



# Mémoire de Master

Présenté au

Département : Génie Électrique

Domaine : Sciences et Technologies

Filière: Electronique

Option : Electronique des systèmes embarqués

## Thème

**Réalisation et commande d'un robot mobile autonome de surveillance à base d'une carte Raspberry PI**

Présenté par :

**Djeffal Nour El Houda**

**Soutenu le 03/07/2023**

Devant la commission composée de :

Mr : Haroun Smail	M.C.B	Univ. Bouira	Encadrant
Fekik Arezki	M.C.A	Univ. Bouira	Président
Benziane Mourad	M.A.A	Univ. Bouira	Examineur

**Année universitaire : 2022/2023**

# Remerciements

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

*J'adresse mes sincères remerciements à mon encadrant monsieur HAROUN SMAIL de m'avoir proposé ce sujet de mémoire, pour sa guidance précieuse, son soutien inébranlable et son expertise inestimable.*

*Mes remerciements vont également au membres du jury, pour leur disponibilité, leurs précieuses remarques et recommandations qui ont grandement enrichi ce travail.*

*Je tiens à exprimer ma gratitude envers mes enseignants et mes mentors qui ont partagé avec générosité leurs connaissances et compétences, et qui ont contribué à mon développement académique et personnel.*

*Et pour finir je souhaite exprimer ma reconnaissance à ma famille, et tous ceux qui m'ont soutenu durant cette aventure. Leur amour, leur encouragement et leurs prières ont été des sources d'inspiration et de motivation inestimables.*

# Dédicace

*Aujourd'hui, je prends la plume pour dédier ces quelques mots à ma chère maman, symboles de ma gratitude éternelle. Tu as été et resteras toujours le pilier solide de ma vie, celle qui m'a guidée avec amour sur le chemin de la réussite.*

*Chaque instant passé à tes côtés a été empreint de tendresse, de soutien et de bienveillance. Ton amour inconditionnel m'a donné la force de dépasser les obstacles, d'oser rêver et de croire en moi-même. Tu m'as enseigné les valeurs fondamentales de respect, d'honnêteté et de persévérance, des valeurs qui continueront à guider ma vie.*

*Aujourd'hui, tandis que je contemple les fruits de mon labeur, je ne peux m'empêcher de penser à toi, qui a planté les graines de ma réussite dans les jardins fertiles de ton amour inconditionnel. Tu as été ma première enseignante, m'apprenant les leçons précieuses de la vie bien avant que je ne pénètre les portes des institutions académiques.*

*A toi ma chère grand-mère, ceci est ma profonde gratitude pour ton éternel amour, que ce rapport soit le meilleur cadeau que je puisse t'offrir malgré le destin voulait que tu partes aussi tôt avant d'assister à ma réussite.*

**-NOUR EL HOUDA-**

# **SOMMAIRE**

# Sommaire

Remerciements .....	i
Dédicace.....	ii
Sommaire .....	iv
Liste des Figures.....	vii
Liste des Tableaux.....	viii
Résumé .....	x
Glossaire .....	xii
Introduction générale.....	1

## **Chapitre I: Généralités sur la robotique mobile**

I.1.Introduction.....	4
I.2. Généralités sur la robotique .....	5
I.2.1. Définition de la robotique.....	5
I.2.2. Domaine d'application .....	5
I.2.2.1. La robotique d'intervention : .....	5
I.2.2.2. La robotique de service professionnelle : .....	6
I.2.2.3. La robotique personnelle : .....	6
I.3. Généralités sur les robots.....	6
I.3.1. Etymologie du terme robot .....	6
I.3.2. Définition .....	8
I.3.3. Architecture d'un robot .....	8
I.3.4. Classification des robots.....	9
I.4. Robots mobiles de surveillances .....	15
I.4.1. Robots mobiles de surveillances commercialisés.....	15
I.4.2. Robots mobiles de surveillances dans la littérature et la recherche.....	17
I.7. Conclusion .....	17

## **Chapitre II: Conception matérielle du robot à réaliser**

II.1. Introduction .....	19
II.2. Schéma synoptique et fonctionnement du système .....	19
II.3. Constitution matérielle du robot .....	22
II.3.1. Partie mécanique et actionneurs .....	22

II.3.1.1. Le châssis.....	22
II.3.1.2. Moteur à courant continu DC .....	24
II.3.1.3. Le Driver Moteur L298N .....	26
II.3.1.4. Servomoteur.....	29
II.3.1.5. Buzzer.....	31
II.3.2. Unité de traitement et de contrôle .....	31
II.3.2.1. Raspberry PI .....	31
II.3.2.2. Le Microcontrôleur ESP32 .....	34
II.3.3. Les capteurs.....	36
II.3.3.1. Capteur Ultrasonique HC-SR04.....	36
II.3.3.2. Capteur Gyroscopique – accélérométrique.....	38
II. 3.4. Système de vision .....	39
II.3.4.1. Webcam .....	39
II.3.5. Batteries d’Alimentation .....	40
II.3.5.1. Batterie Lithium-Polymère .....	40
II.3.5.2. Power-Bank .....	40
II.4. Conclusion .....	41

## **Chapitre III: Réalisation matérielle et logicielle du robot**

III.1. Introduction .....	43
III.2. Réalisation pratique du robot .....	43
III.2.1. Réalisation de la structure mécanique du robot .....	43
III.2.1.1. Conception et dessin de la structure du robot.....	43
III.2.1.2. Découpage de la structure .....	45
III.2.1.3. Assemblage du corps du robot.....	45
III.2.2. Réalisation du circuit électronique du robot .....	46
III.3. Principe de fonctionnement des différentes parties du robot .....	48
III.3.1. Sélectionner le chemin.....	49
III.3.1.1. Evitement d’Obstacles automatiquement.....	49
III.3.2. Calibrage du PID.....	52
III.4. Implémentation logicielle.....	53
III.4.1. Outils de développement pour le Raspberry Pi.....	54
III.4.1.1. Linux .....	54
III.4.1.2. Python .....	54
III.4.1.3. Visual Studio Code .....	57

III.4.1.4. SSH .....	58
III.4.2. Logiciel de programmation pour L'ESP32 : Le C++ .....	60
III.5. Organigramme Final du fonctionnement du robot .....	63
III.6. Conclusion.....	65

## **Chapitre IV: Systeme de surveillance visuelle**

IV.1. Introduction .....	67
IV.2. L'application du AI dans la surveillance .....	67
IV.3. Système de Capture d'images et Reconnaissance Faciale .....	69
IV.3.1. Acquisition d'image .....	70
IV.3.2. Détection de visage .....	70
IV.3.3. Le prétraitement .....	71
IV.3.4. Extraction des caractéristiques.....	71
IV.3.5. La classification .....	72
IV.3.6. L'apprentissage .....	72
IV.3.7. La décision.....	72
IV.4. Outils logiciels utilisés.....	72
IV.4.1. Base de donnée .....	72
IV.4.2. OpenCV.....	73
IV.4.3. Numpy.....	74
IV.4.4. Dlib.....	74
IV.4.5. Face_recognition .....	75
IV.6. Conclusion .....	79
Conclusion générale .....	80
Références Bibliographiques .....	83
Références des figures .....	88

# Liste des Figures

<b>Figure 1. 1.</b> Robot marin .....	5
<b>Figure 1. 2.</b> Robot Chirurgicale .....	6
<b>Figure 1. 3.</b> Robot nettoyeur .....	6
<b>Figure 1. 4.</b> Architecture d'un robot.....	8
<b>Figure 1. 5.</b> Robot à chenille qui passe partout.....	11
<b>Figure 1. 6.</b> Le robot réalisé par Boston Dynamics PETMAN.....	11
<b>Figure 1. 7.</b> Robot rampant développé par l'université Carnegie Mellon.....	12
<b>Figure 1. 8.</b> Robot mobile de type uni-Cycle.....	13
<b>Figure 1. 9.</b> Robot mobile de type tricycle .....	13
<b>Figure 1. 10.</b> Robot mobile de type voiture.....	14
<b>Figure 1. 11.</b> Robot de type omnidirectionnels.....	14
<b>Figure 1. 12.</b> Le Rover Curiosity sur la planète Mars envoyé par la Nasa.....	15
<b>Figure 1. 13.</b> Quelques exemples de robots mobiles de surveillance commercialisés .....	16
<b>Figure 2. 1.</b> Schéma synoptique du fonctionnement du système.....	19
<b>Figure 2. 2.</b> Schéma global du robot Rocker-Bogie.....	23
<b>Figure 2. 3.</b> Connexion entre deux mécanismes Lambda.....	24
<b>Figure 2. 4.</b> Structure globale de robot Rocker-Bogie à mécanisme double Lambda.....	24
<b>Figure 2. 5.</b> Construction d'un moteur à courant continu.....	25
<b>Figure 2. 6.</b> Moteur-roue DC 3-12V .....	26
<b>Figure 2. 7.</b> Module Driver Moteur L298N.....	27
<b>Figure 2. 8.</b> Activation de deux moteurs par le L298.....	27
<b>Figure 2. 9.</b> Le pont en H.....	28
<b>Figure 2. 10.</b> Sens du courant en fonction de l'état des interrupteurs d'un pont en H.....	29
<b>Figure 2. 11.</b> Vu interne d'un servomoteur. ....	29
<b>Figure 2. 12.</b> Servomoteur 270° .....	30
<b>Figure 2. 13.</b> Le pilotage d'un servomoteur par des impulsions PWM .....	30
<b>Figure 2. 14.</b> Un buzzer.....	31
<b>Figure 2. 15.</b> Raspberry Pi 4.....	32
<b>Figure 2. 16.</b> Les différents pins et leurs significations du Raspberry pi .....	34
<b>Figure 2. 17.</b> Module ESP32 .....	35
<b>Figure 2. 18.</b> Les pins du microcontrôleur ESP32 .....	35
<b>Figure 2. 19.</b> Capteur Ultrason HC-SR04.....	36
<b>Figure 2. 20.</b> Principe de fonctionnement d'un capteur ultrason.....	37
<b>Figure 2. 21.</b> Le module MPU6050 .....	38
<b>Figure 2. 22.</b> La puce du capteur d'image .....	39
<b>Figure 2. 23.</b> Webcam USB.....	39
<b>Figure 2. 24.</b> Batterie Lithium-Polymère .....	40
<b>Figure 2. 25.</b> Vu de l'intérieur d'une banque d'alimentation .....	41



<b>Figure 3. 1.</b> Plans et dimensions des éléments de la structure du Robot. ....	44
<b>Figure 3. 2.</b> Parties du corps du robot découpées dans du contreplaqué. ....	45
<b>Figure 3. 3.</b> Assemblage final du corps du robot et mise en place des moteurs de roue. ....	45
<b>Figure 3. 4.</b> Câblage du circuit électronique du robot. ....	46
<b>Figure 3. 5.</b> La réalisation finale du robot. ....	47
<b>Figure 3. 6.</b> Organigramme de la logique du fonctionnement principal du robot. ....	49
<b>Figure 3. 7.</b> Les angles de navigation. ....	50
<b>Figure 3. 8.</b> Organigramme de détection d’obstacle. ....	51
<b>Figure 3. 9.</b> Erreur dans le chemin. ....	52
<b>Figure 3. 10.</b> Organigramme de la tâche d’étalonnage PID. ....	53
<b>Figure 3. 11.</b> Logo du système d’exploitation Linux ....	54
<b>Figure 3. 12.</b> Logo de langage de programmation Python ....	55
<b>Figure 3. 13.</b> L’exécution du code en spécifiant le nom du fichier. ....	56
<b>Figure 3. 14.</b> Logo de json. ....	56
<b>Figure 3. 15.</b> Stockage des données dans un fichier json. ....	57
<b>Figure 3. 16.</b> Logo du VScode. ....	57
<b>Figure 3. 17.</b> Présentation d’interface de Visual Studio Code. ....	58
<b>Figure 3. 18.</b> L’accès au SSH. ....	59
<b>Figure 3. 19.</b> Enter L’adresse IP. ....	59
<b>Figure 3. 20.</b> La sélection du système d’exploitation. ....	59
<b>Figure 3. 21.</b> Taper le mot de passe. ....	59
<b>Figure 3. 22.</b> Photo du terminal en Vscode du fonctionnement du robot. ....	59
<b>Figure 3. 23.</b> Organigramme de la tâche de communication. ....	61
<b>Figure 3. 24.</b> Le lancement du moniteur série. ....	61
<b>Figure 3. 25.</b> Envoyer le caractère P. ....	62
<b>Figure 3. 26.</b> Envoyer le caractère S. ....	62
<b>Figure 3. 27.</b> Envoyer le caractère F. ....	62
<b>Figure 3. 28.</b> Trouver le chemin. ....	62
<b>Figure 3. 29</b> Evitement d’obstacle. ....	62
<b>Figure 3. 30.</b> Organigramme final du fonctionnement du robot. ....	64

<b>Figure 4. 1.</b> Reconnaissance facile et digital ....	67
<b>Figure 4. 2.</b> Détection de vol par l’IA ....	68
<b>Figure 4. 3.</b> Optimiser la gestion du trafic avec l’IA. ....	68
<b>Figure 4. 4.</b> Caméra de surveillance. ....	69
<b>Figure 4. 5.</b> Processus d’un système de capture d’image et de reconnaissance faciale. ....	70
<b>Figure 4. 6.</b> Exemple de l’acquisition d’image. ....	70
<b>Figure 4. 7.</b> Détection de visage ....	71
<b>Figure 4. 8.</b> Extraction des points de repère faciaux ....	71
<b>Figure 4. 9.</b> Logo de la bibliothèque d’OpenCV ....	73
<b>Figure 4. 10.</b> Le code Python pour capturer une photo à l’aide d’OpenCV. ....	74
<b>Figure 4. 11.</b> La conception globale de notre système. ....	76
<b>Figure 4. 12.</b> La visualisation sur PowerShell. ....	77

<b>Figure 4. 13.</b> Lancement du programme à partir du fichier client.py.....	77
<b>Figure 4. 14.</b> Alimentation de la webcam.....	77
<b>Figure 4. 15.</b> Capturer par la webcam d'un visage inconnu. ....	78
<b>Figure 4. 16.</b> Enregistrement des images .....	78
<b>Figure 4. 17.</b> Détection d'un visage connu avec affichage de son prénom. ....	78
<b>Figure 4. 18.</b> Récupération des images prises.....	79
<b>Figure 4. 19.</b> Photos réels prises par le robot.....	79

## **Résumé :**

Notre mémoire se focalise sur la réalisation d'un robot mobile autonome de surveillance en utilisant la carte Raspberry Pi. Nous avons exploré divers aspects de ce projet, notamment la robotique, la surveillance, le système de mobilité rocker-bogie, ainsi que la reconnaissance faciale.

En utilisant des capteurs tels que le capteur ultrasonique HC-SR04 et une webcam pour la reconnaissance faciale, nous avons pu concevoir un système capable de détecter les obstacles et de reconnaître les visages. Le Raspberry Pi 4 joue un rôle central en envoyant des commandes à l'ESP32 pour activer les capteurs et contrôler les moteurs des roues.

Notre projet offre des possibilités de surveillance en temps réel et de contrôle à distance, tout en mettant l'accent sur l'aspect autonome du robot. L'ensemble du système est alimenté par une batterie lithium pour assurer une mobilité continue. Grâce à cette réalisation, nous avons pu explorer et mettre en pratique les concepts clés de la robotique, et de la surveillance.

**Mots clés :** Robotique, Robot mobile autonome, Surveillance en temps réel, Système de mobilité rocker-bogie, Commande à distance, Carte Raspberry Pi, Capteur ultrasonique HC-SR04, Esp32, Moteurs, Détection des obstacles, Webcam, Reconnaissance faciale, Batterie lithium.

## **Abstract:**

Our dissertation focuses on the development of an autonomous mobile surveillance robot using the Raspberry Pi board. We have explored various aspects of this project, including robotics, surveillance, the rocker-bogie mobility system, and facial recognition. By utilizing sensors such as the HC-SR04 ultrasonic sensor and a webcam for facial recognition, we have designed a system capable of detecting obstacles and recognizing faces. The Raspberry Pi 4 plays a central role by sending commands to the ESP32 to activate the sensors and control the wheel motors. Our project offers real-time surveillance capabilities and remote control, while emphasizing the autonomous aspect of the robot. A lithium battery to ensure continuous mobility powers our system. Through this accomplishment, we have been able to explore and apply key concepts in robotics and surveillance.

Keywords: Robotics, Autonomous Mobile Robot, Real-time Surveillance, Rocker-Bogie Mobility System, Remote Control, Raspberry Pi Board, HC-SR04 Ultrasonic Sensor, ESP32, Motors, Obstacle Detection, Webcam, Facial Recognition, Lithium Battery.

## ملخص:

تركز أطروحتنا على تصميم روبوت مراقبة متنقل مستقل باستخدام لوحة Raspberry Pi. لقد استكشفنا جوانب مختلفة من هذا المشروع، بما في ذلك الروبوتات، المراقبة ونظام التنقل Rocker-bogie، بالإضافة إلى تكنولوجيا التعرف على الوجوه.

بالاعتماد على مستشعرات خاصة مثل لاقط الموجات فوق الصوتية HC-SR04 وكاميرا صغيرة لالتقاط الصور، تمكنا من تصميم نظام قادر على اكتشاف العوائق والتعرف على الوجوه. يلعب Raspberry Pi 4 دورًا محوريًا في عمل الروبوت حيث يقوم هذا الأخير بإرسال أوامر إلى ESP32 لتنشيط المستشعرات والتحكم في محركات العجلات.

يوفر مشروعنا إمكانية المراقبة والتحكم الآني عن بعد، مع التأكيد على كون الروبوت مستقلاً. يتم تشغيل النظام بأكمله بواسطة بطارية ليثيوم لضمان الحركة المستمرة. بفضل هذا العمل، تمكنا من استكشاف وتطبيق المفاهيم الأساسية للروبوتات والمراقبة.

**الكلمات المفتاحية:** الروبوتات المتنقلة المستقلة، المراقبة الآنية، نظام التنقل Rocker-bogie، التحكم عن بعد، لوحة Raspberry Pi، مستشعر الموجات فوق الصوتية HC-SR04، Esp32، المحركات، كشف العوائق، كاميرا، التعرف على الوجوه، بطارية ليثيوم.

# Glossaire

**ADC:** Analog-to-Digital Converterter  
**BDD:** Base de Données  
**BLE:** Bluetooth Low Energy  
**CMD:** Command  
**CSI:** Camera Serial Interface  
**DAC:** Digital-to-Analog Converterter  
**DC:** Direct Current  
**EN:** Enable  
**GND:** Ground  
**GPIO:** General Purpose Input/Output  
**GPU:** Graphics Processing Unit  
**HDMI:** High-Definition Multimedia Interface  
**HOG:** Histogram of Oriented Gradients  
**I2C:** Inter-Integrated Circuit  
**IHM:** Interface Homme Machine  
**IP:** Internet Protocol  
**JPEG:** Joint Photographic Experts Group  
**JSON:** JavaScript Object Notation  
**LED:** Light-Emitting Diode  
**Li-ion:** Lithium-Ion  
**LPDDR4:** Low-Power Double Data Rate 4  
**Perfboard:** Perforated Board  
**PID:** Proportional-Integral-Derivative  
**PNG:** Portable Network Graphics  
**PWM:** Pulse Width Modulation  
**PWR:** Power  
**RAM:** Random Access Memory

**RGB:** Red Green Blue  
**RX:** Receiver  
**SAR:** Successive Approximation Register  
**SD:** Secure Digital  
**SOC:** System-on-a-Chip  
**SPI:** Serial Peripheral Interface  
**SRAM:** Static Random Access Memory  
**SSH:** Secure Shell  
**SVM:** Support Vector Machine  
**TX:** Transmitter  
**UART:** Universal Asynchronous Receiver-Transmitter  
**USB:** Universal Serial Bus  
**VCC:** Voltage Common Collector  
**WIFI:** Wireless Fidelity  
**WLAN:** Wireless Local Area Network

# **Introduction Générale**

### Introduction générale

La robotique est un domaine en pleine expansion qui présente un potentiel énorme dans de nombreux domaines de la science et de la société. Au fil des années, de nombreuses avancées ont été réalisées dans ce domaine, avec des universités et des chercheurs du monde entier qui explorent de nouvelles frontières pour repousser les limites de ce que les robots peuvent accomplir. Ces machines autonomes programmables ont le potentiel de transformer nos vies de diverses manières, que ce soit dans l'industrie, la santé, l'exploration spatiale, ou encore la surveillance.

Dans ce mémoire, nous développons les concepts clés de la robotique, en mettant l'accent sur la conception et la réalisation d'un robot mobile autonome. Nous abordons également les aspects liés à la surveillance, à la détection des mouvements et à la communication sans fil. Notre objectif est de développer des technologies innovantes en matière de vision par ordinateur, de contrôle de mouvement et de communication pour fournir une surveillance en temps réel.

Une des composantes clés de notre robot est le système Rocker-Bogie, une conception développée par la NASA pour ses Rovers martiens. Ce système de mobilité, qui consiste en six roues disposées de manière spécifique, permet au robot de franchir des obstacles de grande taille tout en maintenant sa stabilité. Cette conception offre une excellente maniabilité et une répartition uniforme du poids sur les roues, ce qui est essentiel pour les déplacements sur des surfaces difficiles.

Le Raspberry Pi 4 joue un rôle central dans notre système en tant qu'unité de commande principale. Il envoie des ordres à l'ESP32, qui agit comme un relais pour activer les capteurs et les moteurs du robot. Cette communication entre le Raspberry Pi 4 et l'ESP32 permet de coordonner les différentes actions du robot.

L'ESP32, quant à lui, reçoit les ordres du Raspberry Pi 4 et génère des signaux de commande. Il communique notamment avec le MPU6050, un capteur d'orientation et de mouvement, pour obtenir des données précises sur l'orientation et la vitesse angulaire du robot. Ces informations sont essentielles pour assurer un mouvement précis et améliorer le contrôle du robot.

Les moteurs DC du robot sont commandés par des drivers L298N, qui sont activés par l'ESP32. Ces drivers permettent de contrôler la rotation des roues du robot, ce qui lui confère une capacité de déplacement autonome.



## Introduction générale

---

En outre, notre robot est équipé d'un capteur ultrasonique HC-SR04, qui est utilisé pour estimer la distance entre le robot et les obstacles environnants. Ce capteur permet au robot de détecter les obstacles et de les éviter en ajustant sa trajectoire.

L'ensemble de robot est alimenté avec une batterie Li-ion qui fournissent environ 11,1V rechargeable tandis que l'unité de commande est alimentée par une autre batterie de 5V.

En plus de ces fonctionnalités, la reconnaissance des objets joue un rôle crucial. Elle permettra à notre robot de détecter et d'analyser les objets présents dans son champ de vision, afin de prendre des décisions appropriées en fonction de la situation. Nous avons exploré différentes bibliothèques et techniques de reconnaissance des objets pour développer cette fonctionnalité.

Ce manuscrit est divisé en quatre chapitres qui portent sur les aspects suivants :

Le premier chapitre présente des généralités sur la robotique : définition, historique, classification...etc.

Le second chapitre est consacré à la description de la conception matérielle utilisé dans notre projet, en décrivant tous les composants utilisés pour la réalisation du robot et son unité de commande. Nous abordons aussi l'étude de l'architecture des robots rocker-bogie à mécanisme double lambda.

Dans le troisième chapitre, on passe à la réalisation matérielle et logicielle de notre robot en présentent les différents organigrammes du code a implémenté.

Dans le quatrième et dernier chapitre est consacré pour le système de surveillance visuelle ou on parle sur les différentes étapes de la reconnaissance faciale ainsi les bibliothèques utilisées avec des tests et des résultats réels sur le projet.

Enfin, nous concluons notre travail avec une conclusion générale et quelques perspectives envisageables dans un futur proche ou plus lointain.

# **Chapitre I**

## **Généralités sur la Robotique mobile**

## I.1.Introduction

Depuis les années 70, des chercheurs et des scientifiques se sont intéressés à la construction d'un nouveau type de machine appelée robot en raison de la variété des tâches ennuyeuses, répétitives, difficiles, voire dangereuses. La machine est un mécanisme électronique, automatique et informatisé. Qui est capable d'effectuer diverses tâches sur commande ou selon des instructions préprogrammées [1].

Ces dernières années, la robotique a eu un impact considérable sur de nombreux aspects de la vie moderne et peut être divisée en deux catégories : les robots industriels et les robots mobiles.

Les robots industriels sont généralement stationnaires et ils sont utilisés dans de nombreuses applications industrielles : assemblage mécanique, soudure, peinture... Les robots mobiles ne sont pas fixes, ils sont divisés en robots marcheurs, à roues, à chenilles selon le mode de déplacement... comme ils peuvent être classés selon les domaines d'application en robots militaires, de laboratoire, industriels et de service.

Les robots mobiles sont largement utilisés dans les environnements industriels, le plus souvent pour des tâches répétitives en suivant un chemin bien défini matérialisé parfois par des lignes sur le sol ou par l'utilisation d'amers artificiels. Cependant actuellement il y'a une forte tendance à élargir les milieux où évoluent les robots à des environnements domestiques. Les types d'applications possibles sont innombrables. Cela peut aller des tâches de nettoyage et d'entretien, à une assistance à une personne handicapée dans des tâches d'exploration et de préhension. On parle alors, de façon générale, de robotique d'intérieur.

Un tel cadre d'utilisation requiert que le système robotisé dispose d'un niveau minimum d'autonomie et de facilités de navigation. Pour cela, le système doit obligatoirement accomplir trois tâches de base qui sont la localisation, la planification et la navigation. Il faut toutefois noter que les deux dernières tâches sont tributaires de la bonne exécution de la première [1].

Nous présenterons dans ce chapitre un aperçu général de la robotique. Nous évoquerons également l'architecture des robots et leurs classifications. Nous terminons ce chapitre par une présentation des robots mobiles de surveillance.

## I.2. Généralités sur la robotique

### I.2.1. Définition de la robotique

La robotique est la science qui étudie l'ensemble des techniques permettant la conception et la réalisation de machines automatiques ou de robots. C'est un domaine multi disciplinaire ; on y trouve des aspects concernant la mécanique, l'informatique, ainsi que l'électronique [2].

En se basant sur l'Intelligence Artificielle naissante, dans son livre "Run-around" Asimov présente les 3 règles de la Robotique :

- Un robot ne peut porter atteinte à un être humain, ni rester inactif laissant ce dernier exposé au danger.
- Un robot doit obéir aux ordres donnés par les êtres humains, sauf si de tels ordres sont en contradiction avec la Première Loi.
- Un robot doit protéger sa propre existence dans la mesure où cette protection n'est pas en contradiction avec la Première ou la Deuxième Loi de la robotique [3].

### I.2.2. Domaine d'application

La robotique est un domaine en plein essor ces dernières années. Les évolutions technologiques continuent de dépasser nos attentes et des solutions techniques qui s'adaptent au moindre problème sont désormais possibles.

En effet, les applications robotiques peuvent être divisées en trois grandes catégories, à savoir : Robotique d'intervention, robotique de service professionnelle et robotique personnelle.

#### I.2.2.1. La robotique d'intervention :

Cette forme de robotique est généralement utilisée pour remplir des tâches dans des environnements difficiles d'accès ou hostiles (industries nucléaire, intervention sur des catastrophes naturelles, ainsi que l'exploration marine ou spatiale...) [4].



Figure 1. 1. Robot marin [1].

### I.2.2.2. La robotique de service professionnelle :

Cette catégorie est généralement utilisée dans l'industrie professionnelle pour accomplir tâches répétitives et/ou hasardeuses, cette robotique demande une certaine précision et Une qualité inaccessible aux opérateurs humains, d'où sa grande utilité [5].



Figure 1. 2. Robot Chirurgicale [2].

### I.2.2.3. La robotique personnelle :

Son utilisation est capable de répondre à un grand nombre de besoins telle la réalisation de tâches domestiques, ainsi que l'assistance aux personnes en perte d'autonomie...etc [5].



Figure 1. 3. Robot nettoyeur [2].

## I.3. Généralités sur les robots

### I.3.1. Etymologie du terme robot

Le terme « robot » découle de langues slaves (Tchèque) et signifie textuellement « esclave » ou bien (Robota) qui signifie travail forcé.

La définition précise d'un robot, et en contrepoint de la robotique, est complexe. Assemblage complexe de pièces mécaniques et de pièces électroniques, le tout piloté par une intelligence artificielle [6].

### I.3.2. Définition

Un robot est un système mécanique composé de corps mobiles reliés par des actionneurs qui lui donnent des capacités de mouvement dans l'espace physique.

Les structures mécaniques utilisés sont de types très divers, qu'il s'agisse de robots manipulateurs constitués d'un bras terminé par un outil ou un organe, ou encore de robots mobile avec des principes de locomotion [7].

### I.3.3. Architecture d'un robot

L'architecture des robots mobiles basé sur les quatre éléments suivants :

- La structure mécanique et la motricité.
- Les organes de sécurité.
- Le système de traitement des informations et gestion des tâches.
- Le système de localisation [8].

La figure suivante représente un schéma synoptique d'un robot en interaction avec son environnement.

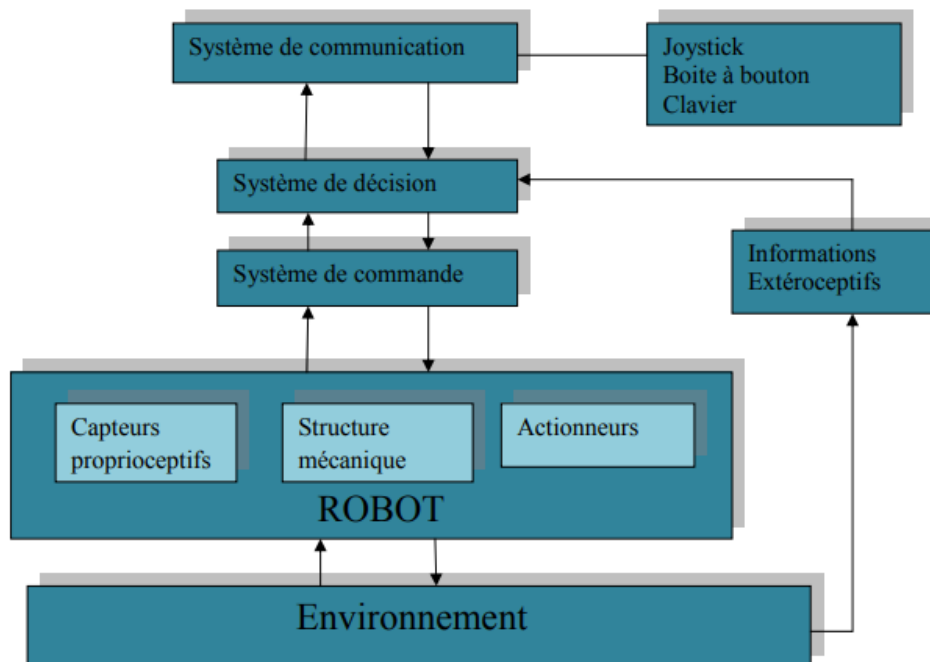


Figure 1. 4. Architecture d'un robot.

### I.3.4. Classification des robots

Les robots sont classés selon plusieurs critères, mais principalement selon trois Critères :

- Système de locomotion.
- Degré d'autonomie.
- Domaine d'application [9].

#### I.3.4.1. Classification selon le degré d'autonomie

Les robots autonomes sont des systèmes dotés de capacités décisionnelles et de moyens de collecte et de traitement de données.

Les robots mobiles peuvent être classés selon les catégories suivantes :

##### a. Robot télécommandé :

Ce sont des robots contrôlés par un opérateur (machine ou humain) qui instruit chacun La tâche de base à faire (avancer, reculer, tourner à droite, etc.) [9].

##### b. Robot semi-autonome :

Ce type de robot effectuent un certain nombre de tâche par eux même d'une façon complètement autonome mais peuvent être interrompu pour recevoir des commandes de control par un opérateur[9].

##### c. Robot autonome :

Un robot est considéré comme autonome s'il peut adapter son comportement à son environnement.

#### I.3.4.2. Classification selon le domaine d'application

L'un des plus grands avantages des robots mobiles est les champs d'application illimités,

Voici quelque domaine d'application :

**a. Les robots industriels et de service :**

Il existe des robots mobiles pour des applications industrielles. Celles-ci concernent principalement le transport et la distribution (usines, mines, hôpitaux, ateliers). Les robots de service sont conçus pour aider les personnes en situation d'handicap moteur...etc. [10]

**b. Les robots militaires :**

Il existe de nombreuses applications militaires pour la robotique mobile. Ce périmètre est Intérêt à fournir des spécifications serrées pour la vitesse du véhicule, la capacité à franchir les obstacles (la robustesse est très importante pour ce type de robot) [10].

**c. Les robots de laboratoires :**

Pour valider les recherches théoriques sur la perception ou la planification motrice, de nombreux chercheurs Laboratoires travaillant dans le domaine de la robotique. Tous les robots sont à l'origine Laboratoire né de la recherche dans ce domaine [11].

**d. Les robots de surveillances et de sécurités :**

Les robots de sécurité se déplacent automatiquement le long des itinéraires de patrouille, détectent les étrangers et transmettent les messages aux agents de sécurité. Les opérateurs derrière les écrans de contrôle appartiennent au passé, les robots eux-mêmes détectent les intrusions et les gardes n'ont qu'à affronter physiquement les intrus potentiels.

Ainsi que Les charges utiles des robots de surveillance sont utilisées pour cartographier ou rechercher des zones, et les charges utiles des robots de sécurité sont utilisées pour localiser des actifs tels que des propriétés c'est comme comparer le comportement de ces deux types de robots revient à comparer un problème global à un problème local de surveillance et de certitude [12].

**I.3.4.3. Classification selon le type de locomotion****I.3.4.3.1. Les robots mobiles à chenilles :**

Les robots mobiles à chenilles ont l'avantage d'une bonne traction et d'une capacité à franchir les obstacles. Son utilisation est destinée à une utilisation sur des terrains irréguliers ou des zones à faible adhérence (présence de boue, d'herbe, etc.).



En général, à des fins militaires, ils sont principalement utilisés comme robots de surveillance ou engins de déminage [12].



**Figure 1. 5.** Robot à chenille qui passe partout [3].

#### **I.3.4.3.2. Les robots mobiles marcheurs :**

Les robots mobiles marcheurs sont conçus pour effectuer une variété de tâches difficiles d'accès, dangereuses ou impossibles pour l'homme.

L'anatomie flexible permet le rapprochement des manipulateurs robotiques. Le mouvement est contrôlé sous la forme de coordonnées articulaires. Un robot bipède oscillant entièrement autonome [11].



**Figure 1. 6.** Le robot réalisé par Boston Dynamics PETMAN [4].

#### I.3.4.3.3. Les robots mobiles rampants :

La reptation est une solution de locomotion aux environnements tunnelisés, entraînant la création de structures filiformes (voir la Figure 1.7). Le système se compose d'un certain nombre de modules, chacun avec plusieurs mobilités, encore une fois, les techniques utilisées sont dérivées de la façon dont les serpents se déplacent [12].

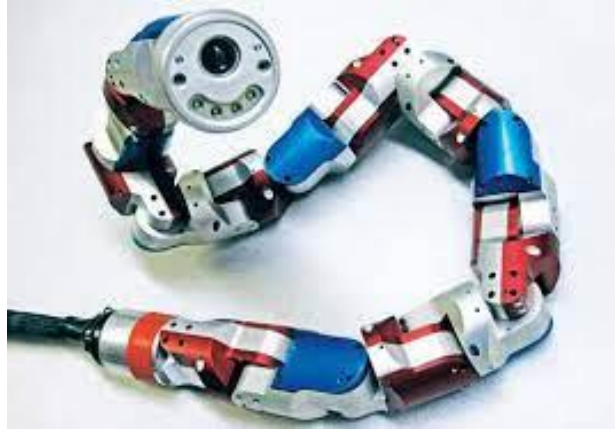


Figure 1. 7. Robot rampant développé par l'université Carnegie Mellon [4].

#### I.3.4.3.4. Les robots mobiles à roues :

La mobilité à roues est la construction mécanique la plus couramment utilisée. Ce type de robot est facile à déplacer, mais nécessite un sol relativement plat.

Il existe plusieurs classes de robots à roues, principalement déterminés par l'emplacement et le nombre de roues utilisées [13].

##### a. Robots uni-cycles :

Les robots monocycles sont propulsés par une ou deux roues indépendantes pour assurer la stabilité à l'aide de capteurs de position (gyroscope) et de déplacement (accéléromètre). Son point d'articulation est sur l'axe reliant les deux roues motrices [13].

Un schéma du robot monocycle est illustré à la figure 1.8 :

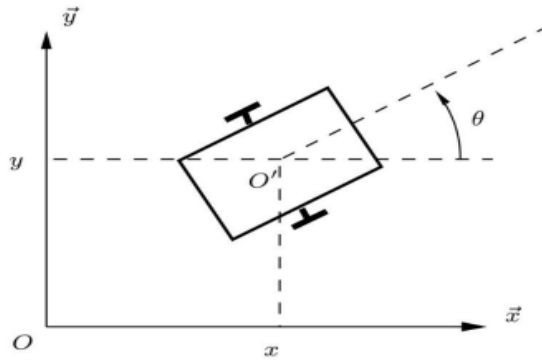


Figure 1. 8. Robot mobile de type uni-Cycle [3].

### b. Robots tricyles :

Un robot à trois roues est composé de deux roues fixes sur le même axe et d'une roue centrale orientable sur l'axe longitudinal (voir la Figure 1.9).

Le mouvement du robot est déterminé par la vitesse des deux roues fixes et l'orientation des roues directrices.

C'est un robot non-holonyme. En effet, il est impossible de le déplacer dans une direction perpendiculaire aux roues fixes [13].

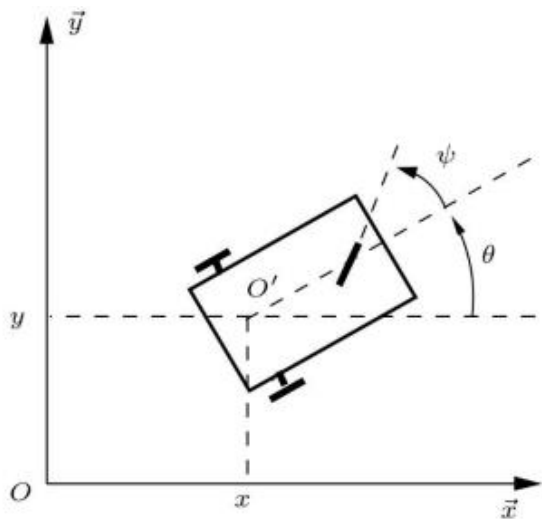


Figure 1. 9. Robot mobile de type tricycle [3].

### c. Robots voitures :

Un robot automobile ressemble à un tricycle, composé de deux roues fixes sur le même axe et de deux roues centrales orientables placées aussi sur le même axe (voir Figure 1.10). Cependant, le robot automobile a un point de pivot supplémentaire, ce qui le rend plus stable [13].

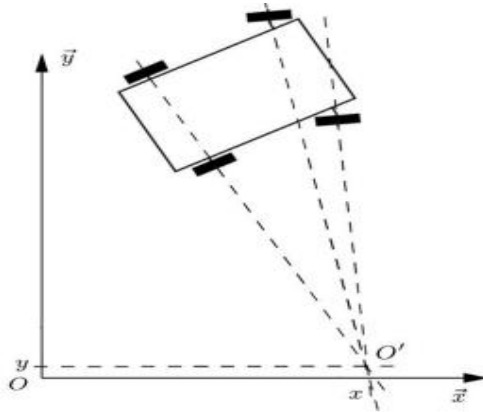


Figure 1. 10. Robot mobile de type voiture [3].

### d. Robots mobiles omnidirectionnels :

Un robot mobile est dit omnidirectionnel s'il peut répondre indépendamment à la vitesse (vitesse de translation le long des axes  $x$  et  $y$ , et vitesse de rotation autour de l'axe  $z$ ) (voir Figure 1.11). Libre de se déplacer dans n'importe quelle direction. Il se compose généralement de trois roues déportées orientables disposées en triangle équilatéral [13].

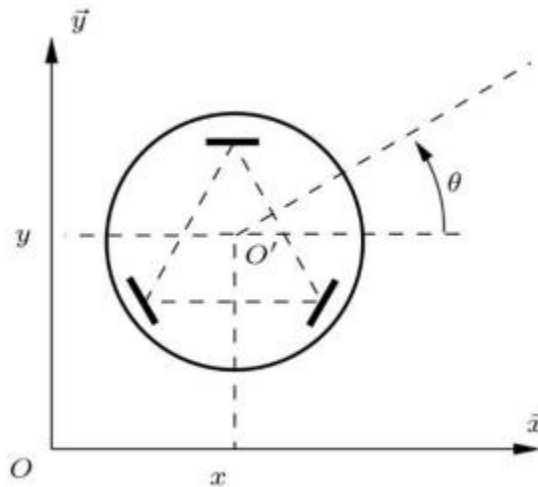
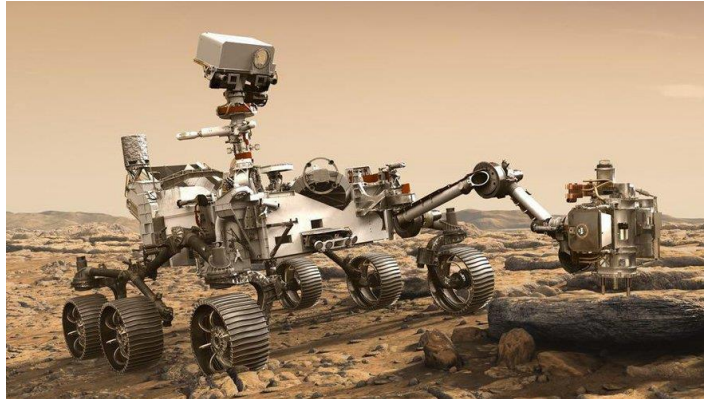


Figure 1. 11. Robot de type omnidirectionnels [3].

### e. Robots explorateurs :

De manière générale, cette catégorie comprend une sous-catégorie de robots mobiles à roues. Ceux-ci sont généralement manipulés à des fins d'exploration. Ils sont appelés rovers.

Par exemple : Mars Science Laboratory (MSL) est une mission d'exploration de la planète Mars [14].



**Figure 1. 12.** Le Rover Curiosity sur la planète Mars envoyé par la Nasa [4].

## I.4. Robots mobiles de surveillances

Un robot mobile de surveillance est un système autonome conçu pour patrouiller, surveiller et collecter des informations dans un environnement donné. Il combine des capacités de déplacement autonome avec des fonctionnalités de surveillance et de communication pour effectuer des tâches de surveillance dans divers contextes, tels que la sécurité, la surveillance industrielle, la surveillance environnementale, etc.

Ces robots de surveillance mobiles sont équipés de capteurs et de caméras qui leur permettent de détecter et de collecter des données sur leur environnement, et utilisent généralement des algorithmes avancés de traitement de données et d'intelligence artificielle pour analyser les informations collectées en temps réel [14].

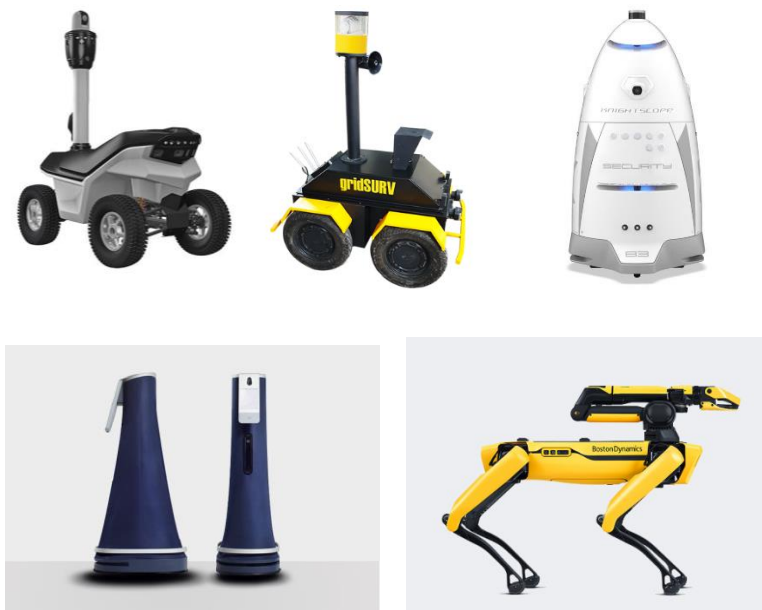
### I.4.1. Robots mobiles de surveillances commercialisés

Les robots commercialisés ont souvent été développés par des entreprises et ont passé des tests approfondis pour s'assurer de leur fiabilité, de leur sécurité et de leur performance. Ils sont conçus

pour répondre à des besoins spécifiques de l'utilisateur, tels que la surveillance de la sécurité, l'inspection de l'industrie, la logistique et le transport, entre autres.

Il existe plusieurs modèles commercialement disponibles de robots mobiles de surveillance. Voici quelques exemples populaires :

- **Knightscope K5** : Le Knightscope K5 est un robot de sécurité utilisé pour patrouiller et surveiller les espaces extérieurs. Il est équipé de caméras à 360 degrés, de capteurs de mouvement, de détecteurs de gaz, de systèmes de reconnaissance de plaque d'immatriculation et de capacités de communication bidirectionnelle.
- **Boston Dynamics Spot** : Spot est un robot quadrupède polyvalent développé par Boston Dynamics. Il peut être utilisé pour diverses applications de surveillance, y compris l'inspection industrielle, la surveillance de chantier, la collecte de données environnementales, etc. Spot est équipé de caméras et de capteurs pour naviguer dans différents environnements et collecter des informations.
- **Cobalt Robotics** : Cobalt Robotics propose une solution de robot de sécurité autonome pour la surveillance intérieure. Leur robot utilise des caméras haute définition, des capteurs de mouvement et des capacités de détection d'objet pour patrouiller dans les espaces intérieurs, détecter les activités suspectes et alerter le personnel de sécurité [12].



**Figure 1. 13.** Quelques exemples de robots mobiles de surveillance commercialisés [5].

### I.4.2. Robots mobiles de surveillances dans la littérature et la recherche

Les robots présentés dans la littérature sont souvent des prototypes ou des concepts développés dans le cadre de projets de recherche. Ils ont souvent été créés pour explorer de nouvelles idées et concepts technologiques, pour tester des algorithmes ou des systèmes de contrôle, ou pour démontrer la faisabilité d'une application. Ils sont généralement moins robustes que les robots commercialisés et n'ont pas été testés pour une utilisation à grande échelle. Cependant, les robots présentés dans la littérature peuvent être une source d'inspiration pour de futurs développements de robots commerciaux.

Ci-dessous quelques exemples de modèles de robots mobiles de surveillance qui ont été décrits dans la littérature et la recherche :

- **Robot de surveillance en milieu hospitalier** : un robot développé par des chercheurs de l'Université de Tokyo pour patrouiller et surveiller les couloirs des hôpitaux [15].
- **Robot de surveillance pour la sécurité incendie** : un robot équipé de capteurs pour détecter les incendies et les fumées développé par des chercheurs de l'université Carnegie Mellon [16].
- **Robot de surveillance pour la détection de mouvements suspects** : un robot équipé de caméras et de capteurs pour détecter les mouvements suspects dans les zones publiques développé par des chercheurs de l'Université de Stanford [17].

Ces exemples représentent une petite fraction des modèles de robots mobiles de surveillance décrits dans la littérature et la recherche. Les chercheurs et les universitaires du domaine de la robotique travaillent constamment sur de nouveaux concepts et modèles pour améliorer les capacités de surveillance des robots dans différents contextes et applications.

## I.7. Conclusion

Ce chapitre s'attache à résumer tous les concepts de base nécessaires à la compréhension dans le domaine de la robotique mobile. Dans ce chapitre, nous avons également abordé l'architecture des robots mobiles et exploré différentes classes selon différents critères. Nous avons également examiné les modèles vendus et développés dans la littérature.

Maintenant que nous avons appris la terminologie de base de la robotique mobile, passons à la phase de conception matérielle de notre propre robot de surveillance mobile dans le chapitre suivant.

# **Chapitre II**

## **Conception matérielle du Robot à réaliser**



### II.1. Introduction

Après avoir focaliser dans le chapitre précédant sur la robotique en générale et sur les robots mobiles de surveillance en particulier, nous allons présenter alors dans ce chapitre la conception matérielle qui s’articule essentiellement autour de la carte Raspberry PI ainsi qu’un microcontrôleur esp32 ces deux composants permettant le contrôle et la communication avec les différents capteurs et actionneurs. Ensemble, ils constituent le cerveau du robot, lui permettant d’interagir avec son environnement et de réaliser des tâches de surveillance autonomes.

Dans un premier temps nous décrivons le fonctionnement global de notre robot soutenu par un schéma synoptique ensuite nous décrivons l’architecture et le fonctionnement du matérielles utilisés.

### II.2. Schéma synoptique et fonctionnement du système

La figure 2.1 représente le schéma synoptique de notre robot de surveillance à réaliser.

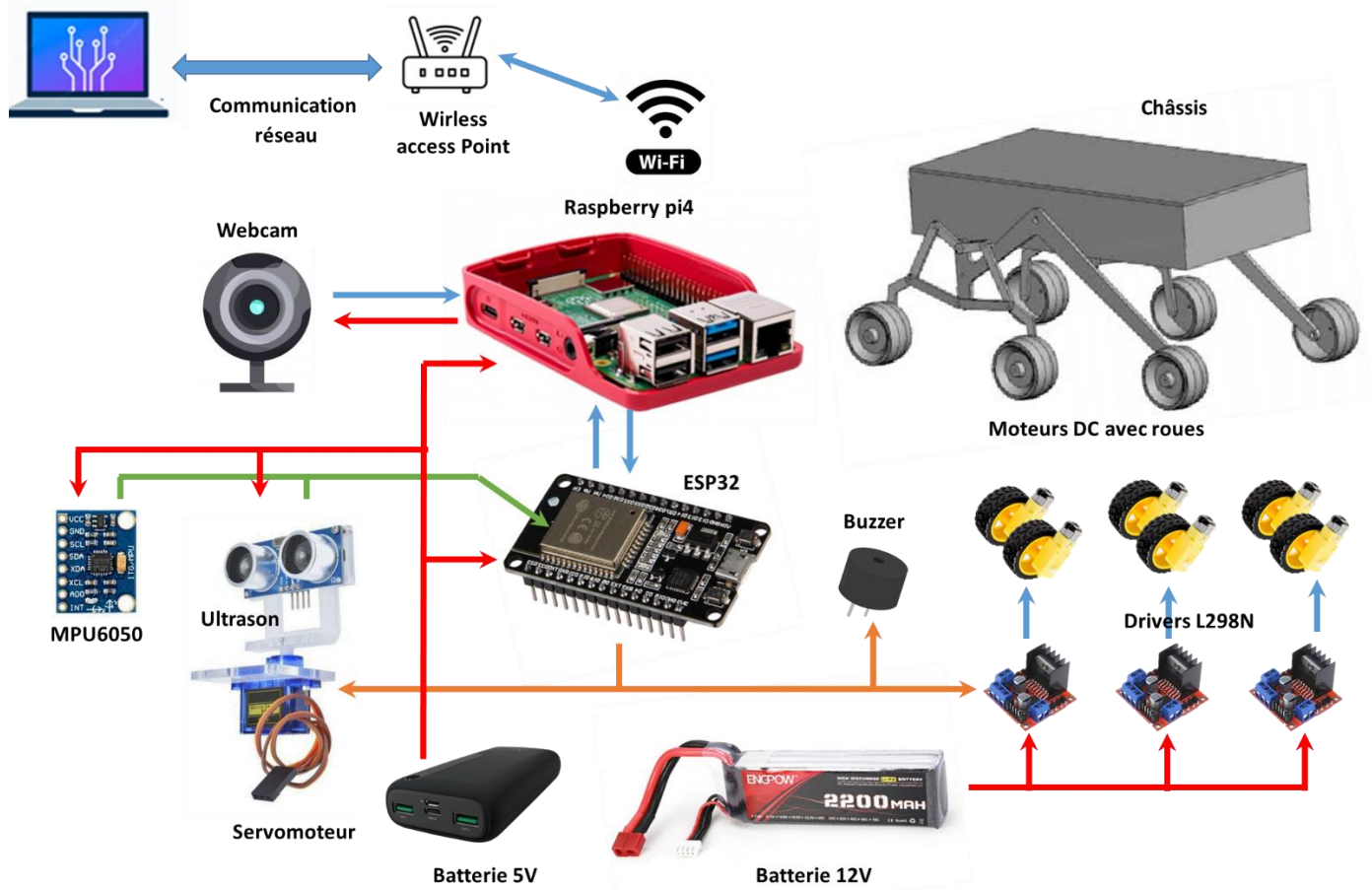


Figure 2. 1. Schéma synoptique du fonctionnement du système.

L'objectif principal de notre réalisation est d'améliorer la sécurité en surveillant les zones à risque et en détectent les activités suspectes par exemple : la surveillance d'une zone industrielle, ainsi notre robot peut fournir une surveillance en temps réel par le développement des technologies innovantes en matière de vision par ordinateur de contrôle de mouvement et de communication sans fil.

La tâche spécifique de notre robot dépendra de l'application pour laquelle il est conçu qui est la surveillance c'est pour cela on peut décomposer le robot en plusieurs unités :

**1. Partie Mécanique** : cela comprend la structure physique du robot tel que :

- Le Châssis : joue un rôle majeur dans la conception d'un robot car il abrite tous les composants électroniques et mécaniques, et il est responsable de la mobilité du robot. Pour notre cas nous avons opté pour la structure Rocker-Bogie a mécanisme double Lambda.
- Les actionneurs et les moteurs-roues : Les moteurs-roues sont responsables de la propulsion du robot et son mouvement dans différentes directions. Les actionneurs, tels que les servomoteurs, permettent le contrôle précis des mouvements des parties mobiles du robot. Notre robot sera équipé des roues actionnées par des moteurs à courant continu. Ces moteurs sont contrôlés par des driver de Type L298N. Un servomoteur peut orienter le capteur ultrason dans différentes directions. Le robot est équipé aussi d'un Buzzer pour émettre des signaux sonores.

**2. Unité de traitement et de contrôle** : Cette partie est responsable du traitement des données, de la prise de décisions et du contrôle du robot. Elle peut être basée sur des microcontrôleurs ou des ordinateurs embarqués plus puissants. Le robot à réaliser sera commandé par une carte Raspberry PI et un microcontrôleur ESP 32.

**3. Les Capteurs** : Les capteurs sont utilisés pour collecter des données sur l'environnement du robot. Notre robot inclut un capteur de mouvement MPU6050 (un module qui combine un gyroscope et un accéléromètre). Un capteur de distance à ultrasons HC-SR04 est monté sur un servomoteur, permettant de mesurer la distance entre le robot et les objets environnants.

**4. Système de vision** : Un système de vision, généralement composé de caméras, est utilisé pour capturer des images et des vidéos de l'environnement. Ces données visuelles sont traitées pour des tâches telles que la détection d'objets, la reconnaissance faciale, la surveillance en temps réel, etc.

5. **Batteries et alimentation** : Les batteries ou autres sources d'alimentation fournissent l'énergie nécessaire au fonctionnement du robot. Il est essentiel d'avoir une autonomie suffisante pour permettre des opérations prolongées.
6. **Partie Informatique** : cette partie comprend les logiciels et les algorithmes qui permettent au robot de prendre des décisions, de communiquer avec d'autres dispositifs, de collecter et de traiter des données. Cette partie sera traité en détail dans le troisième chapitre.

D'après la structure globale du robot on peut maintenant détaillés sur le fonctionnement :

Tout d'abord on a le cerveau de notre réalisation c'est bien que le Raspberry Pi 4 qui permet d'obtenir une grande flexibilité sur les plans logiciel et matériel. Concernant le matériel, le Raspberry Pi 4 permet l'utilisation de différents moyens de communication filaires (SPI, I2C, UART, USB, ...) et sans fil (Wi-Fi, Bluetooth). Ainsi, en utilisant un point d'accès Wi-Fi, les deux appareils Raspberry et le PC sont connectés au même réseau local (WLAN), il est possible de connecter les deux via SSH et d'accéder au Raspberry à distance. Ce mode de connexion permettre de contrôler notre robot à distance comme suit :

Le programme implémenté dans notre carte RPi4 et permet d'obtenir des informations sur l'état du robot plus facilement et instantanément.

Le Raspberry pi 4 envoie un ordre a l'esp32 pour qu'il puisse lire les données capteurs, ce dernier génère un signal de commande au MPU6050 via une communication série (I2C) pour demander des mesures spécifiques, telles que la vitesse angulaire et l'orientation, ce dernier envoie des données a l'esp32 qui est quant à lui il envoie un signal de commande vers les drivers L298N qui sont à leurs tours permettent de commander les moteurs DC qui font tourner les roues.

Grace au capteur de distance HC-SR04, le robot est capable de détecter un obstacle dans sa trajectoire, le servomoteur est commandé en permanence par un signal PWM depuis l'ESP32 pour assurer un balayage par le capteur de 180 degrés.

Le robot est équipé d'une caméra (webcam) connectée au Raspberry Pi, pour capturer des images et des vidéos de l'environnement. La caméra permet de donner un retour visuel et n'est pas utilisé pour le contrôle dans notre cas. Les photos prises par la webcam sont envoyées au Raspberry pour le traitement (détection d'objet et reconnaissance faciale).

Le robot est alimenté avec une batterie rechargeable LiPo qui fournis environ 11,1V pour fournir l'énergie nécessaire aux moteurs, et les autres composants. Ce type de batteries offre une

autonomie suffisante pour les opérations prolongées. Le robot utilise également une autre batterie de 5V (power Bank) pour alimenter le Raspberry Pi, et l'ESP32. Cela permet de maintenir le fonctionnement du système même lorsque la batterie principale est déchargée.

### **II.3. Constitution matérielle du robot**

La constitution matérielle de notre robot est un point essentiel de sa conception. Elle comprend les composants et les éléments physiques qui permettent au robot de fonctionner et d'accomplir ses tâches spécifiques.

Dans cette section, nous allons présenter les principaux éléments qui composent notre robot autonome de surveillance.

#### **II.3.1. Partie mécanique et actionneurs**

La partie mécanique et les actionneurs jouent un rôle crucial dans notre projet robotique. Notre conception repose sur le robot Rocker-Bogie a mécanisme double lambda, qui offre une excellente mobilité et stabilité sur différents types de terrains.

En termes de composants mécaniques, nous avons choisi avec soin des roues de qualité adaptées à notre application spécifique. Ces roues sont actionnées par des moteurs à courant continu, qui sont les principaux acteurs de la propulsion du robot. Nous utilisons également des actionneurs, tels que des servomoteurs, pour contrôler les mouvements précis des parties mobiles du robot, tels que l'orientation du capteur ultrasonique.

##### **II.3.1.1. Le châssis**

Dans notre projet, le choix du châssis de type Rocker-Bogie revêt une importance cruciale pour assurer les performances et la fonctionnalité du robot de surveillance autonome. Le châssis joue plusieurs rôles clés dans notre conception : la stabilité, la capacité de franchissement d'obstacles et de maniabilité.

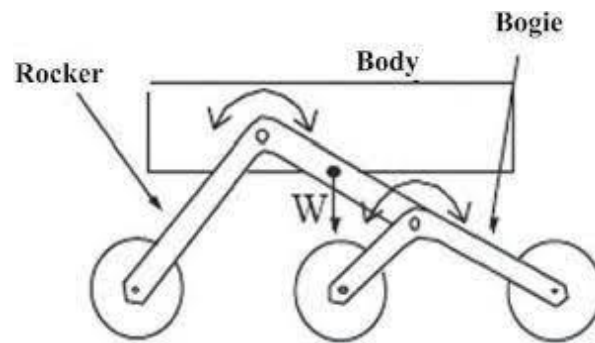
Le robot Rocker-Bogie a mécanisme double lambda permet à notre robot de surmonter les obstacles et de se déplacer en douceur, offrant ainsi une grande flexibilité dans ses mouvements.

Le concept de Rocker-Bogie est un mécanisme de suspension utilisé dans les robots mobiles pour assurer une meilleure mobilité et une capacité de franchissement d'obstacles. Il est caractérisé par l'utilisation de bras de suspension flexibles qui permettent aux roues de s'adapter aux

irrégularités du terrain. Le mécanisme Rocker-Bogie est conçu de telle sorte que lorsqu'une roue rencontre un obstacle, elle se comporte comme un "basculeur" (rocker) en soulevant la roue opposée, ce qui permet de surmonter des obstacles plus grands que le diamètre des roues. De plus, le mécanisme permet au robot de garder au moins trois roues en contact avec le sol en permanence, assurant ainsi une meilleure stabilité et une meilleure répartition du poids [18].

Le terme "rocker" décrit l'aspect de basculement des gros maillons de chaque côté du système de suspension. Il s'agit d'une suspension à 6 roues avec une structure à symétrie bilatérale. Chaque côté a trois roues reliées par deux maillons. La tringlerie principale, appelée culbuteur, a deux articulations, le premier joint est connecté à la roue avant, tandis que l'autre joint est assemblé à une autre liaison appelée bogie. Dans le système, "bogie" fait référence aux liens conjoints qui ont une roue motrice attachée à chaque extrémité.

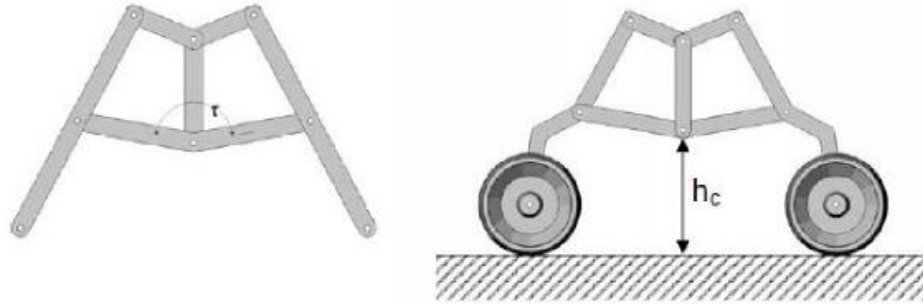
Le principal avantage de la suspension de bogie basculant est que la charge sur chaque roue est presque identique. Sur différentes positions, la force normale des roues se répartit également contrairement aux suspensions souples des 4 roues motrices [16].



**Figure 2. 2.** Schéma global du robot Rocker-Bogie [6].

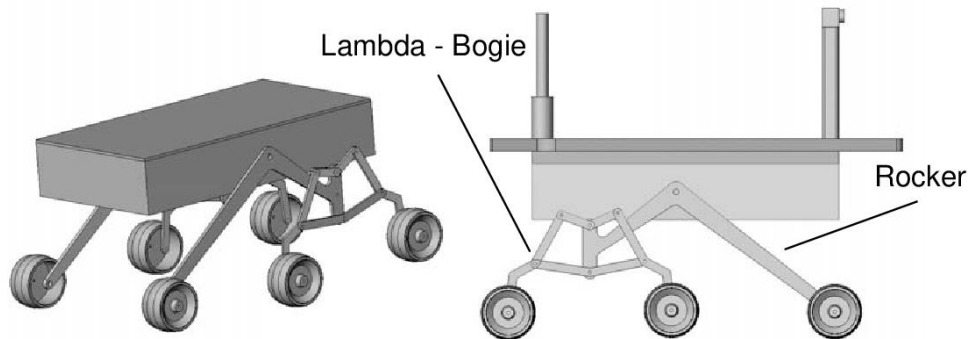
La nouvelle conception du bogie consiste en deux mécanismes lambda qui sont connectés symétriquement. Ainsi, les roues se déplacent en ligne droite mais en sens inverse l'une de l'autre. Cette conception équilibre les forces de réaction sur chaque roue ; par conséquent, la force de traction reste la même pour chaque roue, qu'une roue soit en position haute.

La connexion symétrique de deux mécanismes est un processus critique. Étant donné que les deux côtés du bogie fonctionneront dans une partie linéaire de la courbe, un côté sera à l'opposé de l'autre côté. Lors de la conception de cette connexion, nous devons éviter les configurations singulières du mécanisme.



**Figure 2. 3.** Connexion entre deux mécanismes Lambda [6].

Les mécanismes lambda symétriques sont reliés les uns aux autres par une liaison rigide en forme de V. L'angle  $\tau$  peut être sélectionné géométriquement. La contrainte de cet angle est la garde au sol du bogie ( $h_c=6\text{cm}$ ) et la capacité d'obstacle maximale. Pour nos paramètres, angle de connexion optimal  $\tau = 160$  [16].

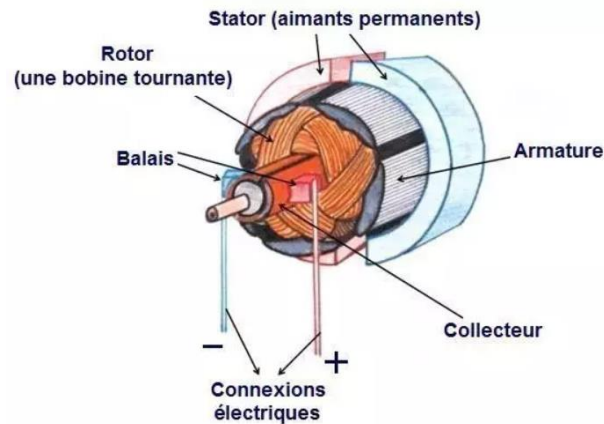


**Figure 2. 4.** Structure globale de robot Rocker-Bogie a mécanisme double Lambda [6].

### II.3.1.2. Moteur à courant continu DC

Les machines à courant continu font partie de la famille des actionneurs (éléments qui produisent une action). Ceux-ci transforment une énergie mécanique en une énergie électrique créant ainsi un courant continu ou transforment une énergie électrique en une énergie mécanique créant un mouvement de rotation donc elles sont réversibles elles peuvent devenir génératrices ou moteur [19].

Un moteur à roues à courant continu (DC) est composé d'une bobine, d'un aimant permanent, d'un commutateur, d'un rotor et éventuellement d'engrenages. Lorsque du courant électrique circule dans la bobine, un champ magnétique est créé, ce qui génère un mouvement rotatif du rotor. Le commutateur assure l'inversion de la polarité de la bobine pour maintenir la rotation [20].



**Figure 2. 5.** Construction d'un moteur à courant continu [7].

### ➤ Principe de fonctionnement:

Le moteur à roues à courant continu fonctionne selon le principe de l'interaction entre un champ magnétique et un courant électrique. Lorsque le courant électrique circule dans la bobine du moteur, un champ magnétique est créé, ce qui génère une force qui fait tourner le rotor. Le commutateur, également appelé collecteur, inverse la direction du courant dans la bobine à des intervalles réguliers, permettant ainsi au rotor de continuer à tourner dans la même direction. Cela permet au moteur de convertir l'énergie électrique en mouvement mécanique rotatif [21].

Pour notre projet nous avons opté pour un moteur à roues à courant continu pour assurer la propulsion et le mouvement du robot. Ces moteurs offrent un contrôle précis et une grande fiabilité, ce qui les rend adaptés à notre application de robot mobile de surveillance.



Figure 2. 6. Moteur-roue DC 3-12V [7].

➤ **Les principales caractéristiques du moteur DC :**

- Tension nominale : 12 V
- Courant nominal : 2 A
- Vitesse à vide : 2000 tr/min
- Couple nominal : 0.5 Nm
- Diamètre de l'arbre : 6 mm
- Poids : 150 g
- Dimensions : 50 mm x 50 mm x 30 mm

### II.3.1.3. Le Driver Moteur L298N

C'est un composant électronique basé sur le composant L298N qui est un double Pont-H (H-Bridge motors Driver), Il contient des éléments électroniques supplémentaires tels que la résistance et le stabilisateur de tension..., qui aident à protéger le pont H contre les dommages résultant d'une tension élevée. Il peut contrôler deux moteurs à courant continu ou un moteur pas-à-pas à 4 fils avec 2 phases. Ce module intègre une entrée de tension. La plage de tensions bascule entre les 3 V et les 35 V, selon la configuration du cavalier régulateur de tension ; et supporte une intensité de 2 A. Avec le cavalier régulateur on peut contrôler des moteurs de 5 à 12 V, aussi on pourra utiliser la broche de 5 V du module pour alimenter la carte ESP32 [22].



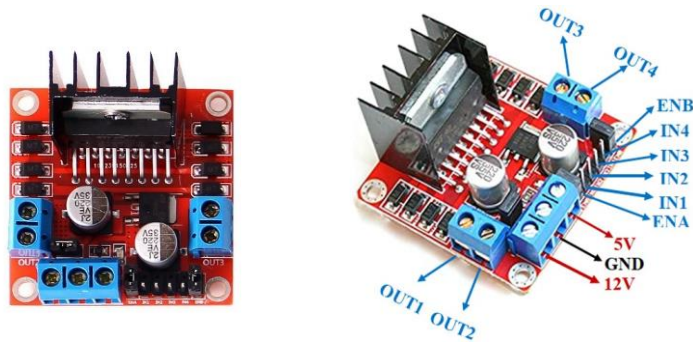


Figure 2. 7. Module Driver Moteur L298N [8].

➤ La commande PWM sur le L298N :

Pour contrôler la vitesse de rotation des moteurs on doit retirer les cavaliers des pins ENA et ENB. On les connecte aux deux sorties PWM de la plaque Arduino de façon qu'on puisse envoyer une valeur entre (0 et 255) V pour gérer la vitesse des moteurs. Avec les cavaliers installés, les moteurs tourneront toujours à la même vitesse [23].

- ENA raccordés à un niveau haut (HIGH) activera MOTOR A
- ENB raccordés à un niveau haut (HIGH) activera MOTOR B.

Carte de contrôle des moteurs (L298N – « double pont en H ») :

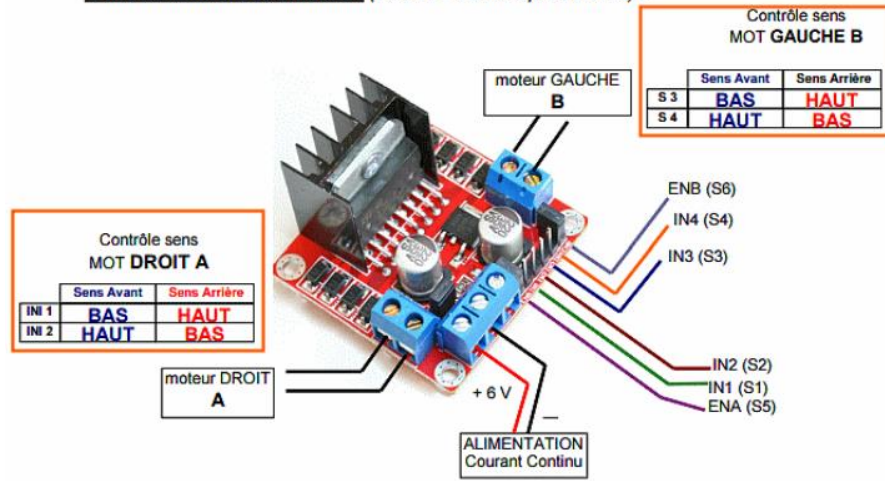


Figure 2. 8. Activation de deux moteurs par le L298 [9].

❖ **Rotation du moteur A**

- IN1 raccordés à 5 V et IN2 à GND MOTORA tournera dans le sens horlogique.
- IN1 raccordés à GND et IN2 à 5V MOTORA tournera dans le sens antihorlogique.

❖ **Rotation du moteur B**

- IN3 raccordés à 5 V et IN4 à GND MOTORB tournera dans le sens horlogique.
- IN3 raccordés à GND et IN4 à 5V MOTORB tournera dans le sens Anti-horlogique.

➤ **Principe de fonctionnement du pont de H :**

Le pont en H est une structure électronique servant à contrôler la polarité aux bornes d'un dipôle. Il est composé de quatre éléments de commutation généralement disposés schématiquement en une forme de **H**, Les commutateurs peuvent être des relais, des transistors, ou autres éléments de commutation en fonction de l'application visée.

Les interrupteurs fonctionnent deux par deux. Le A est associé au D et le B est associé au C. Dans le schéma ci-dessous, rien ne se passe car tous les interrupteurs sont ouverts (ils ne laissent pas passer le courant), donc le moteur est arrêté.

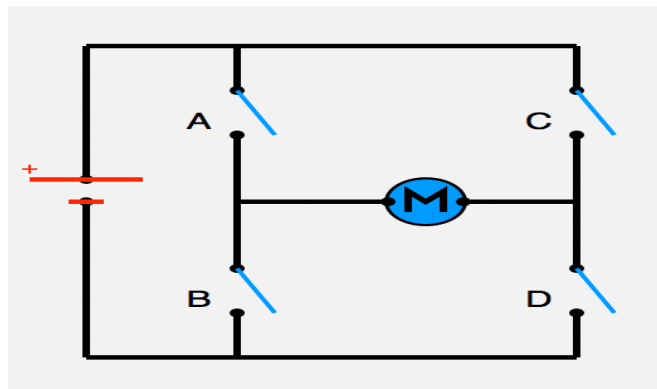


Figure 2. 9. Le pont en H [10].

Voyons maintenant ce qui arrive lorsqu'on actionne en même temps les interrupteurs A et D (schéma de gauche), ou les interrupteurs B et C (schéma de droite) selon (Figure 2.10).

Sur le schéma de gauche, les interrupteurs A et D sont fermés, donc le courant entre par la patte gauche du moteur et sort par sa droite et par conséquent le moteur tourne.

Sur le schéma de droite, les interrupteurs B et C sont fermés, donc le courant entre par la patte droite du moteur et sort par sa gauche ce qui fait tourner le moteur dans le sens inverse [24].

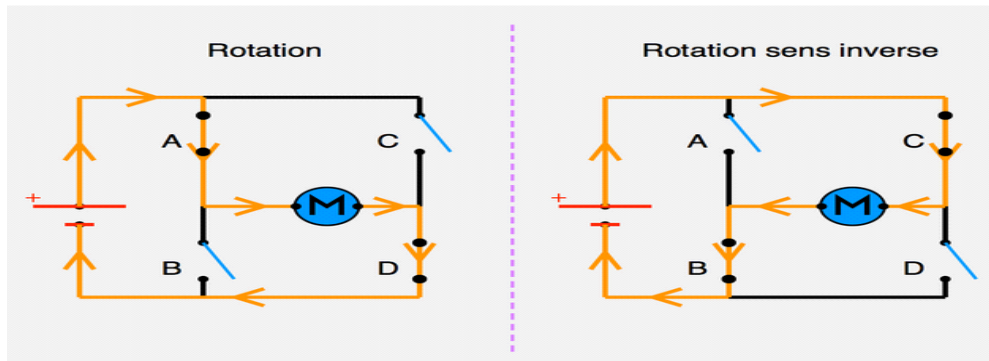


Figure 2. 10. Sens du courant en fonction de l'état des interrupteurs d'un pont en H [11].

#### II.3.1.4. Servomoteur

Un servomoteur est un système qui vise à produire un mouvement précis en réponse à une commande externe C'est un actionneur (système produisant une action). Sur un servomoteur, l'angle de l'axe reste fixé dans une position et peut varier entre (0 et 180°) en fonction du signal envoyé.

Un servomoteur se compose de quatre parties :

- Un moteur à courant continu.
- Un réducteur.
- Un potentiomètre.
- Un circuit de contrôle.

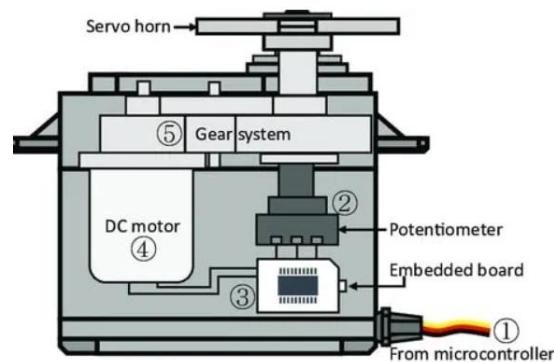


Figure 2. 11. Vu interne d'un servomoteur [12].

Trois fils de connexion sont nécessaires pour faire fonctionner le servomoteur. Deux fils servent pour l'alimentation, et un fil pour recevoir le signal de commande. On reconnaît ces fils par un code de couleur :

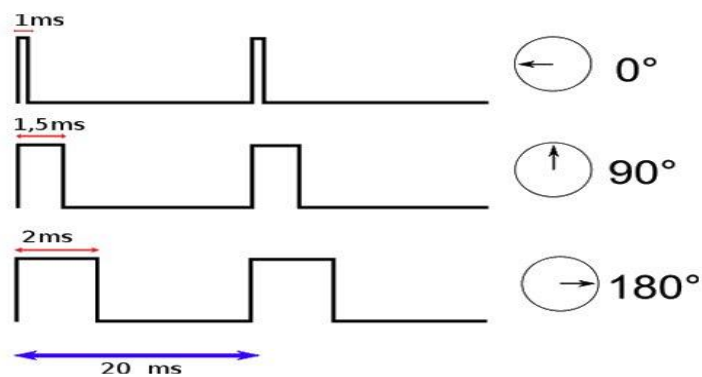
- **Rouge** : pour l'alimentation positive (4.5V à 6V en général).
- **Noir ou Marron** : pour la masse (GND).
- **Orange, jaune ou blanc** : pour le signal de commande [25].



**Figure 2. 12.** Servomoteur 270° [12].

Pour commander un servomoteur, il faut lui envoyer un train d'impulsions dont la période (intervalle de temps entre chaque impulsion) est toujours de 20 ms (millisecondes). Ce qui va varier et qui finalement déterminera la position du bras n'est pas la période, mais bien la durée de l'impulsion [26] :

- Pour une impulsion de 1 ms, le servomoteur se met en position 0°.
- Pour une impulsion de 1,5 ms, le servomoteur se met en position 90°.
- Pour une impulsion de 2 ms, le servomoteur se met en position 180°.



**Figure 2. 13.** Le pilotage d'un servomoteur par des impulsions PWM [12].

### II.3.1.5. Buzzer

Un buzzer est un dispositif électronique qui produit un son ou une tonalité audible lorsqu'il est traversé par un courant électrique. Il s'agit d'un petit composant électromécanique qui vibre rapidement lorsqu'il est traversé par un courant électrique, créant ainsi une onde sonore audible par l'oreille humaine [27].



Figure 2. 14. Un buzzer [13].

### II.3.2. Unité de traitement et de contrôle

L'unité de traitement et de commande joue un rôle essentiel dans notre projet robot. C'est elle qui assure le contrôle et la coordination de toutes les fonctionnalités du robot, permettant ainsi sa navigation autonome et ses capacités de surveillance. Grâce à des cartes électroniques intelligentes telles que l'ESP32 et le Raspberry Pi, nous disposons d'une puissance de calcul et de capacités de traitement avancées.

#### II.3.2.1. Raspberry PI

Dans le cadre de notre projet nous avons opté pour le Raspberry PI 4 comme cerveau de notre système. Le Raspberry Pi est un nano-ordinateur mono-carte basé sur un processeur ARM développé par des professeurs du département d'informatique de l'Université de Cambridge dans le cadre de la Fondation Raspberry Pi.

Cet ordinateur, de la taille d'une carte de crédit, est destiné à encourager l'apprentissage de la programmation informatique ; il permet l'exécution de plusieurs variantes du système d'exploitation y compris Debian (également connue sous le nom de Debian GNU/Linux, est une distribution

Linux composée de logiciels libres et open-source) Mais il fonctionne également avec le système d'exploitation Microsoft Windows : Windows 10 IoT Core sur carte SD.

Avec son petit facteur de forme et sa faible consommation d'énergie, le Raspberry Pi 4 est couramment utilisé pour un large éventail de projets, notamment les centres multimédias, les consoles de jeu rétro, la robotique, la domotique, etc.



Figure 2. 15. Raspberry Pi 4 [14].

➤ **Les caractéristiques du Raspberry Pi 4 :**

- **Processeur :** Broadcom BCM2711, quad-core Cortex-A72 cadencé à 1,5 GHz, ce qui en fait le modèle le plus puissant de la gamme Raspberry Pi.
- **RAM:** 2Go, 4Go ou 8Go (LPPDR4).
- **Ports :** 2 x USB 3.0.  
2 x USB 2.0.  
1 x USB-C (alimentation seulement).  
1 x GPIO 40 pin.  
1 x port quadripôle Audio/Vidéo composite.  
2 x micro-HDMI.
- **Connexion filaire :** Gigabit Ethernet (RJ45).
- **Connexion sans fil :** Bluetooth 5.0, Wi-Fi double bande.
- **Architecture :** ARMv8 64 bits, qui est plus moderne que l'architecture ARMv6 utilisée dans les modèles précédents.
- Lecteur de Carte SD.

- Port caméra CSI pour connecter la caméra Raspberry pi.
- **Alimentation** : 5V DC via un connecteur USB-C (minimum 3A), 5V DC via une en-tête GPIO (minimum 3A), compatible Power over Ethernet (PoE) (nécessite un HAT pour PoE).
- **Compatibilité Vidéo** : la possibilité de gérer jusqu'à **deux écrans en 4K à 60ips [28]**.

➤ **Les broches d'entrée/sorties (GPIO) :**

Un des grands points intéressants du Raspberry pi est qu'il dispose d'un connecteur GPIO sur lequel on peut connecter du matériel externe le model pi B comporte 40 broches (Les broches GPIO sont disposées en deux rangées de 20 broches chacune).

Le GPIO (General purpose Input/Output) du raspberry pi permet d'interagir avec le monde réel en communiquant avec des capteurs, des moteurs et d'autre composants électroniques. Ces 40 broches peuvent être utilisés de manières différentes et se décompose en plusieurs types :

- ❖ **Alimentation (PWR)** : 3.3V ou 5V.
- ❖ **Ground (GND)** : masse 0V.
- ❖ **GPIO** : Les broches GPIO permettent d'entrée/sortie numériques, c'est-à-dire que l'on peut les utiliser pour lire des signaux ou envoyer des signaux numériques à d'autres composants électroniques.
- ❖ **I2C (Inter-Integrated Circuit)**: Le Raspberry Pi dispose de deux broches pour la communication I2C, un protocole de communication série utilisé pour connecter des capteurs et des périphériques externes.
- ❖ **SPI (Serial Peripheral Interface)**: Dispose deux broches pour la communication SPI, permet la communication entre un maitre et un ou plusieurs esclaves.
- ❖ **UART (universal asynchronous receiver / transmitter)**: Dispose de deux broches pour la communication UART, qui permettent la transmission de données série c'est-a-dire la liaison entre l'ordinateur et le port série.
- ❖ **PWM (Pulse width modulation)** : Disponible sur GPIO12, GPIO13, GPIO18, GPIO19. Les broches du GPIO sont alimentés a 3,3V seule les deux broches d'arrivées de courant (en rouge) fournissent une tension de 5v [29].

Les protocoles I2C, Le SPI, L’UART et Le PWM permettent de connecter avec des microcontrôleurs pour réaliser des opérations plus complexes tel que le control du servomoteur, la lecture des mesures prises par des capteurs...etc.

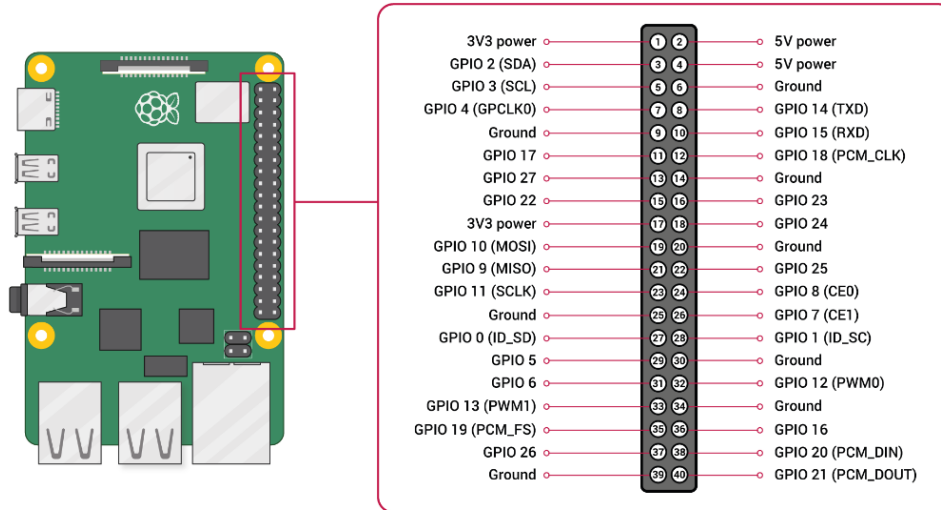


Figure 2. 16. Les différents pins et leurs significations du Raspberry pi [15].

### II.3.2.2. Le Microcontrôleur ESP32

Pour notre projet, nous avons besoin d'un moyen de recevoir les commandes envoyer par le Raspberry c'est pour cela on utilise un ESP32.

La carte ESP32 est développé par la société Espressif avec une série de SoC (System on a Chip), est une carte de développement à faible coût et à faible consommation d'énergie dédié à l'internet des objets (IoT) et les applications embarquées.

Le module ESP32 est une petite carte électronique, appelée microcontrôleur, facile à prendre en main grâce à ses ressemblances avec la carte esp32 qui est bien plus répandue.

L’ESP32 est en ce sens une solution intéressante car elle contient au sein de son architecture un module WiFi intégré. Le deuxième gros avantage de l'ESP32 est la puce Bluetooth intégrée. Il peut également être utilisé en mode "BLE" (Bluetooth Low Energy) et consomme moins d'énergie. L'ESP32 peut également être utilisé comme module de communication en combinaison avec n'importe quel microcontrôleur ou carte de développement en exécutant des programmes écrits dans des langages de programmation tels que C/C++ et MicroPython [30].





Figure 2. 17. Module ESP32 [16].

➤ Les caractéristiques du microcontrôleurs ESP32 :

- **Tension d'alimentation** : 5 Vcc via micro-USB.  
3,3 Vcc via broches Vin.
- **Microprocesseur** : Tensilica LX6 Dual-Core.
- **La Fréquence** : 240MHz.
- **Mémoire SRAM** : 512KB.
- **Mémoire Flash** : 4Mb.
- **E/S Disponible** : 2 x sorties analogiques (DAC) .  
15 x entrées analogiques (ADC) [31].

➤ Présentation des broches de L'ESP32 :

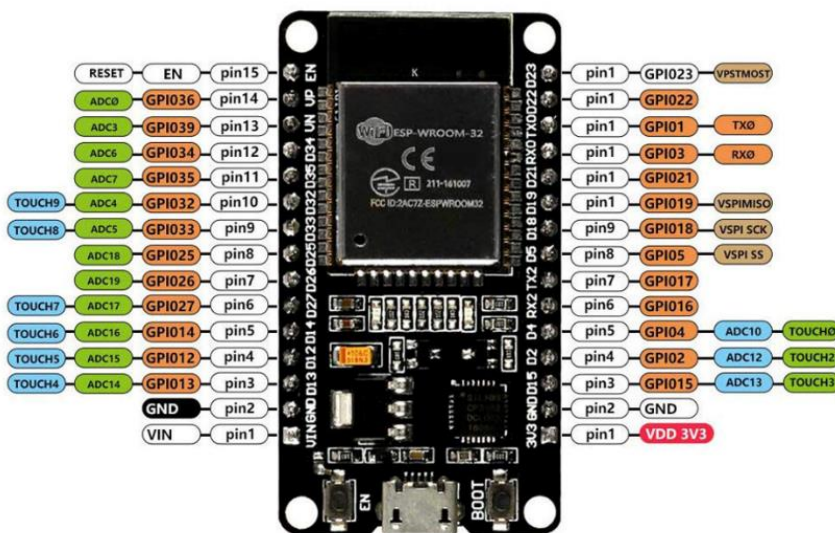


Figure 2. 18. Les pins du microcontrôleurs ESP32 [17].

### II.3.3. Les capteurs

Les capteurs permettent à notre robot de recueillir des informations sur son environnement et d'interagir de manière précise et adaptée. Nous avons intégré une variété de capteurs, tels que le capteur ultrasonique et le capteur MPU6050.

#### II.3.3.1. Capteur Ultrasonique HC-SR04

Un capteur à ultrasons est un dispositif électronique qui mesure la distance d'un objet cible en émettant des ondes sonores ultrasoniques et qui convertit le son réfléchi en un signal électrique. Les capteurs à ultrasons sont composés de deux éléments principaux : l'émetteur (qui émet le son à l'aide de cristaux piézoélectriques) et le récepteur (qui reçoit le son). Les capteurs à ultrasons sont également utilisés dans les systèmes robotiques de détection d'obstacles, ainsi que dans la technologie de fabrication [32].

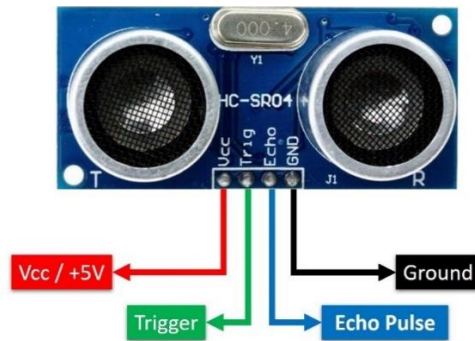


Figure 2. 19. Capteur Ultrason HC-SR04 [18].

- Les caractéristiques du capteur HC-SR04 sont les suivants :
  - Le capteur HC-SR04 est un capteur à bas prix.
  - Sa tension d'alimentation est de 5 volts.
  - L'angle de mesure est de 15°.
  - Sa distance de captation entre 2 cm à 4 m.
  - Sa résolution (précision) est environs a 3mm [33].
- L'Ultrason dispose de quatre (04) broches de connexion sont :

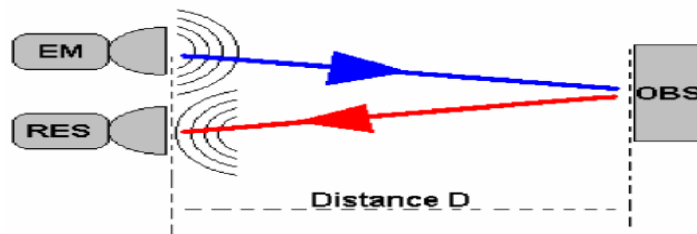
- **VCC, GND** : Alimentation +5 V DC, et Masse d'alimentation
- **Trig** : Entrée de déclenchement de la mesure (Trigger input).
- **Echo** : Sortie de mesure donnée en écho (Echo output).

Le principe de fonctionnement du capteur est entièrement basé sur la vitesse du son. Voilà comment se déroule une prise de mesure :

1. On envoie une impulsion **HIGH** de  $10\mu\text{s}$  sur la broche **TRIGGER** du capteur.
2. Le capteur envoie alors une série de 8 impulsions ultrasoniques à 40KHz (inaudible pour l'être humain).
3. Les ultrasons se propagent dans l'air jusqu'à toucher un obstacle et retournent dans l'autre sens vers le capteur.
4. Le capteur détecte l'écho et clôture la prise de mesure.

Le signal sur la broche **ECHO** du capteur reste à **HIGH** durant les étapes 3 et 4, ce qui permet de mesurer la durée de l'aller-retour des ultrasons et donc de déterminer la distance.

**N.B** : Il y a toujours un silence de durée fixe après l'émission des ultrasons pour éviter de recevoir prématurément un écho en provenance directement du capteur [34].



**Figure 2. 20.** Principe de fonctionnement d'un capteur ultrason [19].

Comme il est décrit précédemment, on utilise ces capteurs pour localiser le robot par rapport à des obstacles statiques ou dynamiques, ce capteur repose sur l'idée de calculer le temps de vol d'un signal ultrasonique depuis son point de départ (Figure 2.). Alors si la distance entre le transducteur et l'obstacle est «  $D$  » donc la distance parcourue par l'onde ultrasonique est «  $2D$  », le temps du parcours est donné par :

$$T = \frac{2D}{V_{son}}$$

Où :

T : le temps entre l'émission et la réception.

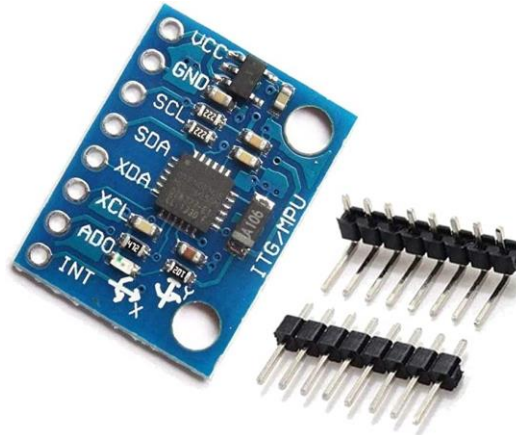
D : distance entre le transducteur et l'obstacle.

V<sub>son</sub> : la vitesse de déplacement des ultrasons dans l'air [35].

### II.3.3.2. Capteur Gyroscopique – accélérométrique

Le capteur gyroscopique est un dispositif capable de mesurer et de conserver l'orientation et la vitesse angulaire d'un objet. Ces capteurs sont plus avancés que les accéléromètres. Ils peuvent mesurer l'inclinaison et l'orientation latérale de l'objet (Robot par exemple).

Lorsque votre robot se retourne, le capteur gyroscopique peut mesurer la vitesse de rotation du robot, en degrés par seconde. Cette vitesse de rotation est mesurée à un taux de 1000 échantillons par seconde. Par conséquent, en multipliant la vitesse par la durée du virage, le capteur gyroscopique peut également indiquer le degré de rotation du robot, en degrés [36].



**Figure 2. 21.** Le module MPU6050 [20].

➤ Les broches de connexions :

- VCC, GND Pour l'alimentation du module MPU6050.
- SDA Pour les données entre le contrôleur et le module MPU6050.
- SCL Pour l'entrée de l'horloge.

- XDA C'est la ligne de données I2C SDA du capteur pour la configuration et la lecture des capteurs externes.
- XCL Il s'agit de la ligne d'horloge I2C SCL du capteur pour la configuration et la lecture des capteurs externes.
- ADO Adresse esclave I2C LSB.
- INT Broche d'interruption pour indiquer que les données sont prêtes.

### II. 3.4. Système de vision

Le système de vision occupe une place centrale dans notre projet, offrant à notre robot la capacité de percevoir et d'interagir avec son environnement, grâce à l'utilisation d'une webcam qui est considéré comme les yeux de notre robot.

#### II.3.4.1. Webcam

Une webcam est une petite caméra connectée à un ordinateur ou à un réseau, par un port USB et un câble. Toutes les webcams fonctionnent globalement de la même manière :

Elles utilisent une puce de capteur d'image qui est le cœur de la webcam pour capturer des images en mouvement et les convertir en flux de chiffres à un intervalle prédéfini (par exemple, une photo est prise toutes les 30 secondes) et la transfère à un autre endroit pour qu'elle puisse être visionnée. Il existe plusieurs méthodes de diffusion de l'image, La plus courante consiste à transformer l'image en un fichier JPEG [37].



**Figure 2. 22.**La puce du capteur d'image [21].



**Figure 2. 23.** Webcam USB [21].

### II.3.5. Batteries d'Alimentation

Dans notre projet, nous avons utilisé une combinaison de deux batteries : une batterie LiPo (lithium-polymère) et d'une power Bank pour l'alimentation de notre système.

#### II.3.5.1. Batterie Lithium-Polymère

Une batterie LiPo (Lithium-Polymer) est un type de batterie rechargeable utilisant une technologie au lithium. Elle se compose de cellules contenant des électrolytes polymères qui facilitent le mouvement des ions entre les électrodes.

Le principe de fonctionnement d'une batterie LiPo repose sur le déplacement des ions lithium entre les électrodes, l'anode et la cathode, pendant le processus de charge et de décharge. Lorsqu'une batterie LiPo est chargée, les ions lithium migrent de l'anode vers la cathode, créant un potentiel électrique. Lorsque la batterie est déchargée, les ions lithium se déplacent de la cathode vers l'anode, libérant de l'énergie électrique [38].



Figure 2. 24. Batterie Lithium-Polymère [22].

Dans notre projet, la batterie LiPo est utilisée comme source d'alimentation pour alimenter différents composants, tels que les moteurs, les capteurs et les microcontrôleurs.

#### II.3.5.2. Power-Bank

Une banque d'alimentation, également appelée batterie externe, est un appareil portable qui peut charger des appareils électroniques tels que des ordinateurs portables lorsqu'une prise électrique n'est pas accessible.



**Figure 2. 25.** Vu de l'intérieur d'une banque d'alimentation [23].

Un power Bank contient une batterie lithium-ion qui stocke l'énergie électrique. La plupart des banques d'alimentation sont équipées d'un port d'entrée micro-USB ou USB-C pour charger la batterie interne et d'un port de sortie USB pour connecter des appareils à charger [39].

## II.4. Conclusion

La conception matérielle est une étape cruciale dans la création d'un système électronique fonctionnel et efficace, cette étape a été abordée en détail dans ce deuxième chapitre. Nous avons commencé par un schéma synoptique suivie d'une explication détaillée sur le fonctionnement du robot à réaliser et enfin en arrivant à la description détaillée des composants utilisés.

Dans le prochain chapitre nous allons entamer la réalisation matérielle et logicielle du robot de surveillance de type rocker-bogie.

# **Chapitre III**

## **Réalisation matérielle et logicielle du Robot**



## III.1. Introduction

La réalisation matérielle et logicielle du robot est une étape cruciale dans le développement d'un projet robotique. Cette phase consiste à passer de la théorie à la pratique, et à mettre en œuvre les différents composants et systèmes qui ont été sélectionnés pour le robot.

Cela implique à la fois une partie matérielle, qui concerne la construction physique du robot, et une partie logicielle, qui consiste la programmation de l'ensemble des comportements et des actions que le robot est capable d'exécuter. Cette phase nécessite une bonne connaissance des différents composants utilisés, des normes de câblage, ainsi que des logiciels et outils de programmation qui seront utilisés pour développer le code source du robot.

Dans ce chapitre, nous aborderons en détail les différentes étapes de la réalisation matérielle et logicielle d'un robot.

## III.2. Réalisation pratique du robot

### III.2.1. Réalisation de la structure mécanique du robot

Avant de commencer à construire notre robot Rocker-bogie basculant, nous devons comprendre sa structure et ses composants. C'est ce que nous avons fait dans le chapitre précédent.

Pour la réalisation finale du corps du robot, le contreplaqué a été choisi en raison de sa disponibilité et de sa facilité de manipulation.

#### III.2.1.1. Conception et dessin de la structure du robot

La conception du robot consiste à utiliser la suspension Rocker-Bogie avec le mécanisme double lambda pour naviguer dans des environnements d'accès difficiles.

Au début, un plan a été dessiné avec des dimensions bien précises. Les dimensions du système sont importantes pour avoir une bonne stabilité mécanique. Ces dimensions sont indiquées dans la figure 3.1.

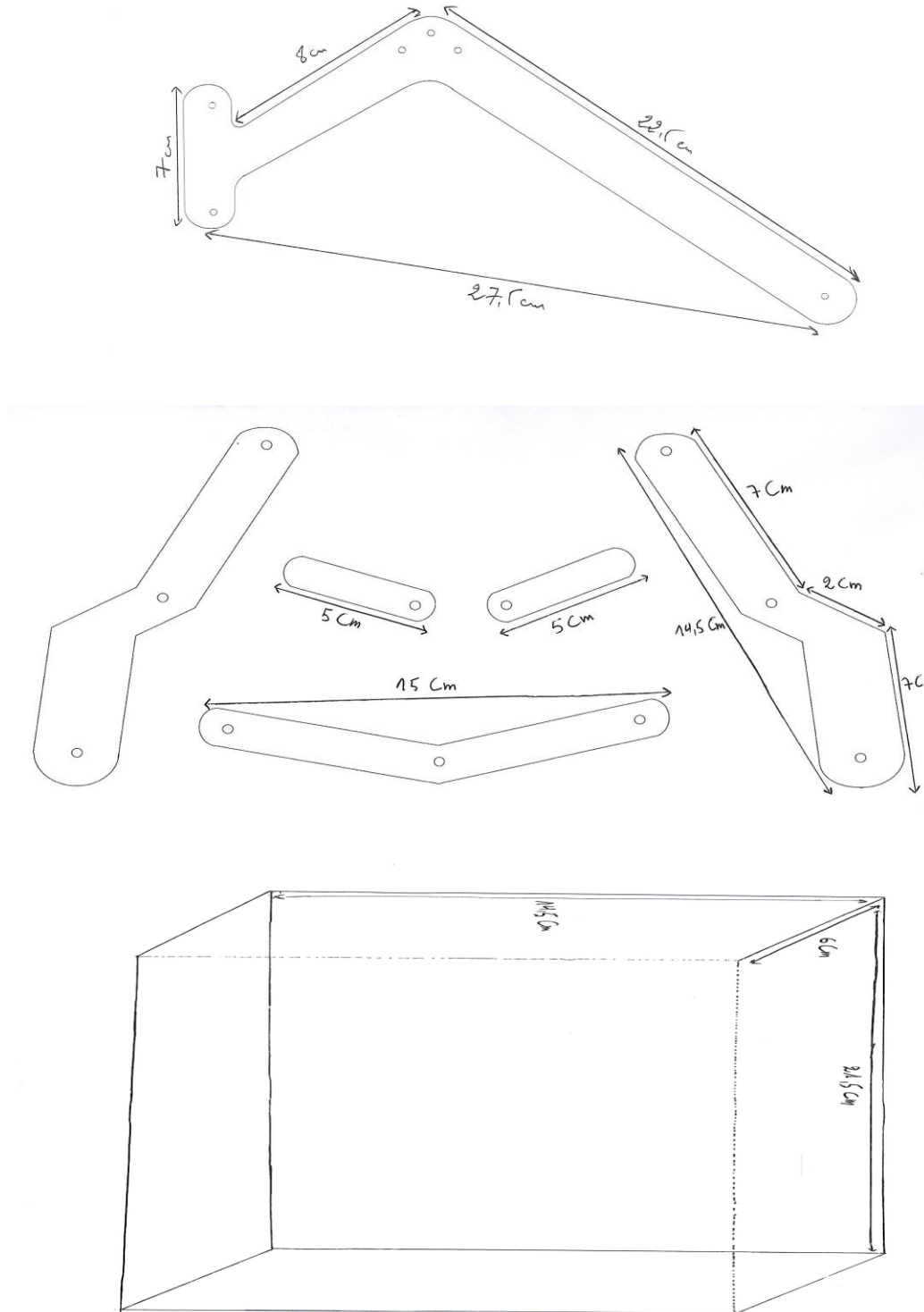


Figure 3. 1. Plans et dimensions des éléments de la structure du Robot.

### III.2.1.2. Découpage de la structure

Les dessins des différentes pièces (bascule, mécanisme lambda, cadre articulé) sont découpés dans du contreplaqué de 5 mm d'épaisseur à l'aide des équipements du découpage manuel.

La partie coupée est illustrée dans la figure suivante :



**Figure 3. 2.** Parties du corps du robot découpées dans du contreplaqué.

### III.2.1.3. Assemblage du corps du robot

Une fois le processus de coupe terminé, nous collectons les différentes parties de robot rocker-bogie en utilisant des boulons de 3 Cm.

La figure ci-dessous montre l'assemblage final du corps du robot avec les moteurs à courant continu sur roues.



**Figure 3. 3.** Assemblage final du corps du robot et mise en place des moteurs de roue.

### III.2.2. Réalisation du circuit électronique du robot

Après avoir présenté tous les éléments électroniques impliqués dans le robot, dans cette partie on utilise le logiciel Fritzing pour présenter un schéma de connexion global des composants électroniques et de la carte Raspberry.

Fritzing est un outil logiciel open source utilisé pour concevoir et prototyper des circuits électroniques. Il fournit une bibliothèque de composants électroniques conviviale pour créer des schémas, il a été initialement créé pour aider les concepteurs et les amateurs à documenter et à partager leurs projets électroniques.

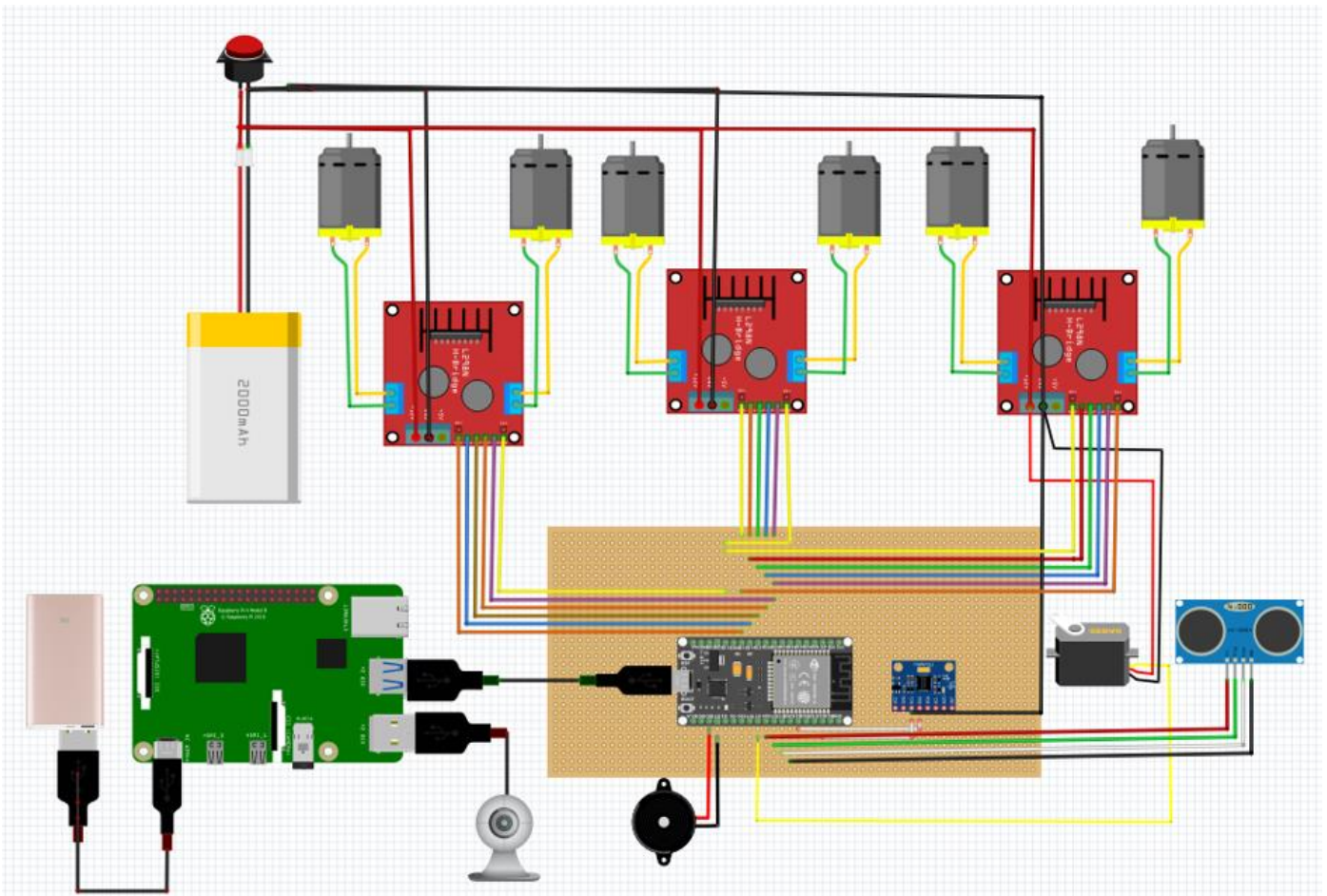
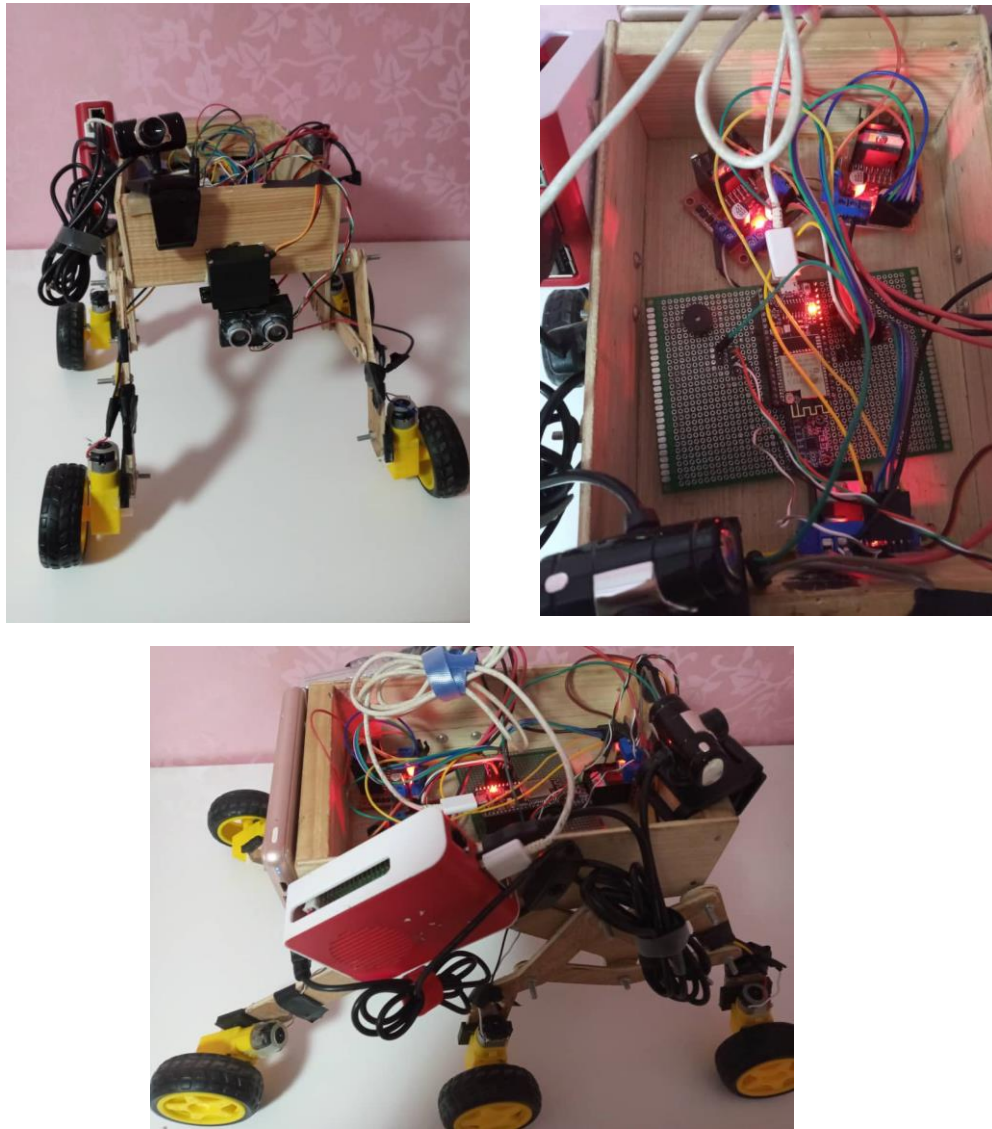


Figure 3. 4. Câblage du circuit électronique du robot.

La figure ci-dessous présente le câblage pratique des modules constituant le robot :



**Figure 3. 5.** La réalisation finale du robot.

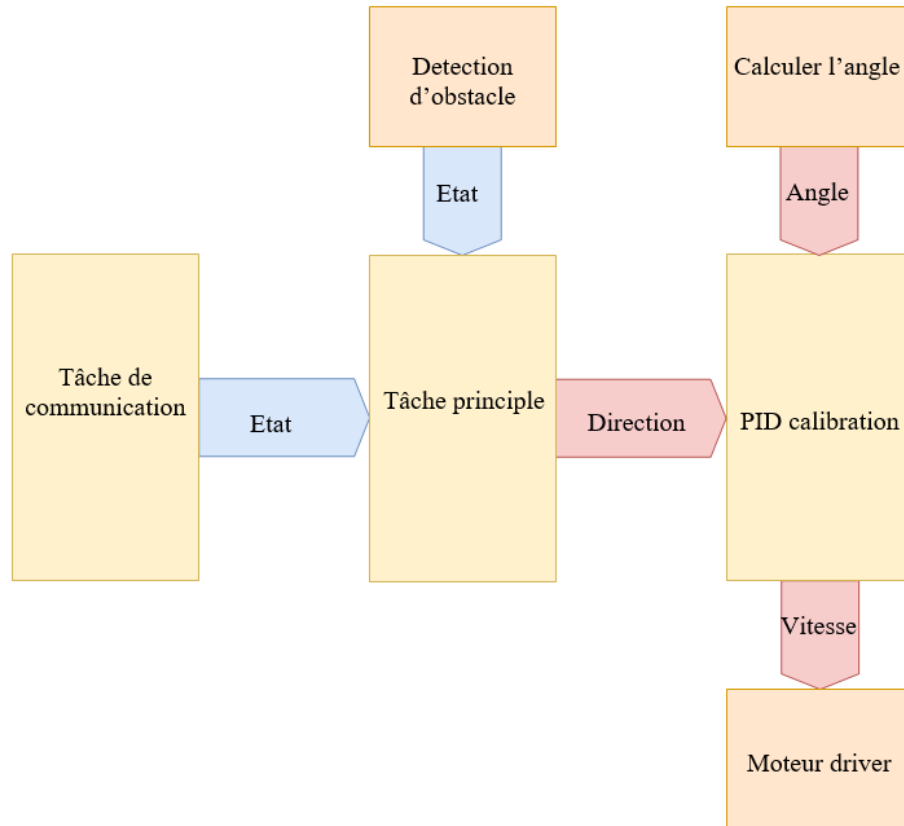
Le câblage électronique du robot dépendra des différents composants qui y sont présents. Voici une explication générale du câblage de notre robot :

- Microcontrôleur : l'ESP32 est le cerveau du robot et se connecte à tous les autres composants électroniques. Le microcontrôleur doit être alimenté à l'aide d'une source d'alimentation appropriée.
- Driver moteur : Le driver moteur, tel que le L298N, permet de contrôler la vitesse et la direction des moteurs. Les sorties de chaque canal du driver moteur doivent être connectées aux bornes des moteurs à courant continu (DC) du robot.

- **Moteurs** : Les moteurs à courant continu (DC) sont utilisés pour fournir la puissance mécanique au robot. Chaque moteur est connecté aux bornes du canal correspondant du driver moteur.
- **Batterie** : La batterie fournit l'alimentation électrique pour le robot. La batterie doit être connectée aux bornes d'entrée du driver moteur pour alimenter les moteurs.
- **Capteurs** : Les capteurs, tels que le MPU6050 ou les capteurs de distance à ultrasons, sont utilisés pour mesurer des grandeurs physiques telles que l'accélération ou la distance. Les capteurs sont connectés aux entrées de l'ESP32.
- **Actionneurs** : Les actionneurs, tels que les servomoteurs ou les buzzer, sont utilisés pour effectuer des actions spécifiques. Les actionneurs sont connectés aux sorties numériques du microcontrôleur.
- **Connexion entre l'ESP32 et le Raspberry Pi** : pour établir une connexion entre les deux on utilise un câble USB ainsi pour connecter le Raspberry et esp32 on les connectes via un port USB.

### **III.3. Principe de fonctionnement des différentes parties du robot**

La Figure 3. 6. Représente l'organigramme de fonctionnement global de notre robot su surveillance autonome.



**Figure 3. 6.** Organigramme de la logique du fonctionnement principal du robot.

### III.3.1. Sélectionner le chemin

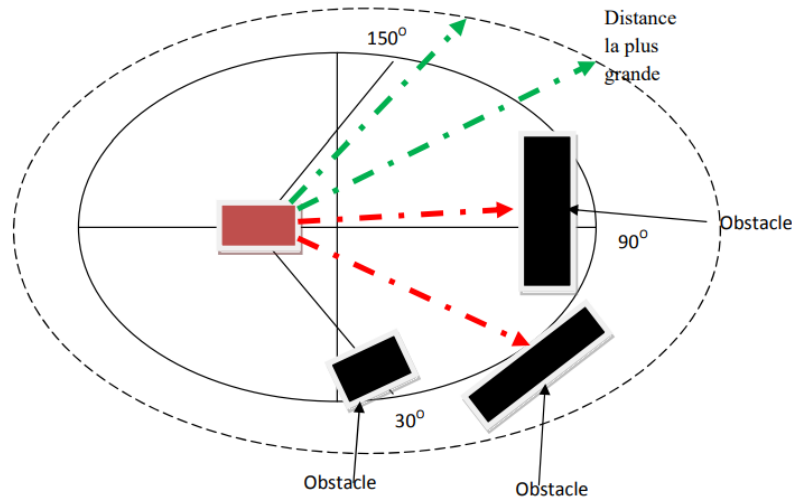
Le robot se déplace aléatoirement dans l'espace à partir de sa position de départ, Le but de ce comportement est la recherche d'un site d'abri tout en évitant les obstacles. L'environnement pour le robot est totalement inconnu.

#### III.3.1.1. Evitement d'Obstacles automatiquement

L'évitement d'obstacles est un comportement de base présent dans quasiment tous les robots mobiles, il est indispensable pour permettre au robot de fonctionner dans son environnement. Il permet d'éviter les obstacles apparus dans le champ de vision du robot, pour cela le robot doit déterminer un autre chemin.

Le comportement d'évitement d'obstacle est déterminé par un capteur US selon l'environnement dans lequel il évolue et selon sa position par rapport aux obstacles rencontrés.

Les directions autour du robot (entre l'angle  $30^\circ$  et  $150^\circ$ ) sont représentées par une distance. Ces distances sont des unités utilisées pour traiter les rotations des moteurs. (La plus grande distance + son angle par rapport au servomoteur) nous permettront de déterminer dans quel sens les moteurs doivent se tourner.



**Figure 3. 7.** Les angles de navigation [24].

Si le robot ne reçoit aucune information de ce capteur au cours de son déplacement, il considère que le chemin est libre c'est-à-dire aucune présence d'obstacle, donc dans cette situation le robot se déplace vers la cible jusqu'à ce qu'un obstacle soit détecté par le capteur, il envoie une impulsion d'une durée de  $10\mu s$  sur une broche de l'esp32 qui correspond à la patte Trig du capteur et on reçoit une impulsion sur une broche de l'esp32 qui correspond à la patte Echo du capteur, d'une durée proportionnelle à la distance qui sépare le capteur de l'obstacle.

L'organigramme de fonctionnement du capteur ultrason est décrit comme suit :



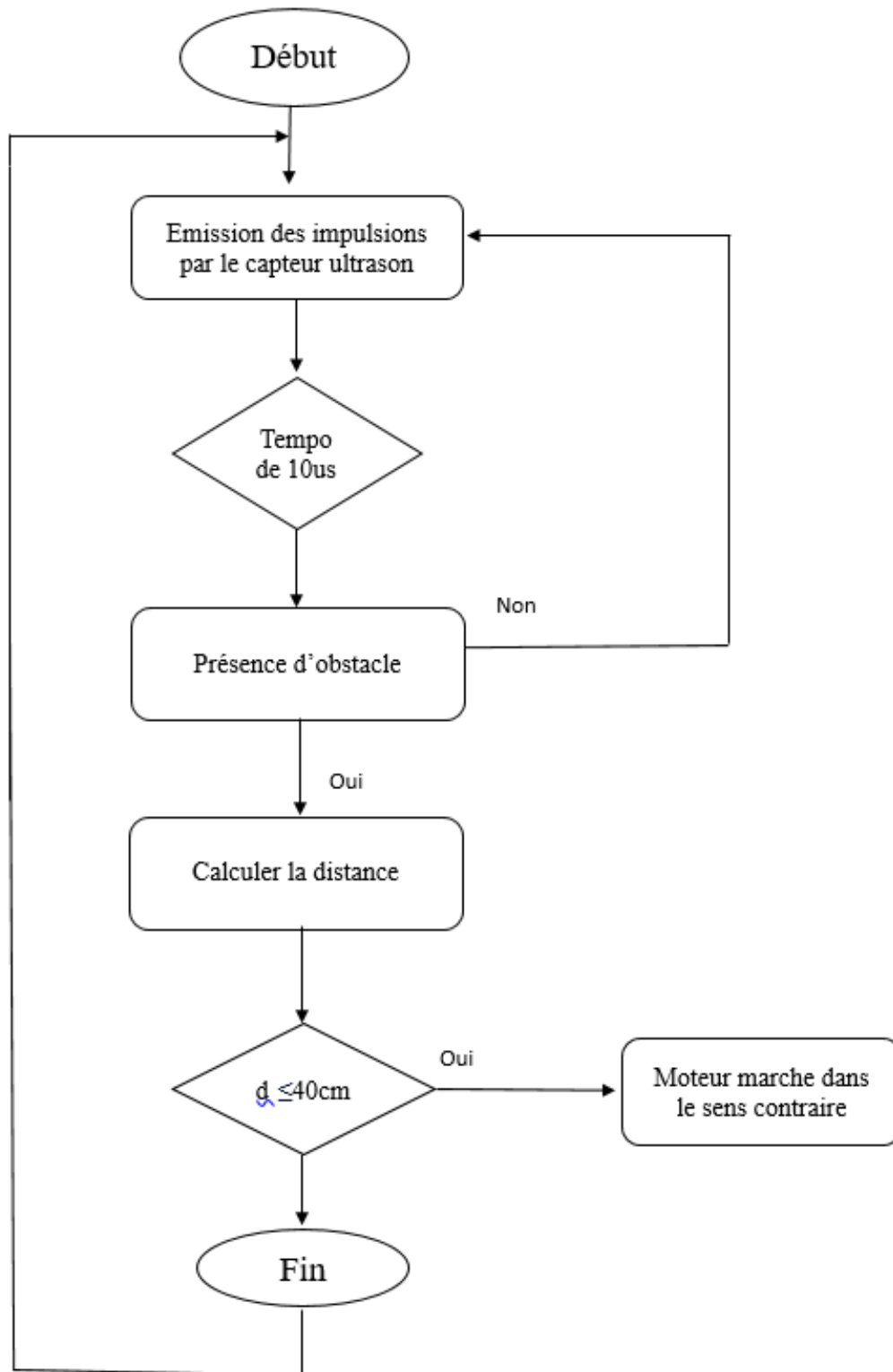
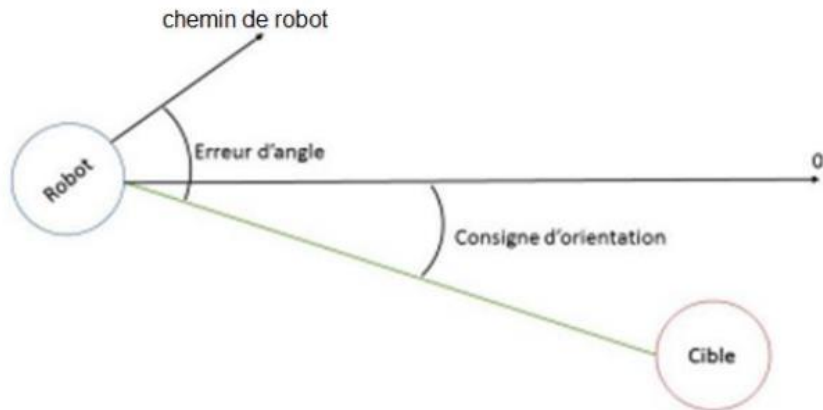


Figure 3. 8. Organigramme de détection d'obstacle.

### III.3.2. Calibrage du PID

Lorsque le robot mobile est désorienté on corrige cette erreur en utilisant un régulateur PID pour contrôler le robot (Figure 3. 9).



**Figure 3. 9.** Erreur dans le chemin [25].

La régulation PID (Proportionnelle, Intégrale, Dérivée) utilisé pour contrôler le mouvement d'un robot. Le contrôleur PID est un algorithme de contrôle qui ajuste la sortie d'un système en fonction de la différence entre la consigne et la mesure en temps réel. Le calibrage PID permet d'ajuster les coefficients du contrôleur pour obtenir une réponse optimale en fonction des conditions spécifiques du système robotique et des objectifs de contrôle. Cela inclut des vibrations réduites, une précision de suivi de trajectoire améliorée et un temps de réponse plus rapide.

L'organigramme suivant présente le principe de fonctionnement du PID :

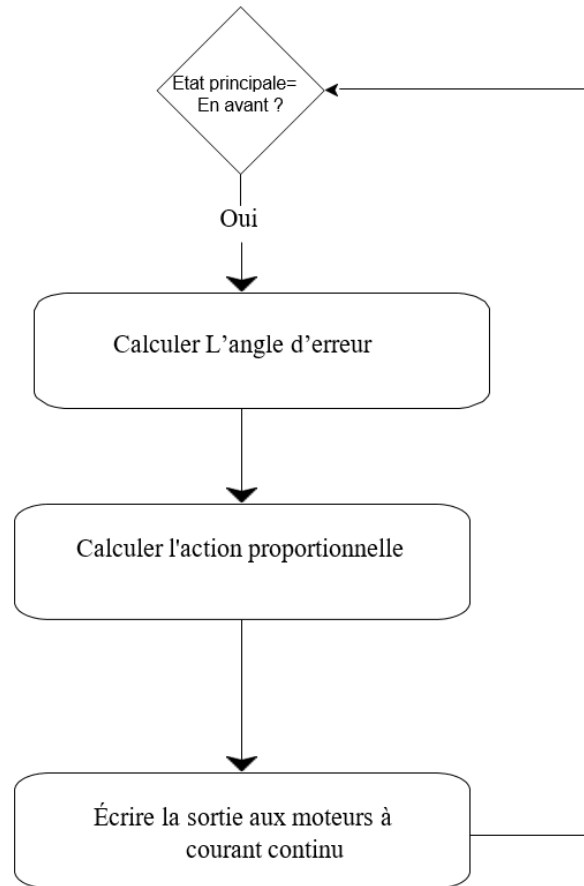


Figure 3. 10. Organigramme de la tâche d'étalonnage PID.

### III.4. Implémentation logicielle

L'implémentation de la partie logicielle du robot a fait appel à plusieurs logiciels de programmation pour implémenter différentes tâches. Nous avons utilisé le C++ pour programmer la carte de contrôle principale, qui est basée sur l'ESP32 ainsi nous avons également utilisé Vscode pour surveiller les messages séries échangés entre l'ESP32 et les capteurs.

Pour la vision et l'analyse d'images nous avons utilisé un Raspberry Pi équipé de Linux et programmé en Python. Visual studio code avec l'extension remote-SSH pour accéder au Raspberry ont été utilisés pour implémenter les programmes en Python. Cette combinaison nous a permis de bénéficier des avantages de chaque logiciel et de réaliser un robot performant et efficace.

### III.4.1. Outils de développement pour le Raspberry Pi

#### III.4.1.1. Linux

Linux est un système d'exploitation open source basé sur Unix. Il est connu pour sa fiabilité, sa sécurité et sa flexibilité, ainsi que pour sa large communauté de contributeurs et d'utilisateurs. Contrairement à certains systèmes d'exploitation propriétaires, Linux peut être installé et utilisé gratuitement, ce qui le rend accessible à tous. De plus, Linux est disponible en plusieurs distributions, ce qui permet aux utilisateurs de choisir celle qui convient le mieux à leurs besoins.



*Figure 3. 11. Logo du système d'exploitation Linux [26].*

Linux est devenu populaire auprès des développeurs, des entreprises et des organisations gouvernementales en raison de sa stabilité et de sa capacité à gérer des charges de travail importantes. En outre, les distributions Linux sont souvent mises à jour régulièrement, offrant ainsi des fonctionnalités et des améliorations de sécurité en permanence.

En tant qu'OS open source, le code source de Linux est accessible et modifiable par tous les utilisateurs et développeurs, ce qui permet une personnalisation et une adaptation importantes pour répondre aux besoins spécifiques. De plus, l'open source garantit une transparence et une sécurité accrues pour les utilisateurs, qui peuvent inspecter le code pour s'assurer qu'il n'y a pas de backdoors ou d'autres vulnérabilités cachées [40].

#### III.4.1.2. Python

Python est un langage de programmation open source, il est utilisé dans de nombreux domaines tels que la science des données, le développement Web, l'automatisation des tâches et l'IA.

Python présente une syntaxe claire, concise et lisible, ce qui le rend accessible même aux programmeurs novices, il est également multiplateforme, ce qui signifie que le même code peut fonctionner sur différentes plates-formes telles que Windows, Mac OS et Linux.



**Figure 3. 12.** Logo de langage de programmation Python [27].

Chaque langage de programmation a ses forces et ses faiblesses pour ce logiciel il dispose de mise à jours chaque de ces mises à jour peut ajouter de nouvelles fonctionnalités très utiles en ligne avec les pratiques de développement modernes, ce langage, bien qu'il soit facile à utiliser, possède les deux fonctionnalités, peut être utilisé pour la création de scripts et l'automatisation pour le développement de logiciels professionnels uniquement, il est donc très polyvalent. Python présente de nombreux avantages, mais il ne convient pas à toutes les tâches.

Par conséquent, il n'est pas assez bon pour la programmation au niveau du système. Enfin, dans les situations où la vitesse de l'application est primordiale, il vaut mieux éviter de l'utiliser, préférant utiliser C ou C++, ou un autre langage du même type.

La bibliothèque ou "library " de Python est l'une des principales raisons de son succès, il s'agit d'un vaste écosystème de logiciels développés par des tiers, cette collection s'est enrichie et élargie au fil des décennies. Plusieurs bibliothèques standard sont proposées, offrant des modules adaptés pour les tâches de programmation les plus courantes telles que l'accès aux fichiers.

Certains modules peuvent même gérer des tâches de programmation de haut niveau requises pour les applications modernes. Cela inclut la lecture et l'écriture de formats de fichiers structurés tels que JSON et XML, la manipulation de fichiers compressés ou l'utilisation de protocoles Web et de formats de données [41].

Nous avons utilisé dans notre projet le python3 car Il apporte de nouvelles fonctionnalités importantes et pratiques.

Pour commencer à utiliser Python, vous devez d'abord l'installer sur votre ordinateur. Une fois il est installé, vous pouvez l'utiliser via l'interface de ligne de commande ou un environnement de développement intégré (IDE) tel que Visual Studio Code pour exécuter des programmes Python que vous créez à l'aide de fichiers **.py** dans votre code et exécuter ces fichiers à partir de la ligne de commande ou IDE.

Le code Python peut être écrit dans un éditeur de texte ou un IDE et exécuté en spécifiant le nom du fichier de script après la commande python. Par exemple, dans ce cas, le fichier de script est nommé "main.py" et peut être exécuté avec la commande suivante :

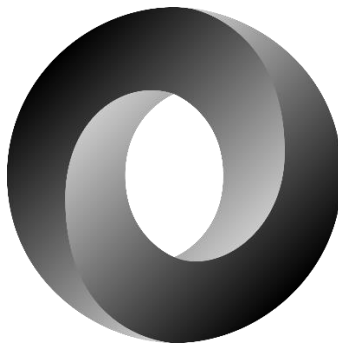
```
python main.py
```

**Figure 3. 13.** L'exécution du code en spécifiant le nom du fichier.

#### III.4.1.2.2. Json

JSON est l'acronyme de JavaScript object, il s'agit d'un format de fichier open-standard pour l'échange de données.

Les données peuvent être organisées, stockées dans un format lisible par l'homme et facilement accessibles, quelle que soit la langue utilisée. Les données sont généralement stockées dans des fichiers avec l'extension. Json sous forme de paires clé-valeur dans un format de texte brut [42].



**Figure 3. 14.** Logo de json [28].

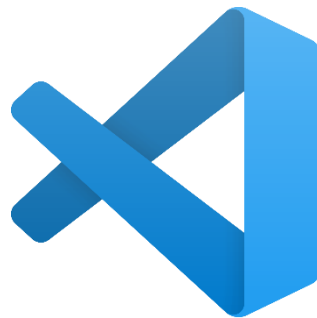
Dans notre cas on a utilisé json pour stocker l'adresse IP ainsi le numéro de port (Figure 3.19).

```
{ } client_config.json ●  
C: > Users > AN > OneDrive > Bureau > Dead  
1 {  
2   "ip": "192.168.155.75",  
3   "port": 8000,  
4   "images": ["nour.jpeg"],  
5  
6
```

**Figure 3. 15.** Stockage des données dans un fichier json.

### III.4.1.3. Visual Studio Code

Visual Studio Code (VS Code) est un éditeur de code source léger et gratuit développé par Microsoft pour Windows, Linux et macOS. Il offre des fonctionnalités avancées pour la programmation, telles que la complétion de code, la détection d'erreurs et la refactorisation. Il prend en charge de nombreux langages de programmation, tels que C++, C#, Python, Java, JavaScript, PHP, Ruby et bien d'autres. Visual Studio Code dispose également d'une grande communauté de développeurs qui ont créé des extensions pour ajouter des fonctionnalités supplémentaires à l'éditeur [43].



**Figure 3. 16.** Logo du VScode [29].

Vscode comme de nombreux autres éditeurs de code adopte une interface utilisateur commune et la disposition d'un explorateur sur la gauche, affichant tous les fichiers et dossiers auxquels vous avez accès, et un éditeur sur la droite, affichant le contenu des fichiers que vous avez ouverts.

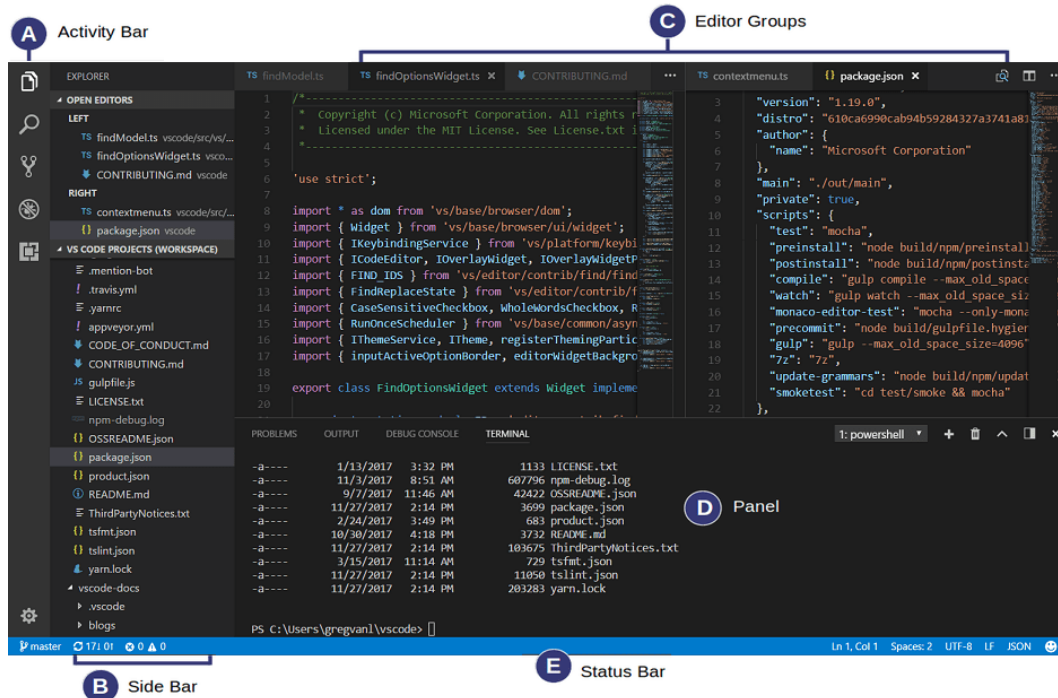


Figure 3. 17. Présentation d'interface de Visual Studio Code.

#### III.4.1.4. SSH

SSH (Secure Shell) désigne à la fois un logiciel et un protocole de communication informatiques. Ce protocole possède par ailleurs la particularité d'être entièrement chiffré. Cela signifie donc que toutes les commandes que vous exécuterez via SSH seront totalement secrètes.

Le SSH a été créé en 1995 avec pour principal but de permettre la prise de contrôle à distance d'une machine à travers une interface en lignes de commande.

Aujourd'hui, SSH est principalement utilisé à travers l'implémentation libre Open SSH qui est présent dans la plupart des distributions Linux [44].

Avant de commencer, on doit créer un réseau WiFi via le point d'accès créé sur notre smartphone. Nous avons également besoin d'un environnement de programmation, dans ce cas, le code Visual Studio qui lance SSH qui permet d'accéder au Raspberry selon les étapes suivantes :

1. Tout d'abord on doit se connecter à un hôte distant via SSH dans Visual Studio Code vous permet de vous connecter à distance à un serveur Linux et de travailler sur des fichiers à l'aide de l'interface Visual Studio Code.



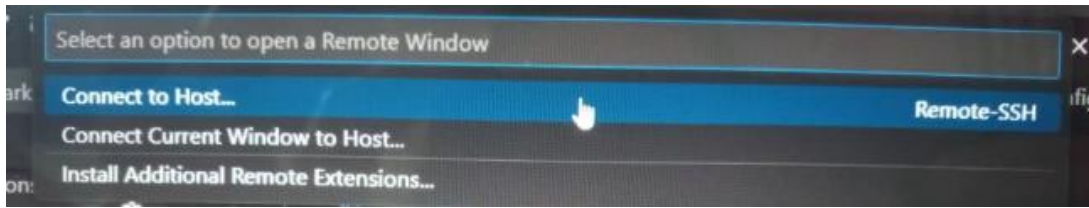


Figure 3. 18. L'accès au SSH.

2. Une fois on a lancer le SSH, on passe à écrire l'adresse IP.

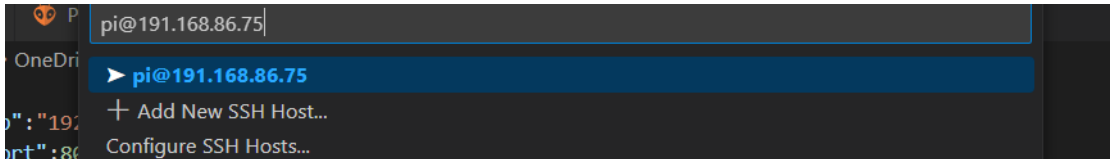


Figure 3. 19. Enter L'adresse IP.

3. Sélectionnez la plateforme de l'hôte distant 192.168.86.75 ou vous devez spécifier le type de système d'exploitation de l'hôte distant pour que la connexion fonctionne correctement.

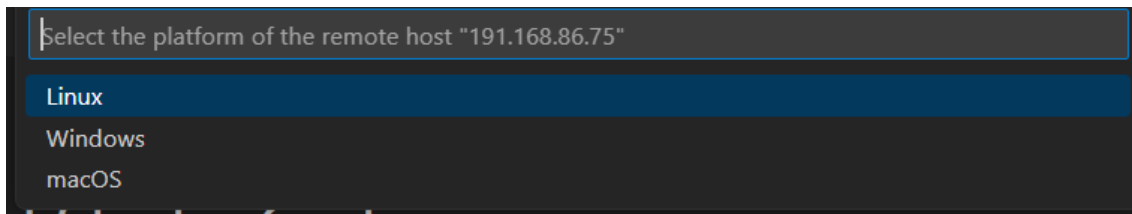


Figure 3. 20. La sélection du système d'exploitation.

4. Une fois nous avons choisi Linux, on tape maintenant le mot de passe.

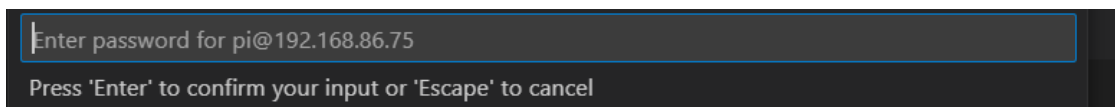


Figure 3. 21. Taper le mot de passe.

5. On choisit le fichier main.py et on lance le programme, on voit que le robot à trouver l'esp32 donc il lui a envoyer l'ordre de commencer.

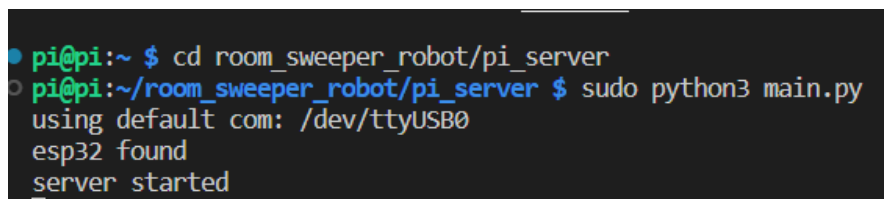


Figure 3. 22. Photo du terminal en Vscode du fonctionnement du robot.

### III.4.2. Logiciel de programmation pour L'ESP32 : Le C++

Le C++ est un langage de programmation de haut niveau utilisé pour créer des logiciels tels que des systèmes d'exploitation, des navigateurs Web. Il est largement utilisé pour le développement de logiciels complexes nécessitant une haute performance, une utilisation intensive de la mémoire et une manipulation de bas niveau des ressources système.

Le C++ est une évolution du langage de programmation C, qui ajoute des fonctionnalités supplémentaires telles que la programmation orientée objet, la surcharge d'opérateurs, Il permet également l'utilisation de bibliothèques logicielles existantes pour faciliter le développement de logiciels. L'environnement de programmation le plus couramment utilisé pour le C++ est Visual Studio Code (VSCODE) [45].

La programmation du robot a été réalisée en C++. L'organigramme de communication entre Raspberry Pi et ESP32 est présenté dans la figure 3.28.

La communication entre le Raspberry Pi et l'ESP32 fonctionne de la manière suivante :

Le Raspberry Pi envoie une commande sous forme de caractère via une liaison série (USB) vers l'ESP32, ce dernier reçoit la commande et l'interprète en conséquence :

- Si le Raspberry Pi envoie le caractère "P", cela signifie que l'ESP32 doit actionner les capteurs ainsi que les drivers pour que le robot avance.
- Si le Raspberry Pi envoie le caractère "S", cela signifie que l'ESP32 doit s'arrêter.
- Si le Raspberry Pi détecte un visage qui n'est pas dans sa base de données, il envoie le caractère "F" à l'ESP32.
- L'ESP32 reçoit le caractère "F" et actionne le buzzer pour alerter l'utilisateur de la détection d'un visage intrus.

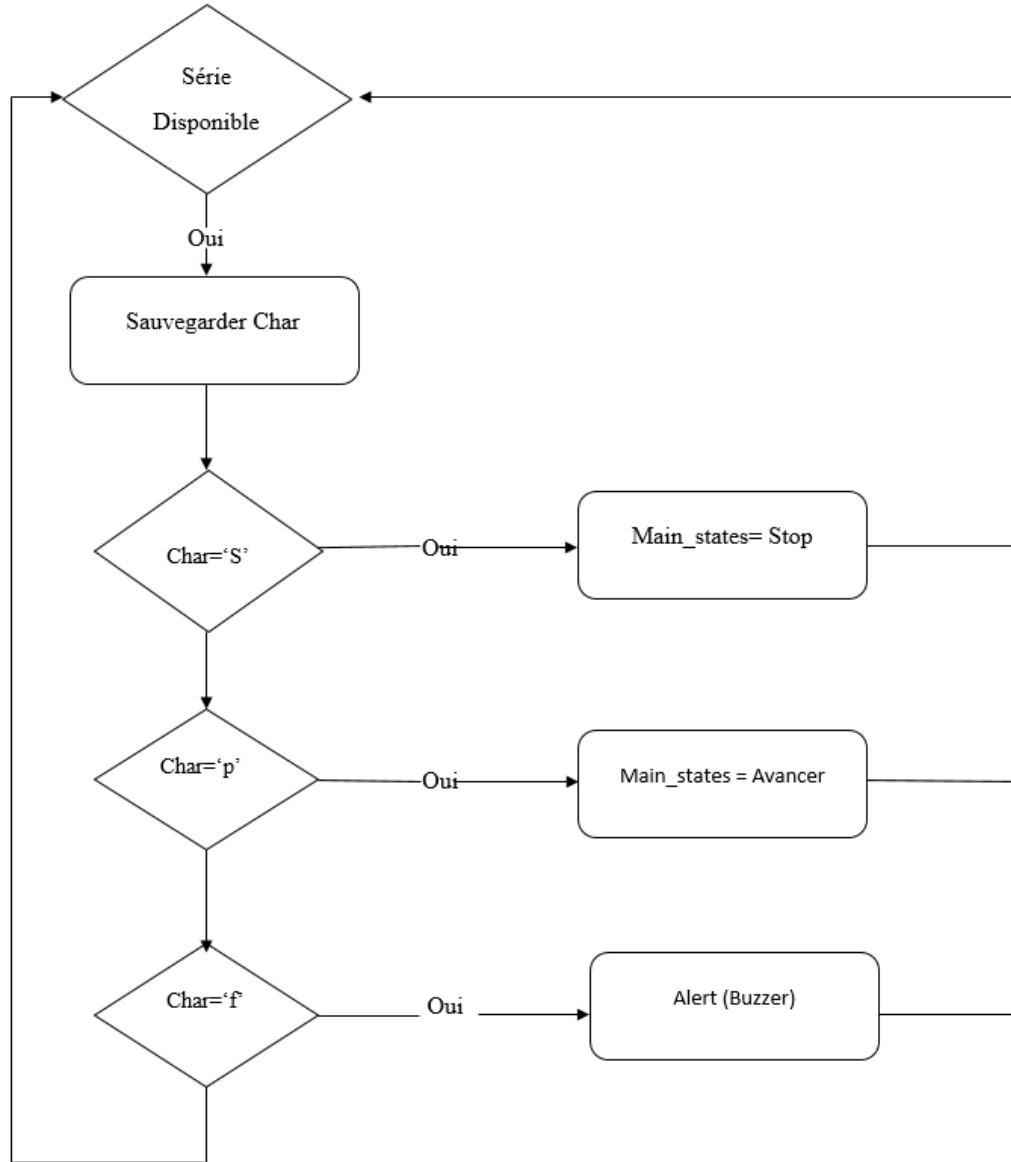


Figure 3. 23. Organigramme de la tâche de communication.

Les figures suivantes présentent le fonctionnement du serial monitor de l'esp32 :

```

* Executing task in folder robot_control: C:\Users\AN\.platformio\penv\Scripts\platformio.exe device monitor
--- Terminal on COM3 | 9600 8-N-1
--- Available filters and text transformations: colorize, debug, default, direct, esp32_exception_decoder, hexlify, log2file, nocontrol, printable, send_on_enter, time
--- More details at https://bit.ly/pio-monitor-filters
--- Quit: Ctrl+C | Menu: Ctrl+T | Help: Ctrl+T followed by Ctrl+H
state :stop
state :stop
state :stop
state :stop
  
```

Figure 3. 24. Le lancement du moniteur série.

```
state :stop
state :stop
state :stop
state :stop
state :stop
state :stop
state :stop
state :stop
state :stop
state :stop
state :stop
state :stop
```

Figure 3. 25. Envoyer le caractère P.

```
state :forward
state :forward
state :forward
state :forward
state :forward
state :forward
state :forward
state :forward
state :forward
state :forward
state :forward
state :forward
```

Figure 3. 26. Envoyer le caractère S.

```
state :face_found
state :face_found
state :face_found
state :face_found
state :face_found
state :face_found
state :face_found
state :face_found
state :face_found
```

Figure 3. 27. Envoyer le caractère F.

Quand le robot rencontre un obstacle il cherche d’abord un chemin pour esquiver. Si un chemin est trouvé, le robot tourne soit à gauche ou à droite pour l’éviter en fonction de la direction la plus appropriée.

```
state :find_path
state :find_path
state :find_path
state :find_path
state :find_path
state :find_path
```

Figure 3. 28. Trouver le chemin.

```
state :big_left
state :big_left
state :big_left
state :big_left
state :big_left
state :big_left
state :big_left
state :big_left
```

```
state :big_right
state :big_right
state :big_right
state :big_right
state :big_right
state :big_right
state :big_right
state :big_right
```

```
state :dodge_left
state :dodge_left
state :dodge_left
state :dodge_left
state :dodge_left
state :dodge_left
state :dodge_left
state :dodge_left
```

Figure 3. 29 Evitement d’obstacle.

### III.5. Organigramme Final du fonctionnement du robot

Lorsque le robot détecte un obstacle, il décide de tourner à gauche ou à droite selon un certain angle comme suit :

Si le robot rencontre un obstacle qui nécessite un virage à 60 degrés, il peut s'agir d'un petit obstacle qui peut être facilement évité sans nécessiter un grand virage. Cela permettra au MPU6050 d'enregistrer la nouvelle valeur d'angle, permettant au robot d'avancer.

En revanche, lorsque le robot rencontre un obstacle qui nécessite un virage de 90 degrés, cela veut dire y'a une présence d'un obstacle important ce qui nécessite un virage plus important pour éviter l'obstacle sans risquer de le heurter et enregistre le nouvel angle grâce au MPU6050.

Lorsque le robot détecte un visage inconnu, il s'arrête et émet un bip pendant 10 secondes avant de continuer à avancer.

Le robot calculera la distance s'il doit choisir la direction gauche ou droite :

Si la distance à gauche est supérieure à la distance à droite, choisissez le chemin de gauche, sinon choisissez le chemin de droite.

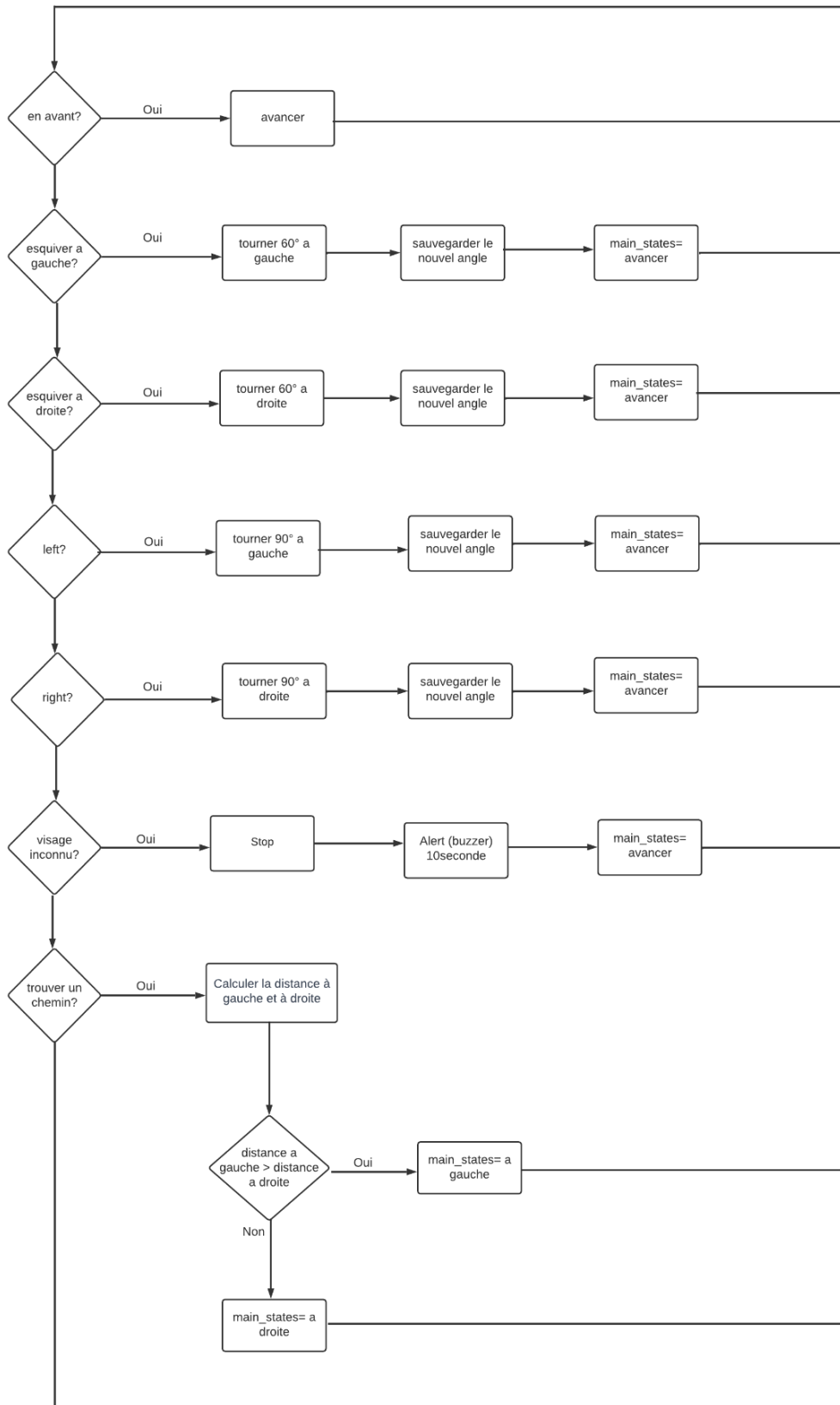


Figure 3. 30. Organigramme final du fonctionnement du robot.

### III.6. Conclusion

Dans ce chapitre, nous avons abordé les différents composants matériels nécessaires pour la construction du robot, ainsi que les différentes étapes de montage et de câblage de ces composants. Nous avons également présenté le choix et l'installation des logiciels nécessaires pour programmer et contrôler le robot.

La réalisation matérielle du robot nous a permis de comprendre le rôle et l'interaction de chaque composant électronique et mécanique. La programmation du robot, quant à elle, nous a permis de définir son comportement et ses actions en fonction des différentes situations rencontrées.

Dans le prochain et dernier chapitre nous aborderons une partie cruciale qui est le traitement d'image ou nous allons utiliser une caméra pour capturer des images et comment nous pouvons les traiter pour détecter des objets spécifiques, tels que des visages.

# **Chapitre IV**

## **Systeme de surveillance visuelle**



## IV.1. Introduction

Dans ce chapitre, nous allons décrire en détail le fonctionnement de notre système de reconnaissance faciale, en expliquant les différentes étapes de la détection des visages ainsi que les technologies et bibliothèques logicielles utilisées. Enfin, nous présentons brièvement les résultats obtenus lors des tests du robot.

## IV.2. L'application du AI dans la surveillance

La vision artificielle est de plus en plus utilisée dans la surveillance, car elle permet une surveillance continue, en temps réel, de grandes zones à faible coût, et peut fournir des alertes en cas d'incidents ou de comportements suspects. Voici quelques exemples d'applications de la vision artificielle dans la surveillance :

- **Détection d'intrusion** : Les caméras de surveillance équipées d'algorithmes de vision artificielle peuvent détecter les intrus ou les personnes qui pénètrent dans une zone restreinte. Les systèmes de détection peuvent être configurés pour déclencher une alerte si une personne est détectée.
- **Reconnaissance faciale** : Les caméras de surveillance équipées de la technologie de reconnaissance faciale peuvent identifier les individus qui pénètrent dans une zone restreinte. Cette technologie est couramment utilisée dans les aéroports, les gares et les bâtiments gouvernementaux [46].



**Figure 4. 1.** Reconnaissance facile et digital [30].

- **Analyse du comportement** : Les algorithmes de vision artificielle peuvent analyser les comportements des personnes filmées, tels que les mouvements inattendus, les rassemblements de personnes, ou encore les personnes qui se comportent de manière agressive. Les systèmes peuvent déclencher des alertes en cas de comportements suspects [47].



Figure 4. 2. Détection de vol par l'IA [31].

- **Surveillance de la circulation** : Les caméras équipées de la technologie de vision artificielle peuvent détecter les infractions au code de la route, comme le non-respect des feux de signalisation, le franchissement de la ligne continue ou encore la vitesse excessive. Les systèmes de surveillance peuvent déclencher des alertes en cas d'infractions [48].

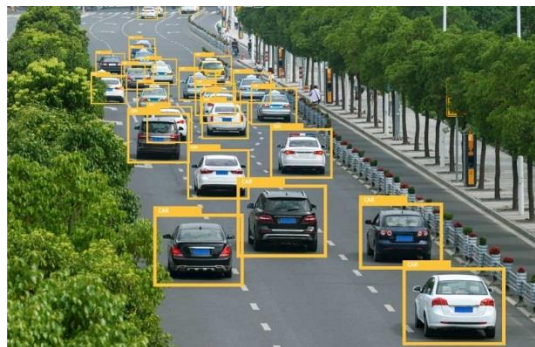


Figure 4. 3. Optimiser la gestion du trafic avec l'IA [32].

- **Surveillance de la sécurité** : Les caméras équipées de la vision artificielle peuvent surveiller les zones à risque, comme les installations industrielles, les aéroports, les gares ou les centres commerciaux, pour détecter les activités suspectes et prévenir les incidents de sécurité [49].



**Figure 4. 4.** Caméra de surveillance [33].

### IV.3. Système de Capture d'images et Reconnaissance Faciale

Le système de capture de photos est un ensemble de dispositifs, de logiciels et de techniques permettant d'acquérir des images à partir d'une source, telle qu'une webcam, un appareil photo ou une caméra. L'objectif principal du système de capture de photos est de capturer des images de haute qualité pour une utilisation ultérieure, telle que l'affichage, le stockage, le partage ou le traitement.

Un système de reconnaissance faciale est une application logicielle visant à reconnaître une personne grâce à son visage de manière automatique. C'est un domaine de la vision par ordinateur consistant à reconnaître automatiquement une personne à partir d'une image de son visage. La reconnaissance de visage a de nombreuses applications dans la vidéo-surveillance, la biométrie, la robotique.

Dans notre travail, on s'intéresse à la reconnaissance de visages à partir d'une séquence vidéo, puisqu'elle est plus demandée et plus pertinente [50]

Le principe de fonctionnement de base de notre système de reconnaissance est montré dans la figure suivante :

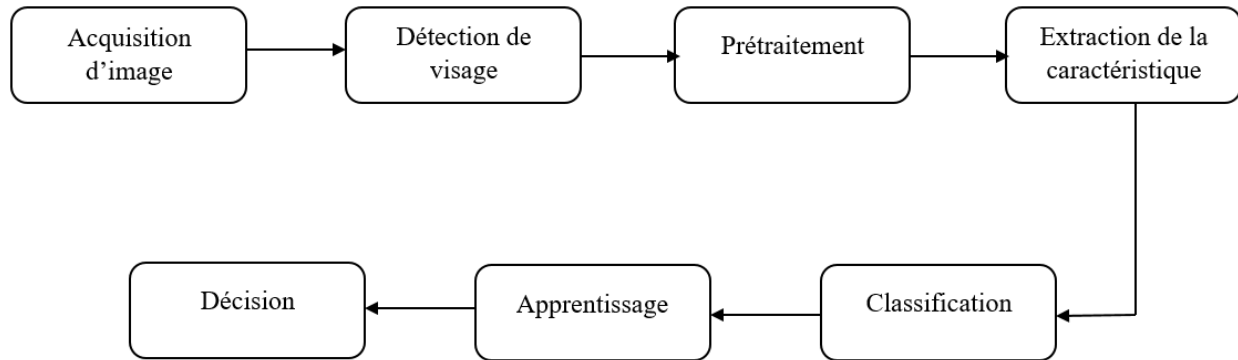


Figure 4. 5. Processus d'un système de capture d'image et de reconnaissance faciale.

### IV.3.1. Acquisition d'image

Les étapes de l'acquisition sont les suivantes :

- **Activation de la caméra :** Il s'agit d'allumer et d'initialiser la caméra pour qu'elle soit prête à capturer des images.
- **Capture de l'image :** Utilisation des bibliothèques OpenCV pour capturer une image à partir de la caméra.
- **Stockage de l'image :** L'image capturée doit être stockée dans une base de données, en fonction des besoins du système [51].

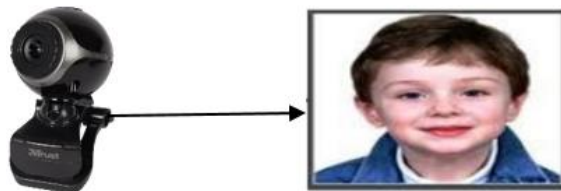


Figure 4. 6. Exemple de l'acquisition d'image [34].

### IV.3.2. Détection de visage

La détection des visages est la première étape de tout processus de reconnaissance, elle a pour but de localiser les visages dans une scène. L'algorithme doit localiser les visages dans l'image en utilisant des techniques telles que la détection de visages basée sur des modèles dans notre cas on a utilisé HOG, SVM [52].

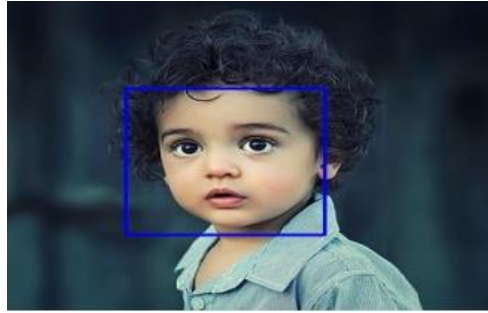


Figure 4. 7. Détection de visage [35].

### IV.3.3. Le prétraitement

Le prétraitement permet d'éliminer le bruit - causé par la qualité du dispositif d'acquisition et l'éclairage - présent dans les images brutes et d'améliorer la qualité de celles-ci par plusieurs méthodes comme : la convolution, la normalisation, les filtres, l'égalisation...etc [53].

### IV.3.4. Extraction des caractéristiques

Le but de cette étape est d'extraire l'information faciale pertinente à partir de l'image du visage préalablement normalisée, l'information extraite forme le modèle à partir duquel se fera la comparaison entre les visages. Cela implique généralement l'identification et l'extraction de traits spécifiques du visage, tels que la forme du visage, la position des yeux, la taille du nez, les contours des lèvres, etc [54].

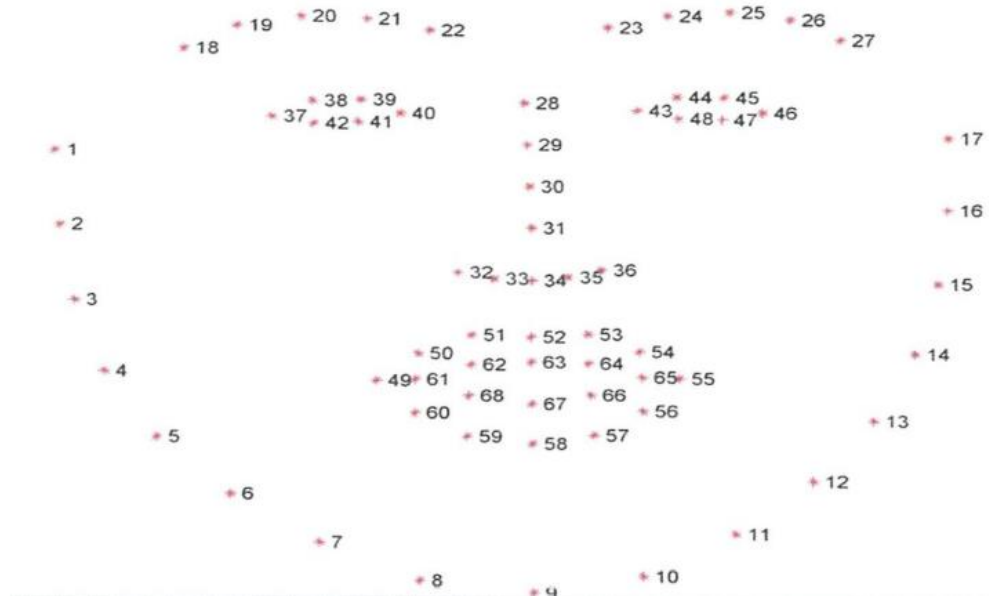


Figure 4. 8. Extraction des points de repère faciaux [36].

### IV.3.5. La classification

La classification est une étape essentielle de la reconnaissance faciale qui permet de déterminer l'identité d'un visage à partir des caractéristiques extraites. Elle repose sur l'utilisation d'un modèle pré-entraîné qui compare les caractéristiques du visage avec celles présentes dans une base de données [55].

### IV.3.6. L'apprentissage

C'est l'étape où on fait apprendre les individus au système, elle consiste à mémoriser les paramètres, après extraction et classification dans une base de données bien ordonnées pour faciliter la phase de reconnaissance et la prise d'une décision, elle est en quelque sorte la mémoire du système [56].

### IV.3.7. La décision

La décision dans le contexte de la reconnaissance faciale se réfère à la détermination de l'identité d'un visage à partir des caractéristiques extraites et des modèles de classification entraînés. Une fois que les caractéristiques d'un visage inconnu ont été extraites, elles sont comparées aux caractéristiques des visages connus dans la base de données [57].

## IV.4. Outils logiciels utilisés

### IV.4.1. Base de donnée

Une base de données est un ensemble organisé de données qui sont stockées de manière structurée, afin de permettre une gestion efficace et une récupération facile des informations. Dans le contexte de la reconnaissance faciale, une base de données peut être utilisée pour stocker des informations sur les visages, Cette base de données peut être consultée lors de la phase de reconnaissance pour comparer les caractéristiques du visage capturé avec celles présentes dans la base de données, afin d'identifier et de différencier les visages connus des visages inconnus.

-Tout d'abord, on prend des photos des personnes qu'on veut l'introduire dans notre data-base.

-L'étape suivante consiste à introduire sous chaque image le nom de la personne.

- Enfin, on enregistre toutes les photos qui sont des format jpeg ensuite stocker dans un fichier json.

#### IV.4.2. OpenCV

OpenCV (Open Source Computer Vision Library) est une bibliothèque open source populaire de traitement d'images et de vision par ordinateur. Il offre des fonctionnalités puissantes pour capturer, éditer, traiter et analyser des images et des vidéos et il est écrit en C++ et prend en charge plusieurs langages de programmation, dont Python [58].



**Figure 4. 9.** Logo de la bibliothèque d'OpenCV [37].

Dans notre cas, nous avons utilisé `opencv` pour accéder à la webcam et alimenter un flux vidéo en direct. OpenCV fournit des fonctionnalités pour détecter et se connecter aux webcams disponibles sur notre système. Une fois connecté, on peut utiliser les capacités d'OpenCV pour capturer des images individuelles à partir du flux vidéo en direct de votre webcam et en temps réel. Le format d'image pris par la caméra Web est BGR (bleu, vert, rouge). Dans ce format, chaque pixel est représenté par trois canaux de couleur : bleu, vert et rouge. Chaque canal a des valeurs de pixel de 0 à 255 représentant l'intensité d'une couleur particulière dans le pixel.

Pour capturer une photo, voici les étapes générales pour capturer une photo à l'aide d'OpenCV :

1. Importez la bibliothèque OpenCV dans votre programme.
2. Initialisez la capture vidéo en utilisant la fonction `cv2.VideoCapture(0)` pour accéder à la webcam par défaut.
3. À chaque itération de la boucle, utilisez la fonction `cap.read()` pour lire une image de la webcam.
4. Affichez l'image capturée dans une fenêtre à l'aide de la fonction `cv2.imshow()`.
5. Enregistrez l'image capturée en utilisant la fonction `cv2.imwrite()` en spécifiant le nom et le chemin de sauvegarde de l'image.

```
import cv2

# Initialise la capture vidéo
cap = cv2.VideoCapture(0)

while True:
    # Lit une image de la webcam
    ret, frame = cap.read()

    # Affiche l'image capturée
    cv2.imshow('Webcam', frame)

    # Capture la photo
    cv2.imwrite('nour.jpg', frame)
    break
```

**Figure 4. 10.** Le code Python pour capturer une photo à l'aide d'OpenCV.

### IV.4.3. Numpy

Numpy est une bibliothèque pour le langage de programmation python. Dans notre cas les images capturées par la webcam ou chargées à partir de fichiers sont également représentées sous forme de tableaux numpy. Chaque pixel de l'image est représenté par une valeur numérique, généralement un entier ou un flottant, qui peut être stocké dans un tableau multidimensionnel numpy. Cela permet d'effectuer des opérations sur l'image, telles que le redimensionnement, le recadrage, le filtrage ou la normalisation des valeurs des pixels.

### IV.4.4. Dlib

Dlib est une bibliothèque logicielle populaire en vision par ordinateur et en apprentissage automatique. Elle propose une large gamme d'algorithmes et de modèles pour diverses tâches liées à la vision par ordinateur, y compris la reconnaissance faciale.

Dlib offre plusieurs fonctionnalités utiles pour la reconnaissance faciale telles que :

- Détection de visage : Dlib propose des algorithmes basés sur le modèle HOG (Histogram of Oriented Gradients) est utilisé pour extraire des informations sur la texture et la forme des régions faciales détectées. Il analyse les gradients de luminosité dans différentes directions pour capturer les contours et les structures du visage. Ces informations sont ensuite utilisées



pour créer des vecteurs de caractéristiques compacts qui représentent les visages de manière distinctive.

- Extraction de points de repère faciaux : Dlib offre également des fonctionnalités pour extraire les points de repère faciaux à partir des visages détectés. Ces points de repère correspondent à des caractéristiques spécifiques du visage, comme les yeux, le nez, la bouche, les sourcils, etc.

L'extraction des points de repère se fait en utilisant le SVM (Support Vector Machine) est un algorithme d'apprentissage supervisé utilisé pour classifier les visages en fonction de leur identité (reconnaissance faciale), il est entraîné à partir d'un ensemble d'exemples de visages connus pour apprendre à discriminer les différentes classes d'identité. Lors de la phase de test, le SVM est utilisé pour prédire l'identité des visages en se basant sur les vecteurs de caractéristiques extraits des nouvelles images capturées.

#### **IV.4.5. Face\_recognition**

La bibliothèque face\_recognition est un outil basé sur dlib qui offre des fonctionnalités avancées de reconnaissance faciale. Elle utilise des techniques telles que la création de vecteurs de caractéristiques faciales et l'apprentissage automatique pour identifier et différencier les visages.

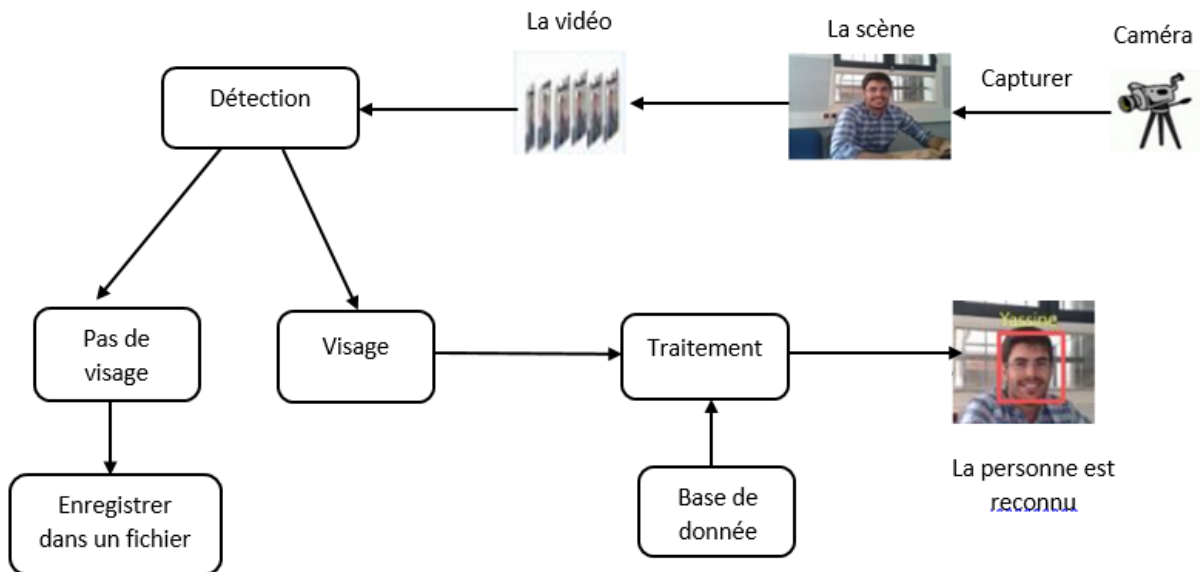
Lorsqu'il s'agit de reconnaissance faciale avec face\_recognition, voici comment cela fonctionne :

- Création de vecteurs de caractéristiques : À partir des points de repère faciaux, face\_recognition utilise un algorithme pour créer un vecteur de caractéristiques uniques pour chaque visage. Ce vecteur capture les informations distinctives du visage et peut être utilisé pour comparer les visages ultérieurement.
- Comparaison des visages : Une fois que les vecteurs de caractéristiques sont créés pour les visages connus, face\_recognition peut les comparer avec les vecteurs de caractéristiques des visages inconnus.
- Identification des visages : En utilisant les résultats de la comparaison, face\_recognition peut identifier les visages correspondants à ceux présents dans la base de données ou déterminer si un visage est inconnu.

#### **IV.5. Fonctionnement du système**

Notre programme est divisé en deux parties : client et serveur.

- Partie client est responsable de la capture des images à partir de la webcam, du prétraitement de ces images, de l'extraction des caractéristiques faciales et de l'envoi des données au serveur pour le traitement ultérieur.
- Partie serveur serait chargée de recevoir les données du client, d'effectuer la détection et la reconnaissance faciale, ainsi que de renvoyer les résultats au client.



**Figure 4. 11.** La conception globale de notre système.

Le fonctionnement de notre système est expliqué par les étapes suivantes :

#### Etape 01 :

- L'ouverture de PowerShell sur notre système, PowerShell est un environnement de ligne de commande disponible sur les systèmes Windows.
- Ensuite, nous devons connaître l'adresse IP du Raspberry Pi ainsi que le mot de passe requis pour y accéder. L'adresse IP est généralement fournie par notre réseau local ou configurée manuellement sur le Raspberry Pi.
- Une fois que nous disposons de ces informations, nous pouvons nous connecter au Raspberry Pi en utilisant PowerShell. Pour cela, nous servons de la commande "ssh" suivie de l'adresse IP du Raspberry Pi.
- Une fois connecté, nous pouvons accéder au répertoire où se trouve le fichier "main.py" .
- Une fois connecté, nous exécutons le fichier main.py pour démarrer le serveur.

- Le serveur est maintenant prêt à recevoir les données de la partie cliente.

```
PS C:\Users\AN> ssh pi@192.168.91.75
pi@192.168.91.75's password:
Linux pi 5.15.84-v7l+ #1613 SMP Thu Jan 5 12:01:26 GMT 2023 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon May 29 09:39:39 2023
pi@pi:~$ cd room_sweeper_robot/pi_server
pi@pi:~/room_sweeper_robot/pi_server$ sudo python3 main.py
Using default com: /dev/ttyUSB0
esp32 found
server started
```

**Figure 4. 12.** La visualisation sur PowerShell.

Etape 02 : Après la première étape de la visualisation sur PowerShell on suit maintenant les étapes suivantes :

- L'ouverture de la commande (CMD) sur notre système.
- L'exécution de la commande python client.py pour démarrer le programme client.
- Le programme client va alors initialiser la webcam connectée à notre système.
- La webcam commence à capturer des images en temps réel.
- Sauvegarder toute les images capturer dans un fichier appeler Image.
- Lorsqu'un visage est détecté s'il était dans sa BDD il affiche son prénom, sinon il mentionne que c'est un visage inconnu.

```
C:\Users\AN\Nour Robot\client>client.py start
sending message: start
```

**Figure 4. 13.** Lancement du programme à partir du fichier client.py.



**Figure 4. 14.** Alimentation de la webcam.

```
start
starting recognition loop
Found camera
no face
no face
no face
no face
no face
no face
no face
no face
no face
no face
no face
no face
no face
no face
no face
no face
no face
no face
no face
no face
saving...
```

**Figure 4. 16.** Enregistrement des images

```
no face
no face
no face
no face
unknown face: Unknown
no face
unknown face: Unknown
no face
no face
no face
no face
no face
no face
no face
no face
no face
saving...
no face
```

**Figure 4. 15.** Capturer par la webcam d'un visage inconnu.

```
found face: nour
found face: nour
found face: nour
found face: nour
found face: nour
found face: nour
found face: nour
found face: nour
found face: nour
saving...
```

**Figure 4. 17.** Détection d'un visage connu avec affichage de son prénom.

- Une fois nous voulons récupérer les images prises durant toute la durée que nous étions pas présent il suffit juste de taper la commande « getpictures».

```
C:\Users\AN\Nour Robot\client>client.py getpictures
New message with 3 length
3
Receiving 3 images.
New message with 19 length
19
05292023_130027.png
New image with 319423 length
New message with 19 length
19
05292023_124015.png
New image with 331102 length
New message with 19 length
19
```

Figure 4. 18. Récupération des images prises.

- Après avoir reçu les photos capturer on ouvre le fichier image.



Figure 4. 19. Photos réels prises par le robot.

## IV.6. Conclusion

Dans ce chapitre, nous avons expliqué en détail le fonctionnement du système de reconnaissance faciale. Nous avons combiné diverses bibliothèques logicielles telles que Dlib, face\_recognition et OpenCV pour reconnaître les visages, extraire les points de repère faciaux et comparer les visages avec des bases de données. Nous avons aussi implémenté l'algorithme HOG en utilisant le modèle SVM.

Les résultats obtenus lors des tests ont montré une grande précision et un taux de réussite élevé dans la reconnaissance et l'identification des visages.

# **Conclusion Générale**

### Conclusion générale

Notre projet de réalisation et de commande d'un robot mobile autonome de surveillance à base d'une carte Raspberry Pi a été une expérience passionnante et enrichissante dans le domaine de la robotique. Nous avons exploré différents aspects, tant du côté matériel que logiciel, pour concevoir un robot capable de surveiller et de détecter des objets.

Nous avons commencé par introduire le domaine de la robotique et l'importance croissante de cette technologie dans notre société. Nous avons ensuite examiné le concept du système Rocker-Bogie, qui offre une mobilité efficace et stable sur différents terrains.

En ce qui concerne le côté matériel, nous avons utilisé la carte Raspberry Pi 4 comme cerveau de notre robot, offrant une grande flexibilité en termes de connectivité et de contrôle. Nous avons intégré des capteurs tels que le MPU6050 pour la détection des mouvements, le HC-SR04 pour l'estimation de la distance par rapport aux obstacles, ainsi qu'une webcam pour la capture d'images.

Du côté logiciel, nous avons exploré des algorithmes de reconnaissance faciale pour identifier les personnes enregistrées dans la base de données. Nous avons également utilisé des bibliothèques de vision par ordinateur telles qu'OpenCV, Dlib et Face\_recognition pour la détection d'objets et le traitement d'images.

L'objectif principal de ce projet de fin d'étude était de réaliser un robot explorateur de type Rocker-Bogie à commande sans fils. Cet objectif a été atteint, mais comme tout œuvre humaine, ce travail reste perfectible. Nous proposons donc un certains perspectives et améliorations à apporter que nous n'avons pas eu la chance de les accomplir faute de temps et de moyens.

Ces perspectives se résument dans les points suivants :

- Utilisation de capteurs plus performants comme les capteurs laser (LIDAR) qui permet une meilleur localisation et détection des dimensions de l'environnement du robot ce qui donne une meilleure prise de décision dans le choix de trajectoire automatique.
- Intégration de la connectivité Internet : En ajoutant une connectivité Internet, le robot pourrait envoyer des notifications en temps réel, stocker les données collectées dans le cloud et être contrôlé à distance depuis n'importe où dans le monde.

## Conclusion générale

---

- Intégration de la vision nocturne ou de capteurs infrarouges : Pour améliorer la surveillance dans des conditions de faible luminosité ou la détection d'intrusions nocturnes, on peut ajouter des capteurs de vision nocturne ou des capteurs infrarouges à notre robot. Cela permettra une surveillance continue et efficace, indépendamment des conditions d'éclairage.



# **Références Bibliographiques**

### Références Bibliographiques

- [1] Etude et conception d'un Robot marchant par HEDDOUCHE Kamel Université Université Mohamed Khider Biskra, année 2014.
- [2] H. Joachim et R. Joachim, localization of a mobile robot by matching 3D laser-rangeimage and preicated sensor imges, October 24-26 éd, Proceedings of the vehicules, 1994 ;pp 345-350.
- [3] TIM GODISAR, DANIEL CARLTON SMITH, KYLE SWISHER, « LUNABOTICS DRIVE SYSTEM DESIGN PROPOSAL », Lane Department of Computer Science and Electrical Engineering ,2011.
- [4] Introduction à la robotique par Laetitia Matignon Université de Caen, France.
- [5] Methode de trackage des robot mobiles, In proceedings of the 1998 IEEE international, 1998.
- [6] Pierre Tournassoud, Planification de trajectoire et contrôle en robotique, application aux robots mobiles et manipulateurs, Bibliothèque de la faculté de technologie, université de TLEMCEM, 1992.
- [7] J. José, B. Gonzales, Vision Panoramique pour la robotique mobile : stéréovision et localisation par indexation d'images, Toulouse : Thèse de Doctorat de l'Institut National Polytechnique de Toulouse, 2004.
- [8] SLIMANE Nouredine <<SYSTEME DE LOCALISATION POUR ROBOTS MOBILES>> SPECIALITE : CONTROLE Discipline : Robotique Pour l'obtention du grade de DOCTEUR D'ETAT DE L'UNIVERSITE DE BATNA Soutenu le : 23 / 11 / 2005
- [9] Tropato Jean-Charles, « La robotique industrielle et de service : La place de l'Italie dans le Monde », Rapport d'étude. Ambassade de France en Italie, 2011.
- [10] FILLIAT David, « Robotique Mobile », Ecole Nationale superieur de Techniques avancées, Paris Tech, 2013.
- [11] Captain Nick Ham, Oregon Wing, Director of Aerospace Education, INTRODUCTION TO ROBOTIC, 2011.

## Références Bibliographiques

---

- [12] H. Joachim et R. Joachim, Localization of a mobile robot by matching 3D laser-range-images and predicted sensor images, October 24-26 éd., Proceedings of the Intelligent Vehicles, 1994, pp. 345-350.
- [13] B. Bayle, « Robotique Mobile », Ecole Nationale Supérieure de Physique de Strasbourg, Université Louis Pasteur, 2007.
- [14] Captain Nick Ham, Oregon Wing, Director of Aerospace Education, INTRODUCTION TO ROBOTIC, 2011.
- [15] A. Shibata, T. Wada, T. Tanie. "RoboCare: An Integrated Approach to Assisted Living Based on a Mobile Robot." In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2001.
- [16] M. Alipour, W. J. Wolfe, C. Atkeson. "FIREBOT: A Firefighting Robot for Structure Fires." In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2006.
- [17] D. Kragic, L. Hager. "SentryBot: A Robotic Surveillance System for Intruder Detection". In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2004.
- [18] P. Panigrahi, A. Barik, Rajneesh R. & R. K. Sahu, "Introduction of Mechanical Gear Type Steering Mechanism to Rocker Bogie", Imperial Journal of Interdisciplinary Research (IJIR) Vol-2, Issue-5, ISSN: 2454-1362, 2016.
- [19] Fabrice DESCHAMP « Cours BAC S SI – Convertir L'énergie – Machine à Courant Continu » Sciences de l'ingénieur. Lycée Jacques Prevert.
- [20] Austin Hughes, "Electric Motors and Drives: Fundamentals, Types and Applications," Newnes, 2013.
- [21] T.J.E. Miller, "Direct Current Machines," Butterworth-Heinemann, 2012.
- [22] BELKACEM, A., DIDOUH, M. « Commande à distance d'un robot mobile via un Smartphone Androïde ». Mémoire de Master en Réseaux mobiles et services de télécommunications. Université Aboubakr Belkaïd de Tlemcen. 12/06/2017.
- [23] Paolo Aliverti, "Motor Control Shield with L298N," Arduino Srl, N/A.

## Références Bibliographiques

---

[24] Krishna Rungta, "Motor Control Using L298N: Understanding and Implementing H-Bridge Motor Drivers," Independently published, 2019.

[25] Richard L. Boyle, Thomas E. Love, "Servo Motors and Industrial Control Theory," CRC Press, 2017.

[26] Kevin M. Lynch, Frank C. Park, "Modern Robotics: Mechanics, Planning, and Control," Cambridge University Press, 2017.

[27]"Buzzer and Bell How-To Guide" by Jameco Electronics

[28] Simon Monk, "Raspberry Pi Cookbook: Software and Hardware Problems and Solutions," O'Reilly Media, 2020.

[29] Matt Richardson & Shawn Wallace "Getting started with Raspberry Pi "publié par Maker Media Inc.2014

[30] R. p. magazine, THE OFFICIAL ESP32 PROJECTS BOOK VOLUME 2, Magpi, 2016.

[31] Espressif Systems, "ESP32 Technical Reference Manual," 2019.

[32] UCEC. "HC-SR04 Ultrasonic Distance Sensor." UCEC, N/A.

[33] ElecFreaks. "HC-SR04 Ultrasonic Sensor." ElecFreaks, N/A.

[34] O. DOUIBI, A.GHAMRI. «Réalisation d'un robot mobile autonome» Mémoire d'ingénieur d'état en électronique, Université Biskra, Juin 2003.

[35] DJEDOU, N., BOUKERN, D. « Étude et réalisation d'un RADAR de détection ». Mémoire de Master en Réseaux et Systèmes de Télécommunications. Université Aboubakr Belkaïd de Tlemcen. 19/06/ 2017.

[36] InvenSense Inc., "MPU-6050 Product Specification Revision 3.4," 2012.

[37] "Webcams - Digital Cameras & Camcorders." B&H Photo Video, <https://www.bhphotovideo.com/c/browse/Webcams-Digital-Cameras-Camcorders/ci/21225/N/4196380294> (consulté le 16 mai 2023).

[38] "Lithium Batteries and Other Electrochemical Storage Systems" de Pierre T. Amstutz et Laurence A. Girault.

## Références Bibliographiques

---

- [39] Design and Construction of a Power Bank with Lithium-ion Battery" - Par A. K. A. Akinlabi.
- [40] Michael Kerrisk, "The Linux Programming Interface: A Linux and UNIX System Programming Handbook," No Starch Press, 2010.
- [41] Eric Matthes, "Python Crash Course: A Hands-On, Project-Based Introduction to Programming," No Starch Press, 2019.
- [42] David Flanagan, "JavaScript: The Definitive Guide: Master the World's Most-Used Programming Language," O'Reilly Media, 2018.
- [43] Bruce Johnson, "Visual Studio Code: An Efficient IDE for Web Development," Packt Publishing, 2019.
- [44] Daniel J. Barrett, Richard E. Silverman, Robert G. Byrnes, "SSH, The Secure Shell: The Definitive Guide," O'Reilly Media, 2005.
- [45] Bjarne Stroustrup, "The C++ Programming Language," Addison-Wesley Professional, 2013..
- [46] Stan Z. Li, Anil K. Jain, "Handbook of Face Recognition," Springer, 2011.
- [47] John O. Cooper, Timothy E. Heron, William L. Heward, "Applied Behavior Analysis," Pearson, 2020.
- [48] Martin Treiber, Arne Kesting, "Traffic Flow Dynamics: Data, Models and Simulation," Springer, 2013.
- [49] IFPO (International Foundation for Protection Officers), "Security Supervision and Management: Theory and Practice of Asset Protection," Butterworth-Heinemann, 2015.
- [50] R. Lambert, «Différence entre Intelligence Artificielle, Machine Learning et Deep Learning,» 2019. [En ligne]. Available: <http://penseeartificielle.fr/difference-intelligence-artificiellemachine-learning-deep-learning/>. [Accès le 10 06 2023].
- [51] Shan Liu, Heming Sun, Wenjun Zeng, "Image and Video Processing in the Compressed Domain," CRC Press, 2014.
- [52] Asit Kumar Datta, Madhura Datta, Pradipta Kumar Banerjee, "Face Detection and Recognition: Theory and Practice," CRC Press, 2016.

## Références Bibliographiques

---

[53] Salvador García, Julián Luengo, Francisco Herrera, "Data Preprocessing in Data Mining," Springer, 2015.

[54] Mark S. Nixon, Alberto S. Aguado, "Feature Extraction and Image Processing," Academic Press, 2012.

[55] Richard O. Duda, Peter E. Hart, David G. Stork, "Pattern Classification," Wiley, 2000.

[56] Christopher M. Bishop, "Pattern Recognition and Machine Learning," Springer, 2006.

[57] Charles Yoe, "Principles of Risk Analysis: Decision Making Under Uncertainty," CRC Press, 2011.

[58] Bradski, G., & Kaehler, A. (2008). "Learning OpenCV: Computer Vision with the OpenCV Library." O'Reilly Media.

### Références des figures

- [1] Robotique Mobile – David FILLIAT, En vue de l'obtention du diplôme d'ingénieur, École Nationale Supérieure de Techniques Avancées ParisTech
- [2] Ishak MOUMENE, Etude comparative de quelques algorithmes d'évitement d'obstacles en robotique mobile, En vue de l'obtention du diplôme MASTER en Electronique, Université Abderrahmane Mira de Bejaia, 2013.
- [3] B. M. Claus et S. A. Claus , Optimal landmark selection for triangulation of robot position, vol. 23, Journal of Robotics and Autonomous Systems, 1998, pp. 277-292.
- [4] H. Joachim et R. Joachim, Localization of a mobile robot by matching 3D laser-range-images and predicted sensor images, October 24-26 éd., Proceedings of the Intelligent Vehicles, 1994, pp. 345-350.
- [5] Veruggio, Gianmarco, EURON Roboethics Roadmap, 2007.
- [6] P. Panigrahi, A. Barik, Rajneesh R. & R. K. Sahu, "Introduction of Mechanical Gear Type Steering Mechanism to Rocker Bogie", Imperial Journal of Interdisciplinary Research (IJIR) Vol-2, Issue-5, ISSN: 2454-1362, 2016.
- [7] Fabrice DESCHAMP « Cours BAC S SI – Convertir L'énergie – Machine à Courant Continu » Sciences de l'ingénieur. Lycée Jacques Prevert.
- [8] Rungta, Krishna. "Motor Control Using L298N: Understanding and Implementing H-Bridge Motor Drivers." Independently published, 2019.
- [9] Texas Instruments. "L298N - Dual Full-Bridge Driver." Texas Instruments Datasheet, N/A.
- [10] TutorialsPoint. "L298N Motor Driver." TutorialsPoint, [https://www.tutorialspoint.com/l298n\\_motor\\_driver/index.htm](https://www.tutorialspoint.com/l298n_motor_driver/index.htm) (consulté le 19 mars 2023).
- [11] "L298N Motor Driver Tutorial." Circuit Digest, <https://circuitdigest.com/microcontroller-projects/l298n-motor-driver-arduino-tutorial> (consulté le 20 mars 2023).
- [12] "Servo Motors." Adafruit Learning System, <https://learn.adafruit.com/adafruit-arduino-lesson-14-servo-motors> (consulté le 18 février 2023).

## Références Bibliographiques

---

- [13] "Buzzer and Bell How-To Guide" by Jameco Electronics
- [14] Simon Monk, "Raspberry Pi Cookbook: Software and Hardware Problems and Solutions," O'Reilly Media, 2020.
- [15] M.Legouge, «Quel OS installer sur un Raspberry Pi,» 2019. [En ligne]. Available: <https://www.clubic.com/raspberry-pi/article-850017-1-os-installer-raspberry-pi.html>. [Accès le 01 07 2019].
- [16] K.Skaarhoj, «systeme exploitation ESP32,» 2018. [En ligne]. Available: <https://www.ionos.fr/digitalguide/serveur/know-how/systeme-exploitation-ESP32/>. [Accès le 15 04 2023].
- [17] "ESP32 Technical Reference Manual." Espressif Systems, [https://www.espressif.com/sites/default/files/documentation/esp32\\_technical\\_reference\\_manual\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32_technical_reference_manual_en.pdf) (consulté le 15 Avril 2023).
- [18] "Ultrasonic Sensors." MaxBotix Inc., [https://www.maxbotix.com/Ultrasonic\\_Sensors.htm](https://www.maxbotix.com/Ultrasonic_Sensors.htm) (consulté le 24 mai 2023).
- [19] "Ultrasonic Sensor Working Principle and Applications." Circuit Digest, <https://circuitdigest.com/tutorial/ultrasonic-sensor-working-principle> (consulté le 24 mai 2023).
- [20] "MPU6050 Accelerometer and Gyroscope Sensor." Adafruit Learning System, <https://learn.adafruit.com/mpu6050-accelerometer-gyro-on-raspberry-pi> (consulté le 24 mai 2023).
- [21] "Webcams - Digital Cameras & Camcorders." B&H Photo Video, <https://www.bhphotovideo.com/c/browse/Webcams-Digital-Cameras-Camcorders/ci/21225/N/4196380294> (consulté le 16 mai 2023).
- [22] "Lithium-Ion Batteries: Fundamentals and Applications" par Yuanhua Lin et Kunzhang Zhu - un livre qui couvre les principes de base des batteries au lithium, les matériaux des électrodes et des électrolytes, la conception et l'ingénierie des batteries, les applications et les tendances futures.
- [23] Yong Liu, "Power Bank Technology and Application," Springer, 2017.



## Références Bibliographiques

---

- [24] "Autonomous Obstacle Avoidance: An Overview." Robotics Business Review, <https://www.roboticsbusinessreview.com/rbr50/insights/autonomous-obstacle-avoidance-overview/> (consulté le 2 juin 2023).
- [25] "PID Controller - A Brief Introduction." Control Guru, <https://controlguru.com/pid-basics/pid-controller-a-brief-introduction/> (consulté le 2 juin 2023).
- [26] Michael Kerrisk, "The Linux Programming Interface: A Linux and UNIX System Programming Handbook," No Starch Press, 2010.
- [27] "Python.org - Official Website." Python Software Foundation, <https://www.python.org/> (consulté le 2 juin 2023).
- [28] "JSON.org - The Official JSON Website." JSON.org, <https://www.json.org/> (consulté le 3 juin 2023).
- [29] "Visual Studio Code - Code Editing. Redefined." Visual Studio Code, <https://code.visualstudio.com/> (consulté le 2 juin 2023).
- [30] "Facial Recognition - Technology Overview." Electronic Frontier Foundation, <https://www.eff.org/pages/facial-recognition> (consulté le 18 juin 2023).
- [31] "Behavior Analysis Certification Board." Behavior Analysis Certification Board, <https://www.bacb.com/> (consulté le 20 juin 2023).
- [32] "Traffic Analysis and Management - Federal Highway Administration." Federal Highway Administration, <https://www.fhwa.dot.gov/trafficinfo/> (consulté le 20 juin 2023).
- [33] "Surveillance for Safety - National Safety Council." National Safety Council, <https://www.nsc.org/safety/safety-topics/surveillance> (consulté le 20 juin 2023).
- [34] "Image Acquisition and Processing with LabVIEW," National Instruments, <https://www.ni.com/fr-fr/shop/select/image-acquisition-and-processing-with-labview-book>.
- [35] "A Gentle Introduction to Object Detection With Deep Learning," Towards Data Science, <https://towardsdatascience.com/a-gentle-introduction-to-object-detection-with-deep-learning-7798521805a>.

## Références Bibliographiques

---

[36] "A Comprehensive Guide to Feature Extraction Techniques in Machine Learning," Analytics Vidhya, <https://www.analyticsvidhya.com/blog/2021/09/a-comprehensive-guide-to-feature-extraction-techniques-in-machine-learning/>.

[37] "OpenCV Documentation" - <https://docs.opencv.org/>