

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne démocratique et populaire

وزارة التعليم العالي و البحث العلمي
Ministère de l'enseignement supérieur et de la recherche scientifique

جامعة سعد دحلب البليدة
Université SAAD DAHLAB de BLIDA

كلية التكنولوجيا
Faculté de Technologie

قسم الإلكترونيك
Département d'Électronique



Mémoire de Master

Mention Électronique
Spécialité Automatique

présenté par

SI-AHMED Yasmine

&

OUCHEFOUN Nassima

Conception d'un robot mobile avec vision intelligente embarquée

Proposé par : Mr. KAZED Boualem

Année Universitaire 2016-2017

C'est un devoir agréable d'exprimer en quelques lignes la reconnaissance qu'on doit à tous ceux qui ont contribué de loin ou de près à l'élaboration de ce travail, qu'ils trouvent ici nos vifs respects et nos profondes gratitude.

En premier lieu, on remercie Dieu, le tout puissant, pour nous avoir donné, la patience, la volonté et la force nécessaires pour achever ce travail.

On tient à remercier Monsieur B. KAZED pour son encadrement fructueux, ses remarques et suggestions, mais aussi par ses encouragements et son suivi.

On remercie les membres du jury pour avoir bien voulu examiner et juger ce travail.

Enfin, On tient à remercier tous ceux, qui nous ont aidé, soutenu et encourager aux moments opportuns. A tous, un grand MERCI !

Dédicace

Je dédie ce mémoire à mes chers parents ma mère Nadia et mon père Youssef pour leur patience, leur amour, leur soutien et leurs encouragements.

A ma très chère Katie.

A mes sœurs, Imy et Naima.

A mes frères Zizou et A.Latif.

A mon très cher Dady.

Merci d'avoir été toujours là pour moi.

Yasmine.

Dédicace

Je dédie ce mémoire à mes chers parents ma mère OURDIA et ma grand mère FATIMA pour leur patience, leur amour, leur soutien et leurs encouragements.

A ma chère sœur, IMANE

A mon frère ISMAIL

A mes chers tantes et oncles

A mes amis et mes camarades de classe

Merci d'avoir été toujours là pour moi.

Nassima.

ملخص: هذا المشروع يتضمن تصميم ، تنفيذ و التحكم في الروبوت المتحرك. من اجل كشف و تتبع الاجسام المتحركة في الفيديو حسب لونها. و التعرف على ملامح الوجه. هذا و بفضل الرؤية بالكمبيوتر الذي يسمح للروبوت بفهم ما يرى عند الاتصال بالرازبري باي و الكاميرة الخاصة به. سيمثل الكمبيوتر الراس المدبر للروبوت يتكفل بتتبع الجسم ذات اللون الازرق، التعرف على ملامح الوجه ثم حساب احداثيات الموقع و المسافة ليتم تحويلها الى سرعة. ارسالها الى الاردوينو الذي ستتكلف باستلام الاوامر و تنفيذها و ذلك من خلال تشغيل المحركات ليتم التتبع من اجل تحقيق متطلبات المشروع.

كلمات المفاتيح التعرف على ملامح ألوجه، كمبيوتر الرازبري باي، معالجة الصور بالكمبيوتر ،روبوت متحرك

Résumé : Ce projet concerne la conception, la réalisation et la commande d'un robot mobile à deux roues motrices, en temps-réel pour la détection, le suivi d'objets, et la reconnaissance faciale à l'aide de la bibliothèque OpenCV conçue pour la vision par ordinateur.

L'approche adoptée consiste à utiliser le nano ordinateur Raspberry Pi connecté en CSI avec son module caméra. Cet ordinateur se charge d'effectuer les différents traitements d'images nécessaires sur le flux vidéo diffusé, et transmettre la position calculée de l'objet traqué à la carte Arduino relié par le bus I²C en esclave afin de le capturer et l'emmener à la personne connue trouvée.

Mots clés : Robot mobile; OpenCV ; Suivi d'objets; Reconnaissance faciale; Raspberry Pi ; Traitements d'images ; le bus I²C

Abstract: This project concerns the design, the production and the command of a mobile robot with two wheels, for the detection, object tracking and the facial recognition in real time with the help of the OpenCV library designed for the computer vision.

The adopted approach here requires the usage of the nano computer raspberry pi connected in CSI with its own camera module. This computer is responsible to make the different images processing needed on the video streams, and transmitting the calculated position of the tracked object to the Arduino board connected by the i²c bus as a slave in order to capture it and then take it to the recognized person

Keywords : Mobile robot; object tracking; facial recognition; OpenCV; Raspberry pi ; Images processing ; I²C bus.

Listes des acronymes et abréviations

USB: Universal Serial Bus

CD-Rom: Compact Disc - Read Only Memory

Pixels: *Picture Elements*

BGR: Blue, Green, Red

BVR: Bleu, Vert, Rouge

HSV: Hue, Saturation, Value

CC: courant Continue

Tr/min: Tour/minute

V: Volts

mA: milliampère

Cm: Centimètre

°C: °Celsius

PWM: Pulse Width Modulation

MHz: Mega hertz

KB: Kilo Bytes

KO: kilo Octets

SoC: system on Chip

CSI: Camera Serial Interface

MP: Mega Pixels

HD: Haute Definition

FPS: Frame per Second

OpenCV: *Open source Computer Vision*

BSD: *Berkeley Software Distribution license*

IDE: Integrated Development Environment

I²C: *Inter Integrated Circuit*

Mac OS: *Macintosh Operating System*

openBR : *Open source Biometrics*

Table des matieres

Chapitre 1 : Généralités sur la robotique mobile

1.1 Introduction	2
1.2. Les robots mobiles	3
1.2.1. Présentation générale des robots mobiles	3
1.2.2. Classification des robots mobiles	4
1.2.3. Les robots mobiles à roues	5
a. La classe des robots mobiles à roues	6
a.1. Robot unicycle	6
a.2. Robot tricycle	6
a.3. Robot de type voiture	6
a.4. Robot à roues omnidirectionnelles.....	6
1.2.4. Robot mobile autonome	7
a. La perception de l'environnement d'évolution	7
b. La décision	7
c. La commande du mouvement	8
1.2.5. La navigation des robots mobiles autonomes	8
a. La navigation	8
b. Les types de navigation	9
b.1. La navigation réactive	9
b.2. La navigation par carte	9
1.3. Conclusion	9

Chapitre 2 : Traitement numérique des images.

2.1. Introduction.....	11
2.2. Les images numériques	11
2.3. L'espace des couleurs	12
2.3.1. L'espace de couleur BGR.....	12
2.3.2. L'espace de couleur HSV.....	12
2.3.3. L'espace de couleur niveau de gris « grayscale ».....	12

2.4.	La détection d'objets selon la couleur en temps réel	13
2.4.1.	La conversion de l'image de l'espace BGR à HSV	13
2.4.2.	La binarisation	14
2.4.3.	Le masque de l'objet	14
2.4.4.	Opérations sur l'image binaire	15
a.	Ouverture morphologique.....	15
2.4.5.	Difficultés de la méthode.....	15
2.5.	La reconnaissance du visage	16
2.5.1.	La détection de visages dans une vidéo	16
2.5.2.	La normalisation	17
2.5.3.	Extraction des caractéristiques et classification	17
2.5.4.	La décision	18
2.6	Conclusion	18

Chapitre 3 : Présentation générale du projet.

3.1.	Introduction	20
3.2.	Les Solutions matérielles	20
3.2.1.	Châssis et roues	20
3.2.2.	L'alimentation électrique	21
3.2.3.	Les moteurs électriques	21
a.	Les moteurs MAXON	21
b.	Le moteur LEGO	22
3.2.4	La pince	22
3.2.5.	Le servomoteur	23
3.2.6.	Le capteur GP2D15	23
3.2.7.	Les cartes électroniques programmables	24
a.	La carte Arduino	24
b.	Le Raspberry Pi	27
b.1.	Le module caméra Raspberry Pi	28
c.	La répartition des taches	29
3.2.8.	La carte puissance	30
3.3.	Solutions logicielles	31

3.3.1. Pour le Raspberry Pi.....	31
a. Le langage de programmation PYTHON	31
b. La bibliothèque OpenCV	31
3.3.2. Pour la carte Arduino	32
3.4. Communication entre les deux cartes	33
3.4.1. Introduction	33
3.4.2. Transmission des données	33
3.4.3. Mode de transmission des données	33
a. Le sens des échanges de données.....	34
b. Le mode de transmission des données	35
c. La synchronisation	36
3.4.4. Les protocoles de communication	36
a. Présentation générale du protocole I ² C	36
b. Raccordement du protocole I ² C	37
c. Le protocole I ² C	38
3.5. Conclusion	41

Chapitre 4 Résultats du projet

4.1. Introduction	43
4.2. Détection et suivi d'objets de couleur	43
4.2.1. introduction	43
4.2.2. La détection de couleur avec OpenCV	43
a. Organigramme de la détection de couleur	44
b. Résultats obtenus	45
c. Les problèmes de la détection de couleur	46
4.3. La reconnaissance faciale	47
4.3.1 Introduction	47
4.3.2. La détection de visages avec OpenCV	47
a. La méthode de Viola et Jones	47
b. Organigramme de détection de visages	49
c. Résultats obtenus	50
d. Les problèmes de la détection de visages	51
d.1. Le temps de calcul	51
d.2. La rotation	52

4.3.3. La reconnaissance de visages avec OpenCV	53
a. Le choix de la méthode	53
b. La base de données	53
c. Organigramme de la reconnaissance faciale	55
d. Résultats obtenus	56
e. Les problèmes de la reconnaissance de visages	57
f. Conclusion	57
Conclusion générale et perspectives	59
Référencés bibliographiques	61

Liste des figures

Chapitre 1

Figure 1.1. Les robots mobiles dans les différents domaines.....	4
Figure1.2. Les classes des robots mobiles à roues : roues unicycle, tricycle, voiture, omnidirectionnelle.....	7

Chapitre 2

Figure 2.1.Exemple de pixels d'une image numérique en couleur	11
Figure 2.2.Exemple d'une séquence d'images	13
Figure 2.3. Les différents types de binarisation (seuillage)	14
Figure 2.4. Principe de fonctionnement de base d'un système de reconnaissance faciale...	16
Figure 2.5 : Les caractéristiques générales d'un visage humain.....	17

Chapitre 3

Figure3.1. Le robot mobile	21
Figure 3.2. Les moteurs MAXON à courant continu	22
Figure 3.3. Le moteur type LEGO	23
Figure 3.4. La pince fermée et ouverte.	23
Figure 3.5. Le servomoteur type HITEC	24
Figure 3.6. La carte Arduino Méga.....	26
Figure 3.7. La carte Arduino Uno.	26
Figure 3.8. La carte Raspberry Pi 3 model B.....	29
Figure 3.9. Le module camera Raspberry Pi V1.3.....	29
Figure 3.10. La carte puissance L298.....	31
Figure 3.11. Le mode simplexe, Half-duplex et full-duplex.....	35
Figure 3.12 le mode parallèle et le mode série pour la transmission des données.....	36
Figure 3.13 Raccordement du protocole I ² C	39
Figure.3.14. Le chronogramme des lignes SDA et SCL pour le bus I ² C.....	39

<i>Figure 3.15. Validité des données en fonction des signaux SDA et SCL.....</i>	<i>40</i>
<i>Figure 3.16. La condition d'acquittement ACK ou de non-acquittement NACK.....</i>	<i>42</i>

Chapitre 4

<i>Figure 4.1. Résultats obtenus.....</i>	<i>46</i>
<i>Figure 4.2. L'amélioration de l'image binaire.....</i>	<i>47</i>
<i>Figure 4.3. Détection de visages</i>	<i>51</i>
<i>Figure 4.4. Problèmes de détection de visages</i>	<i>52</i>
<i>Figure 4.5. La base de données</i>	<i>54</i>
<i>Figure 4.6. La reconnaissance faciale du système</i>	<i>56</i>

Liste des tableaux

Chapitre 1 :

Tableau 1.1. Domaine d'application de la robotique mobile.....5

Chapitre 3 :

Tableau 3.1. Les caractéristiques techniques du capteur GP2D15.....24

Tableau 3.2. Les caractéristiques techniques des carte Arduino Uno et Méga.....26

Tableau 3.3 Les caractéristiques su Raspberry Pi 3 model B27.

Tableau 3.4 Les caractéristiques du module caméra Raspberry Pi v1.3.....29

Tableau 3.5. . Caractéristique L298N.....30

Chapitre 1 :

Généralités sur la robotique

mobile

« Parfois c'est dans les généralités qu'il y'a le plus de contenu, parce qu'elles ne ligotent pas notre imagination, et lui permettent de vagabonder à sa guise »

Anna la douce - - Dezső Kosztolányi-

CHAPITRE 1 : Généralités sur la robotique mobile

1.1. Introduction

L'homme a toujours eu peur d'accomplir des tâches dangereuses, le dégoût d'accomplir des tâches sales, et surtout la flemme d'accomplir les tâches répétitives sans cesse, c'est pour cela qu'il devait penser très vite à une solution pour remplacer sa présence indispensable. Pour mieux faire il fallait imiter au maximum le système humain tout en éliminant ses défauts : la fatigue, la peur, la flemme, les capacités etc.

Le besoin, et la volonté des humains de trouver ces substituts pour leurs présences dans des situations où ces derniers sont incapables d'effectuer le travail, des situations plus dangereuses, de haute précision ou répétitives ont été parmi les principales motivations de recherches dans les systèmes autonomes et la robotique qui est considérée aujourd'hui comme la source la plus importante d'innovations, susceptibles d'améliorer la productivité et la sécurité dans les secteurs militaires et industriels.

Au croisement de la mécanique, l'électronique et l'informatique qu'on appelle mécatronique avec l'automatique, la vision, le design et pleins d'autres disciplines on retrouve la robotique, la science qui s'intéresse aux robots.

Les robots aujourd'hui ont un impact considérable sur la vie moderne, de la fabrication industrielle aux soins de santé, le transport et l'exploration de l'espace jusqu'au plus profond de la mer.

En robotique, on distingue deux grands pôles d'intérêt, la robotique de manipulation et la robotique mobile. Les robots manipulateurs ont une base fixe on les retrouve dans les chaînes d'assemblage industrielles (peintures, soudage, emboutissage, etc.) contrairement aux robots mobiles qui peuvent se déplacer.

CHAPITRE 1 : Généralités sur la robotique mobile

1.2. Les robots mobiles

En définissant le mot « robot » de la manière la plus simple on peut dire que c'est un appareil automatique qui peut effectuer des fonctions normalement effectuées par des humains.

1. 2.1. Présentation générale des robots mobiles

Un robot mobile est une machine automatique capable de se mouvoir dans un environnement donné. Cette catégorie englobe tous les robots à base mobile, ils en existent plusieurs types, ils sont généralement classifiés selon le degré d'autonomie c'est-à-dire les capacités décisionnelles afin d'atteindre les buts qui lui ont été assignés, les types de locomotion, tout ce qui concerne les moyens de déplacements d'un point à un autre, et enfin le domaine d'application c'est à dire le milieu dans lequel ils évoluent sur terre, dans les airs ou encore sous l'eau.

Le terme "robots mobiles" bien que désignant l'ensemble des robots à bases mobiles, est généralement employé pour désigner les robots mobiles terrestres à roues. Les autres robots mobiles sont en effet le plus souvent désignés par leur type de locomotion, et leurs domaines d'application: les robots marcheurs, marins, sous-marins, drones volants, militaires, agricoles etc.

L'intérêt indéniable de la robotique mobile réside dans sa mobilité qui ouvre des applications dans de nombreux domaines et qui nous a permis d'augmenter considérablement nos connaissances sur la localisation et la navigation de systèmes mobiles autonomes.

CHAPITRE 1 : Généralités sur la robotique mobile

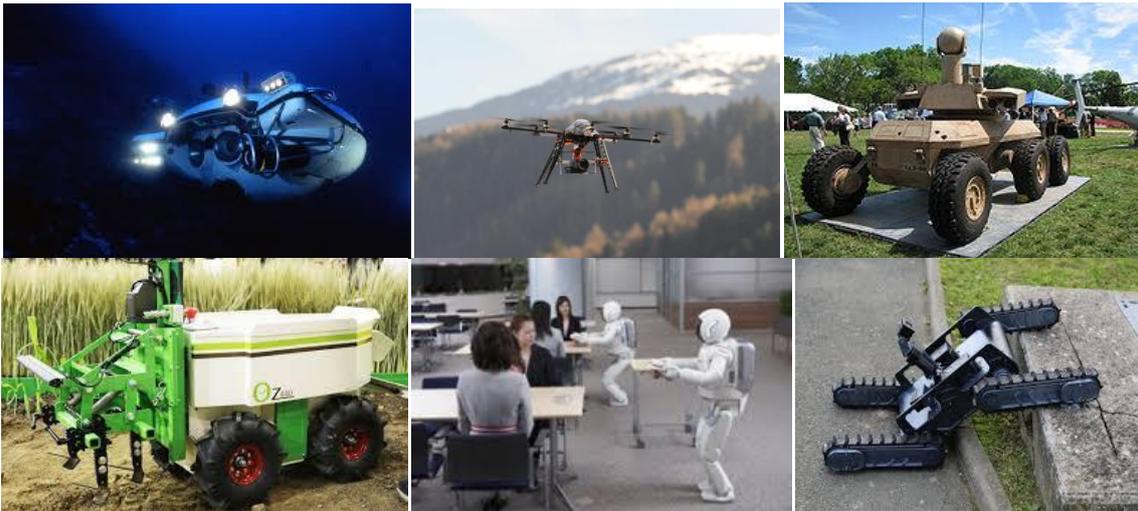


Figure1.1. Les robots mobiles dans les différents domaines.

1. 2.2 Classifications des robots mobiles

Comme dit précédemment, la classification des robots mobiles se fait suivant plusieurs critères (degré d'autonomie, système de locomotion, domaine d'application).

- *degré d'autonomie* :

Le robot doit prendre ses propres décisions, cela signifie qu'il doit être capable à la fois de percevoir correctement son environnement, mais également de savoir comment réagir en conséquence, suivant son niveau d'autonomie.

- *système de locomotion* :

Dans le domaine de la robotique mobile, l'étude et la structure de locomotion de la plateforme utilisée, possèdent une place prépondérante afin de garantir une bonne performance du robot mobile autonome en extérieur. De nombreuses approches et solutions peuvent être envisagées, il convient d'apporter un soin particulier quant à leur sélection.

CHAPITRE 1 : Généralités sur la robotique mobile

- *Domaine d'application*

Domaine d'applications	Exemples
Sécurité civile	Neutralisation d'activé terroriste Démunage
Militaire	Surveillance, patrouille Pose d'explosifs
Chimique	Manipulation de matériaux toxiques surveillance de site
Médecin	Aide aux handicapés physiques, aux malades Assistance d'urgence
Lutte contre l'incendie	Aide aux handicapés physiques, aux aveugles Assistance d'urgence
Sous-marine	Recherche de navires immergés Inspection des fonds marins
Agricole	Traite, moisson, traitement des vignes Cueillette de fruits
Espace	Exploration
Industriel	Convoyage surveillance
Industrie nucléaire	Manipulation de matériaux radioactifs Surveillance de sites

Tableau 1.1. Domaine d'application de la robotique mobile.

1. 2.3. Les robots mobiles à roues

Dans ce mémoire on s'intéresse aux robots terrestres autonomes à roues. La mobilité par roues est la structure mécanique la plus adaptée, cette structure assure selon l'agencement et les dimensions des roues un déplacement dans toutes les directions avec une accélération et une vitesse importante.

a. La classe des robots mobiles à roues

Il existe plusieurs classes de robots à roues déterminées, généralement, par la position et le nombre de roues utilisées. Nous citerons ici les quatre classes principales :

CHAPITRE 1 : Généralités sur la robotique mobile

a. 1. Robot unicycle

Un robot mobile de type unicycle est actionné par deux roues motrices indépendantes à l'arrière, et possède éventuellement une roue folle à l'avant (roue non commandée) destinée uniquement à assurer la stabilité du système. Le mouvement est donné par la vitesse des deux roues, son centre de rotation est situé sur l'axe reliant les deux roues motrices. Figure [1.2. (a)]

a. 2. Robot tricycle

Un robot mobile de type tricycle est constitué de deux roues fixes placées sur un même axe et d'une roue centrée orientable placée sur l'axe longitudinal. Le mouvement du robot est donné par la vitesse des deux roues fixes et par l'orientation de la roue orientable. Son centre de rotation est situé à l'intersection de l'axe contenant les roues fixes et de l'axe de la roue orientable. Figure [1.2. (b)]

a. 3. Robot de type voiture

Un robot mobile de type voiture est semblable au tricycle, il est constitué de deux roues fixes placées sur un même axe et de deux roues centrées orientables placées elles aussi sur un même axe. Le robot mobile de type voiture est cependant plus stable puisqu'il possède un point d'appui supplémentaire. Figure [1.2. (c)].

a. 4. Robot à roues omnidirectionnelles

Un robot mobile omnidirectionnel est un robot qui peut se déplacer librement dans toutes les directions. Il est en général constitué de trois roues décentrées orientables placées en triangle équilatéral. Figure [1.2. (d)].

CHAPITRE 1 : Généralités sur la robotique mobile

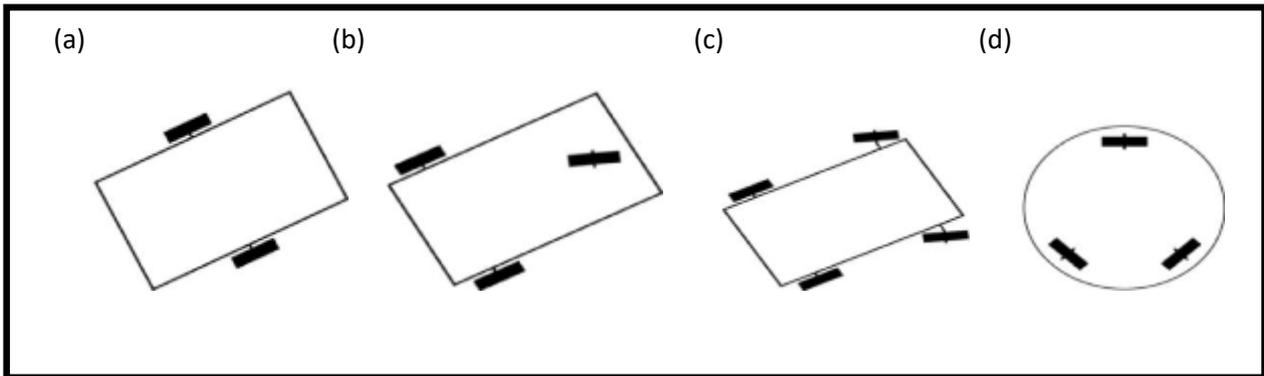


Figure 1.2. Les classes des robots mobiles à roues : roues unicycle (a), tricycle (b), voiture (c), omnidirectionnelle (d).

1. 2.4 Robots mobiles autonomes

Afin d'être autonome, un robot mobile doit posséder de nombreuses capacités de perception, de décision et d'action qui lui permettent d'agir de manière autonome dans son environnement en fonction de la perception qu'il en a et de ses buts, mais également de savoir comment réagir en conséquence suivant son niveau d'autonomie.

a. La perception de l'environnement d'évolution

La perception est la base de tout système autonome. En robotique, elle désigne la capacité du système à recueillir, traiter et mettre en forme des informations utiles au robot pour agir et réagir dans le monde qui l'entoure.

Pour cela, un robot est équipé de capteurs proprioceptifs qui fournissent des informations sur le robot lui-même, et extéroceptifs qui fournissent des informations sur ce qu'il y a autour de lui (son environnement).

b. La décision

Le robot doit décider de comment agir pour atteindre son but, doit être capable de se déplacer d'un point à l'autre en trouvant des chemins efficaces et sécuritaires afin d'éviter des collisions avec les obstacles.

CHAPITRE 1 : Généralités sur la robotique mobile

c. La commande du mouvement

Le robot doit actionner ses moteurs pour réaliser la trajectoire désirée, en fonction des décisions prises par l'étape de décision.

1. 2.5 La navigation des robots mobiles autonomes

La navigation autonome des robots est la capacité à évoluer sans aide dans l'environnement de travail. La complexité dépend donc de l'environnement dans lequel ces robots doivent évoluer.

Plusieurs domaines d'applications sont concernés par la navigation : systèmes de transport intelligent, services, assistance, surveillance etc.

a. La navigation

La navigation d'un robot mobile est un des grands problèmes clés de la communauté de la robotique mobile. Le problème consiste à intégrer sur un robot mobile réel, toutes les fonctions nécessaires pour qu'il puisse exécuter des déplacements à travers un environnement donné en utilisant des informations perçues par ses capteurs.

La navigation autonome doit pouvoir répondre aux trois mêmes questions que nous nous posons tous inconsciemment lors de nos déplacements :

- Où suis-je ?
- Où dois-je aller ?
- Comment m'y rendre ?

C'est au robot mobile de planifier son parcours, et de déterminer avec quels mouvements il va atteindre son objectif.

b. Les types de navigation

Les stratégies de navigation permettant à un robot mobile de se déplacer pour rejoindre son but, sont extrêmement diverses, de même que les classifications qui peuvent en être faites.

CHAPITRE 1 : Généralités sur la robotique mobile

Afin de situer les différentes méthodes de navigation, nous reprenons ici une classification établie par Trullier[1]. Cette classification présente l'avantage de distinguer les stratégies : sans modèles internes et les stratégies avec modèle interne.

Elle comporte cinq catégories cernée dans deux types de navigation : la navigation réactive et la navigation par carte :

b. 1. La navigation réactive :

Se basant sur les informations émanant des capteurs du robot plutôt, que sur la connaissance préalable de l'environnement externe. Ces méthodes présentent une bonne réactivité c'est à dire une bonne capacité à s'adapter à une situation ou un problème donné, et une bonne robustesse.

b. 2. La navigation par carte :

Cette stratégie autorise une navigation globale, elle permet de rejoindre un but arbitraire dans l'environnement. Elle s'appuie sur un modèle interne du monde (des lieux), une carte qui supporte une planification.

1. 3. Conclusion :

Dans ce chapitre, on a présenté quelques généralités sur la robotique mobile, spécialement à roues vu que c'est le type choisi pour la réalisation de ce projet. La navigation, et ces Types réactive et par carte en générale qui aident le robot à se mouvoir dans son environnement.

Chapitre 2

Traitement numérique des images

Chapitre 2 Traitement numérique des images

2.1. Introduction

Le traitement d'images est une science récente qui a pour but d'offrir aux spécialistes de différents domaines, comme au grand public, des outils de manipulation de ces données digitales issues du monde réel.

Dans cette partie nous nous attachons au problème de détection de suivi d'objet, visages et la reconnaissance faciale qui représentent des tâches préliminaires dans ce projet.

2.2 Les images numériques

Nous allons à présent manipuler des images numériques. Pour cela il est important de savoir que le terme désigne toute image stockée sur un support informatique : clé USB, disque dur, CD-Rom, téléphone portable, etc.

Elles sont constituées d'une multitude de pixels organisés en lignes (y) et colonnes (x). Chaque pixel possède une couleur.

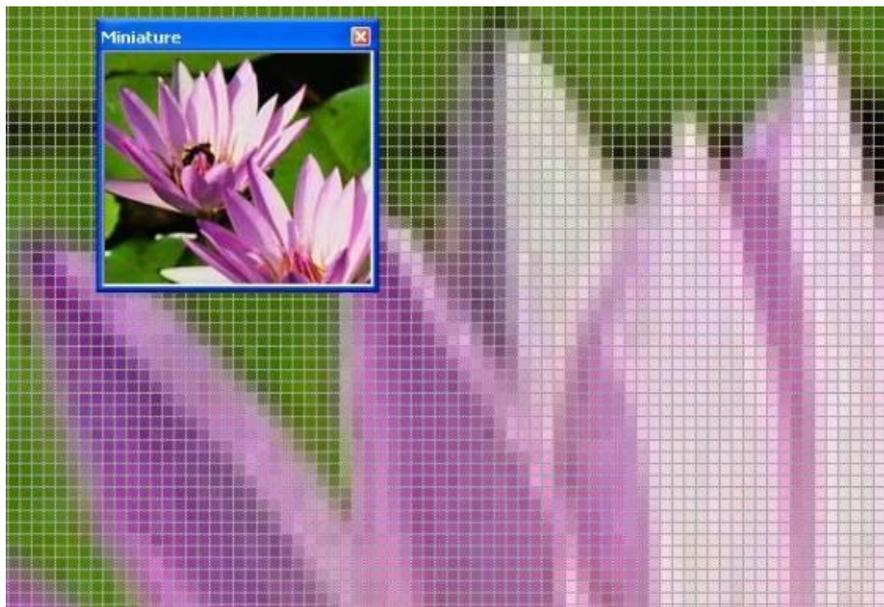


Figure 2.1. Exemple de pixels d'une image numérique en couleur.

2. 3. L'espace des couleurs

L'espace des couleurs est utilisé pour représenter les couleurs sous forme numérique. Dans ce chapitre nous allons utiliser l'espace BVR (Bleu, Vert, Rouge), TSV (Teinte, Saturation, Valeur) et niveau de gris (grayscale).

2.3.1. L'espace des couleurs BVR

C'est un modèle de couleur additif, il permet de définir plus de 16 millions de couleurs. Elles sont obtenues par une combinaison linéaire des valeurs des couleurs primaires : Bleu, Vert et Rouge. *L'intensité de chaque couleur est codée sur 1 octet permettant 256 niveaux différents. Ceci nous permet d'obtenir en fin du compte $256 \times 256 \times 256 = 16,777,216$ codes de couleurs différents pour chaque pixel.*

Pour des raisons historiques, il a été choisi l'espace par défaut de la bibliothèque OpenCV.

2.3.2. L'espace des couleurs TSV

C'est une autre façon de représenter une couleur, Le modèle TSV décrit les couleurs en termes de teinte, de saturation et de valeur (luminosité).

- *La teinte (T)* : Dans les systèmes de description de couleur informatique, la teinte est la forme pure d'une couleur, c'est-à-dire sans adjonction de blanc. Elle est codée suivant l'angle qui lui correspond sur le *cercle des couleurs* (de 0° à 360°).
- *Saturation (S)* : Décrit la quantité du blanc dans la couleur pure (0 à 100%).
- *La valeur (V)* : Décrit l'obscurité de la couleur, plus la *valeur* d'une couleur est faible, plus la couleur est sombre (de 0 à 100%).

2.3.3. L'espace de couleur niveau de gris « grayscale

Dans cet espace de couleur, les pixels de l'image sont représentés par plusieurs *niveaux de gris*. La valeur de luminosité de chaque pixel d'une image en ce mode est comprise entre 0 (noir) et 255 (blanc).

Chapitre 2 Traitement numérique des images

2.4. La détection d'objets selon la couleur en temps réel

Une séquence vidéo est composée d'images fixes défilant suffisamment vite pour que l'œil humain ne perçoive pas une succession d'images mais ait une illusion de mouvement. La cadence est de 25 images par seconde.

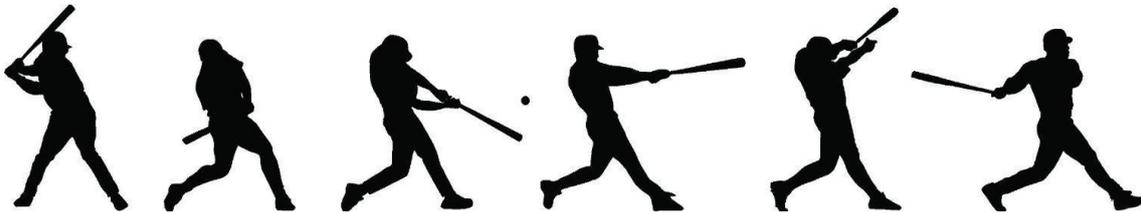


Figure 2.2. Exemple d'une séquence d'images.

La détection est souvent la phase initiale du suivi, plus coûteuse en temps de calcul que le suivi lui-même. Il existe, d'innombrables méthodes plus ou moins robustes, variantes, améliorées d'anciennes méthodes, ou nouvelles approches.

L'étape de détection consiste à trouver dans l'image, des caractéristiques qui pourraient correspondre aux objets que l'on souhaite suivre.

L'idée est d'utiliser un ensemble de traitements d'image pour extraire la couleur de la zone de l'objet, et le suivre.

2.4.1. La conversion de l'image de l'espace BVR à TSV

La bibliothèque OpenCV capture les images et les vidéos *en format BVR* par défaut. Dans des projets où la description de couleur joue un rôle essentiel, une conversion vers l'espace HSV est fortement conseillée, surtout que l'apparence de couleur d'un objet peut fortement varier avec l'éclairage de la scène, et au moment que *le BVR décrit les couleurs en termes de quantité de Bleu, vert et rouge* c'est-à-dire qu'il n'est pas le plus robuste aux variations des conditions, le modèle TSV quant à lui décrit les couleurs de façon similaire à la façon dont l'œil humain a tendance à les percevoir.

Chapitre 2 Traitement numérique des images

2.4.2. La binarisation

La binarisation appelée aussi seuillage est souvent la première étape dans les systèmes de traitement et d'analyse d'images, c'est la technique de classification la plus simple. Son but est de diminuer la quantité des informations présente dans l'image, et de ne garder que les informations jugées pertinentes. Elle consiste à séparer les pixels en fonction de la valeur de leur intensité, en deux classes distinctes, généralement représentées par deux couleurs, le blanc et le noir. Notre première classe correspondra à notre couleur à isoler (le bleu), la seconde à toutes les autres couleurs. Nous verrons sur notre image binarisée tous les éléments jaunes apparaître en blanc sur un fond noir.

Comme montré ci-dessous [figure 2.3], il existe plusieurs types de seuillage selon le nombre des seuils.

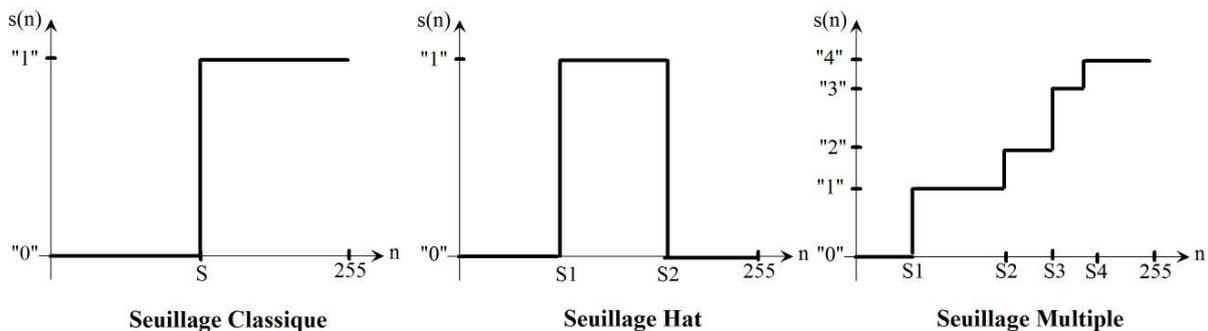


Figure 2.3. Les différents types de binarisation (seuillage).

2.4.3. Le masque de l'objet

On appelle masque de l'objet mobile, la projection ou la région occupée par cet objet dans le plan image. L'obtention de ce masque permet de se focaliser sur les zones d'intérêt de l'image. Dans notre image binarisée le masque est rempli en blanc et correspond à l'objet traqué, et tout le reste est en noir. Dans notre cas tous les pixels seront remplacés un à un suivant un intervalle de valeurs TSV sélectionnées. Ainsi, si un pixel à une valeur comprise dans l'intervalle il prendra la valeur 1 (blanc), et si sa valeur est en dehors de l'intervalle, il prendra la valeur 0 (noir). [Figure 2.3. seuillage Hat].

Chapitre 2 Traitement numérique des images

2.4.4. Opérations sur l'image binaire

a. Ouverture morphologique

Après l'obtention d'une image binaire, qui nous indique où se trouve la couleur traquée par des taches blanches et tous le reste en noir. Afin de suivre cet objet il faut l'isoler.

Une image binarisée est bruitée par des pixels éparpillés qui appartiennent à l'intervalle des valeurs TSV, mais qui ne présentent pas l'objet traqué. Afin de s'en débarrasser on fait appel à une ouverture morphologique.

L'ouverture morphologique est le résultat d'une érosion suivie d'une dilatation de l'ensemble érodé donc : une application successive de deux operateurs morphologiques :

- *Une érosion* : L'opération d'érosion consiste à éliminer les points blancs isolés. Cette transformation nous permet de supprimer les pixels isolés (qui appartiennent à l'intervalle des valeurs TSV) mais qui ne correspondent pas à l'objet cherché.
- *Une dilatation* : L'opération de dilatation consiste à éliminer les points noirs isolés au milieu des parties blanches. Cette transformation nous permet de renforcer les groupes denses de pixels de l'objet cherché.

2.4.5. Difficultés de la méthode

C'est une méthode assez simple à mettre en œuvre, la rapidité de cette technique la rend très attrayante pour les applications en temps réel, mais en raison de sa simplicité, de nombreux problèmes peuvent entraîner l'échec du suivi.

- l'apparence d'un objet peut varier fortement avec l'éclairage de la scène ce qui rend la détection des couleurs dépendante de la luminosité.
- Si un autre objet possède la même couleur on ne peut les différencier, excepte s'il y a une grande différence de taille entre les deux objets.

Chapitre 2 Traitement numérique des images

2.5. La reconnaissance du visage

La reconnaissance faciale est un domaine très actif dans les domaines de la Vision et de la Biométrie, puisqu'il a été étudié vigoureusement pendant 25 ans et produit finalement des applications en sécurité, en robotique, en interfaces homme-ordinateur, appareils photo numériques, jeux et divertissement.

Un système de reconnaissance faciale doit pouvoir identifier des visages présents dans une image ou une vidéo de manière automatique. Pour identifier un visage d'une personne spécifique, il faut une progression d'étapes :

- la détection de visage : Repérer la présence d'un visage dans une image numérique.
- la reconnaissance du visage : C'est la comparaison du visage repéré avec la base de données des visages enregistrés précédemment.

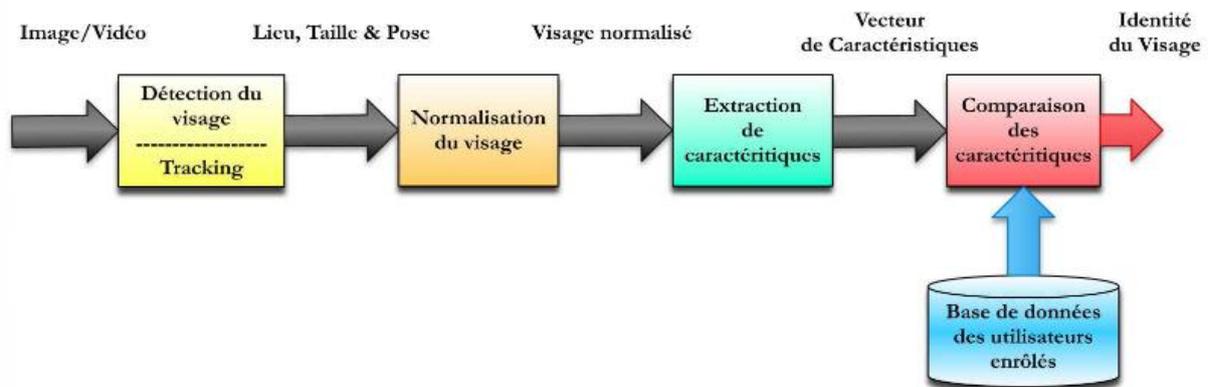


Figure 2.4. Principe de fonctionnement de base d'un système de reconnaissance faciale.

2.5.1. La détection de visages dans une vidéo

La détection de visages est la première étape dans le processus de reconnaissance faciale. Son efficacité a une influence directe sur les performances du système. C'est un cas spécifique de détection d'objets, dont la tâche principale est de détecter des visages humains dans une image numérique. Il s'agit alors de repérer la présence de caractéristiques de visages humains (le front, le nez, les yeux, la bouche, le montant), c'est à dire la structure

Chapitre 2 Traitement numérique des images

générale d'un visage dans l'image, réunis sur une surface suffisamment petite pour qu'on puisse considérer qu'ils appartiennent à la même personne [figure 2.5].

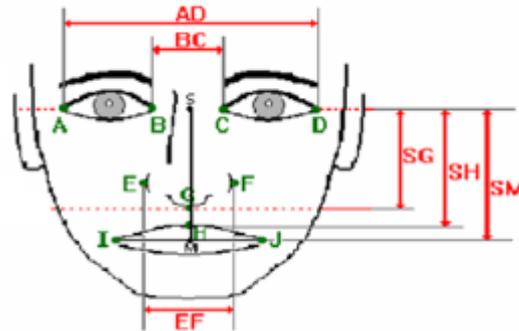


Figure 2.5 Les caractéristiques générales d'un visage humain.

C'est un problème de classification où on assigne l'image à la classe visage ou à la classe non visage. Ce problème est rendu d'autant plus délicat qu'un grand nombre de facteurs viennent compliquer sa solution : la couleur, la forme, la présence de lunettes ou d'une moustache, l'orientation (face ou profil), l'expression faciale (rire, peur, colère), la complexité du décor, les variations de poses, les conditions de lumières généralement inconnues, etc.

2.5.2. La normalisation

Elle permet de préparer l'image du visage repéré dans la partie précédente de telle sorte qu'elle soit exploitable, en lui faisant subir plusieurs transformations.

2.5.3. Extraction des caractéristiques et classification

Le résultat de l'étape de détection est l'obtention de la partie d'image à traiter. on extrait les informations qui seront sauvegardées en mémoire pour être utilisées et comparées plus tard dans la phase de décision avec une base données qui contient une ou plusieurs photos étiquetées des individus à reconnaître.

Chapitre 2 Traitement numérique des images

Le choix de ces informations utiles revient à établir un modèle pour le visage, elles doivent être discriminantes, on peut les distinguer, et non redondantes c'est-à-dire les informations ne doivent pas se répéter.

2.5.4. La décision

Cette étape consiste à comparer les modèles de visages enregistrés à une base de données, pour trouver la meilleure correspondance possible pour l'identification des personnes connues.

2.6 Conclusion

Dans ce chapitre, nous avons d'abord abordé certaines définitions primaires concernant le traitement d'images, le codage des couleurs, la binarisation et quelques opérations sur les images binaires dans le but de les améliorer ou d'extraire maximum d'informations pertinentes. Ensuite présenter la méthode de détection d'objets à base de couleur qu'on a trouvé idéal pour le suivi d'un objet d'une couleur unique. Et enfin citer les principales étapes de détection et reconnaissance faciale dans un flux vidéo. En gros nous avons présenté les principales taches du projet réalisé.

Chapitre 3

Présentation générale du projet

« On ne peut agir efficacement qu'en travaillant du concret »

(Edmund Burke, homme politique)

CHAPITRE 3 Présentation générale du projet

3.1. Introduction

Le robot doit être capable en premier temps à l'aide d'une camera orientée vers le bas de détecter, suivre un cylindre à base de sa couleur, dans une table, puis le capturer avec une pince actionnée par un moteur LEGO. Ensuite et grâce à un autre servomoteur qui oriente la camera vers le haut, afin de trouver une personne connue pour lui donner l'objet capturé, on passe à la détection des visages, reconnaissance de ces derniers, et enfin le déplacement du robot vers la personne connue détectée.

Après avoir fixé les objectifs principaux de notre projet, il est temps de le réaliser. Dans ce qui suit on va présenter les différents composants et logiciels choisis pour sa réalisation.

3.2. Les solutions matérielles

Notre choix des composants repose essentiellement sur la disponibilité, et le coût, tout en respectant les objectifs principaux du projet.

3.2.1. *Châssis et roues*

Un robot a besoin d'un corps, un châssis qui assure la stabilité et permet de fixer les différentes pièces pour former la morphologie finale.

Le robot mobile construit, est un robot à deux roues motrices indépendantes actionnées par deux moteurs à courant continu type MAXON.

CHAPITRE 3 Présentation générale du projet

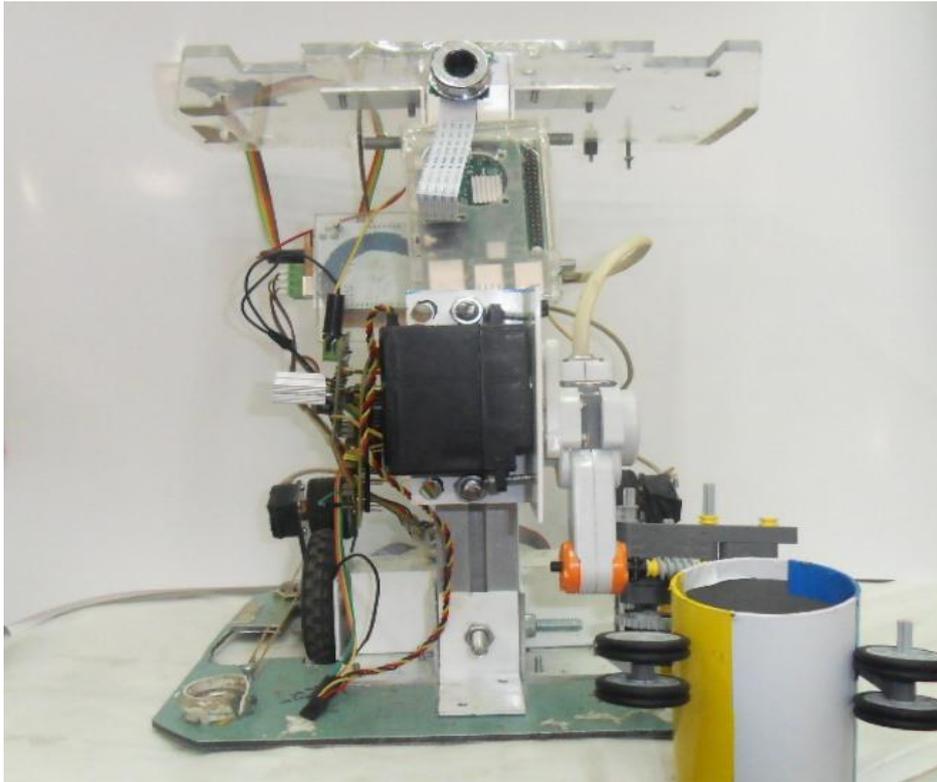


Figure 3.1. Le robot mobile.

3.2.2. L'alimentation électrique

Un robot ne peut être autonome sans alimentation, le choix est varié et dépend de la consommation énergétique des différents éléments à intégrer.

Dans ce projet, on utilise une batterie de 12V, qui se charge d'alimenter les différents moteurs, servomoteurs, cartes électroniques, et capteurs.

3.2.3. Les moteurs électriques

Dans ce projet on a utilisé

- un moteur type MAXON à courant continu, qui nous assure les déplacements de la base mobile dans tous les sens, à droite, à gauche, marche avant, marche arrière, rotation.
- Un moteur type LEGO qui commande l'ouverture et la fermeture de la pince pour la capture de l'objet traqué.

CHAPITRE 3 Présentation générale du projet

a. Les moteurs MAXON CC



Figure 3.2. Les moteurs électriques MAXON à courant continu.

b. Le moteur LEGO

Les moteurs LEGO disposent d'un capteur de rotation intégré permettant aux programmes de maîtriser avec précision le nombre de tours ainsi que la vitesse (max 177 Tr/mn). Cette fonctionnalité intégrée nous permet d'estimer avec une forte précision l'ouverture et la fermeture de la pince de notre projet.

En raison du connecteur spécial de ce moteur (type de prise de téléphone non standard), certaines modifications ont été faites afin de le commander avec la carte électronique Arduino.

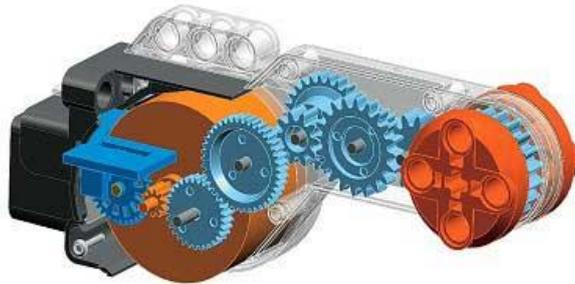


Figure 3.3. Le moteur type LEGO.

CHAPITRE 3 Présentation générale du projet

3.2.4 La pince

Une fois l'objet coloré détecté et atteint, le rôle de la pince est maintenant de le capturer et de le mettre dans sa place destinée. Pour ce projet on a utilisé une pince à base de pièces de LEGO actionnée par le moteur LEGO.



Figure 3.4. La pince fermée et ouverte.

3.2.5 Le servomoteur

En effectuant une rotation, un servomoteur est capable d'atteindre et de maintenir des positions prédéterminées, et a l'avantage d'être asservi en position angulaire. Le plus souvent, l'angle est compris entre 0 et 180°.

Dans ce projet on a utilisé un servomoteur type HITEC : pour l'orientation de la caméra (vers le bas et vers le haut).



Figure 3.5. Le servomoteur type HITEC.

CHAPITRE 3 Présentation générale du projet

3.2.6. Le capteur GP2D15

Puisque le robot va se déplacer dans des différents coins, il risque d'être entravé par des obstacles, alors l'utilisation d'un capteur de distance effectuant la détection de ces obstacles rencontrés sur le chemin devient nécessaire. Dans ce chapitre on s'intéresse aux capteurs de distance GP2D15

Paramètres GP2D15	Gamme
Tension d'alimentation	4,5 à 5,5V
La sortie	Sortie numérique
La portée	Fixée à 24Cm \pm 3Cm

Tableau 3.1. Les caractéristiques techniques du capteur GP2D15.

3.2.7. Les cartes électroniques programmables

Pour réaliser les opérations principales demandées, la mise en relation électrique des composants électroniques cités est importante. Pour cela l'implémentation des cartes électroniques est obligatoire.

a. La carte Arduino

Arduino est le nom d'une gamme de cartes à microcontrôleur programmable et à plusieurs entrées et sorties numériques et analogiques. Elle permet de piloter un système à partir du programme mis dans sa mémoire. Elles utilisent toutes un même logiciel de programmation multiplateforme appelé logiciel Arduino également.

On peut l'associer à des capteurs (de lumière, de température, de position, etc.), des actionneurs (moteurs, pompe, etc.), des organes de sortie (lampe, chauffage, etc.), des circuits de puissance, une alimentation (piles, panneaux solaires, etc.), etc.

Il existe toute une gamme de carte Arduino chacune ses caractéristiques, utilisées selon le besoin de l'utilisateur. Dans ce projet on s'intéresse à la carte Arduino Méga et Uno.

CHAPITRE 3 Présentation générale du projet

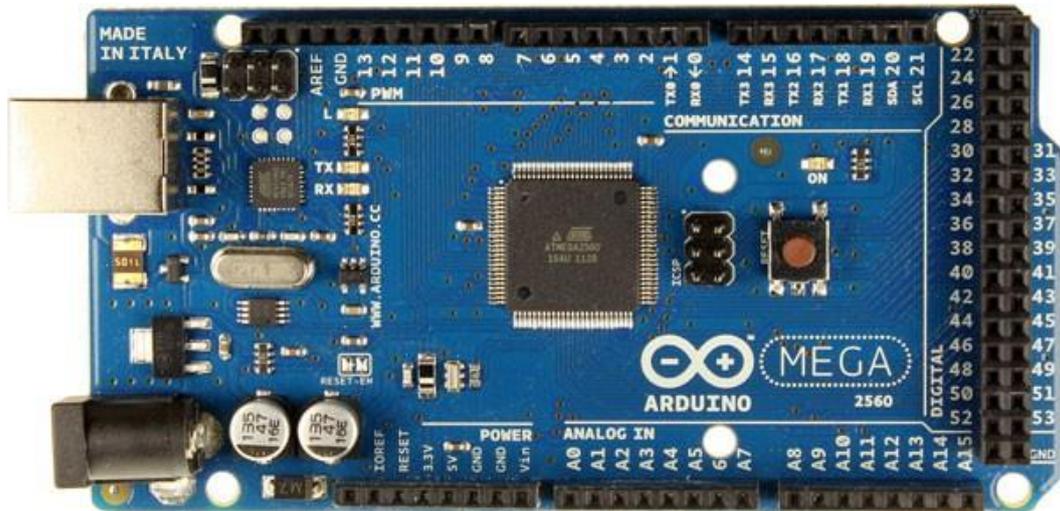


Figure 3.6. La carte Arduino Méga

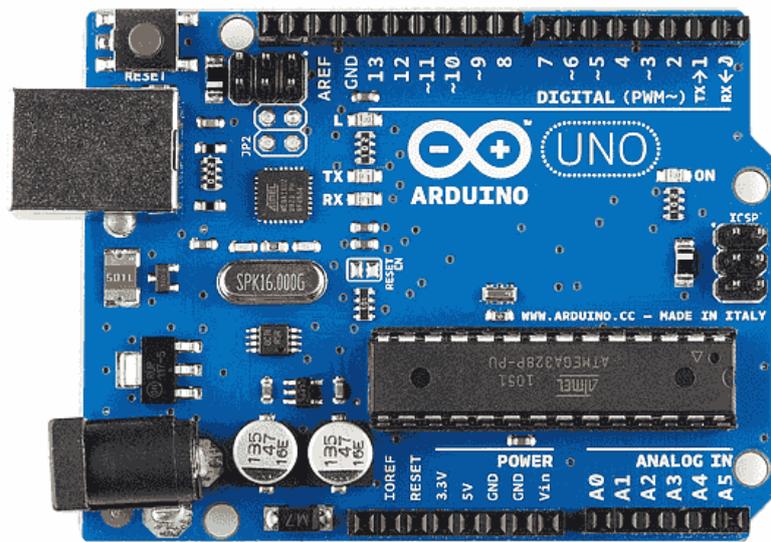


Figure 3.7. La carte Arduino Uno

CHAPITRE 3 Présentation générale du projet

Les caractéristiques techniques des deux cartes UNO et MEGA :

	Carte Arduino Uno	Carte Arduino Méga
Microcontrôleur	ATmega328	ATmega2560
Tension de fonctionnement	5 V	5 V
Tension d'entrée (recommandée)	7 à 12 V	7 à 12 V
Tension d'entrée (limites)	6 à 20 V	6 à 20 V
Broches E/S numériques	14 (dont 6 fournissent la sortie PWM)	54 (dont 14 fournissent la sortie PWM)
Broches d'entrée analogiques	6	16
Courant alternatif par broche d'E/S	40 mA	40 mA
Courant continu pour la broche de 3,3 V	50 mA	50 mA
Mémoire Flash	32 KB (dont 0.5 KB utilisés par le chargeur initial de programme)	256 Ko (dont 8 Ko utilisés par le chargeur initial de programme)
SRAM	2 KB	8 Ko
EEPROM	1 KB	4 Ko
Vitesse de l'horloge	16 MHz	16 MHz

Tableau 3.2. Les caractéristiques techniques des carte Arduino Uno et Méga.

CHAPITRE 3 Présentation générale du projet

b. Le Raspberry Pi

Raspberry Pi est un ordinateur miniature, mono carte entièrement fonctionnel, un système sur puce (SoC), qui par défaut fonctionne sur un système d'exploitation Linux spécialement conçu pour lui, appelé Raspbian, et ils en existent d'autres.

À l'origine, le Raspberry Pi a été créé au Royaume-Uni pour servir d'ordinateur bon marché destiné à l'enseignement des bases de l'informatique, aux grand public. Sorti dans sa première version en 2011, Raspberry Pi n'a cessé d'évoluer jusqu'aujourd'hui avec des différentes versions (*A, B, B+, Pi 2, Zéro et Pi 3*) pour devenir l'un des plus célèbres ordinateurs du monde.

Dans ce projet, on s'intéresse à la dernière version Raspberry Pi 3 model B.

<i>Paramètres</i>	<i>Raspberry Pi 3 model B</i>
<i>Sortie</i>	<i>29/02/2016</i>
<i>SoC</i>	<i>BCM2837</i>
<i>CPU</i>	<i>Quad Cortex A53 @ 1.2GHZ</i>
<i>GPU</i>	<i>400MHZ VideoCore IV</i>
<i>RAM</i>	<i>1 GB SDRAM</i>
<i>Stockage</i>	<i>Micro-SD</i>
<i>Ethernet</i>	<i>10/100</i>
<i>Sans fil</i>	<i>802.11n / Bluetooth 4.0</i>
<i>Sortie vidéo</i>	<i>HDMI/Composite</i>
<i>Sortie audio</i>	<i>HDMI / Headphone</i>
<i>GPIO</i>	<i>40</i>
<i>Instruction set</i>	<i>ARM v8- A</i>

Tableau 3.3. Les caractéristiques du Raspberry Pi 3 model B

CHAPITRE 3 Présentation générale du projet



Figure 3.8. La carte Raspberry Pi 3 model B

b.1. Le module camera Raspberry Pi

Le module caméra Raspberry Pi est un produit officiel de la fondation Raspberry Pi. Le modèle a été lancé en 2013. Il se branche directement sur le connecteur CSI du Raspberry Pi. Dans ce projet on utilise la version 1.3 qui est capable de livrer une image de résolution de 5MP ou un enregistrement vidéo HD 1080p à 30fps.



Figure 3.9. Le module camera Raspberry Pi V1.3.

CHAPITRE 3 Présentation générale du projet

<i>Paramètres</i>	<i>Gamme</i>
La taille et poids	<i>Environ 25 x 24 x 9mm, 3g</i>
Touche resolution	<i>5 Méga pixels</i>
Modes video	<i>1080p30, 720p60 et 640 x 480p60 / 90</i>
Capteur	<i>Omni Vision OV5647</i>
Résolution du capteur	<i>2592 x 1944</i>
Distance focale	<i>3,60 mm ± 0,01</i>
Champs de vision horizontal	<i>53,50° ± 0,13 °</i>
Champs de vision vertical	<i>41,41° ± 0,11 °</i>
Ratio focal (F-Stop)	<i>2,9</i>

Tableau 3.4. Les caractéristiques du module caméra Raspberry Pi v1.3.

c. La répartition des taches

Dans ce projet, l'Arduino et Raspberry Pi forment une association importante. Chaque carte est censée transmettre à l'autre, les bonnes informations afin d'assurer le bon fonctionnement du système.

Pour profiter des avantages de chaque carte, il est important de distribuer les taches selon la puissance, et la robustesse de chacune. L'Arduino est adaptée pour un type de travail répété, mais Raspberry Pi peut faire des choses plus complexes.

Etant donné que le Raspberry Pi est 40 fois plus rapide que l'Arduino, et grâce à son module camera, il va s'occuper de la partie traitement d'images, quant à l'Arduino, elle assure le bon fonctionnement des actionneurs et des capteurs du projet selon les informations reçues.

CHAPITRE 3 Présentation générale du projet

3.2.8. La carte puissance

Les cartes Arduino ont des limites à respecter : les sorties, tout comme les entrées, ne peuvent délivrer que 5V et 40mA chacune. Si on branche directement un appareil consommant plus sur une de ces sorties, on risque de griller la carte. Pour brancher des moteurs, il faut y ajouter une carte de puissance [figure 3.10]. Elle se branche sans souder sur les pins, et nous permet de conduire deux moteurs à courant continu, en contrôlant la vitesse et la direction de chacun d'eux indépendamment.

Le composant central de la carte est un L298N, c'est un double pont en H, à haute tension et à courant élevé, il permet de faire tourner les moteurs dans un sens ou dans l'autre sans avoir à modifier les branchements.



Figure 3.10. La carte puissance

<i>Paramètres</i>	<i>Gammes</i>
<i>Alimentation</i>	+6V +35v
<i>Courant maximal</i>	2 A
<i>La puissance de la sortie</i>	+5V 7V (alimentation interne 5V)
<i>Puissance maximale</i>	25W
<i>Température de travail</i>	25°C +130°C

Tableau 3.5. Les caractéristiques du composant L298N.

CHAPITRE 3 Présentation générale du projet

3.3. Solutions logicielles

Après la conception mécanique de notre robot, il est temps de lui donner la vie, lui permettre de se déplacer, d'interagir et voir le monde extérieur. En fait, la machine ne comprend pas directement notre langage pour exécuter nos ordres. En effet, la machine ne résonne qu'avec des états logiques « 0 » et « 1 ». On parle de langage binaire, que nous les humains trouvons sa compréhension une tâche fastidieuse.

C'est pour cela qu'on fait souvent appel à une solution intermédiaire : un traducteur automatique : compilateur ou interpréteur qui consiste à analyser, traduire et transcrire à notre place les instructions de code source (langage humain) en langage binaire compréhensible par la machine.

3.3.1. Pour le Raspberry pi

a. La langage de programmation PYTHON



Étant donné que Raspberry Pi fonctionne sous Linux, il est possible d'installer n'importe quel langage, mais en raison de documentations, d'exemples, et de projets réalisés, on a finalement opté pour le PYTHON un langage simple, lisible et moins complexe.

PYTHON est un langage de programmation interprété, dont la première version est sortie en 1991. Créé par **Guido van Rossum**.

Malgré qu'un langage compilé se révèle bien plus rapide qu'un langage interprété, mais la facilité d'apprentissage, la bonne documentation, une pléthore de bibliothèques pour de nombreux besoins, et aussi la rapidité de développement a fait que ce langage a connu une croissance remarquable ces dernières années.

b. La bibliothèque OpenCV



Les langages de programmation ne disposent pas à la base de fonctions de traitement photo ni vidéo. Pour cela, il faut utiliser une bibliothèque qui ajoute ces fonctions et l'importer dans le programme.

CHAPITRE 3 Présentation générale du projet

Dans ce projet on a choisit une bibliothèque multi plate-forme (Windows, GNU/Linux, Android, et MacOS.) open source initialement développée par Intel sous licence BSD, OpenCV.

Comme son nom l'indique **Open source Computer Vision**, est libre et spécialisée dans la vision par ordinateur, aussi bien dans le monde de la recherche que celui de l'industrie. Cette bibliothèque possède de nombreuses fonctionnalités intégrées très diversifiées, permettant de créer des programmes partant des données brutes pour aller jusqu'à la création d'interfaces graphiques basiques. Elle propose un ensemble de plus de 2500 algorithmes de vision par ordinateur pour les langages C, C++, et PYTHON, qui permettent de gagner du temps sur le codage de tâches de bas niveau.

Ses capacités d'être utilisée pour la détection d'objets : de couleurs, et de personnes, le suivi du mouvement, et voir même la reconnaissance faciale en temps réel sur un flux vidéo, étaient les principales raisons pour notre choix de se familiariser avec l'OpenCV.

La bibliothèque OPENCV sera utilisée dans notre projet pour détecter, suivre la couleur jaune en mouvement, et la reconnaissance faciale en utilisant le module camera du Raspberry pi.

3.3.2. Pour la carte Arduino



Pour pouvoir programmer la carte Arduino, on doit installer l'IDE Arduino sur notre ordinateur. Ce logiciel peut fonctionner sur différentes plates-formes – Windows, Mac et Linux. De plus, il nous permet de programmer la carte via l'interface USB sans qu'il soit nécessaire d'utiliser de matériel particulier, avec un langage proche du C/C++, dont les bases sont faciles d'accès et demande très peu de ressources.

CHAPITRE 3 Présentation générale du projet

3.4. Communication entre les deux cartes

3.4.1. *Introduction*

Le Raspberry Pi et l'Arduino ne sont plus à présenter. Ils ont changé notre manière de pratiquer l'électronique et l'informatique embarquée. C'est pour cela qu'on a pensé de faire travailler les deux cartes ensemble ce qui nous permettra ainsi de profiter des avantages de chacune de ces cartes. Le Raspberry Pi sera configuré comme maître, la carte Arduino Uno et Mega comme esclaves. Le protocole I2C se chargea de les relier, ils pourront communiquer de manière bidirectionnelle mais en half duplex.

3.4.2 *Transmission des données*

Beaucoup d'applications nécessitent de pouvoir communiquer avec d'autres périphériques ou systèmes. Les systèmes de transmission de données véhiculent l'information entre une source et un destinataire en utilisant un support physique.

Les signaux transportés peuvent être soit directement d'origine numérique comme dans les réseaux de données, soit d'origine analogique (parole, image...) mais convertis sous une forme numérique. La tâche du système de transmission est d'acheminer le signal ou l'information de la source vers le destinataire avec le plus de fiabilité possible.

3.4.3 *Mode de transmission des données*

Pour une transmission de données sur une voie de communication entre deux machines, la communication peut s'effectuer de différentes manières. La transmission est caractérisée par:

- Le sens des échanges des données.
- Le mode de transmission des données.
- La synchronisation.

CHAPITRE 3 Présentation générale du projet

a. Le sens des échanges de données

Selon le sens des échanges on distingue trois modes de transmission :

- *La liaison Simplex* : c'est une liaison dans laquelle les données circulent dans un seul sens, c'est à dire de l'émetteur vers le récepteur.
- *La liaison half-duplex* : semi duplex ou parfois appelée liaison à l'alternat, est une liaison dans laquelle les données circulent dans un sens ou l'autre mais pas les deux simultanément.
- *La liaison full-duplex* : appelée aussi duplex intégral est une liaison dans laquelle les données circulent de façon bidirectionnelle et simultanément.

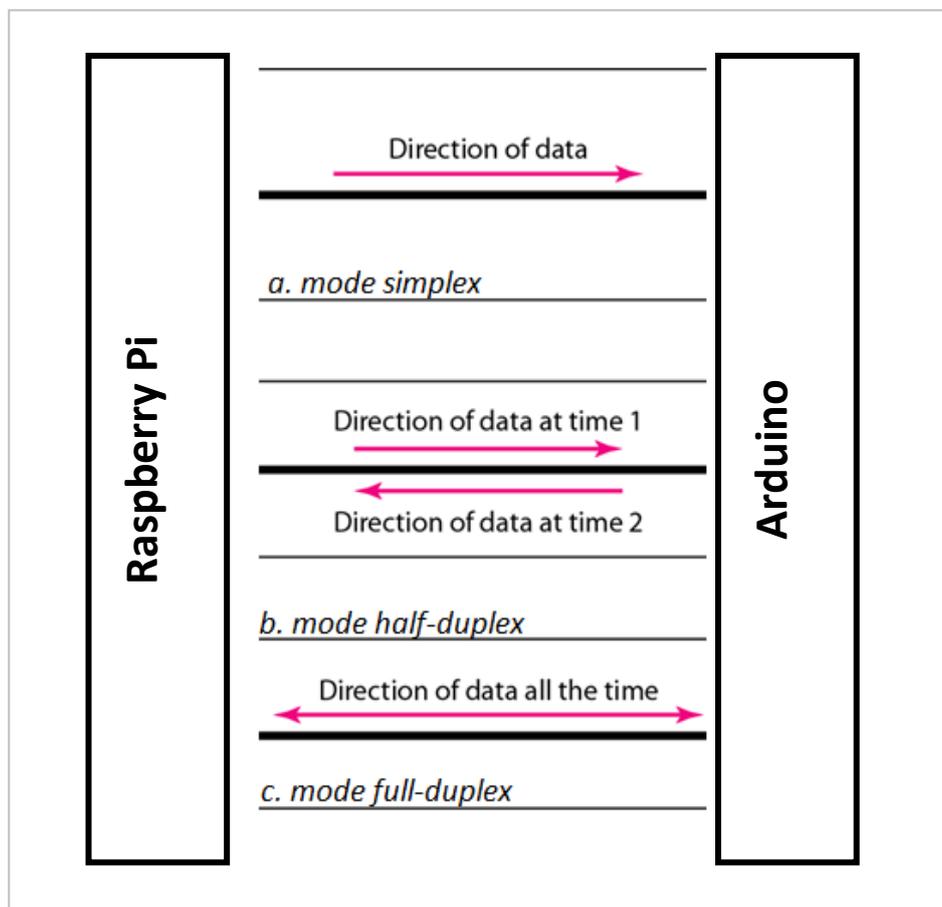


Figure 3.11. Le mode simplexe, Half-duplex et full-duplex.

CHAPITRE 3 Présentation générale du projet

b. Le mode de transmission des données

L'information élémentaire à transmettre est le mot binaire constitué de « N » bits. Deux techniques de transmission pour acheminer l'information sont mises en œuvre :

- *La liaison série* : tous les « N » bits d'un même mot sont transmis successivement sur une même ligne de transmission.
- *La liaison parallèle* : La transmission parallèle est caractérisée par un transfert simultané de tous les « N » bits d'un même mot sur « N » voies différentes.

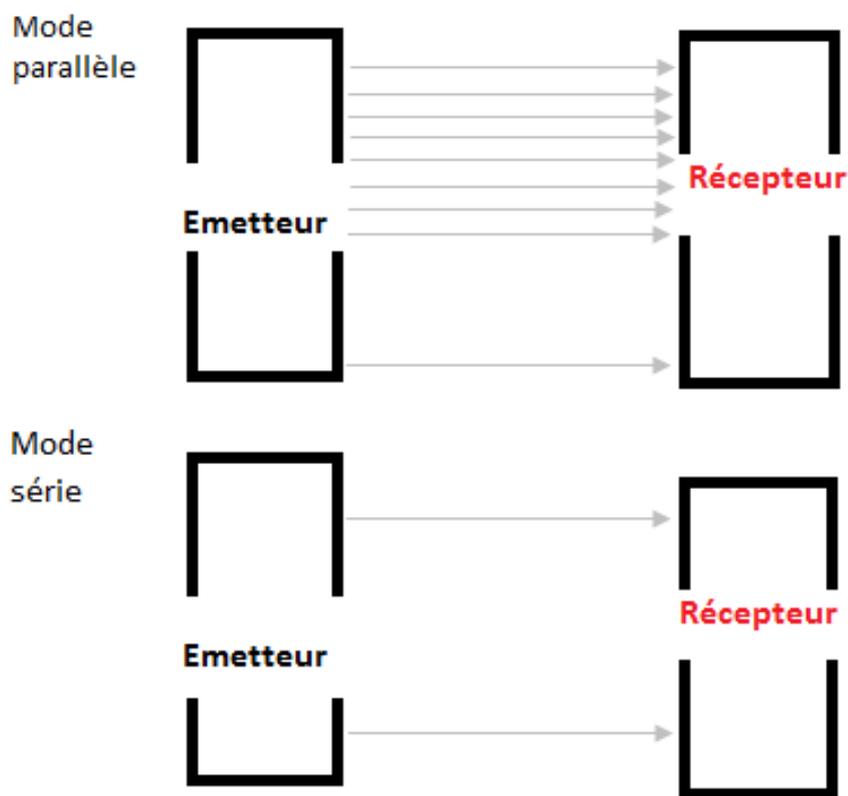


Figure 3.12. Le mode parallèle et le mode série pour la transmission des données

Comme le montre la figure ci-dessus [Figure 3.12] le mode parallèle transmet huit octets sur 8 lignes différentes dans le même temps que le mode série transmet seulement huit bits (un octet) sur une seule voie.

CHAPITRE 3 Présentation générale du projet

Dans notre projet on utilise une liaison série pour transmettre les données entre Raspberry Pi et l'Arduino.

c. La synchronisation

Le mode série est le plus utilisé, un seul fil transporte l'information, les bits sont envoyés successivement, le temps entre deux caractères n'étant pas défini, il en résulte un problème de synchronisation entre l'émetteur et le récepteur, distinguer et reconnaître les séquences de bits utiles. Pour y remédier à ce problème :

- *Transmission synchrone* : l'émetteur et le récepteur sont cadencés à la même fréquence d'horloge c'est-à-dire la fréquence de lecture correspond à la fréquence d'envoi. Le matériel le plus lent impose donc le rythme des communications.
- *Transmission asynchrone* : Chaque caractère est précédé d'un « START bit » et suivi d'un « STOP bit ». L'avantage des transmissions asynchrones réside dans la simplicité de la méthode.

3.4.4 Les protocoles de communication

Un protocole de communication définit l'ensemble des règles qui précisent les modalités de fonctionnement d'une communication entre deux systèmes.

Il existe une multitude de protocoles, on s'intéresse dans ce chapitre au protocole I²C utilisé dans notre projet.

a. Présentation générale du protocole I²C

Le protocole I²C (Inter Integrated Circuit) développé au début des années 80, par Philips Semiconductor pour les applications de domotique et d'électronique domestique, est un bus de *type série*, Les données sont envoyées bit par bit par groupe d'octets appelés « des trames » sur la ligne SDA, *synchrone* et il fonctionne sur un système *bidirectionnel half-duplex*, les informations transitent dans les deux sens sur un même fil mais pas en même temps. Chaque circuit est reconnu par son adresse et peut être soit transmetteur (Le circuit

CHAPITRE 3 Présentation générale du projet

qui envoie la donnée sur le bus) soit receveur (Le circuit qui reçoit la donnée du bus) de l'information. Ces circuits peuvent être : Un ordinateur, un microcontrôleur, un microprocesseur, une mémoire, un périphérique (clavier, écran,...) etc.

Le bus I²C appelé aussi le *Two Wire BUS* a besoin de deux fils SDA (Serial DATA) et SCL (Serial CLOCK) qui véhiculent les informations entre les différents circuits connectés :

- Le SDA (Serial DATA Line) : c'est la ligne où passent les données dans les deux sens.
- Le SCL (Serial CLOCK Line) : c'est la ligne où passe le signal d'horloge pour la synchronisation.

Notons qu'il est nécessaire de rajouter la masse (GND) qui sert à référencer les deux signaux d'informations SCL et SDA.

Le protocole de la liaison est du type MAITRE/ESCLAVE. Dans notre projet on va identifier la carte Raspberry Pi comme MAITRE et Arduino Uno et Méga comme ESCLAVE.

b. Raccordement du protocole I²C

On utilise les broches SDA et SCL. Il suffit de raccorder ces broches des cartes utilisées ensemble (et partager la même masse) pour constituer le Bus I²C.

- Pour Arduino Méga : Les pins 20 et 21 pour SDA et SCL.
- Pour Arduino Uno : Les pins A4 et A5 pour SDA et SCL.
- Pour la carte Raspberry Pi : Les pins 3 et 5 pour SDA et SCL.

CHAPITRE 3 Présentation générale du projet

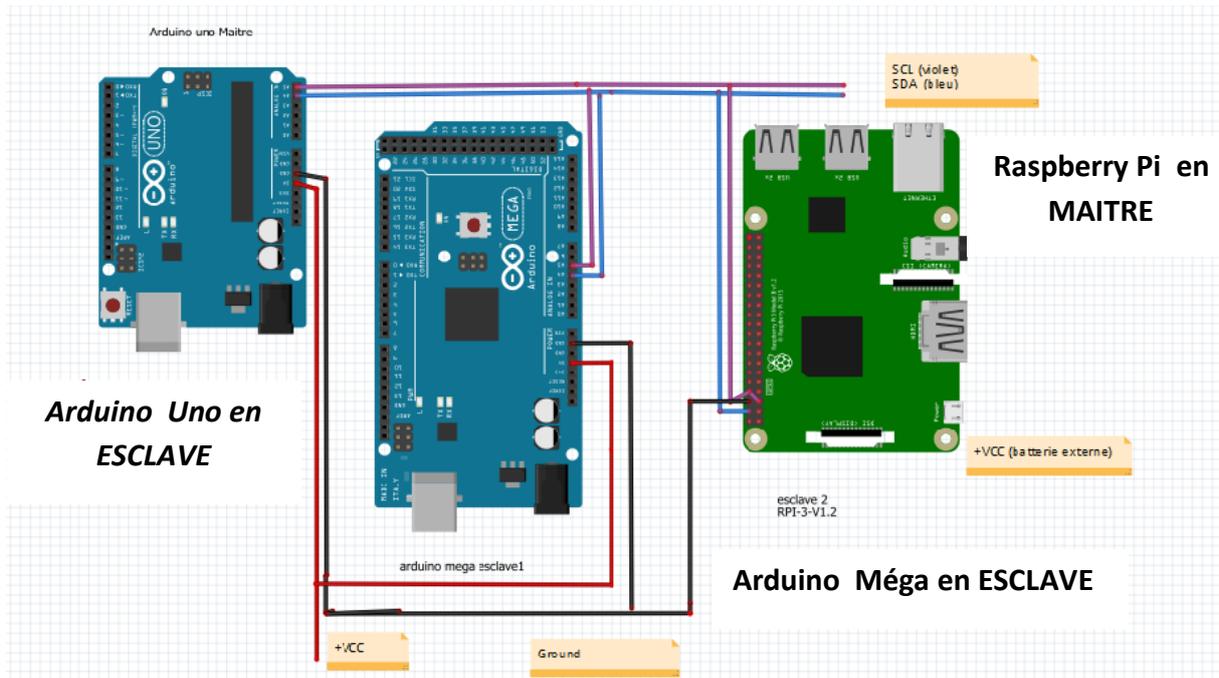


Figure.3.13. Raccordement du protocole I²C

c. Le protocole I²C

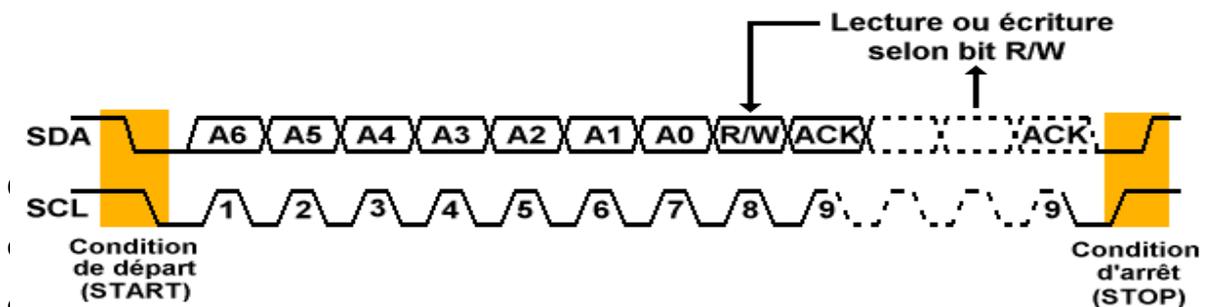


Figure 3.14. Le chronogramme des lignes SDA et SCL pour le bus I²C

Avant de tenter de prendre contrôle du bus, il faut tout d'abord s'assurer qu'il est libre, c'est à dire les deux lignes sont en repos (à l'état HAUT). Pour pouvoir communiquer :

CHAPITRE 3 Présentation générale du projet

- Les conditions de départ et d'arrêt du protocole du bus I²C :

La condition « Start » : La condition de départ des échanges de données condition:

- Pour la ligne SDA : passage de l'état « HAUT » à l'état « BAS ». Un front descendant.
- Pour la ligne SCL : niveau « HAUT ».

La condition « Stop » : Détermine la fin des échanges de données condition d'arrêt :

- Pour la ligne SDA : passage de l'état «BAS » à l'état « HAUT ». Un front montant
- avec la ligne SCL : maintenir l'état « haut ».

Notons que:

- les conditions de départ et d'arrêt sont toujours générées par le maître du bus.
- La condition d'arrêt peut survenir même au milieu de la transmission d'un octet, pour abandonner la transmission et libérer le bus pour les autres circuits.
- seulement si la ligne SCL est à l'état bas, que le circuit qui émet les données, peut modifier l'état de la ligne SDA.

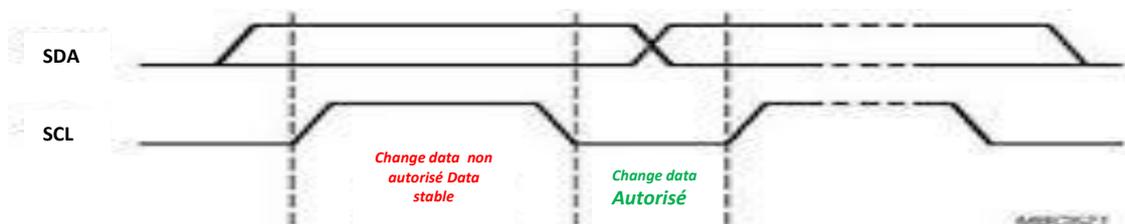


Figure 3.15. Validité des données en fonction des signaux SDA et SCL

Le rôle du signal d'horloge SCL alors, est de rythmer l'envoi des données. Chaque bit du signal de données SDA sera envoyé à chaque front descendant du SCL.

CHAPITRE 3 Présentation générale du projet

- *La transmission d'adresse :*

Après le lancement des conditions de départ le maître doit choisir avec quel esclave est censé Recevoir/Envoyer des données, pour cela il doit tirer son adresse. Dans ce but, le premier octet que transmet le maître n'est pas une donnée mais les bits d'adressage (A6-A0) suivi d'un bit R/W.

Sa direction : Maître vers esclave

Notons qu'il y'a certaines adresses qu'elles ne sont pas utilisées pour l'adressage de composants. Elles ont été réservées par Philips pour effectuer certaines fonctions spéciales. («0000 0xxx», « 11111XXX» etc.)

- *Le bit R/W :*

Les bits d'adressage (7 bits) sont suivis d'un 8eme pour former l'octet appelé le bit Read/Write, pour indiquer le mode d'action voulu par le maître :

- R/W = 0 : Envoie des données : une donnée est envoyée à un esclave.
- R/w=1 : Lecture des données : l'esclave revoie une donnée au maître.

Sa direction : Maître vers esclave.

- *Le bit ACK :*

Chaque octet de donnée transmis est suivi par l'émission d'un bit d'acquiescement (ou ACK) qui sert d'un *accusé de réception*.

- ACK = 0 : Le maître (Raspberry Pi) peut continuer les opérations. (ACK)
- ACK = 1 : Une erreur de sélection le maître génère la condition d'arrêt (NACK)

Sa direction : du récepteur vers l'émetteur

Le récepteur de l'échange doit procéder à l'acquiescement(ACK) pour indiquer à l'émetteur (qu'il soit maître ou esclave) que l'information a bien été reçue.

CHAPITRE 3 Présentation générale du projet

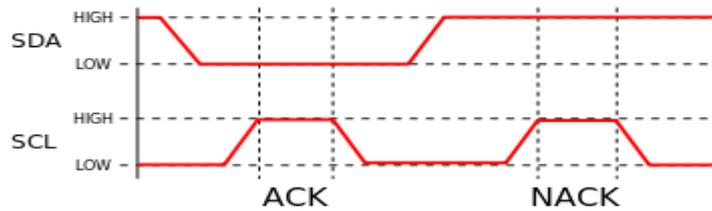


Figure 3.16. La condition d'acquittement ACK ou de non-acquittement NACK.

3.5 Conclusion

Dans ce chapitre, on a présenté les principales tâches que notre robot est censé faire, les solutions matérielles et logicielles qu'on a jugé les mieux adaptées pour ce projet. On a même spécifié le bus de données utilisé pour associer les deux cartes et assurer le transfert d'informations utiles entre la carte maître et esclave.

Chapitre 4

Résultats du projet

« Que la stratégie soit belle est un fait, mais n'oubliez pas de regarder le résultat. »

Winston Churchill

4. 1. Introduction

Ce dernier chapitre est consacré aux méthodes utilisées et résultats trouvés, après la mise en œuvre des différentes solutions matérielles et logicielles proposées.

Comme mentionné dans le chapitre précédent, les tâches du robot mobile se divisent en deux parties essentielles :

- La détection et le suivi d'objet de couleur jaune.
- La reconnaissance faciale d'une personne.

4. 2. Détection et suivi d'objet de couleur

4. 2.1. Introduction

Pour la détection et le suivi d'objets en mouvement, plusieurs méthodes existent. En raison de simplicité le choix est tombé sur la détection d'objets selon la couleur. C'est une méthode assez simple à mettre en œuvre, il suffit de reconnaître les bordures supérieures et inférieures de la couleur.

Le traçage est programmé en PYTHON en utilisant la bibliothèque OpenCV pour la vision par ordinateur, et le Raspberry Pi pour effectuer les différents traitements nécessaires.

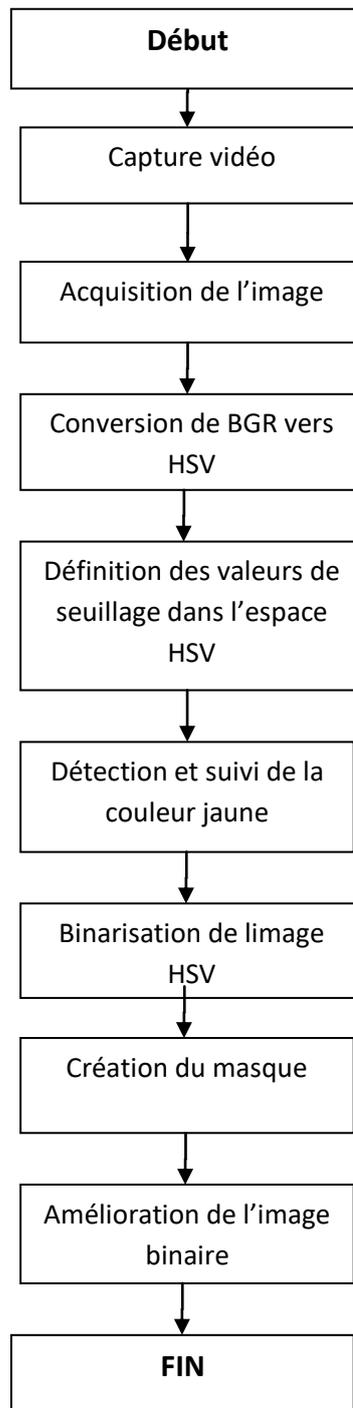
4.2.2. La détection de couleur avec OpenCV

Détecter et reconnaître une couleur, est une tâche sans effort pour nous les humains, mais c'est loin d'être le cas pour un ordinateur. Le module caméra Raspberry Pi ne fait qu'envoyer des données brutes, qui vont être lues et traitées pour pouvoir enfin détecter la couleur voulue.

Vu qu'on a une seule couleur à détecter, et afin de diminuer la quantité des informations présente dans l'image et garder que les pertinentes c'est à dire les pixels de la couleur jaune, on a effectué une binarisation sur les images.

Chapitre 4 : Résultats du projet

a. Organigramme de la détection de couleur



Organigramme 4.1. Organigramme de la détection de couleur

Chapitre 4 : Résultats du projet

b. Résultats obtenus

Avec un temps d'exécution de 0.019585s, le système effectue 51 détections/sec, c'est largement suffisant pour une vidéo de 30 images/sec.

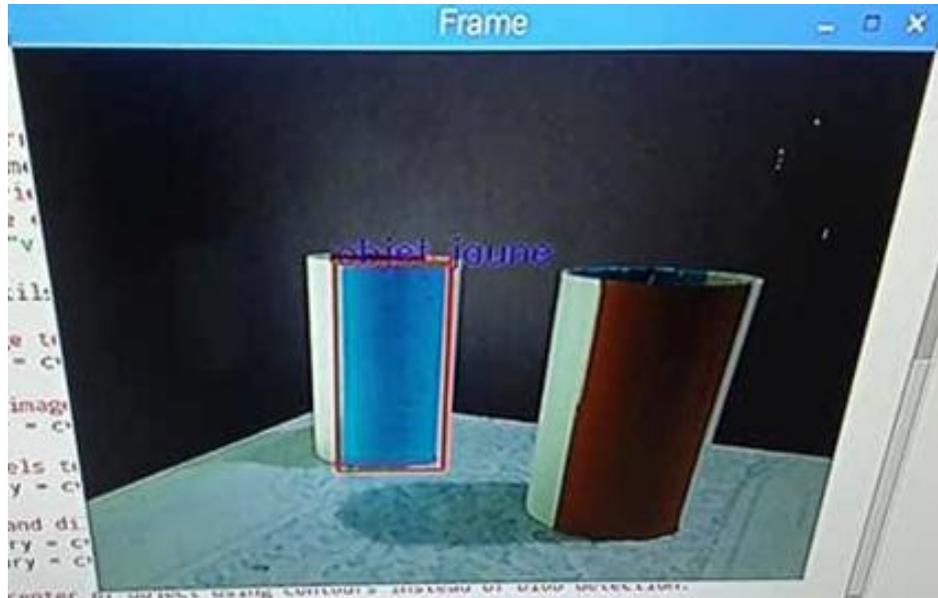


Figure 4.1. Résultats obtenus

Chapitre 4 : Résultats du projet

c. Les problèmes de la détection de couleur

- Trouver l'intervalle des paramètres HSV qui contient que la couleur voulue.
- la distinction des objets est impossible lorsque leur taille est similaire.
- Les résultats changent fortement en fonction de l'éclairage.
- L'image binaire est légèrement bruitée.

La bibliothèque OpenCV propose quelques techniques pour éliminer au maximum le bruit de détection dans une image binaire. Après plusieurs tests on a finalement opté pour l'application d'une ouverture morphologique, afin de supprimer les pixels en blanc éparpillés dans l'image qui n'appartiennent pas à l'objet, ensuite de renforcer ceux qui appartiennent pour avoir une meilleure précision.



Figure 4.2. L'amélioration de l'image binaire.

4. 3. La reconnaissance faciale

4. 3.1. Introduction

Cette partie est consacrée à l'identification des personnes par reconnaissance faciale. Le programme de reconnaissance faciale sera codé en PYTHON. Il utilisera les bibliothèques OpenCV, l'ordinateur Raspberry Pi et son module caméra v1.3.

4. 3.2. La détection de visages avec OpenCV

Comme il a déjà été mentionné, la première étape dans la reconnaissance des visages est la détection des visages.

a. La méthode de Viola et Jones

La méthode de Viola et Jones fait partie des toutes premières méthodes capables de détecter efficacement et en temps réel des objets dans une image. C'est l'une des méthodes les plus connues et les plus utilisées, en particulier pour la détection de visages et la détection des personnes. Elle peut également être utilisée pour détecter le sourire, les yeux, le nez et même d'autres types d'objets comme les voitures ou les avions.

- Principe de la méthode :

La méthode de Viola et Jones consiste à balayer une image à l'aide d'une fenêtre de détection de taille initiale 24x24pixels (dans l'algorithme original) et de déterminer si un visage y est présent. Lorsque l'image a été parcourue entièrement, la taille de la fenêtre est augmentée et le balayage recommence, jusqu'à ce que la fenêtre fasse la taille de l'image. L'augmentation de la taille de la fenêtre se fait par un facteur multiplicatif de 1.25. Le balayage, quant à lui, consiste simplement à décaler la fenêtre d'un pixel. Ce décalage peut être changé afin d'accélérer le processus, mais un décalage d'un pixel assure une précision maximale.

Ce qui différencie cette méthode des autres est notamment :

- l'utilisation d'images intégrales qui permettent de calculer plus rapidement les caractéristiques.
- La sélection par boosting des caractéristiques.

Chapitre 4 : Résultats du projet

- La combinaison en cascade de classifieurs boostés, apportant un net gain de temps d'exécution.

- *Apprentissage du classifieur :*

L'apprentissage du classifieur est une étape préliminaire et très importante. Il s'agit d'entraîner le classifieur afin de le sensibiliser à ce que l'on veut détecter, ici des visages. Pour cela, il est mis dans deux situations :

La première où une énorme quantité de cas positifs lui sont présentés la deuxième où, à l'inverse, une énorme quantité de cas négatifs lui sont présentés. Il se présente sous la forme d'un fichier XML.

- *Image intégrale :*

C'est une représentation sous la forme d'une image, de même taille que l'image d'origine, elle contient en chacun de ses points la somme des pixels situés au-dessus et à gauche du pixel courant.

L'image intégrale $ii(x, y)$ est définie à partir de l'image $i(x', y')$ par :

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y')$$

La valeur de l'image intégrale aux coordonnées (x, y) est la somme de tous les pixels au-dessus et à gauche de (x, y) , y compris ce dernier point.

- *Algorithme d'apprentissage basé sur Adaboost :*

Afin de sélectionner les meilleures caractéristiques, Viola et Jones utilisent une méthode de boosting. Le boosting est un principe qui consiste à construire un classifieur fort à partir d'une combinaison pondérée de classifieurs faibles, c'est-à-dire, donnant en moyenne une réponse meilleure qu'un tirage aléatoire.

L'algorithme est de type itératif, à nombre d'itérations déterminé. À chaque itération, l'algorithme sélectionne une caractéristique, qui sera ajoutée à la liste des caractéristiques

Chapitre 4 : Résultats du projet

sélectionnées aux itérations précédentes, et le tout va contribuer à la construction du classifieur fort final.

b. Détection de visages à l'aide des classifieurs de haar

Pour détecter des objets dans une image, une vidéo ou en temps réel, la bibliothèque OpenCV fournit la fonction « *cvHaarDetectObjects* » basée sur la méthode de détection d'objets de Viola et Jones [10].

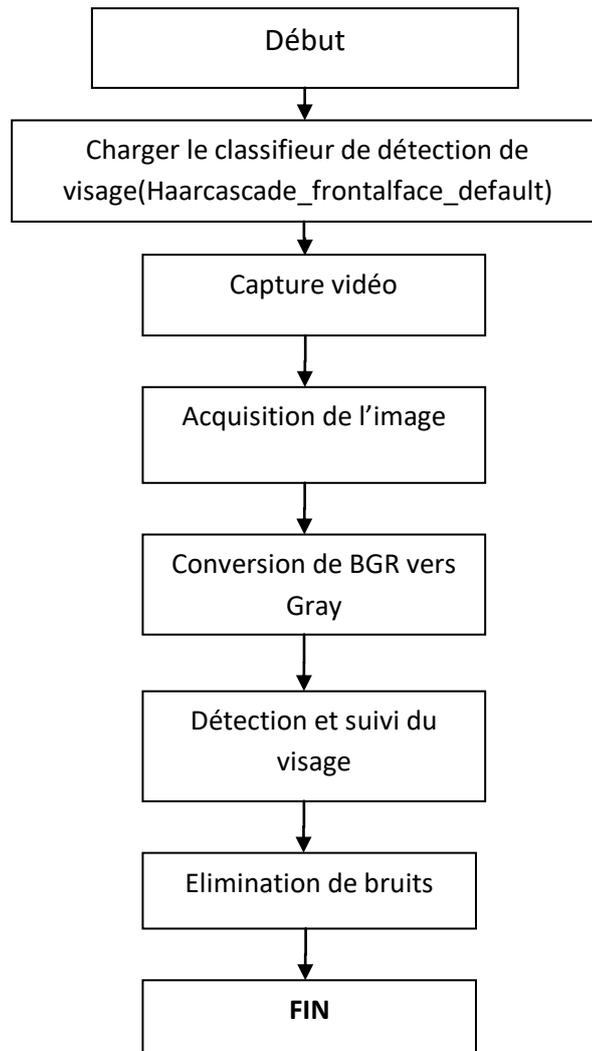
La bibliothèque OpenCV est livrée avec plusieurs classifieurs différents pour la détection de visage dans des poses frontales, sous forme des fichiers XML.

- *Haarcascade_frontalface_alt.xml*
- *Haarcascade_frontalface_alt2.xml*
- *Haarcascade_frontalface_alt_tree.xml*
- *Haarcascade_frontalface_default.xml*

Afin de classer de manière objective et de tirer le meilleur classifieur pour notre projet, on s'est basé sur quelques tests trouvés sur internet sur une base d'images contenant au total 858 photos des visages prises dans différentes conditions. Les résultats ont montrés que le meilleur classifieur pour la détection de visage (frontale) est le **Haarcascade_frontalface_default** avec un pourcentage de 92.78% bonnes détections face à 7.22% de mauvaises. Etant donné qu'il offre les meilleures performances pour la détection de visages, il sera choisi pour notre projet.

Chapitre 4 : Résultats du projet

c. Organigramme de détection de visage



Organigramme 4.2. Organigramme de détection de visage

d. Résultats obtenus

L'approche de Viola et Jones, présente de très bons résultats. On a pu réaliser un taux de détection correcte de 95% et un taux de fausse détection de 1%.

Le programme génère une boîte englobante rectangulaire du visage très fiable lorsque le visage est de face ou légèrement de profil.

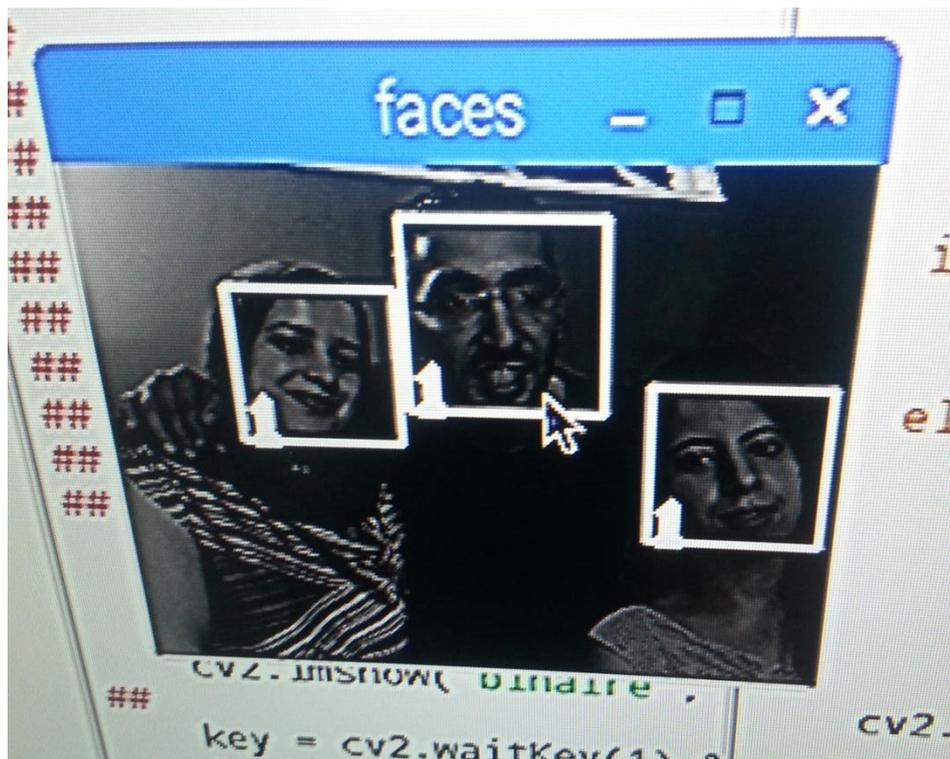


Figure 4.3. Détection de visages

e. Les problèmes de la détection de visages

d. 1. Le temps de calcul

La méthode de Viola et Jones est basée sur une approche par recherche exhaustive sur l'ensemble de l'image. Le système doit être assez rapide pour une détection en temps réel, pour s'approcher de la perfection il doit effectuer au moins 30 détections/sec, ce n'est pas le cas pour ce système. Cependant le moyen le plus simple pour accélérer un peu le

Chapitre 4 : Résultats du projet

programme, étant donné que toutes l'image doit être balayée pixel par pixel, ce serait de travailler sur une image de résolution plus petite 480x240.

d. 2. La rotation

Le système de détection est défaillant en présence de forte rotation. Une des limitations de la méthode de Viola et Jones est son manque de robustesse à la rotation, et sa difficulté à apprendre plusieurs vues d'un même objet. Une première amélioration qui permet de corriger ce défaut qui consiste à apprendre une cascade dédiée pour chaque orientation de vue à été proposé par Viola et Jones [11], par la suite plusieurs autres améliorations ont été proposées pour apporter une solution à ce problème [12] [13].

d.3. Présence d'éléments structurels

Le système rencontre parfois des problèmes à détecter une personne avec des moustaches quand l'éclairage est faible.

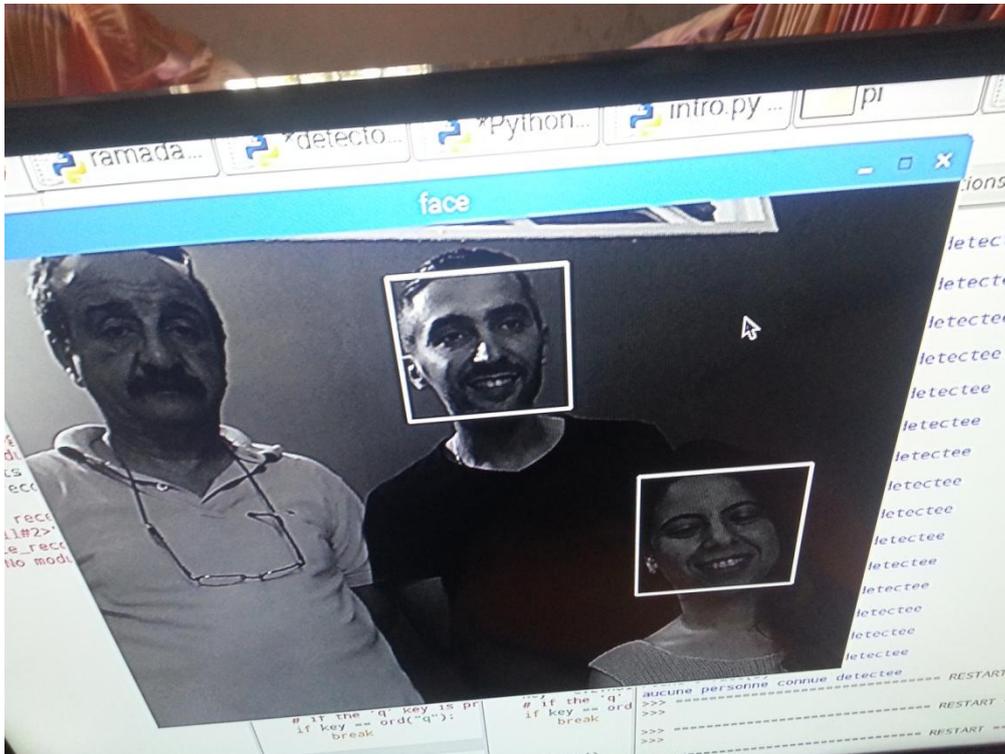


Figure 4.4. Problèmes de détection de visages

4. 3.3. La reconnaissance de visages avec OpenCV

La détection de visages est une caractéristique fantastique que la bibliothèque OpenCV possède, et qui constitue la base d'une opération plus avancée: la reconnaissance faciale.

L'approche adoptée par OpenCV pour y parvenir à reconnaître un visage, est de former le programme en lui fournissant un ensemble de photos classées qui vont servir d'une base de données pour comparer la similarité du visage détecté avec les visages déjà existants afin d'identifier l'individu en question.

a. Le choix de la méthode

Par défaut OpenCV contient 3 algorithmes de reconnaissance faciale :

- Eigenfaces [14]: **createEigenFaceRecognizer()**
- Fisherfaces [15] : **createFisherFaceRecognizer ()**
- LBPH (Local Binary Patterns Histograms) [16] : **createLBPHFaceRecognizer()**

Nous avons choisi d'utiliser le Recognizer LBPH puisqu'il a l'avantage de pouvoir fonctionner avec un échantillon d'images réduit.

b. La base de données

La base de données de ce projet a été construite à l'aide d'un programme PYTHON. Ce programme prend 21 photos pour utilisateur « i » en niveau de gris, redimensionnées, au format JPG, et étiquetées à chaque exécution.

Chapitre 4 : Résultats du projet

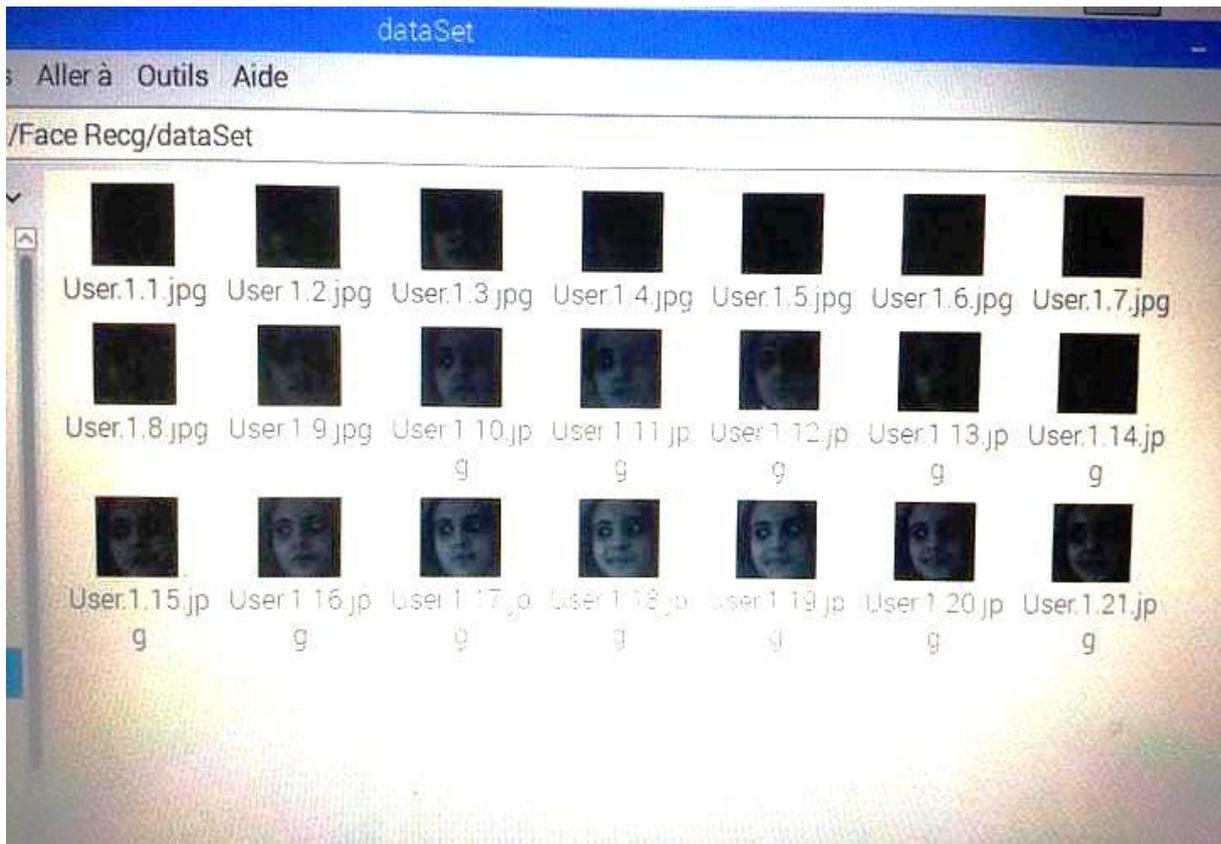
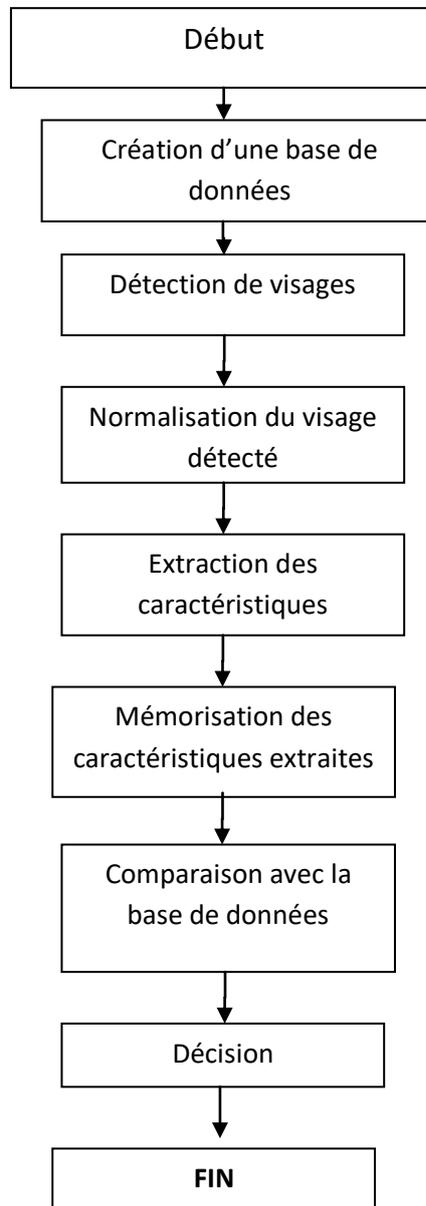


Figure 4.5. La base de données

c. Organigramme de la reconnaissance faciale



Organigramme 4.3. Organigramme de reconnaissance faciale

d. Résultats obtenus

Le programme est capable de reconnaître une personne spécifique dans une vidéo qui pourrait également contenir d'autres personnes. Le programme a reconnu notre personne choisie quand celle-ci est proche et fait face à la caméra.



Figure 4.5. La reconnaissance faciale du système

e. Les problèmes de la reconnaissance des visages

Etant donné que ce n'est pas un système parfait, le résultat est correct mais doit encore être optimisé, la reconnaissance des visages parfois échoue. Par conséquent des individus inconnus (non enregistrés dans la base de données) sont classés comme connus.

Le temps de calcul est lent pour un système de détection en temps réel, mais redimensionner la taille de l'image à 112x92 a donné de bons résultats coté rapidité.

f. Conclusion

Pour la détection de couleur, le programme détecte et suit parfaitement la couleur bleu d'une manière rapide et fluide. Mais reste très sensible aux variations d'éclairage, et aux présences d'autres objets de la même couleur.

Pour une première expérience, la réalisation d'une procédure de détection et de reconnaissance de visage à été largement facilitée par l'utilisation de la bibliothèque OpenCV.

- la reconnaissance faciale parfois échoue face aux diverses variations d'éclairage, des expressions faciales, l'apparence du visage à travers la présence ou l'absence des éléments structurels (moustache).

Conclusion générale

Conclusion générale et perspectives

De nombreuses applications en vision par ordinateur nécessitent la détection et le suivi des objets en mouvement dans une séquence d'images. Le but de ce travail est la réalisation d'un robot mobile avec vision intelligente embarquée.

La première partie du projet, est de détecter et suivre un objet mobile dans une séquence vidéo selon sa couleur, suivi d'une détection de visage et reconnaissance facial en deuxième partie.

Le Raspberry Pi présente le composant le plus important de ce projet, grâce à sa simplicité et sa robustesse pour le traitement d'images numériques, l'utilisation et la vitesse du protocole I²C pour le transfert des données entre les différentes cartes, ont fait de cette réalisation un projet réussi.

La bibliothèque OpenCV a largement facilité la réalisation de ce projet et cela revient à la richesse de différentes fonctions qui la compose surtout dans le traitement des images et des vidéos.

La vision intelligente est un domaine vaste, riche, et surtout très prometteur. Au cours de ces dernières années ce domaine était une cible des recherches scientifiques. Ses applications vont au delà des objectifs de ce mémoire.

Au final, malgré les progrès très conséquents dans le domaine du traitement d'image et la diminution du temps de calcul des processeurs, le suivi robuste du visage et sa reconnaissance demeurent insatisfaisant. En effet, il reste de nombreux verrous scientifiques. Notamment celui de surmonter la forte variabilité d'apparence du visage au cours du temps. Cette variabilité est liée à des facteurs multiples difficilement modélisables tels que les changements d'illumination, les mouvements complexes, les occultations etc.

Conclusion générale et perspectives

Perspectives

Plusieurs améliorations et perspectives peuvent s'envisager dans la continuité des travaux menés dans cette thèse.

1. Pour la détection et suivi d'objets selon la couleur :

Elargir notre travail de détection et suivi des objets en mouvement avec plusieurs couleurs, essayer les différentes méthodes et techniques de détection d'objets dans une scène vidéo, et faire une étude comparative, pour les différentes conditions de travail.

2. Pour la reconnaissance faciale :

L'objectif futur est de minimiser l'influence des différents facteurs tels que les changements d'illumination, les mouvements complexes, les occultations, et de créer un système robuste de reconnaissance faciale, avec une base de données plus large et riche en informations.

Nous pensons également tester une librairie plus avancée qu'OpenCV pour la reconnaissance faciale qui pourrait s'avérer bien plus performante tel qu' OpenBR.

L'utilisation de l'application dans des projets plus importants et utiles tels que la surveillance des lieux, et l'autorisation d'accès aux domiciles.

Références bibliographiques :

Références bibliographiques :

[1] O. Trullier and J. A. Meyer. Biomimetic navigation models and strategies in animats. *AI Communications*, 10:79–92, 1997.

[2] F.Bouffera. contribution des outils de l'intelligence artificielle dans la robotique mobile. Doctorat des sciences. Université d'Oran. Nov. 2014

[3] H.NASSER. Navigation autonome d'un véhicule par la planification de chemins via des données GPS. Université libanaise. Option télécommunication et informatique. Printemps 2012

[4] HACHEMI. Fouzia La détection et suivi des objets en mouvement dans une scène vidéo en utilisant la bibliothèque OpenCV. Université Abou Bakr Belkaid- Tlemcen Faculté des Sciences Département d'Informatique. Option : Système d'Information et des Connaissances (S.I.C)

[5] Lucie Masson. Suivi temps-réel d'objets 3D pour la réalité augmentée. Vision par ordinateur et reconnaissance de formes [cs.CV]. Université Blaise Pascal - Clermont-Ferrand II, 2005. Français.

[6] Jacek Czyz¹ , Branko Ristic² , Benoit Macq¹. Détection et suivi simultanés dans une séquence vidéo en couleur par filtres particuliers.

[7] Y.TANGUY, P, TURELLE, Asservissement linéaire, 1980

[8] D.Blackburn, M. Bone, P. J Phillips. "Face recognition vendor test 2000". Tech. rep. <http://www.frvt.org>, 2001.

Références bibliographiques

- [9] BOUDJELLAL Sofiane, Détection et identification de personnes par méthode biométrique, Option : Télédétection, Université Mouloud MAMMARI de TIZI-OUZOU (UMMTO)
- [10] Paul Viola, Michael Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features", ACCEPTED CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION 2001
- [11] M. Jones et P. Viola, Fast Multi-View Face Detection, IEEE CVPR, 2003.
- [12] A. L. C. Barzak, To Ward and Efficient Implementation of a Rotation Invariant Detection using Haar-like Features. Proceeding of the 2009 IEEE/RSJ international conference on Intelligent robots and systems. Nouvelle-Zélande, Dunedin. 2005, P31-36.
- [13] M. Kolsch et M. Turk, Analysis of Rotational Robustness of Hand Detection with a Viola-Jones Detector, ICPR, Vol.3, 2004.
- [14] Timo Ahonen, Abdenour Hadid, and Matti Pietikäinen. Face description with local binary patterns: Application to face recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 28(2006):2037–2041, 2006.
- [15] Peter N. Belhumeur, Joao P. Hespanha, and David J. Kriegman; Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection, IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 19, NO. 7, pp. 711-720, JULY 1997