1. Introduction :**

Nous allons maintenant définir les ordinateurs monocartes, puis on va examiner la série Raspberry Pi. Cependant, avant cela nous allons nous familiariser avec d'autres familles de SBC populaires.

### **2.2. Ordinateurs monocartes :**

Un ordinateur monocarte (en abrégé SBC) est un système informatique complet sur une seule carte de circuit imprimé (en abrégé PCB).

La carte comporte généralement un ou plusieurs processeurs, de la mémoire vive, d'entrées/sorties (E/S), d'un port Ethernet pour la mise en réseau et de ports USB pour l'interfaçage avec des périphériques USB.

Quelques ordinateurs monocartes sont également équipés de Wi-Fi et de Bluetooth. Les SBC exécutent des distributions de systèmes d'exploitation tels que Ubuntu, Windows, Debian, etc. Ces distributions de systèmes d'exploitation ont des versions spécialement conçues pour être utilisées avec ces SBC.

Contrairement aux ordinateurs traditionnels, un SBC n'est pas modulaire et son matériel ne peut pas être mis à niveau car tous les composants (tels que le CPU, la RAM, le GPU) sont intégrés sur un seul circuit imprimé.

Les SBC sont utilisés comme ordinateurs à bas prix dans les domaines suivants : dans le monde universitaire, la recherche et diverses autres industries.

L'utilisation des SBC dans les systèmes intégrés est assez répandue, et de nombreuses personnes, organisations de recherche et entreprises ont développé et commercialisé des produits entièrement fonctionnels et utilisables basés sur les SBC. La plupart de ces produits sont financés par des fonds publics.

Le principal avantage des SBC est l'intégration d'entrées/sorties générales (GPIO).

L'avantage principal des SBC est la présence de broches d'entrée/sortie à usage général (GPIO). Ces broches fournissent des fonctionnalités telles que divers bus (Serial Peripheral Interface (SPI), I2C et SMBus), des entrées/sorties numériques, des entrées analogiques et des modulations de largeur d'impulsion (PWM).

Presque tous les SBCs populaires ont des GPIO sous une forme ou une autre. En raison de leur petit facteur de forme et de leur GPIO embarqué, ils sont populaires dans les écoles, les universités, les centres de formation, les camps d'entraînement et

## **Chapitre 2 : Les ordinateurs monocartes (SBC)**

les espaces de fabrication. Ils sont fréquemment utilisés dans les domaines suivants réseaux de capteurs et de l'Internet des objets (IoT).

Pour résumer, les avantages des SBC sont les suivants :

- Faible coût
- Taille réduite
- Faible consommation d'énergie
- Possibilité de réseau et d'E/S embarqués

Cependant, les SBCs ont leurs propres inconvénients. Comme tous les composants d'un SBC se trouvent sur le même PCB, il peut être très difficile de réparer si un composant est endommagé pour des raisons mécaniques ou électriques. Pour la même raison, nous ne pouvons même pas mettre à jour quoi que ce soit sur un SBC. Ce sont les seuls inconvénients majeurs des SBC.

Le Micro computer Trainer MMD-1, conçu par John Titus en 1976, est le premier véritable micro-ordinateur monocarte basé sur le microprocesseur Intel C8080A. Il a été appelé Dyna-micro pendant la phase de prototypage, et les unités de production ont été appelées

MMD-1 (abréviation de Mini-Micro Designer 1).

### **2.3. Les différentes familles SBC :**

#### **2.3.1. La famille Beagleboard :**

La Fondation BeagleBoard.org est une organisation basée aux Etats-Unis. C'est une entité à but non lucratif, et son objectif est de fournir une éducation et une collaboration autour de la conception, le développement, le test et l'utilisation de matériel et de logiciels open source dans le domaine des systèmes embarqués.

Ils ont développé plusieurs SBCs portant le nom de beagles (une race populaire de chiens domestiques).

Nous trouverons une liste des SBC actuels qu'ils ont développés et qui sont en production à l'adresse suivante : <http://beagleboard.org/boards>.

Nous pouvons également trouver des produits et des accessoires liés aux cartes Beagle à la même adresse.

## Chapitre 2 : Les ordinateurs monocartes (SBC)

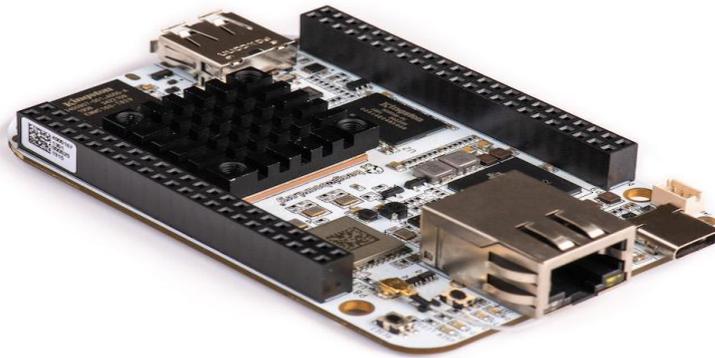


Figure 2.1 : La carte BeagleBone AI

### 2.3.2. ASUS Tinkerboard :

La Tinkerboard ASUS est conçue et fabriquée par ASUS (une multinationale basée à Taiwan).

Sa taille, son agencement et ses broches sont compatibles avec les cartes Raspberry troisième génération de Raspberry Pi.

Nous pouvons trouver plus de détails sur toutes les éditions de la Tinkerboard d'ASUS à l'adresse : <https://tinkerboard.asus.com/index.html>.

La photo suivante montre la vue de dessus d'une Tinkerboard ASUS :



Figure 2.2: tinker board

### 2.3.3. NVIDIA Jetson:

NVIDIA Jetson est une famille de modules utilisés pour la vision par ordinateur, l'IA et le traitement de la parole.

Le meilleur membre avec lequel les débutants peuvent commencer est le Jetson Nano. La page Web du kit de développement Jetson Nano à l'adresse <https://developer.nvidia.com/embedded/jetson-nano-developer-kit>.

## Chapitre 2 : Les ordinateurs monocartes (SBC)

Voici une vue de côté du kit de développement :



Figure 2.3 : NVIDIA Jetson nano kit

### 2.4. Raspberry Pi :

Raspberry Pi est une série de SBC à bas prix et de la taille d'une carte de crédit, développée par la Fondation Raspberry Pi au Royaume-Uni.

L'objectif du développement de Raspberry Pi était de promouvoir l'enseignement des compétences informatiques de base et de la programmation dans les écoles, ce qu'il a très bien fait.

Le Raspberry Pi a étendu son empreinte bien au-delà de son objectif initial en s'imposant sur le marché des systèmes embarqués et de la recherche informatique dans les applications universitaires et industrielles.

La Fondation Raspberry Pi propose des téléchargements pour de nombreuses distributions de systèmes d'exploitation populaires.

Nous pouvons utiliser une variété de langages de programmation tels que Python, C, C++ et Java avec le Raspberry Pi.

#### 2.4.1. Les modèles de Raspberry Pi :

La carte Raspberry Pi existe en plusieurs modèles. De plus, de nombreux accessoires sont associés à ces modèles.

Nous pouvons trouver la liste actuelle des modèles en cours de production sur la page des produits de la Fondation Raspberry Pi (<https://www.raspberrypi.org/products/>).

## **Chapitre 2 : Les ordinateurs monocartes (SBC)**

Malheureusement, la page ne contient aucune information sur les cartes de produits abandonnés de la famille Raspberry Pi.

En outre, le Raspberry Pi est également disponible sous une forme plus souple destinée aux applications industrielles et embarquées. Cette forme est connue sous le nom de module de calcul. Le module de calcul a également de nombreuses itérations. Un kit de prototypage de module de calcul est également mis à disposition par la fondation. Vous pouvez en savoir plus sur les modules de calcul et le kit de prototypage sur la page des produits Raspberry Pi dont nous avons parlé précédemment.

Comme nous l'avons vu, il existe de nombreux modèles de cartes Raspberry Pi, nous utiliserons les modèles Raspberry Pi 4B 4 GB et Raspberry Pi Zero W avec en-tête pour nos exemples de vision par ordinateur. Cependant, ces exemples peuvent également être exécutés sur d'autres modèles de cartes de Raspberry Pi. Ceci est dû au fait que tous les logiciels que nous utilisons (le système d'exploitation, le langage de programmation et la bibliothèque OpenCV) sont entièrement rétrocompatibles.

### **2.4.1.1. Raspberry Pi modèle 4B :**

#### **2.4.1.1.1. Introduction :**

Le Raspberry Pi 4 Modèle B (Pi4B) est le premier d'une nouvelle génération d'ordinateurs Raspberry Pi supportant plus de RAM et avec des performances de CPU, GPU et I/O considérablement améliorées, le tout dans un facteur de forme, une enveloppe de puissance et un coût similaire à ceux de la génération précédente.

Le même facteur de forme, la même enveloppe énergétique et le même coût que la génération précédente de Raspberry Pi 3B+.

Le Pi4B est disponible avec 1, 2 ou 4 gigaoctets de LPDDR4 SDRAM.[8]

#### **2.4.1.1.2. Caractéristiques:**

##### **2.4.1.1.2.1. Hardware:**

- Quad core 64-bit ARM-Cortex A72 running at 1.5GHz
- 1, 2 and 4 Gigabyte LPDDR4 RAM options
- H.265 (HEVC) hardware decode (up to 4Kp60)
- H.264 hardware decode (up to 1080p60)
- VideoCore VI 3D Graphics
- Supports dual HDMI display output up to 4Kp60

## Chapitre 2 : Les ordinateurs monocartes (SBC)

### 2.4.1.1.2.2. Interfaces:

- LAN sans fil 802.11 b/g/n/ac
- Bluetooth 5.0 avec BLE
- 1x carte SD
- 2 ports micro-HDMI prenant en charge deux écrans jusqu'à une résolution de 4Kp60.
- 2 ports USB2.
- 2 ports USB3.
- 1 port Gigabit Ethernet (prend en charge PoE avec un adaptateur PoE HAT supplémentaire).
- 1 port caméra Raspberry Pi (MIPI CSI à 2 voies).
- 1 port d'affichage Raspberry Pi (MIPI DSI à 2 voies).
- 28x GPIO utilisateur prenant en charge diverses options d'interface :
  - Jusqu'à 6x UART
  - Jusqu'à 6x I2C
  - Jusqu'à 5 SPI
  - 1x interface SDIO
  - 1x DPI (affichage parallèle RVB)
  - 1x PCM
  - Jusqu'à 2x canaux PWM
  - Jusqu'à 3x sorties GPCLK

### 2.4.1.1.2.3. Software :

- Jeu d'instructions ARMv8
- Pile logicielle Linux mature
- Développée et maintenue activement
- Support récent du noyau Linux
- Nombreux pilotes développés en amont
- Un user land stable et bien supporté
- Disponibilité de fonctions GPU utilisant des API standard.

Le diagramme suivant montre tous les connecteurs et composants importants d'une carte Raspberry Pi :

## Chapitre 2 : Les ordinateurs monocartes (SBC)

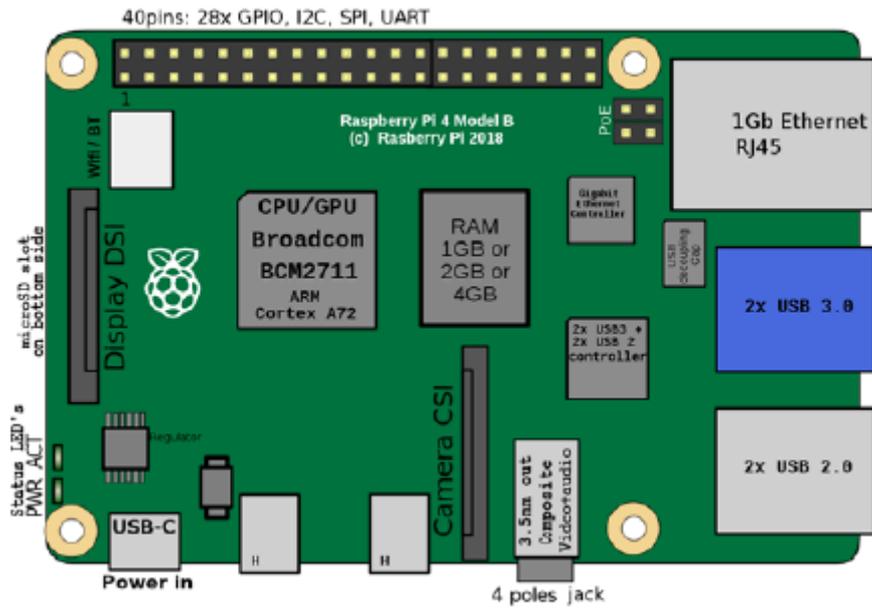


Figure 2.4 : Raspberry Pi 4B vue de dessus.

La photo suivante montre la vue de dessus du Raspberry Pi modèle 4B :

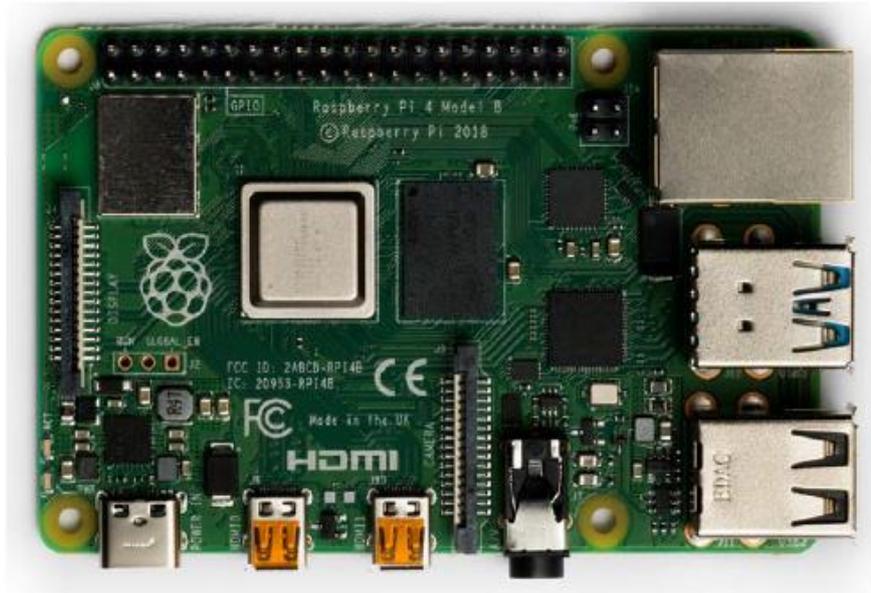


Figure 2.5: La vue de dessus du Raspberry Pi 4B.

Voici une photo du modèle sous un angle :

## Chapitre 2 : Les ordinateurs monocartes (SBC)



Figure 2.6 : Raspberry Pi 4B sous un angle.

### 2.4.1.1.2.4. Spécifications mécaniques:

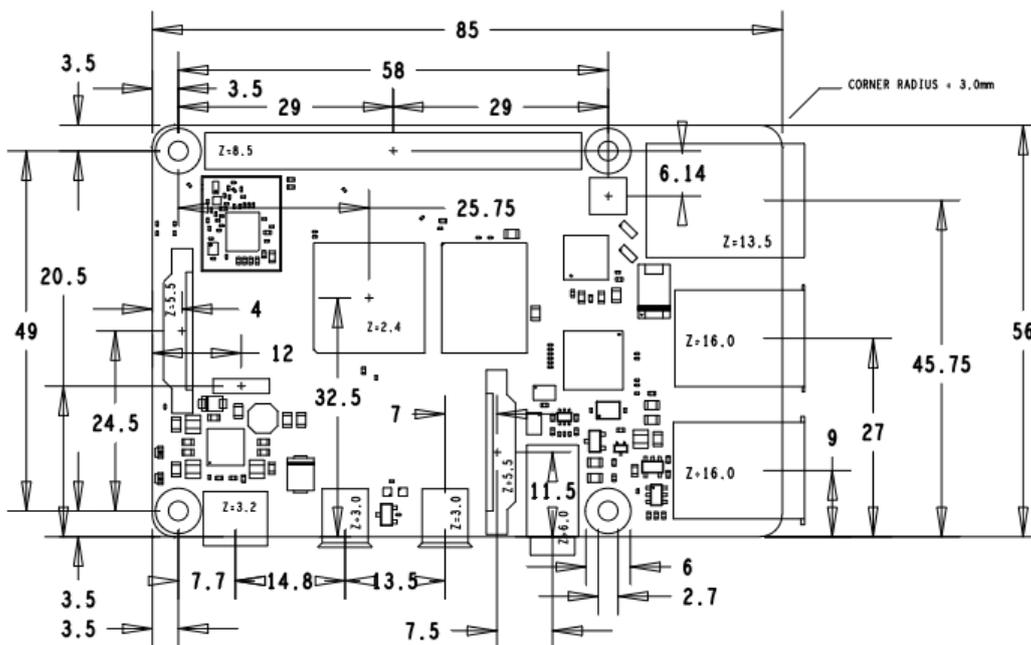


Figure 2.7 : Dimensions mécaniques.

### 2.4.1.1.2.5. Spécifications électriques :

Veillez noter que VDD IO est la tension de la banque GPIO qui est liée au rail d'alimentation 3.3V de la carte.

## Chapitre 2 : Les ordinateurs monocartes (SBC)

Symbol	Paramètres	Min	Max	Unité
VIN	5V Input Voltage	-0.5	6	V

Tableau 2.1 : Valeurs limites absolues.

Symbole	Paramètres	Conditions	MIN	Typique	MAX	Unité
VIL	Tension basse d'entrée	VDD_IO=3.3V	-	-	TBD	V
VIH	Haute tension d'entrée	VDD_IO=3.3V	TBD	TBD	-	V
IIL	Courant de fuite d'entrée	TA = +85°C	-	-	TBD	μA
CIN	Capacité d'entrée	-	-	-	-	pF
VOL	Tension de sortie basse	VDD_IO=3.3V, IOL=-2MA	-	-	TBD	V
VOH	Haute tension de sortie	VDD_IO=3.3V, IOL=2MA	TBD	-	-	V
IOL	Courant de sortie faible	VDD_IO=3.3V, VO=0.4V	TBD	-	-	mA
IOH	Courant de sortie élevé	VDD_IO=3.3V, VO=2.3V	TBD	-	-	mA
RPU	Résistance Pullup	-	TBD	-	TBD	kΩ
RPD	Résistance Pulldown	-	TBD	-	TBD	kΩ

Tableau 2.2 : Caractéristiques du courant continu.

Pin	Symbole	Paramètre	Min	Typique	Max	Unité
Digital outputs	<i>t<sub>montée</sub></i>	10-90% temps de montée	-	TBD	-	ns
Digital outputs	<i>t<sub>descente</sub></i>	90-10% temps de descente	-	TBD	-	ns

Tableau 2.3 : Caractéristiques des broches d'E/S numériques.

## Chapitre 2 : Les ordinateurs monocartes (SBC)



Figure 2.8 : Caractéristiques des entrées-sorties numériques.

### 2.4.1.1.2.6. Alimentation électrique :

Le Pi4B a besoin d'une alimentation USB-C de bonne qualité capable de fournir 5V à 3A. Si les périphériques en aval consomment moins de 500mA, une alimentation 5V, 2.5A peut être utilisée.



Figure 2.9: USB Type-C pin.

### 2.4.1.1.3. Périphériques :

#### 2.4.1.1.3.1. GPIO Interface:

Le Pi4B met à disposition 28 GPIO BCM2711 via un connecteur standard à 40 broches du Raspberry Pi. Ce connecteur est rétrocompatible avec toutes les cartes Raspberry Pi précédentes dotées d'un connecteur à 40 broches.

## Chapitre 2 : Les ordinateurs monocartes (SBC)

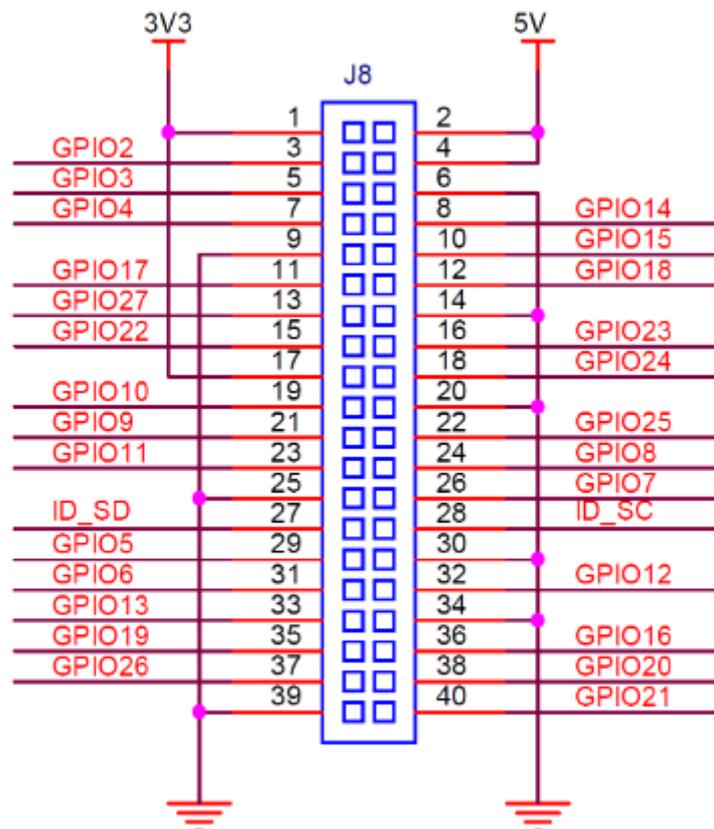


Figure 2.10 : Brochage du connecteur GPIO.

ID\_SD et ID\_SC sont des pins réservés pour HAT ID EEPROM.

### 2.4.1.1.3.2. Interface parallèle d'affichage (DPI) :

Une interface parallèle standard RVB (DPI) est disponible sur les GPIO. Cette interface parallèle jusqu'à 24 bits peut supporter un écran secondaire.

### 2.4.1.1.3.3. Interface SD/SDIO :

Le Pi4B dispose d'un connecteur de carte SD dédié qui supporte le mode 1,8V, DDR50 (avec une bande passante maximale de 50 Mégaoctets / sec). En outre, une interface SDIO est disponible sur les broches GPIO.

### 2.4.1.1.3.4. Interfaces caméra et écran :

Le Pi4B est équipé d'une caméra Raspberry Pi 2 voies MIPI CSI et d'un écran Raspberry Pi 2 voies MIPI DSI.

Ces connecteurs sont rétrocompatibles avec les anciennes cartes Raspberry Pi et prennent en charge tous les périphériques de caméra et d'affichage Raspberry Pi disponibles.

## Chapitre 2 : Les ordinateurs monocartes (SBC)

### 2.4.1.1.3.5.USB:

Le Pi4B possède 2 prises USB2 et 2 prises USB3 de type A. Le courant USB en aval est limité à environ 1,1 A environ pour l'ensemble des quatre prises.

### 2.4.1.1.3.6. HDMI:

Le Pi4B possède 2 ports micro-HDMI, qui prennent tous deux en charge CEC et HDMI 2.0 avec des résolutions allant jusqu'à 4Kp60.



Figure 2.11: Câble HDMI.



Figure 2.12 : Ports HDMI, mini-HDMI, et micro-HDMI.

### 2.4.1.1.3.7. Audio et composite (sortie TV):

Le Pi4B prend en charge une sortie audio analogique de qualité proche de celle d'un CD et une sortie TV composite via un jack TRS à 4 anneaux.

La sortie audio analogique peut alimenter directement un casque de 32 Ohms.

### 2.4.1.1.3.8. Plage de températures et thermales :

La plage de température ambiante de fonctionnement recommandée est de 0 à 50 degrés Celsius.

Pour réduire la puissance thermique en cas de fonctionnement au ralenti ou de charge légère, le Pi4B réduit la vitesse d'horloge et la tension du CPU.

## **Chapitre 2 : Les ordinateurs monocartes (SBC)**

En cas de charge plus importante, la vitesse et la tension (et donc la production thermique) sont augmentées. Le régulateur interne réduit à la fois la vitesse et la tension du CPU afin de s'assurer que la température du CPU ne dépasse jamais 85 degrés C.

Le Pi4B fonctionnera parfaitement bien sans refroidissement supplémentaire et est conçu pour des performances de pointe.

Le Pi4B fonctionnera parfaitement bien sans refroidissement supplémentaire et est conçu pour des performances rapides - en prévoyant une utilisation légère en moyenne et en augmentant la vitesse du CPU lorsque cela est nécessaire.

Si un utilisateur souhaite charger le système en permanence ou le faire fonctionner à une température élevée à pleine performance, un refroidissement supplémentaire peut être nécessaire.

### **2.4.1.2. Raspberry pi Zero w:**

#### **2.4.1.2.1. Description:**

Sept ans jour pour jour après le lancement du Raspberry Pi (qui s'est vendu à cent mille exemplaires le premier jour), la Fondation Raspberry Pi a annoncé le Raspberry Pi Zero W, une nouvelle variante du Raspberry Pi Zero avec LAN sans fil et Bluetooth, au prix de 10 dollars seulement.

#### **2.4.1.2.2. L'histoire :**

En novembre 2015, le diminutif Raspberry Pi d'entrée de gamme à 5 dollars, le Raspberry Pi Zero, a été lancé. Son coût a été divisé par cinq par rapport au modèle A original : il était suffisamment bon marché pour qu'un exemplaire soit même collé sur la couverture du MagPi, au risque de provoquer une insurrection civile dans les kiosques à journaux du pays.

Au cours des quinze mois qui ont suivi, le Zero s'est doté d'un connecteur d'appareil photo et a trouvé sa place dans tous les appareils, des salles d'arcade miniatures aux skateboards électriques. Nombre de ces cas d'utilisation nécessitent une connectivité sans fil.

## Chapitre 2 : Les ordinateurs monocartes (SBC)



Figure 2.13: Raspberry pi zero w.

L'indicateur homebrew "People in Space" dans le hall d'entrée de Pi Towers est un exemple typique, avec un dongle sans fil officiel accroché à l'unique port USB : les utilisateurs finissent souvent par ajouter un hub USB pour leur permettre de connecter un clavier, une souris et un adaptateur réseau, et ce hub peut facilement coûter plus cher que la Zero elle-même.

La Zero W résout ce problème en intégrant davantage de fonctionnalités dans le produit de base. Elle utilise la même puce sans fil Cypress CYW43438 que le Raspberry Pi 3 Model B pour fournir une connectivité LAN sans fil 802.11n et Bluetooth 4.0.

### 2.4.1.2.3. Caractéristiques :

- CPU monocœur de 1 GHz
- 512 MO DE RAM
- Port Mini-HDMI
- Port Micro-USB On-The-Go
- Alimentation Micro-USB
- Connecteur 40 broches compatible HAT
- Connecteurs vidéo composite et de réinitialisation
- Connecteur pour caméra CSI
- LAN sans fil 802.11n
- Bluetooth 4.0

Le Zero W constitue un meilleur ordinateur polyvalent, car il est moins probable que vous ayez besoin d'un hub : si vous utilisez des périphériques Bluetooth, vous pourriez bien vous retrouver avec rien du tout branché sur le port USB. Et bien sûr, c'est une excellente plateforme pour expérimenter des applications IoT.

## Chapitre 2 : Les ordinateurs monocartes (SBC)

### 2.4.1.2.4. Assignment des broches GPIO :

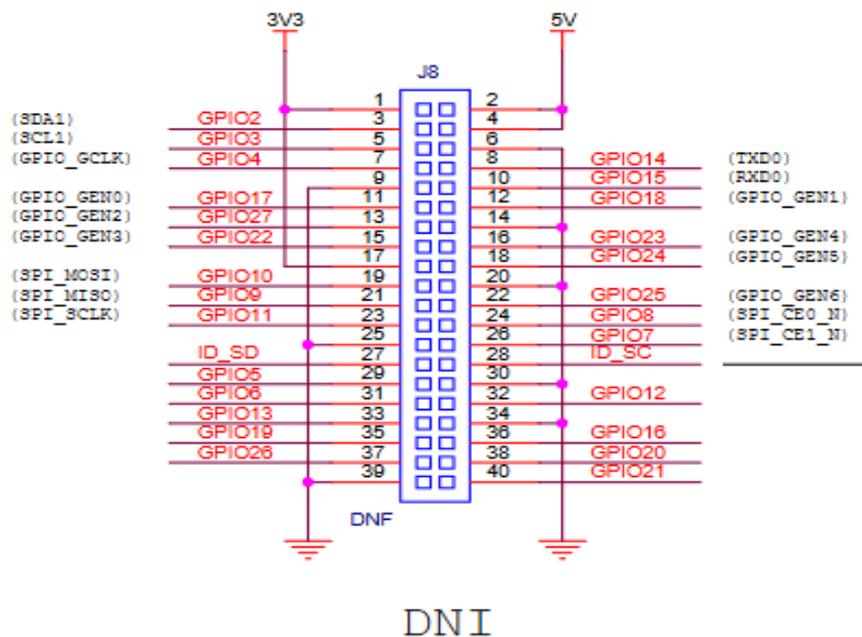


Figure 2.14: Assignment des broches GPIO de Raspberry pi zero w.

On observe que Raspberry pi 4B a la même assignation GPIO de Raspberry pi zero w.

### 2.4.1.3. Raspbian Os :

Raspberry Pi OS est un système d'exploitation gratuit basé sur Debian, optimisé pour le matériel Raspberry Pi, et est le système d'exploitation recommandé pour une utilisation normale sur un Raspberry Pi. Le système d'exploitation est livré avec plus de 35 000 paquets : des logiciels précompilés regroupés dans un format agréable pour une installation facile sur votre Raspberry Pi.

Le système d'exploitation Raspberry Pi est en cours de développement actif, l'accent étant mis sur l'amélioration de la stabilité et des performances d'autant de paquets Debian que possible sur Raspberry Pi.

## Chapitre 2 : Les ordinateurs monocartes (SBC)

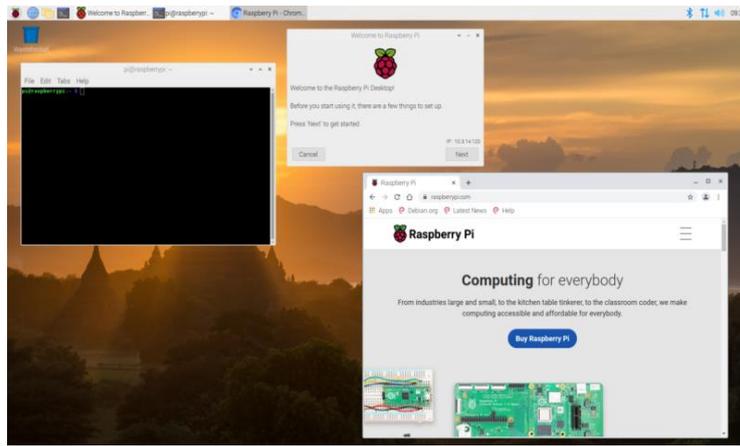


Figure 2.15: Raspberry pi os.

### 2.5. Le meilleur ordinateur monocarte :

- Meilleur ordinateur monocarte 2021 : **Raspberry Pi 4.**
- L'ordinateur monocarte le plus puissant 2021 : **UDOO Bolt.**
- Meilleur ordinateur monocarte ARM pour l'intelligence artificielle, la robotique, l'utilisation sur PC et plus encore : **Nvidia Jetson Xavier NX.**
- Meilleur ordinateur monocarte économique pour l'intelligence artificielle : **Nvidia Jetson Nano.**
- Meilleur PC monocarte pour Android : **Odroid N2.**
- Meilleure alternative au Raspberry Pi : **Rock Pi 4.**

### 2.6. Conclusion :

A travers ce chapitre, nous avons présenté les différents ordinateurs monocartes qu'ils sont utilisés dans le parcours académique et professionnelle on a concentré sur la famille Raspberry pi ainsi que ses caractéristiques et spécifications électrique et mécanique. Puis nous avons vu le système d'exploitation de la carte Raspberry pi. Et à la fin on a vu les meilleurs ordinateurs monocartes par rapport à l'utilisation.

## **Chapitre 3 : Application et Résultats**

---



## Chapitre 3 : Application et Résultats

### 3.1. Introduction :

Le but principal de ce projet est de diminuer le nombre des morts sur la route jusqu'au possible, et de protéger les utilisateurs de la route de la fatigue au moment de conduite. Et de détecter la fatigue du conducteur.

### 3.2. Liste de matériel :

Matériel	Description	Prix
Logitech Webcam C120	-Capteur CMOS de qualité -Capture vidéo : 640 x 480 pixels -Capture d'images jusqu'à 1,3 mégapixel (par interpolation logicielle) -Affichage jusqu'à 30 images par seconde (sur les systèmes conformes à la configuration recommandée) -Certifiée USB 2.0 -Mise au point manuelle.	2300 DA
Raspberry pi 4B	-Econome d'énergie. -Deux ports USB 2.0. -Deux ports USB 3.0. -Equipé avec l'Ethernet Gigabit port ainsi réseau sans fil et Bluetooth. -RAM de 4 GB. -Deux micro HDMI supportent 4K.	11 034,84 DA TTC
Souris	-Sans fil	1000 DA
Clavier	-Sans fil	1900 DA
Carte mémoire	-Capacité de 32 GB. -Transmission de données : 10MB/S	1800 DA

Tableau 3.1 : le matériel utilisé dans l'application.

## Chapitre 3 : Application et Résultats

L'emplacement de la caméra dans la voiture :



Figure 3.1 : L'emplacement de la caméra dans la voiture.

### 3.3. Les paquets nécessaires :

- . **Scipy** : Nous aurons besoin du paquet **SciPy** pour pouvoir calculer la distance euclidienne entre les points de repère du visage dans le calcul du ratio d'aspect de l'œil.
- . **Imutils** : Nous aurons également besoin du package **imutils**, ma série de fonctions de vision par ordinateur et de traitement d'images pour faciliter le travail avec OpenCV.
- . **Dlib** : est une bibliothèque logicielle polyvalente écrite dans le langage de programmation C++. Sa conception est fortement influencée par les idées de la conception par contrat et de l'ingénierie logicielle basée sur les composants.
- . **Math** : Python dispose d'un ensemble de fonctions mathématiques intégrées, dont un module mathématique étendu, qui nous permet d'effectuer des tâches mathématiques sur des nombres.
- . **Argparse** : Le module **argparse** permet d'écrire facilement des interfaces de ligne de commande conviviales. Le programme définit les arguments dont il a besoin, et **argparse** trouvera comment les analyser à partir de `sys.argv`.
- . **NumPy** : est une bibliothèque Python utilisée pour travailler avec des tableaux. Elle dispose également de fonctions permettant de travailler dans le domaine de l'algèbre linéaire, de la transformée de Fourier et des matrices.

### 3.4. L'organigramme de surveillance des yeux pour détecter la fatigue :

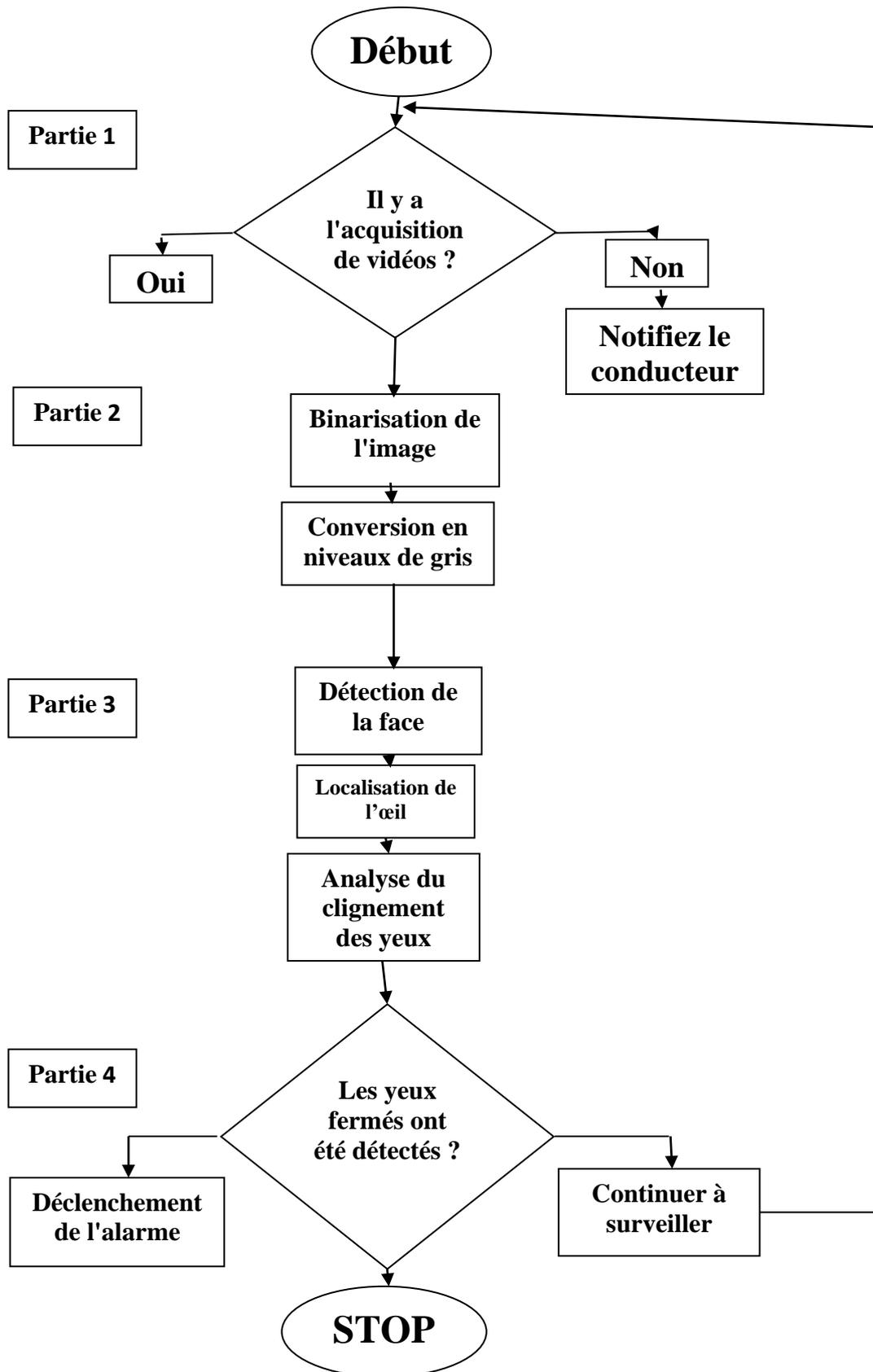


Figure 3.2 : Algorithme de surveillance des yeux pour détecter la fatigue.

## Chapitre 3 : Application et Résultats

### 3.4.1. Algorithme :

Les principales étapes de la détection de l'état des conducteurs peuvent être divisées en quatre blocs principaux, comme le montre l'organigramme de la figure 3.2.

- Dans la première partie, l'objectif est de détecter l'état du conducteur. Nous avons considéré qu'il est très important de vérifier si le dispositif qui est utilisé pour détecter les yeux (dans ce cas, la caméra vidéo) est fonctionnel ou non.
- Dans la deuxième partie, on vérifie si les images sont claires et correctement collectées. Puis on a converti l'image de base couleur à base niveau de gris.
- Dans la troisième partie, le système décide si les yeux du conducteur sont fermés ou ouverts par la détection de la face et la localisation des yeux et le système calcule l'état des yeux.
- Dans la quatrième partie, la décision est prise de poursuivre la surveillance ou de déclencher l'alarme.

### 3.4.2. Méthodologie :

Partie 1 : Acquisition vidéo

Une webcam montée à bord du véhicule se chargera des images.

Partie 2 : L'image couleur capturée par la caméra vidéo sera convertie en une image en niveaux de gris.

Dans les images réelles, il n'y a pas de pixels complètement blancs ou complètement noirs, car il s'agit d'une image en niveaux de gris.



Figure 3.3 : Image à échelle de gris pour la détection des caractéristiques des bords et des lignes.

Partie 3 : Détection des visages et localisation des yeux

Partie 4 : Déclenchement de l'alarme

### Chapitre 3 : Application et Résultats

Après avoir appliqué les étapes mentionnées ci-dessus, le système va faire une analyse et décider de l'état du conducteur. Si la somnolence est détectée, l'alarme se déclenche.

#### 3.5. Résultats de la simulation et discussion :

Chaque œil est divisé en six coordonnées dans l'axe (x, y), la numérotation commençant dans le coin inférieur gauche et se poursuivant dans le sens des aiguilles d'une montre.

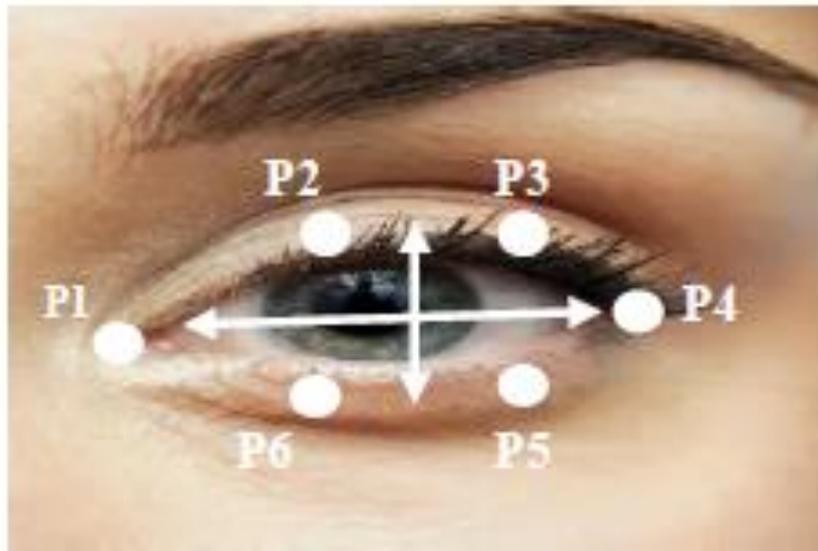


Figure 3.4 : Six marqueurs faciaux associés à l'œil.

Sur la base de l'image ci-dessus, des points clés tels que le rapport d'aspect de l'œil (EAR) peut être calculés :

$$EAR = \frac{\|p2 - p6\| + \|p3 - p5\|}{2\|p1 - p4\|}$$

La formule de calculer l'aspect de l'œil droite sur python :

```
EARd= math.sqrt(np.square(shape[39,0]-shape[41,0])+np.square(shape[39,1]-shape[41,1]))
```

Pour chaque image identifiée dans la vidéo, le rapport d'aspect de l'œil est calculé. Cette technologie est utilisée pour identifier le nombre de flashes.

Lorsque le conducteur ferme les yeux, l'EAR descend approximativement au niveau 0, et l'ouverture des yeux revient à la normale.

Lorsque l'EAR enregistre une valeur inférieure à un seuil défini dans les 5 secondes, l'état de somnolence est identifié et un signal d'alarme est déclenché.

### Chapitre 3 : Application et Résultats

Comme le montre la figure 3.5 ci-dessous, (L'axe des X : Nombres des images enregistrés, L'axe des Y : la valeur de EAR) du côté gauche, l'œil est complètement ouvert et le rapport d'aspect de l'œil est donc presque constant dans le temps. Cependant, lorsque la personne cligne des yeux, le rapport d'aspect de l'œil diminue fortement, s'approchant de 0.

Comme l'illustre la représentation du signal, le rapport d'aspect de l'œil est constant, diminue progressivement lorsque la personne cligne des yeux et revient au premier état, indiquant qu'un seul clignement s'est produit.

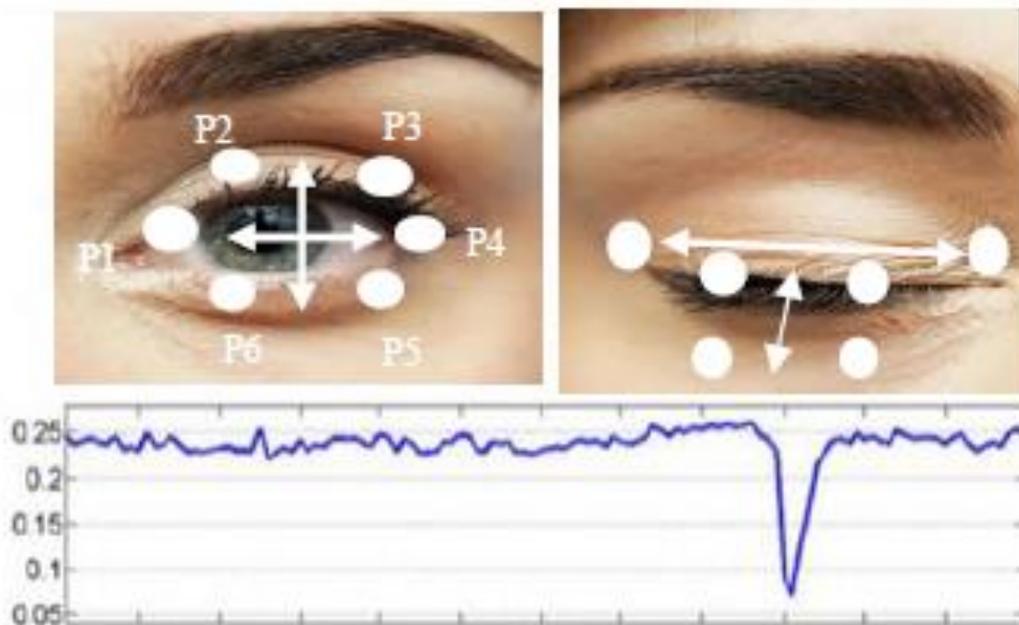


Figure 3.5 : Visualisation des repères des yeux lorsqu'ils sont fermés et ouverts.

Nous avons défini deux valeurs. La première valeur représente la valeur par défaut de 0,25. Si le EAR diminue en dessous de cette valeur par défaut et augmente ensuite, cela signifie qu'un flash a été enregistré.

La deuxième valeur indique que 60 images successives (valeur par défaut) ont été enregistrées avec une valeur inférieure à 0,25 (valeur par défaut).

Après l'enregistrement de `Eye_Aspect_Ration_trigger = 60`, le système génère le signal d'alarme.

a) `Eye_Aspect_Ratio_threshold = 0.25`

b) Déclenchement du rapport d'observation des yeux = 60

### 3.6. Résultats d'expériences et d'analyses :

#### 3.6.1. Détection des visages et des yeux :

Pour détecter la somnolence, nous avons utilisé Python comme langage de programmation et appliqué l'algorithme de Hog en utilisant dlib et OpenCV.

### Chapitre 3 : Application et Résultats

Une étape importante dans la détection de la somnolence du conducteur est d'identifier les zones d'intérêt du visage.

Comme nous l'avons montré ci-dessus, dlib a permis d'identifier les régions d'intérêt qui comprennent les yeux, le nez, la bouche et les sourcils.

Le détecteur facial dlib a identifié 68 coordonnées spécifiques à la structure du visage.

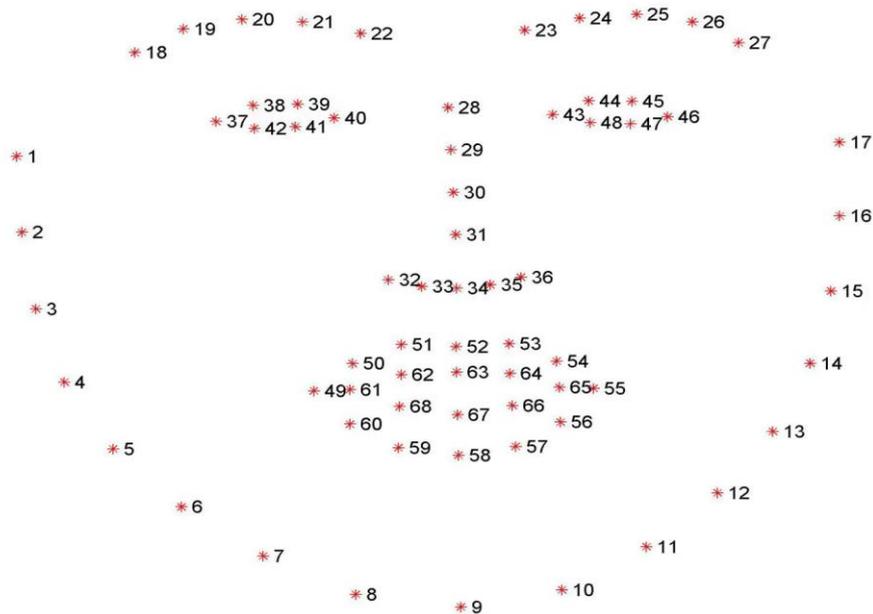


Figure 3.6 : Visualisation des 68 coordonnées des points de repère du visage à partir du jeu de données iBUG 300-W.

Les coordonnées 68 sont positionnées sur les axes x et y.

Ainsi, la zone d'intérêt des zones identifiées est la suivante :

L'œil droit : [36,42]

Œil gauche : [42,48]

Sourcil droit : [17,22]

Sourcil gauche : [22,27]

Nez : [27,35]

Bouche : [48,68]

La figure ci-dessous introduire la détection de visage et des yeux :

## Chapitre 3 : Application et Résultats

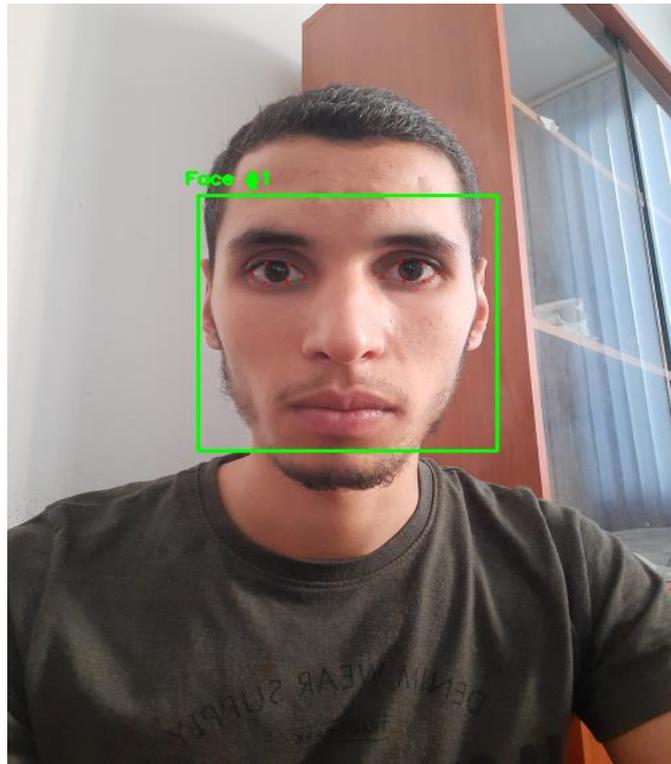
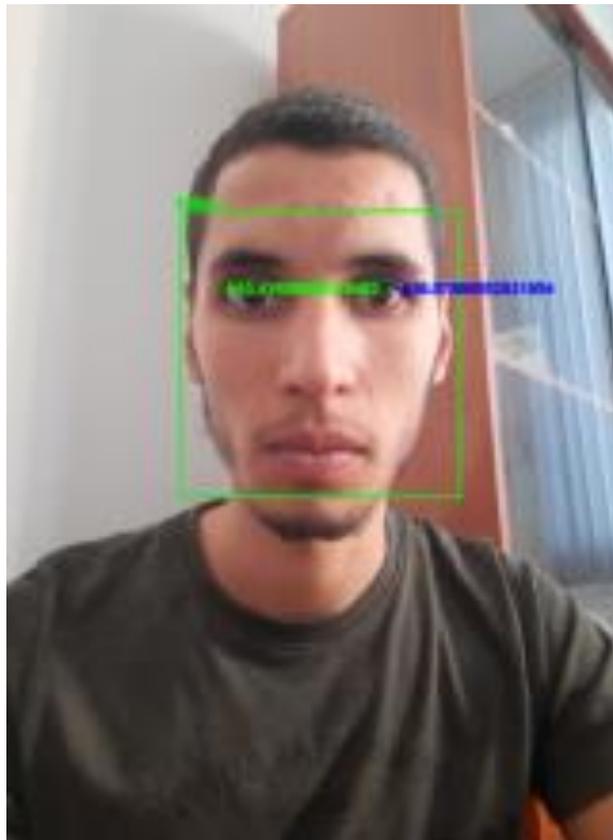


Figure 3.7 : Détection du visage et des yeux.

### 3.6.2. Détection de la fatigue :



## Chapitre 3 : Application et Résultats

Figure 3.8 : les yeux dans l'état active.

A l'état active on a obtenu ces résultats :

EAR droite = 63.63961030678928

EAR gauche = 56.753854494650845

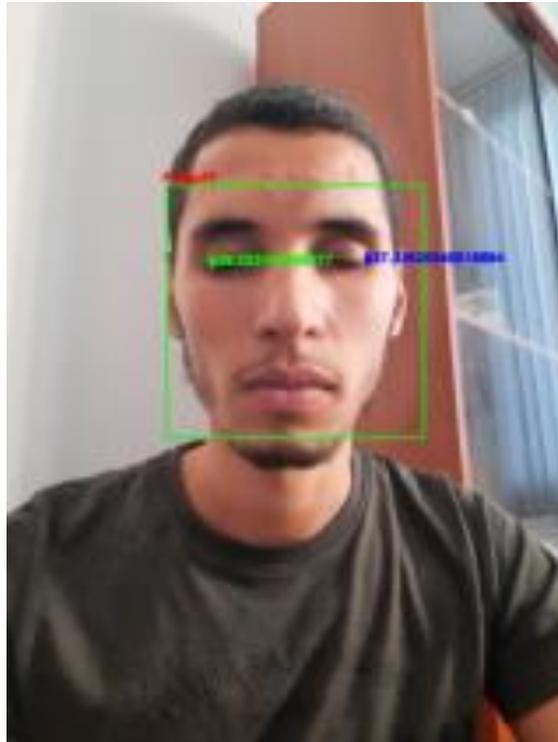


Figure 3.9 : les yeux à l'état fatigüe.

Quand les yeux sont fermés on a obtenu ces résultats :

EAR droite= 57.97413216254298

EAR gauche=54.589376255824725

### 3.7. Conclusion :

Dans ce chapitre, nous avons vu les différents périphériques utilisés avec le Raspberry pi 4B, nous avons vu aussi les paquets et les bibliothèques importer dans le programme python et on a discuté sur comment on détecte la fatigue à partir de la valeur de Eye Aspect Ratio.

## Chapitre 3 : Application et Résultats

## *Conclusion Générale :*

Dans cette mémoire, nous avons présenté les méthodes de traitement et d'analyse d'images.

Nous avons identifié le seuil où l'application offre les meilleurs résultats d'identification.

Grâce à ce système installé à l'intérieur des voitures, le nombre d'accidents de la route causés par la fatigue, la somnolence ou l'imprudence au volant pourrait être fortement réduit.

Pour identifier la somnolence des conducteurs, il a d'abord fallu identifier les régions d'intérêt du visage : les yeux, le nez, la bouche et les sourcils.

Après avoir identifié les régions d'intérêt, nous avons analysé la zone des yeux.

L'application mesure la distance euclidienne des yeux à la fois verticalement et horizontalement.

Suite au résultat EAR (Eye Aspect Ratio) obtenu, il est déterminé si l'œil est fermé ou ouvert.

La contribution apportée par cette étude est la localisation et le suivi du regard du conducteur, afin d'identifier si son attention est distraite pendant la conduite.

Pour une plus grande précision, à l'avenir, une caméra vidéo infrarouge pourrait être utilisée pour aider à identifier les yeux la nuit ou dans un environnement à faible luminosité si le conducteur porte des lunettes de soleil.

## *References:*

- [1] Analysis and forecast of the state and level of accidents on the roads of the Russian Federation and ways to reduce it. [Electronic resource] //URL: <https://e-koncept.ru/2016/96251.htm> (date of the application:18.02.2019)
- [2] Driver Fatigue Detection Using Multitask Cascaded Convolutional Networks, Xiaoshuang Liu<sup>1</sup>, Zhijun Fang<sup>1</sup>(&), Xiang Liu<sup>1</sup>, Xiangxiang Zhang<sup>1</sup>, Jianrong Gu<sup>2</sup>, and Qi Xu<sup>3</sup> School of Electronic and Electric Engineering, Shanghai University of Engineering Science, Shanghai, China {xslu, zjfang, xliu, M020215124}@sues.edu.cn <sup>2</sup> Information Center, Shanghai University of Engineering Science, Shanghai, China rong154@sues.edu.cn <sup>3</sup> College of Information Engineering, Shanghai Maritime University, Shanghai, China qixu@shmtu.edu.cn
- [3] Gollapudi, Sunila Learn Computer Vision Using OpenCV (With Deep Learning CNNs and RNNs) || Object Detection and Recognition
- [4] Manaswi, Navin Kumar, Deep Learning with Applications Using Python || Convolutional Neural Networks
- [5] Ashwin Pajankar,Raspberry Pi Computer Vision Programming Design and implement computer vision applications with Raspberry Pi, OpenCV, and Python 3, 2nd Edition.
- [6] <https://beagleboard.org/ai>
- [7] <https://tinkerboard.asus.com/index.html>
- [8] Raspberry Pi 4 Model B Datasheet Copyright Raspberry Pi (Trading) Ltd. 2019
- [9] Prateek Joshi, David Millán Escrivá, Vinícius Godoy, OpenCV By Example, Enhance your understanding of Computer Vision and image processing by developing real-world projects in OpenCV 3.
- [10] <https://www.educative.io/edpresso/what-is-feature-extraction>.
- [11] Dr. Adrian Rosebrock, Deep Learning for Computer Vision with Python, Starter Bundle, 1st Edition (1.1.0).
- [12] Shamshad Ansari, Building Computer Vision Applications Using Artificial Neural Networks, With Step-by-Step Examples in OpenCV and TensorFlow with Python.
- [13] <https://www.orbit-dz.com/product/rpi-zero-w/>
- [14]<https://www.electronicsdatasheets.com/manufacturers/raspberry-pi/parts/raspberry-pi-zero-w>.
- [15] <https://www.microdistri.com/logitech-webcam-c120-p-1102.html>
- [16] <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/>
- [17] <https://yadom.fr/raspberry-pi-4-model-b-4gb.html?src=raspberrypi>

[18][https://www.cdiscount.com/photo-numerique/r-carte+micro+sd+sandisk+32+go.html#\\_his\\_](https://www.cdiscount.com/photo-numerique/r-carte+micro+sd+sandisk+32+go.html#_his_)

[19] Contributions to driver fatigue detection based on eye-tracking, Ana-Maria Băiașu, Cătălin Dumitrescu, Department of Telematics and Electronics for Transports (Faculty of Transports) University Politehnica (of Bucharest) Bucharest, 060042, Romania. Received: May 28, 2020. Revised: December 13, 2020. Accepted: January 5, 2021. Published: January 18, 2021.

[20] C.Vertan, C. F. (2013). Achiziția și prelucrarea imaginilor color. Bucuresti: Editura Politehnica Press.

[21] Maior, C. B. S. et al. (2020) ‘Real-time classification for autonomous drowsiness detection using eye aspect ratio’, Expert Systems with Applications. doi: 10.1016/j.eswa.2020.113505.

[22] Cech, J. and Soukupova, T. (2016) ‘Real-Time Eye Blink Detection using Facial Landmarks’, Center for Machine Perception, Department of Cybernetics Faculty of Electrical Engineering, Czech Technical University in Prague.

[23] <https://pyimagesearch.com/>

[24] <https://www.simplilearn.com/image-processing-article>