الجمهورية الجزائرية الديمقراطية الشعبية
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR
ET DE LA RECHERCHE SCIENTIFIQUE

جامعة يحى فارس بالمدية
UNIVERSITE YAHIA FARES DE MEDEA

FACULTE DE LA TECHNOLOGIE

Département du Génie Électrique

Projet de Fin d'Etudes
En vue d'obtention de diplôme:

Master

En : Électronique, Option: Instrumentation

Commande Visuelle en 2D d'un Bras Manipulateur

Préparé et Soutenu par :

M. BENGHERBIA Ismail

Proposé et Dirigé par :

M. GUEDMANI Mohammed

M. EULDJI Rafik

Année Universitaire 2017-2018

Dédicace

Je dédie ce travail :

- ✓ A mes très chers parents Pour leur amour inestimable, leur confiance, leur soutien, leurs sacrifices et toutes les valeurs qu'ils ont su m'inculquer,
- ✓ A mon marie avec tout mon amour pour son soutien tout au long temps pour terminer mes études,
- ✓ A ma sœur, et à ses trois filles, ainsi qu'à mes deux frères pour leurs tendresses, leurs complicités et toute l'affection qu'ils m'ont donnée et pour leurs précieux encouragements,
- ✓ A toute mes familles Bengherbia et Tobbal,
- ✓ A tous mes collègues de Mester Instrumentation promo 2018,
- ✓ L'ensemble du personnel de la DD-M pour leur compréhension, leur collaboration, en particulier à Monsieur Laidi Belkacem, chef service pour son soutien et ses encouragements,
- ✓ A tous mes amies.

Ismail

Remerciements

الحمد لله ربي العالمين

En premier lieu, je remercie ALLAH, le tout puissant, pour m'avoir donné, la patience, la volonté et la force nécessaires pour achever ce travail.

Je souhaite tout d'abord remercier mon encadreur Monsieur ***GUEDMANI Mohammed****, pour avoir accepté de diriger ce mémoire et de sa patiente durant la période de l'encadrement.*

Je tiens à remercier : Monsieur ***EULDJI Rafik****, pour avoir consacré une partie de leur temps précieux pour lire et corriger cette mémoire, et de sa patiente durant la période de l'encadrement leur projet.*

Je tiens à remercier aussi à adresser nos remerciements aux membres de jury nous avoirs l'honneur et le plaisir d'accepter et juger notre travail.

Je vous remercie tous, votre soutien et vos encouragements me donnent la force de continuer.

Et Enfin, je m'excuse a tous les personnes qui j'oublier ses noms et qui ne sont pas été citées.

ملـخص:

الهدف من هذا العمل، هو دراسة وتحسين ذراع روبوت تسلسلي، انطلاقا من الهيكل الميكانيكي للروبوت والبطاقة الالكترونية للتحكم فيه، مع دمجه بكاميرا ليستطيع التميز بين الألوان. ويكون التحكم في الروبوت عن طريق استعمال بطاقة Arduino، وباستعمال MatLab. يتم انتقال المعلومات من الكمبيوتر عبر المنفذ التسلسلي. الخطوة الأخيرة من عملنا هي تحليل الاختبارات عن طريق المحاكاة التي أجريت بالمخبر Instrumentation والتي اثبتت نجاح المشروع .

الكلمات المفتاحية: ذراع تسلسلي آلي، الألوان، أردوينو.

Abstract:

The purpose of this work is to study and improve a serial arm manipulator based on the mechanical structure of the robot and its electronic card control it, associated with a camera to distinguish the colors. the control of the robot will go through the use of the cardArduino, and using Matlab, information is transferred from the computer via the serial port. The final step of our work is to analyze the simulation tests, as well as the practical experiments by the Instrumentation Lab, which proved the success of the project.

Key words: Serial arm manipulator, Colors, Arduino.

Résumé:

L'objectif de ce travail est l’étude et d’amélioration d’un bras manipulateur série basé sur la structure mécanique du robot et sa carte de contrôle électronique, et de l’intégrer d’une caméra pour distinguer les couleurs. que le contrôle du robot passera par l’utilisation de la carte Arduino,et à l'aide de Matlab. Les informations sont transférées depuis l'ordinateur via le port série. La dernière étape de notre travail consiste à analyser les tests par simulation, ainsi que les expériences pratiques menées sur laboratoire d’Instrumentation, qui ont prouvés le succès du projet.

Mots clés : Bras manipulateur série, Coleurs, Arduino.

Table des matières

Table des matières:

Dédécas i
Remerciements ii
Résumé iii
Table des matières iv
Table des figures viii
Liste des tableaux ix
Liste des abréviations x

Introduction générale 2

Chapitre I : Généralité sur les Robots Industriels

Généralité 5
I. Introduction 5
I.1. Définitions 5
I.2. Historique 6
I.3. La contribution de la robotique dans l'industrie 7
I.4. L'impact de la robotique sur la production 7
I.5. Différents composants d'une cellule robotisée 7
I.5.1. Unité opérationnelle 7
I.5.2. Unité informationnelle 8
I.6. Structure générale d'un bras manipulateur 9
I.6.1. Base 9
I.6.2. Porteur 9
I.6.2.a.Segment 9
I.6.2.b. Articulation 9
I.6.3. Actionneur 10
I.6.4. Organe terminal 10
I.7. Système de traitement 10
I.8. Système mécanique articulé 10
I.9. Catégories de robots industriels 11
I.9.1. Robots mobiles 11

I.9.2. Robots manipulateurs 11
I.10. Classification et catégorie de robots 12
I.11. Catégorie des robots 13
I.12. Différent les manipulateurs 13
I.12.1. masse du robot 13
I.12.2. Précision et répétitivité 13
I.12.3. Comportement dynamique 14
I.13. Domaines de la robotique 14
I.14. Avantages et inconvénients des robots 14
I.14.1. Avantages 15
I.14.2. Inconvénients 15
I.15. Conclusion 15

Chapitre II : Modélisation des robots manipulateurs

II. Introduction 17
II.1. Modélisation des robots manipulateurs 17
II.2. Modèle géométrique direct (MGD) 18
II.2.1. Principe 19
II.3. Modèle géométrique inverse (MGI) 19
II.4. Calcule du modèle géométrique direct 20
II.5.Conclusion 24

Chapitre III : Traitement d'images et la vision par ordinateur

III. Introduction 26
III.1. Définition de l'image 26
III.2. Acquisition d'une image 26
III.3. Caractéristiques d'une image numérique 27
III.3.1. Pixel 27
III.3.2. Dimension 28
III.3.3. Résolution 28
III.3.4. Bruit 28
III.3.4.a. Illustration des différents types de bruit 28
III.3.5. Histogramme 29
III.3.6. Luminance 29

III.3.7. Contraste 29
III.4. Images en couleurs 29
III.4.1. Energie lumineuse-Classification fréquentielle des ondes lumineuses 30
III.5. Système de traitement d'images 30
III.5.1 Capteurs de vision 31
III.5.2. Le positionnement de la caméra 31
III.5.3. Acquisition et numérisation 32
III.5.4. Pré-traitements 33
III.5.5. Traitement d'images avec MatLab 33
III.5.5.a. Types des images 34
III.5.6. Visualisation 35
III.6. Domaines d'application 35
III.7. Asservissement visuel 36
III.7.1. Asservissement visuel 2D 36
III.7.2. Asservissement visuel 3D 38
III.8. L'avantage d'un système de vision industrielle 39
III.9. L'architecture de commande du notre bras 40
III.10. Conclusion 40

Chapitre VI : Réalisation

IV. Introduction 42
IV.1. Chaine d'acquisition 42
IV.2. Schéma de bloc du bras 43
IV.2.1. Capteur de la vision et ses réglages 43
IV.2.2. Le système de communication 46
IV.2.2.a. Rappels sous MatLab 46
IV.2.2.b. L'organigramme 49
IV.2.3. Transfert des données 49
IV.2.3.a. La liaison RS232 50
IV.2.3.b. Paramètres de la liaison RS232 50
IV.2.3.c. Format des trames 50
IV.2.4. Commande par Arduino Uno 51
IV.2.4.a. Présentation de la carte ARDUINO 51
IV.2.4.b. Langage ARDUINO 52

IV.2.5. Partie actionneur 53
IV.2.5.a. Définition 53
IV.2.5.b. Familles d'actionneurs 54
IV.2.5.c. Les Servomoteurs 54
IV.2.5.d. Brochage du servomoteur 55
IV.2.5.e. Les Commande de servomoteurs 55
VI.3. Dispositif expérimental 57
VI.4. Conclusion 58

Conclusion générale 60
Référence ..... A
Annexe ..... B

Table des figures :

Figure I.1 : Bras manipulateur dans une chaine de production 5
Figure I.2 : Robotiques industrielle (a-UNIMATE, b-PUMA) 6
Figure I.3 : Structure fonctionnelle d'un robot 8
Figure I.4 : Constituants d'un bras manipulateur 9
Figure I.5 : Symbole de l'articulation (a-rotoïde, b-prismatique) 10
Figure I.6 : Robots mobiles - Vorkuta, la Russie. Mars 2016 11
Figure I.7 : Bras manipulateur –LGCGM, Laboratoire. Septembre 2015 11

Figure II.1 : Modèle géométrique directe et inverse 17
Figure II.2 **:** Liaisons selon Danavit –Hartenberg 20
Figure II.3 : Placement des repères et notation du robot 21

Figure III.1 : Forme du pixel 27
Figure III.2 : Les bruits 28
Figure III.3 : Le spectre électromagnétique 30
Figure III.4 : Schéma d'un système de traitement d'images 30
Figure III.5 : Caméra embarquée et caméra déportée 32
Figure III.6 : Représentation l'image du niveau de gris 34
Figure III.7 : Représentation l'image en couleur 35
Figure III.8 : Asservissement visuel 36
Figure III.9 : Asservissement visuel 2D 37
Figure III.10 : Asservissement visuel 3D 38
Figure III-11 : Structure d'un asservissement visuel du bras 40

Figure IV.1 : Schéma synoptique de composition du bras 42
Figure IV.2 : Caméra 43
Figure IV.3 : Réglage de la caméra 44
Figure IV.4 : Centralisé le plan de travail (Table) 44
Figure IV.5 : Modèle générique de caméra 45
Figure IV.6 : Centralisation de la table 45
Figure IV.7 : Application d'acquisition d'image sous MatLab 45
Figure IV.8 : Détection des couleurs rouge 47

Figure IV.9 : Détection des couleurs Bleu 47
Figure IV.10 : Détection des couleurs bleus 48
Figure IV.11 : Trame de communication série 51
Figure IV.12 : Catre Arduino Uno 51
Figure IV.13 : Le constituant d'un servomoteur 53
Figure IV.14 : Principe de fonctionnement de servomoteur 55
Figure III.15 : Chronogramme du commande de Servomoteur 56
Figure IV.16 : Photo de la réalisation du projet 57
Figure IV.17 : Test de la détection des objets rouge 59

Liste des tableaux :

Tableau II.1 : Paramètres DENAVIT-HARTENBERG du notre bras 21

Tableau IV-1 : Différentes commandes et leurs descriptions 52

Liste des abréviations

Liste des abréviations :

2D	Deux dimensions deux axes x, y
3D	Trois dimensions trois axes x, y et z
ASCII	American Standard Code for Information Inter change (Code standard pour l'échange d'information)
CCD	Charge Coupled Device
DDL	Degré De Liberté
D-H	Danavit -Hartenberg
in	Inch (’’)
LGCGM	Laboratoire de Génie Civil et Génie Mécanique
MGD	Modèle Géométrique Direct
MGI	Modèle Géométrique Inverse
OT	Organe Terminal
ppp	point par pouce
pt	Petit point = pixel
PWM	Pulse width modulation (Modulation de la largeur des pics)
RS232	Recommended Standard 232 (interface série standard)
RVB	Rouge, Vert et Bleu (RGB en Anglais Red, Green and Blue)
SCARA	Selective Compliance Articulated Robot Arm
SMA	Système Mécanique Articulé
USART	Universal Synchronous Asynchronous Receiver Transmitter (émetteur-récepteur asynchrone universel)
USB	Universal Serial Bus (bus universel en série)

Introduction Générale

Introduction Générale

L'utilisation des robots sériels pour des opérations d'usinage est devenue un enjeu de développement des robots ces dernières années. L'objectif est de proposer une alternative aux machines-outils. Ainsi, des robots gros porteurs sont proposés par les fabricants et concepteurs de robots. Toutefois, les robots développés et améliorés présentent des précisions visualisation de suivi des tâches particulières dans tout son espace de travail, même si leur rigidité a été améliorée par rapport aux robots utilisés pour des opérations moins exigeantes.

En effet, les comportements géométrique, statique et dynamique de ces robots ne permettent pas de réaliser des usinages avec une grande précision. Le comportement géométrique observé sans sollicitation de la structure par une tâche particulière.

Dans ces travaux, nous nous intéressons à l'amélioration visuelle en 2D de bras manipulateur du robot dans le cadre d'opération d'usinage. En effet, ce type d'opération impose la programmation hors ligne des trajectoires par rapport à un repère pièce. Après positionnement de ce repère dans l'espace de travail, le suivi de ces trajectoires doit permettre d'obtenir l'objet avec ses réglages de la caméra.

Nous proposons dans ces travaux d'apporter une contribution sur les robots industriels, la modélisation des paramètres géométriques d'un bras manipulateur série afin d'améliorer la maîtrise de la précision de réalisation d'une tâche. Dans un premier temps, nous allons présenter le contexte de l'utilisation des robots sériels pour l'usinage et ses contraintes.

Finalement, nous réalisons le programme de détections des objets pour conduire à des nouveaux paramètres du bras manipulateurs.

Dans le cadre de la présente étude, Ce mémoire s'articule autour de quatre chapitres complémentaires. Premier chapitre, il y aura une généralité sur les robots industriels en donnant tous ce qu'on aura besoin comme étude théorique de base le long de ce projet.

Ensuite, dans le deuxième chapitre une description sur le modèle géométrique direct et inverse de notre bras manipulateur.

Dans le troisième chapitre on donnera une description de technique traitement des images et caractéristique d'une image. Nous décrirons en détail les différentes architectures existantes des systèmes asservissement visuel.

Ces trois derniers chapitres constituent une préparation au projet qui nous a été soumis au quatrième et dernier chapitre qui est consacrée pour la réalisation pratique, la mise en ligne des différents programme MatLab et Arduino, les comportements de chaque système et les différentes méthodes d'optimisation du procédé avec tous les détails utiles.

A la fin notre mémoire sera achevé par une conclusion générale et des perspectives, ainsi qu'une bibliographie et des annexes.

Chapitre I :

Généralité sur les Robots Industriels

Généralité :

I. Introduction :

Ce chapitre est dédié principalement à des généralités à la robotique. Dans un premier temps on définit la robotique d'une façon générale en s'intéressant sur la robotique industrielle sa contribution dans l'industrie et son impact sur la production tout en donnant des exemples sur quelques robots industriels.

I.1. Définitions :

Un robot est un système mécanique poly-articulé, par des actionneurs et commandé par un calculateur, qui est destiné à effectuer une grande variété de tâches **[1]**.

Un robot est un dispositif mécatronique (alliant mécanique, électronique et informatique) accomplissant automatiquement des tâches diverses. C'est une machine intelligente fonctionnelle qui nécessite une autonomie de mouvements **[1]**.

Le système des robots est composé d'un bras manipulateur permettant d'assembler des composants industriels et de son ordinateur de commande. Ce robot est le robot d'assemblage le plus rependu dans l'industrie des années 70.

L'aspect de la robotique industrielle ou bras manipulateur est le contrôle de conformité d'assemblage (absence / présence / position), contrôle dimensionnel, détection, localisation, comptage et tri, l'automatisation des chaines de production (Figure I.1) et la manipulation des objets. En plus, elle inspecte les produits, rapidement et d'une manière précise.



Figure I.1 : Bras manipulateur dans une chaine de production.

I.2. Historique : La robotique est passée par plusieurs générations comme suit **[2]** :

- 1947: Premier manipulateur électrique télé-opéré,
- 1954: Premier robot programmable,
- 1961: Utilisation d'un robot industriel, commercialisé par la société UNIMATE (USA) (Figure I.2-a), le premier fut utilisé dans les usines de General Motors,
- 1961: Premier robot avec contrôle en effort,
- 1963: Utilisation de la vision pour commander un robot,
- 1978 : un nouveau robot est conçu par Unimation Inc avec l'aide de General Motors. Ensemble ils conçurent le robot PUMA 500. Le robot PUMA (Programmable Universal Machine for Assembly) a été conçu par Vic Schienman et fut financé par General Motors et par The Massachussets Institute of Technology, (Figure I.2-b)
- 1979: Le robot HILARE. Les chercheurs du L.A.A.S. de Toulouse (France) étudièrent la planification des trajectoires d'un robot mobile ponctuel, dans un environnement totalement connu,
- 1981: Le robot VESA. Ce robot, construit à l'I.N.S.A (France). de Rennes, est équipée d'un arceau de sécurité pour réaliser la détection d'obstacles dans un environnement totalement inconnu,
- 1984: Le robot FLAKEY. Ce robot, conçu et construit au Stanford Research Institute et le reflet des améliorations apportées par 14 années de développement. Le robot FLAKEY est équipé de deux roues motrices avec encodeurs, mais sa vitesse maximale est de 66 cm/s au lieu de quelques centimètres par seconde. Ce robot est capable de naviguer dans des environnements réels,
- 1993: Les robots ERRATIC et PIONNER. a été conçu par Kurt Konolige, au Stanford Research Institue, comme un robot mobile de faible coût pour ses cours de robotique,



a) UNIMATE b) PUMA

Figure I.2 : Robotiques industrielle

Avec l'apparition de la robotique industrielle, les robots étaient conçus pour remplacer les ouvriers dans les tâches pénibles, répétitives ou dangereuses (peinture, soudure…).

I.3. La contribution de la robotique dans l'industrie :

La robotique possède de nombreuses formes de différentes disciplines. Les robots sont intensivement utilisés dans le domaine de l'industrie, où ils effectuent des tâches dangereuses avec rigueur. Avec le progrès de la robotique, la technologie des robots industriels s'améliore et leur coût diminue.

Ils ont été développés pour intervenir dans les milieux à risques, par exemple dans les centres nucléaires ou dans des environnements créant un fort taux de corrosion.

I.4. L'impact de la robotique sur la production :

Les robots remontent la chaine de production lorsque les équipements ont atteint un niveau élevé de fiabilité mais nécessitent des ressources humaines importantes, la meilleure solution d'automatisation consiste à utiliser des outils flexibles basant sur les robots.

- Du point de vue contrôle du processus, il sera meilleur en facilitant le nettoyage du robot, zones de rétention minimales et protection de la connectique,
- Du coté rentabilité, ils améliorent la productivité en augmentant la vitesse de production,
- Une grande flexibilité avec multiplicité des options de fixation ce qui nous amène à un espace de travail plus réduit avec et une solution plus simple mécaniquement avec Lancement de produits et de packaging innovants,
- En parlant du packaging le conditionnement des produits étant plus simple ce qui influera sur l'hygiène d'une façon positive,
- Lors de la production, il y a lieu d'un système de perception permettant de contrôler le produit dans le processus de fabrication dans toutes les positions.

I.5. Différents composants d'une cellule robotisée :

Dans un robot en fonctionnement, on peut distinguer plusieurs ensembles interactifs.

I.5.1. Unité opérationnelle :

Exerce les actions commandées en empruntant la puissance nécessaire à la source d'énergie. Cette partie, qui constitue le robot physique, intègre la structure mécanique (segments, articulations, architecture…), les modules d'énergie (amplificateurs, variateurs,

servovalves...), les convertisseurs d'énergie (moteurs, vérins.), les chaines cinématiques de transmission mécanique (réducteurs, vis à billes, courroies crantées …), les capteurs (caméra,…), et enfin l'effecteur, ou organe terminal, qui est en interaction avec l'environnement.

I.5.2. Unité informationnelle :

Reçoit les instructions décrivant la tâche à accomplir, les mesures relatives à l'état interne de la structure mécanique qui constitue le bras manipulateur et les observations concernant son environnement. Elle élabore en conséquence les commandes de ses différentes articulations en vue de l'exécution de ses tâches. Les systèmes actuels fonctionnent en interaction permanente selon le cycle information-décision-action. (Figure I.3)

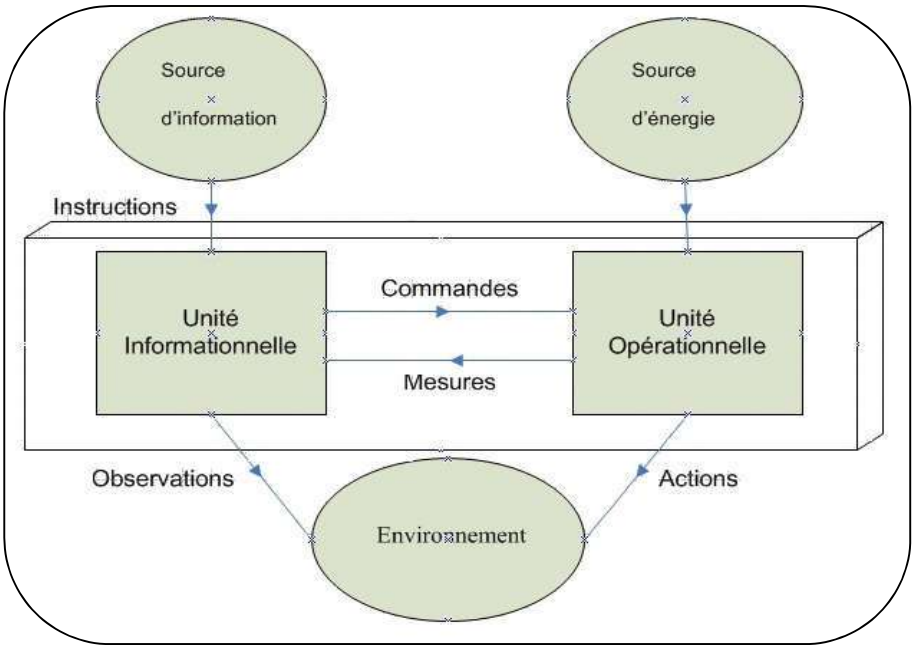


Figure I.3 : Structure fonctionnelle d'un robot.

La manipulation d'un objet est l'application de forces sur cet objet, dans le but de le déplacer. La manipulation peut être rigide, dextre ou souple, dépendant de la rigidité de l'objet et des outils utilisés (pince ou doigts par exemple). Dans ce document, nous entendrons par manipulation à un bras, toutes les tâches du bras doivent manipuler des objets, incluant le cas où ils manipulent le même objet.

Un cas de manipulation considéré à l'unanimité comme coordonné est l'utilisation d'un bras manipulateur pour manipuler un seul objet. Les poses, les vitesses des organes

terminaux, et les forces et moments qu'ils exercent sont liés à cause de la chaine cinématique fermée.

I.6. Structure générale d'un bras manipulateur :

Un bras manipulateur est constitué d'un ensemble de composants, ayant chacun un rôle bien spécifique. (Figure I.4)

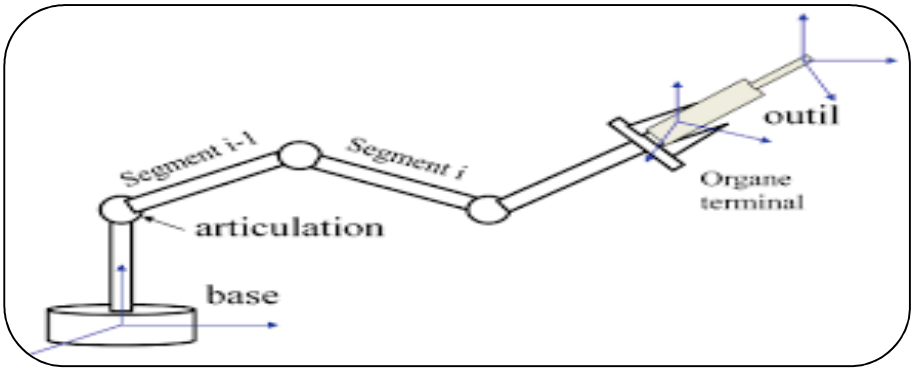


Figure I.4 : Constituants d'un bras manipulateur.

I.6.1. Base :

La base du manipulateur est fixée sur le lieu du travail. Ceci est le cas de la quasi-totalité des robots industriels.

I.6.2. Porteur :

Il représente l'essentiel du système mécanique articulé (segment, articulation, actionneur, l'organe terminal), il a pour rôle d'amener l'organe terminal dans une situation donnée imposée par la tâche. Il est constitué de :

I.6.2.a.Segment :

Corps solides rigides susceptibles d'être en mouvement par rapport à la base du porteur, et les uns par rapport aux autres

I.6.2.b. Articulation :

Une articulation lie deux corps successifs en limitant le nombre de degré de liberté, de l'un par rapport à l'autre.

Il existe deux types articulations, rotoïde (Figure I.5-a) de type pivot, notée R, réduisant le mouvement entre deux corps à une rotation autour d'un axe commun et articulations prismatique (Figure I.5-b) de type glissière, notée P, réduisant le mouvement entre deux corps à une translation le long d'un axe commun. La situation relative entre les deux corps est mesurée par la distance le long de cet axe.

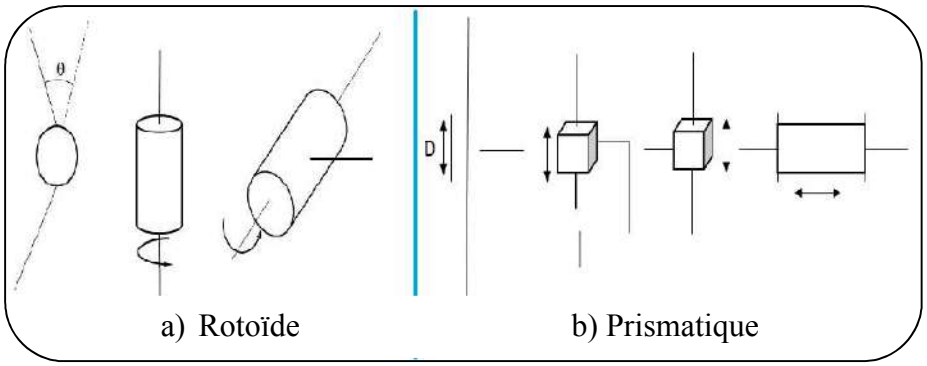


Figure I.5 : Symbole de l'articulation.

I.6.3. Actionneur :

Pour être animé, la structure mécanique articulée comporte des moteurs le plus souvent associés à des transmissions (courroies crantées), l'ensemble constitue les actionneurs. Les actionneurs utilisent fréquemment des servomoteurs. On trouve de plus en plus de moteurs à commutation électronique (sans balais), ou, pour de petits robots, des moteurs pas à pas.

I.6.4. Organe terminal :

Un organe terminal peut être multifonctionnel, au sens où il peut être équipé de plusieurs dispositifs ayant des fonctionnalités différentes. Il peut aussi être monofonctionnel, mais interchangeable. Un robot, enfin, peut-être multi-bras, chacun des bras portant un organe terminal différent. On utilisera indifféremment le terme organe terminal, préhenseur, outil ou effecteur pour nommer le dispositif d'interaction fixé à l'extrémité mobile de la structure mécanique, exemple : pistolet pour la soudure dans les robots industriels.

I.7. Système de traitement :

C'est lui qui gère l'ensemble des tâches. Il admet trois rôles essentiels : le rôle de l'information, qui consiste à collecter l'information venant des capteurs. Ensuite, le rôle de la décision : en partant d'une tâche définie et en tenant compte des données du système et de l'environnement, il établit les actions adéquates. Finalement, le rôle de la communication.

I.8. Système mécanique articulé :

Un système mécanique articulé (SMA) est un ensemble de solides reliés entreeux par des liaisons (pivot, glissière, rotule, sphérique…) animées avec des joints mécaniques.

Dans le SMA, certaines liaisons sont motorisées. On parlera de liaisons actives, c'est le cas des liaisons pivots. D'autres liaisons sont non motorisées ; elles sont appelées passives telles que (les liaisons rotules, etc…) **[3]**.

I.9. Catégories des robots industriels :

Il existe deux types des robots : robots mobiles et robots manipulateurs.

I.9.1. Robots mobiles :

Ce sont des robots capables de se déplacer dans un environnement (Figure I.6). Ils sont équipés ou non de manipulateurs suivant leur utilisation, (les robots explorateurs, les robots de services …).



Figure I.6 : Robots mobiles - Vorkuta, la Russie. Mars 2016.

I.9.2. Robots manipulateurs :

Des robots ancrés physiquement à leur place de travail et généralement mis en place pour réaliser une tâche précise répétitive, (tels que les robots industriels, médicaux …).

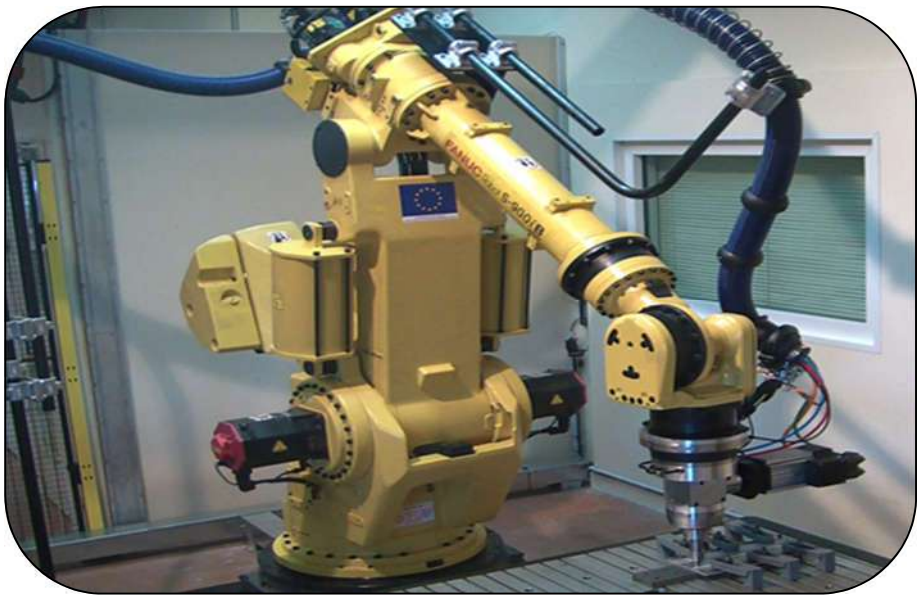


Figure I.7 : Bras manipulateur - LGCGM, Laboratoire. Septembre 2015.

Il existe quatre (04) types du bras manipulateur.

- **Manipulateur à commande manuelle :**

Commandé et dirigé par humain.

- **Manipulateur automatique :**

Qui exerce des mouvements de soudure sans l'intervention de l'homme.

- **Robots programmables :**

Ils répètent les mouvements qu'on leur a appris ou programmés sans informations sur l'environnement ou la tâche effectuée. On peut aussi faire la distinction entre robots « playback » qui reproduit la tâche apprise et robots à commande numérique qui peuvent être programmés hors-ligne. Pour certains robots, par exemple les robots de peinture, qui doivent suivre une trajectoire complexe qu'il est difficile d'exprimer mathématiquement, un opérateur humain spécialiste de la tâche effectue la trajectoire en guidant le bras du robot à l'aide d'un « pantin », et l'entièreté de la trajectoire est mémorisée par le robot **[4]**.

- **Robots intelligents :**

On trouve actuellement des robots qui sont capables d'acquérir et d'utiliser certaines informations sur leur environnement (systèmes de vision, détecteurs de proximité, capteurs d'efforts,...). Les robots de troisième génération sont capables de comprendre un langage oral proche du langage naturel et de se débrouiller de façon autonome dans un environnement complexe grâce à l'utilisation de l'intelligence artificielle **[4]**.

I.10. Classification et catégorie de robots :

Selon le type de tâches, il existe 2 classifications de robot :

- **Robots de substitution :**

Pour les tâches répétitives et précisément définies (donc à commande manuelle ou programmables),

- **Robots de coopération :**

Les mouvements sont pilotés directement par l'homme, le robot apportant l'assistance à l'effort, à la précision du geste…

I.11. Catégorie des robots :

Catégorie selon le type d'architecture :

- **Robots à structure sérielle :**

SCARA, cylindriques, sphériques, cartésiens, anthropomorphes… Exemples d'applications : Packaging/Conditionnement, Peinture/Pulvérisation, Soudage avec process intégré, Vissage, Palettisation, Positionnement, Chargement, Polissage...

- **Robots à structure parallèle :**

(3, 4, 5 et 6 degrés de liberté) : par exemple Robots Delta Exemples d'applications : Assemblage / Manutention / Encaissage, Prise et dépose à la volée « Pick and place ».

I.12. Différent les manipulateurs :

Il y a quatre aspects, qui différent les robots classique en série « les manipulateurs » et les robots parallèles (son architecture est une chaîne cinématique de corps qui peut être fermée ou complexe. **[1]**.

I.12.1. Masse du robot :

Dans l'architecture d'un manipulateur, l'organe terminal se trouve à l'extrémité de la chaîne articulée. Chaque actionneur doit avoir, la puissance nécessaire pour mettre en mouvement, tous les corps et actionneurs en aval.

Dans les cas des robots parallèles, les actionneurs se situent le plus souvent au voisinage des points d'articulation de la base, si ce n'est sur la base elle-même, ce que donne un rapport masse transportable / masse du robot beaucoup plus élevé.

I.12.2. Précision et répétitivité :

Pour le manipulateur une erreur de positionnement, sur une ou plusieurs répercussion sur la position finale de l'organe terminal. Ce problème erreurs, ne se présente pas pour les robots parallèles, puisque sa morphologie, rigidité remarquable, même avec des structures mobiles très allégées.

I.12.3. Comportement dynamique

Compte tenu de leur structure, les manipulateurs, ont des performances dynamiques limitées. Avec les robots parallèles, le fait de diminuer l'inertie du robot, contribue à diminuer les effets de couplage dynamique, et la commande, peut se contenter d'un model dynamique de simplifié.

I.13. Domaines de la robotique :

Les robots industriels peuvent être utilisés dans plusieurs domaines et peuvent accomplir différentes tâches.

- **I.13.1. Domaine de la production :**
  - Automatisation de la production,
  - Meilleur qualité,
  - Adaptation au volume de production,
  - Constance de la qualité,
  - Flexibilité.
- **I.13.2. Domaine de l'exploration :**
  - Accès difficile (nettoyage, démantèlement),
  - Milieu hostile (déminage…).
- **I.13.3. Autres domaine utilisations des robots :**
  - Industrie agro-alimentaire,
  - Génie génétique,
  - Robotique de service : robot pompiste, robot de construction, robot laveur d'avion...,
  - Robotique médicale : assistance aux personnes handicapées, échographie à distance, intuitive surgical...,
  - Robot coopératif « cobot ».

I.14. Avantages et inconvénients des robots :

Un système robotique consiste non seulement des robots mais aussi d'autres dispositifs et systèmes qui sont utilisés avec le robot pour effectuer la tâche nécessaire.

I.14.1. Avantages :

- Robotique et automatisation peut dans de nombreuses situations d'accroître la productivité, la sécurité, l'efficacité, la qualité et la cohérence des produits,
- Robots peuvent travailler dans un environnement dangereux, sans le besoin de soutien de la vie, ou les préoccupations concernant la sécurité,
- Robots n'ont pas besoin de l'éclairage, la climatisation, de ventilation et de protection contre le bruit,
- Robots travail continuellement, sans ressentir une fatigue ou l'ennui, et ne nécessitent pas une assurance médicale ou de vacances,
- Robots sont de précision répétable à tous les moments, sauf si quelque chose arrive à eux ou ils s'usent,
- Robots peuvent être beaucoup plus précis que les humains. Précision linéaire d'un robot typiquement est de 20 à 10 microns.

I.14.2. Inconvénients :

L'inconvénient des robots est qu'ils manquent de capacité de réagir en cas d'urgence, à moins que les situations comprises et les réponses sont inclut dans le système. Les mesures de sécurité nécessaires pour s'assurer qu'ils ne lèsent pas les opérateurs et n'endommagent les machines qui travaillent avec eux.

- Manque de pouvoirs prendre une décision,
- Peuvent causer des dommages à des autres appareils, et la blessure de l'homme,
- Bien que les robots ont de bonnes certaines caractéristiques mais aussi ont ces caractéristiques limités comme la capacité à degré de liberté, la dextérité, capteurs, système de vision et la réponse en temps réel,
- Robots sont coûteux en raison du : coût initial de l'équipement, le coût d'installation, le besoin de périphériques, le besoin de formation et la nécessité de la programmation.

I.15. Conclusion :

La robotisation à changer le monde depuis les premières créations, le faite d'éliminer le travail manuel son influx d'une manière positive en évitant les flux d'attente, en production elle empêche la mal-conformité du produit en contrôlant la production, de là on peut conclure que l'intégration des bras manipulateurs en industrie est indispensable.

Dans le chapitre suivant on détaille notre modélisation mathématique pour obtenir le modelé géométrique direct et inverse de notre bras manipulateur.

Chapitre II :

Modélisation des Bras Manipulateurs

II. Introduction :

Pour développer une stratégie de commande performante pour un robot (bras manipulateur), il est impératif de connaître la cinématique et la dynamique du bras considéré. Et comme les bras manipulateurs sont des systèmes mécaniques poly articulé, fortement non linéaires, d'une dynamique qui peut être mal définis, ils ont des paramètres variables et sont soumis à des perturbations externes. Pour pouvoir faire exécuter une action où une tâche au bras manipulateur, il est nécessaire de modéliser ce système multi variable pour positionner les différents axes. La modélisation de ces systèmes relève très souvent d'une approximation des phénomènes physiques mis en jeu, c'est à partir de cette représentation approximative que l'on souhaite construire une commande pour le système réel, cette commande doit être alors robuste dans le sens où elle devra assurer une faible sensibilité aux incertitudes des paramètres, à leurs variations et aux perturbations externes.

Les erreurs de modélisation, les incertitudes sur l'estimation des paramètres physiques ainsi que les différentes perturbations externes influent beaucoup sur la qualité du contrôle.

Tous ces facteurs doivent être pris en compte lors de la modélisation et de l'élaboration de la loi de commande pour préserver la qualité et la précision des tâches auxquelles est destiné le bras. En effet la majorité des tâches confiées aux bras sont délicates et exigent une très grande précision sous des trajectoires précisé et rapides.

Dans ce chapitre, nous examinerons divers concepts de la théorie de contrôle utilisés dans la commande des bras. On élaborera le modèle d'un bras manipulateur à cinq degrés de liberté à travers la mise en forme des équations mathématiques régissant le mouvement de ce bras. Nous montrerons comment inclure la dynamique des actionneurs du bras.

II.1. Modélisation des bras manipulateurs :

Pour commander un bras manipulateur où pour simuler son comportement on doit disposer de modèles. Plusieurs niveaux de modélisation sont possibles selon les objectifs, les contraintes de la tâche et les performances recherchées : modèles géométriques, cinématiques et dynamiques. (Figure II.1)

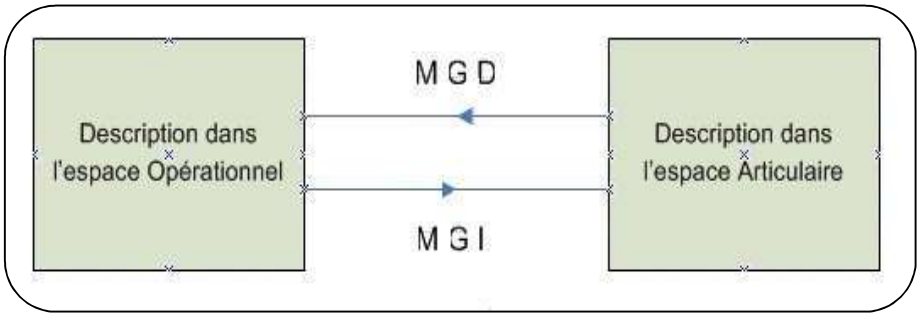


Figure II.1 : Modèle géométrique directe et inverse.

L'obtention de ses différents modèles n'est pas aisée, la difficulté variant selon la complexité de la cinématique de la chaine articulée. Entrent en ligne de compte le nombre de degrés de liberté, le type des articulations mais aussi le fait que la chaine peut être ouverte simple, arborescente ou fermée. Les outils mathématiques que nous présentons dans ce travail utilisent une description des mécanismes fondée sur des notations originales qui permettent d'avoir la même approche quelle que soit la structure.

La conception et la commande des bras nécessitent le calcul de certains modèles mathématiques, tels que : les modèles de transformation entre l'espace opérationnel (dans lequel est définie la situation de l'organe terminal) et l'espace articulaire (dans lequel est définie la configuration du bras). On distingue :

- Les modèles géométriques direct et inverse qui expriment la situation de l'organe terminal en fonction des variables articulaires du mécanisme et inversement,
- Les modèles cinématiques direct et inverse qui expriment la vitesse de l'organe terminal en fonction des vitesses articulaires et inversement,
- les modèles dynamiques définissant les équations du mouvement du bras, qui permettent d'établir les relations entre les couples ou forces exercées par les actionneurs et les positions, vitesses et accélérations des articulations.

II.2. Modèle géométrique direct (MGD):

Le modèle géométrique direct d'un notre bras manipulateur exprime la situation de son Organe Terminal (OT) en f.nction de sa configuration. Donc il exprime la position instantanée de l'OT par rapport au repère fixe lié au bâti. Le modèle géométrique direct est unique et est donne sous forme d'équations explicites **[5]**.

$$: \boldsymbol{N} \rightarrow \boldsymbol{M} \qquad \boldsymbol{q} \rightarrow \boldsymbol{x} = (\boldsymbol{q}) \qquad \text{(II.1)}$$

Ou :

$\boldsymbol{q} = [q_1, q_2, \dots q_n] \in R$: Étant le vecteur des variables articulaires

$X = [x_1, x_2, \dots x_n] \in R$: Étant le vecteur des variables opérationnelles.

Pour calculer le MGD, on suppose que les segments du bras sont parfaitement rigides et les articulations mécaniquement parfaites, il existe plusieurs méthodes pour déterminer le MGD, la plus répandue est celle de Denavit-Hartenberg.**[6]**

C'est-à-dire les coordonnées opérationnelles du bras, en fonction de ses coordonnées articulaires. Dans le cas d'une chaîne ouverte simple, il peut être représenté par la matrice de transformation T_n^0. **[5]**

$$\boldsymbol{T_n^0} = \boldsymbol{T_1^0}(\boldsymbol{q_1})\boldsymbol{T_2^1}(\boldsymbol{q_2}) \dots \dots \dots \boldsymbol{T_n^{n-1}}(\boldsymbol{q_n}) \qquad \text{(II.2)}$$

En réalisant la composition des transformations du repère universel T0 jusqu'au repère T5, sont partitionnées en deux ensemble de matrices, une qui donne la position et l'orientation du porteur :

$$T_0^3 = T_0^1 T_1^2 T_2^3 \quad \text{(II.3)}$$

Et l'autre qui donne la position et l'orientation de l'OT.

$$T_3^5 = T_3^4 T_4^5 \quad \text{(II.4)}$$

On obtient :

$$T_0^5 = [T_0^1 T_1^2 T_2^3]\,[T_3^4 T_4^5] \quad \text{(II.5)}$$

II.2.1. Principe de MDG :

Le principe de cette mét hode est basé sur trois étapes essentielles :

- Fixer des repères à chaque corps du bras,
- Calculer les matrices homogènes entre chaque corps,
- Calculer la matrice homogène entre base et organe terminal.

II.3. Modèle géométrique inverse (MGI) :

Le modèle géométrique direct d'un bras permet de calculer les coordonnées opérationnelles donnant la situation de l'organe terminal en fonction des coordonnées articulaires, le problème inverse consiste à calculer les coordonnées articulaires correspondant à une situation donnée de l'organe terminal. Le modèle s'écrit :

$$\boldsymbol{x} = \boldsymbol{f}^{-1}(\boldsymbol{q}) \quad \text{(II.6)}$$

Lorsqu'elle existe la forme explicite qui donne toutes les solutions possibles. (Il y a rarement unicité de solution) constitue ce que l'on appelle le modèle géométrique inverse (MGI). La détermination du modèle géométrique inverse (MGI) est un problème complexe. On doit inverser un système d'équations non linéaires ce qui n'est pas trivial. Toutefois et selon la structure du manipulateur étudié, il existe des méthodes permettant de résoudre le MGI sous une forme explicite.

II.4. Calcule du modèle géométrique direct :

Méthode de DENAVIT-HARTENBERG (D-H) [6] (Figure II.2**):**

Cette méthode destinée à systématiser la modélisation de n'importe quel type de bras manipulateurs série.

Ses principaux avantages sont :

- Fixer des repères à chaque corps du bras,
- Simplification maximale du modèle géométrique,
- Etablissement d'une norme reconnue par tous.

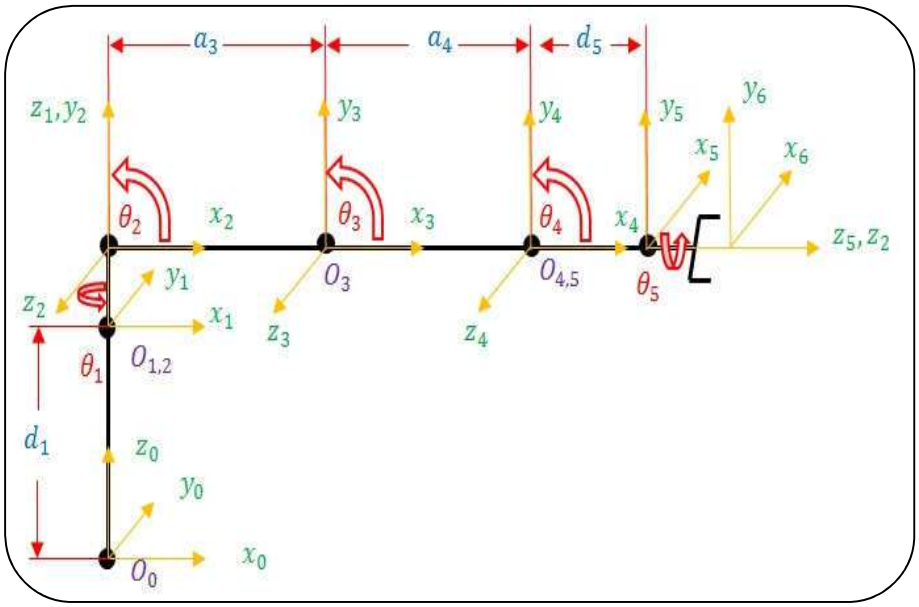


Figure II.2 **:** Liaisons selon Danavit –Hartenberg. **[7]**

Le passage de R_i à R_{i+1} s'exprime en fonction des quatre paramètres suivants :

$\mathbf{a_i}$: Longueur de la perpendiculaire commune aux axes des liaisons l_i et l_{i+1} ,

$\boldsymbol{d_i}$: Distance le long de l'axe, du support de $x_i$au support de x_{i+1} ,

$\boldsymbol{\alpha_i}$: Angle entre Z_i et Z_{i+1} mesurée autour de x_{i+1} ,

$\boldsymbol{\theta_i}$: Angle entre x_i et x_{i+1} mesurée autour de Z_{i+1} .

La position et l'orientation du corps C_i par rapport au corps C_{i+1} sont définies par la matrice de passage homogène $T_{i,i+1}$ du repère R_i (lié à C_i) au repère R_{i+1} (lié à C_i). Cette matrice s'obtient en fonction des quatre paramètres de D-H$\boldsymbol{a_i, d_i, \alpha_i, \theta_i}$.

Notre bras manipulateur (Figure II.3) est disposé de cinq articulations de type rotation et une pince. L'articulation 1ayant comme axe z_1 représente le premier corps et permet une rotation angulaire θ_1 autour de l'axe z_1 dans le plan(x_1, y_1).

L'articulation 2 est définit comme le deuxième corps et son axe est perpendiculaire sur l'articulation 1 provoque une rotation angulaire θ_2 autour de l'axe z_2 dans le plan(x_2, y_2).

Les articulations 3 et 4 sont parallèles à l'articulation 2 (axes de z), et provoque les rotations angulaires θ_3, θ_4 dans les plans $(x_3, y_{3)}\ et\ (x_4, y_{4)}$ successivement.

L'articulation 5 concerne la rotation du poignet de pince. Son axe z_5 qui est verticale sur z_4 provoque la rotion angulaire θ_5 dans le plan(x_5, y_5).

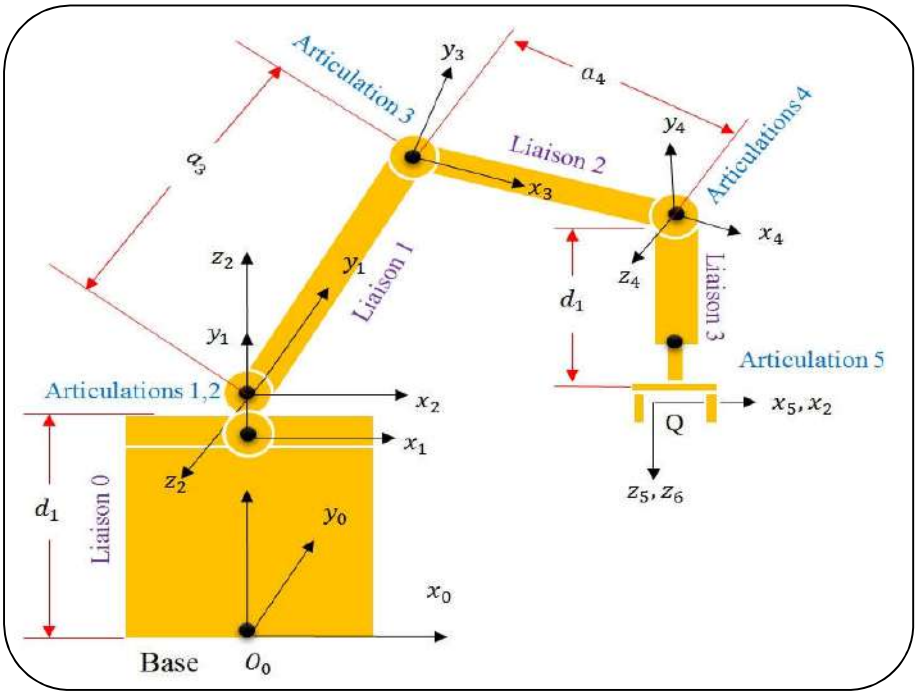


Figure II.3 : Placement des repères et notation du notre bras.

Les paramètres D-H et les matrices de passage homogènes pour notre bras sont représentés dans le tableau suivant :

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	d_1	θ_1^*
2	90	0	0	θ_2^*
3	0	a_3	0	θ_3^*
4	0	a_4	0	$(\theta_4 - 90)^*$
5	-90	0	d_5	θ_5^*
6	0	0	0	$pine$

Tableau II.1 **:** Paramètres DENAVIT-HARTENBERG du notre bras.

On a : $$H = \begin{bmatrix} c_\theta & -s_\theta & 0 & a \\ c_\alpha s_\theta & c_\theta c_\varepsilon & -s_\alpha & -ds_\alpha \\ s_\alpha s_\theta & s_\alpha c_\theta & c_\alpha & dc_\alpha \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{(II.7)}$$

On remplaçant les paramètres du tableau 1 dans l'équation II.1, la transformation de la matrice de T_1 à T_6 est obtenue comme c'est montré au-dessous. Par exemple,T_1 montre la transformation entre les corps 0 et 1(désignant C_i comme ($\cos\theta$) et S_i comme ($\sin\theta$), et…).

$$T_1^0 = \begin{bmatrix} c_{\theta_1} & -s_{\theta_1} & 0 & 0 \\ s_{\theta_1} & c_{\theta_1} & 0 & 0 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{(II.8)}$$

$$T_2^1 = \begin{bmatrix} c_{\theta_2} & -s_{\theta_2} & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s_{\theta_2} & c_{\theta_2} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{(II.9)}$$

$$T_3^2 = \begin{bmatrix} c_{\theta_3} & -s_{\theta_3} & 0 & a_3 \\ s_{\theta_3} & c_{\theta_3} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{(II.10)}$$

$$T_4^3 = \begin{bmatrix} c_{\theta_4} & -s_{\theta_4} & 0 & a_4 \\ s_{\theta_4} & c_{\theta_4} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{(II.11)}$$

$$T_5^4 = \begin{bmatrix} c_{\theta_5} & -s_{\theta_5} & 0 & 0 \\ 0 & 0 & 1 & d_5 \\ s_{\theta_5} & c_{\theta_5} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{(II.12)}$$

$$T_{pince} = \begin{bmatrix} c_{\theta_5} & -s_{\theta_5} & 0 & 0 \\ s_{\theta_5} & c_{\theta_5} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{(II.13)}$$

En utilisant les valeurs de la matrice de transformation ci-dessus, les liens de transformation peuvent être concaténés (multipliés ensemble) dans le but de trouver une seule transformation qui relie le corps (5) au corps (0) :

$$T_5^0 = T_1^0 T_2^1 T_3^2 T_4^3 T_5^4 = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{(II.14)}$$

La simulation d'orientation et de position de notre bras manipulateur de 5ddl peut être calculée en fonction des ongles articulatoires en utilisant les fonctions suivantes **[7]** :

$s_i = \sin\theta_i$; $c_i = \cos\theta_i$

$n_x = \big((c_1c_2c_{3-}c_1s_2s_2)c_{4+}(-c_1c_2s_3 - c_1s_2c_3)s_4\big)c_5 + s_1s_5$

$n_y = \big((s_1c_2c_3 - s_1s_2s_3)c_4 + (-s_1c_2s_3 - s_1s_2c_3)s_4\big)c_5 - c_1s_5$

$n_z = \big((s_2c_3 + c_2s_3)c_4 + (-s_2s_3 + c_2c_3)s_4\big)c_5$

$O_x = -\big((c_1c_3c_3 - c_1s_2s_3)c_4 + (-c_1c_2s_3 - c_1s_2c_3)s_4\big)s_5 + s_1c_5$

$O_y = -\big((s_1c_2c_3 - s_1s_2s_3)c_4 + (-s_1c_2s_3 - s_1s_2c_3)s_4\big)s_5 - c_1c_5$

$O_z = \big((c_2c_3 - s_2s_3)s_4\big)s_5 - \big((s_2c_3 + c_2s_3)c_4\big)$

$a_x = -(c_1c_2c_3 - c_1s_2s_3)s_4 + (-c_1c_2s_3 - c_1s_2c_3)c_4$

$a_y = -(s_1c_2c_3 - s_1s_2s_3)s_4 + (-s_1c_2s_3 - s_1s_2c_3)c_4$

$a_z = (c_2c_3 - s_2s_3)c_4 - (s_2c_3 + c_2s_3)s_4$

$d_x = (-(c_1c_2c_3 - c_1s_2s_3)s_4 + (-c_1c_2s_3 - c_1s_2c_3)c_4)d_5 + (c_1c_2c_3 - c_1s_2s_3)a_4 + c_1c_2a_3$

$d_y = (-(s_1c_2c_3 - s_1s_2s_3)c_4 + (-s_1c_2s_3 - s_1s_2c_3)c_4)d_5 + (s_1c_2c_3 - s_1s_2s_3)a_4 + s_1c_2a_3$

$d_z = (-(s_2c_3 + c_2s_3)s_4 + (-s_2s_3 + c_2c_3)c_4)d_5 + (s_2c_3 + c_2s_3)a_4 + s_2a_3 + d_1$

II.5. Conclusion :

Dans ce chapitre nous avons exposé les modélisations des bras manipulateurs pour décrire les mouvements des articulations d'un notre bras manipulateur, dans le but est de fournir les équations mathématiques qui permettent de programmer, commander et exécuter les tâches désirées.

Afin, on a envisagé la modélisation géométrique du bras manipulateur réalisé, le calcul par la méthode de Denavit-Hartenberg, et l'obtention du modèle géométrique inverse.

Chapitre III :

Traitement d'Images

et la Vision par Ordinateur

III. Introduction :

Dans ce chapitre, nous allons introduire le technique traitement d'images. Après un bref caractéristique d'une image, nous décrirons en détail les différentes architectures existantes des systèmes asservis par vision. Ce descriptif sera suivi d'une présentation des différentes classifications des systèmes d'asservissement visuel 2D, 3D, et la nature des grandeurs asservies et le type de commande.

Le traitement d'images est un domaine très vaste qui a connu, et qui connaît encore, un développement important depuis quelques dizaines d'années.

On désigne par traitement d'images numériques l'ensemble des techniques permettant de modifier une image numérique afin d'améliorer ou d'en extraire des informations.

De ce fait, le traitement d'images est l'ensemble des méthodes et techniques opérant sur celles-ci, dans le but de rendre cette opération possible, plus simple, plus efficace et plus agréable, d'améliorer l'aspect visuel de l'image et d'en extraire des informations jugées pertinentes.

III.1. Définition de l'image :

L'image est une représentation d'une personne ou d'un objet par la peinture, le dessin, la photographie, le film, etc.…. C'est aussi un ensemble structuré d'informations qui, après affichage sur l'écran, ont une signification pour l'oeil humain.

Elle peut être décrite sous la forme d'une fonction I(x, y) analogique continue, définie dans un domaine borné, tel que x et y sont les coordonnées spatiales d'un point de l'image et I est une fonction d'intensité lumineuse et de couleur. Sous cet aspect, l'image est inexploitable par la machine, ce qui nécessite sa numérisation. **[8]**

III.2. Acquisition d'une image :

L'acquisition d'images constitue un des maillons essentiels de toute chaîne de conception et de production d'images. Pour pouvoir manipuler une image sur un système informatique, il est avant tout nécessaire de lui faire subir une transformation qui la rendra lisible et manipulable par ce système. Le passage de cet objet externe (l'image d'origine) à sa représentation interne (dans l'unité de traitement) se fait grâce à une procédure de numérisation.

Ces systèmes de saisie, dénommés optiques, peuvent être classés en deux catégories principales :

- Les caméras numériques,
- Et les scanners.

A ce niveau, notons que le principe utilisé par le scanner est de plus en plus adapté aux domaines professionnels utilisant le traitement de l'image comme la télédétection, les arts graphiques, la médecine, etc. Le développement technologique a permis l'apparition de nouveaux périphériques d'acquisition appelés cartes d'acquisition, qui fonctionnent à l'instar des caméras vidéo, grâce à un capteur *C.C.D*. La carte d'acquisition reçoit les images de la camera, de la T.V. ou du scanner afin de les convertir en informations binaires qui seront stockées dans un fichier. L'une des caractéristiques intéressantes de ces capteurs est la taille du plus petit élément (*pixel*), mais aussi l'inter corrélation de deux éléments voisins : plus cette intercorrélation est faible, meilleure est l'image.

III.3. Caractéristiques d'une image numérique :

L'image est un ensemble structuré d'informations caractérisé par les paramètres suivants :

III.3.1. Pixel :

Contraction de l'expression anglaise " Picture Eléments ": éléments d'image, le pixel est le plus petit point de l'image, (Figure III.1) c'est une valeur numérique représentative des intensités lumineuses.

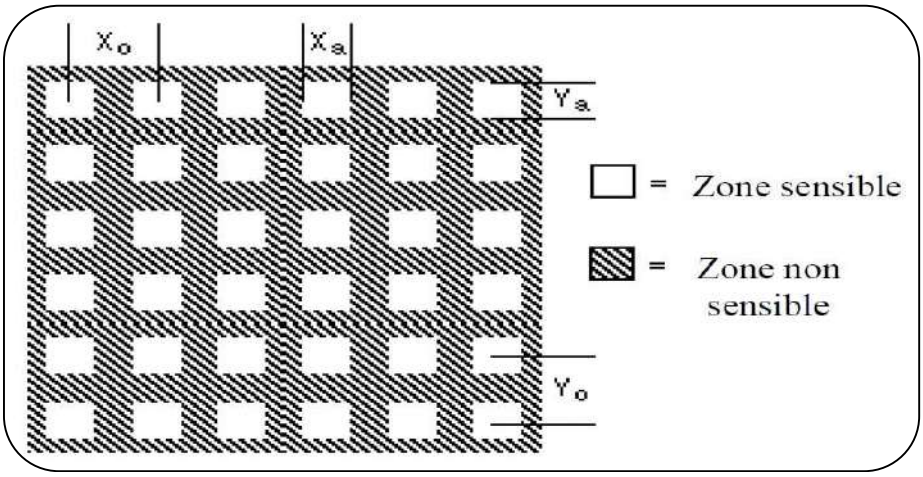


Figure III.1 : Forme du pixel.

Si le bit est la plus petite unité d'information que peut traiter un ordinateur, le pixel est le plus petit élément que peuvent manipuler les matériels et logiciels sur l'image. La lettre A, par exemple, peut être affichée comme un groupe de pixels.

Le pixel a été utilisé par défaut, mais il peut être remplacé par le point (pt) pour le cas d'une image imprimée. Cela ne change rien aux calculs ni aux résultats algébriques.

Par contre si le mot pixel est remplacé par le point, la résolution s'exprimera alors en point par pouce (ppp) ou dot per inch (in) en anglais.

III.3.2. Dimension :

C'est la taille de l'image. Cette dernière se présente sous forme de matrice dont les éléments sont des valeurs numériques représentatives des intensités lumineuses (pixels). Le nombre de lignes de cette matrice multiplié par le nombre de colonnes nous donne le nombre total de pixels dans une image.

Elle correspond à la largeur et hauteur que votre image occupe sur votre écran. Celle-ci s'exprime en pixels.

Dimension = (Nb pixels sur la Longueur) x (Nb pixels sur la Hauteur)

III.3.3. Résolution :

C'est la clarté ou la finesse de détails atteinte par un moniteur ou une imprimante dans la production d'images. Sur les moniteurs d'ordinateurs, la résolution est exprimée en nombre de pixels par unité de mesure (pouce ou centimètre). On utilise aussi le mot résolution pour désigner le nombre total de pixels affichables horizontalement ou verticalement sur un moniteur ; plus grand est ce nombre, meilleure est la résolution.

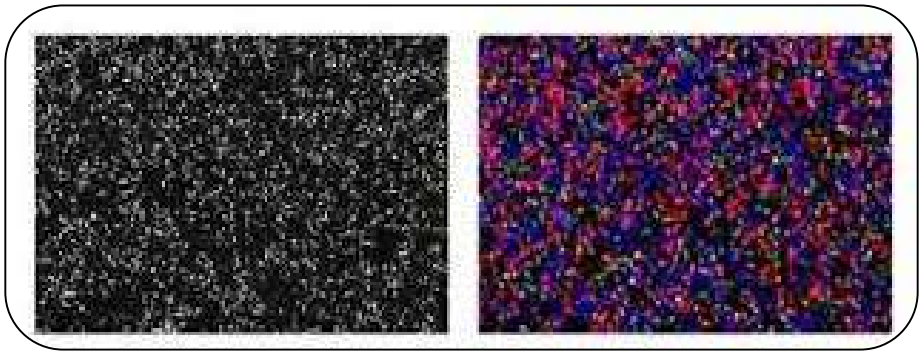
III.3.4. Bruit :

En bref, un bruit (parasite) dans une image est considéré comme un phénomène de brusque variation de l'intensité d'un pixel par rapport à ses voisins, il provient de l'éclairage des dispositifs optiques et électroniques du capteur. **[9]**

III.3.4.a. Illustration des différents types de bruit :

Visuellement, on distingue deux sortes de bruit : (Figure III.2)

- **Le bruit de luminance :** des zones de l'image trop sombres ou trop claires. C'est ce qui ressemble au grain du film argentique,
- **Le bruit de chrominance :** des pixels colorés répartis de manière aléatoire sur la photo.



a) bruit de luminance b) bruit de chrominance.

Figure III.2 : Les bruits.

III.3.5. Histogramme :

L'histogramme des niveaux de gris ou des couleurs d'une image est une fonction qui donne la fréquence d'apparition de chaque niveau de gris (couleur) dans l'image. Il permet de donner un grand nombre d'information sur la distribution des niveaux de gris (couleur) et de voir entre quelles bornes est repartie la majorité des niveaux de gris (couleur) dans le cas d'une image trop claire ou d'une image trop foncée.

Il peut être utilisé pour améliorer la qualité d'une image (Rehaussement d'image) en introduisant quelques modifications, pour pouvoir extraire les informations utiles de celle-ci. Pour diminuer l'erreur de quantification, pour comparer deux images obtenues sous des éclairages différents, ou encore pour mesurer certaines propriétés sur une image, on modifie souvent l'histogramme correspondant.

III.3.6. Luminance :

C'est le degré de luminosité des points de l'image. Elle est définie aussi comme étant le quotient de l'intensité lumineuse d'une surface par l'aire apparente de cette surface, pour un observateur lointain, le mot luminance est substitué au mot brillance, qui correspond à l'éclat d'un objet. Une bonne luminance se caractérise par :

- Des images lumineuses (brillantes),
- Un bon contraste : il faut éviter les images où la gamme de contraste tend vers,
- Le blanc ou le noir ; ces images entraînent des pertes de détails dans les zones sombres ou lumineuses,
- L'absence de parasites.

III.3.7. Contraste :

C'est l'opposition marquée entre deux régions d'une image, plus précisément entre les régions sombres et les régions claires de cette image. Le contraste est défini en fonction des luminances de deux zones d'images. Si L1 et L2 sont les degrés de luminosité respectivement de deux zones voisines A1 et A2 d'une image, le contraste C est défini par le rapport.

III.4. Images en couleurs :

Même s'il est parfois utile de pouvoir représenter des images en noir et blanc, les applications multimédias utilisent le plus souvent des images en couleurs. La représentation des couleurs s'effectue de la même manière que les images monochromes avec cependant quelques particularités. En effet, il faut tout d'abord choisir un modèle de représentation. On peut représenter les couleurs à l'aide de leurs composantes primaires.

Les systèmes émettant de la lumière (écrans d'ordinateurs,...) sont basés sur le principe de la synthèse additive : les couleurs sont composées d'un mélange de rouge, vert et bleu (modèle R.V.B).

III.4.1. Energie lumineuse-Classification fréquentielle des ondes lumineuses :

Les ondes électromagnétiques sont classées et réparties en fonction de leur longueur d'onde ou de leur fréquence (Figure III.3), cette répartition est appelée spectre électromagnétique. Ce spectre est représenté sur la figure suivante, qui consiste en une bande contenant tous les types de rayonnement électromagnétique qui existent dans l'univers.

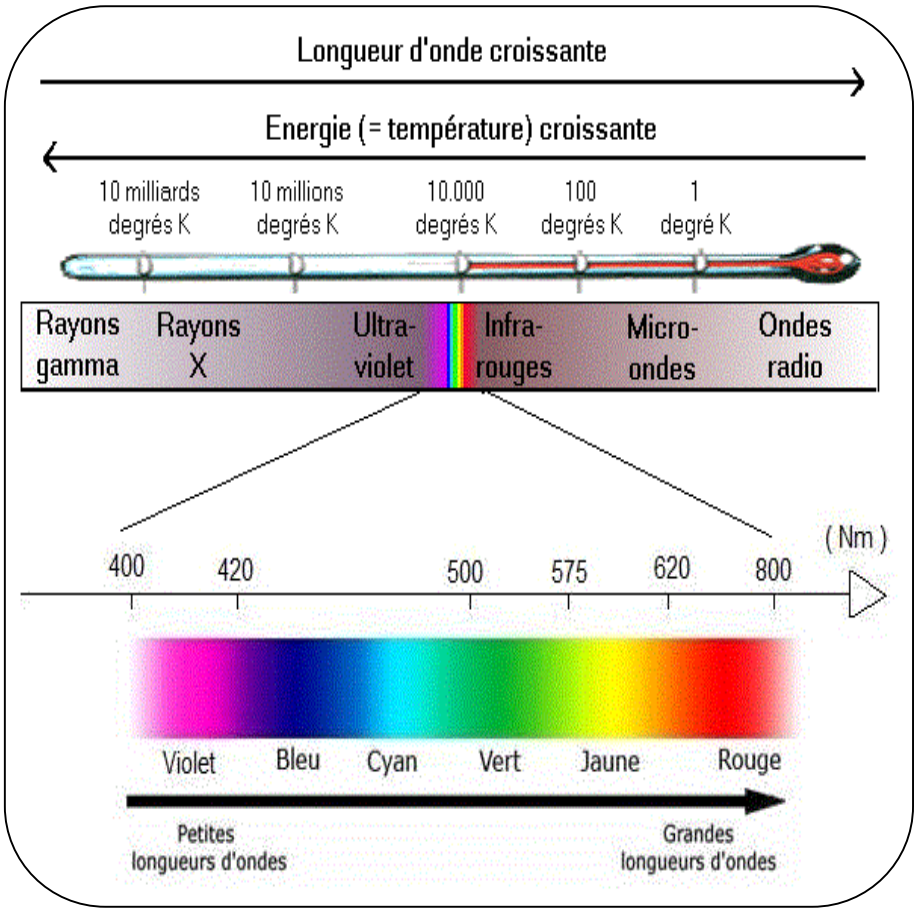


Figure III.3 : Le spectre électromagnétique.

III.5. Système de traitement d'images :

Un système de traitement numérique d'images est composé de : (Figure III.4)

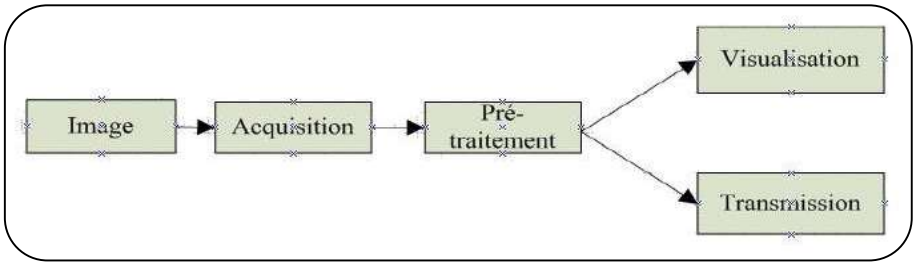


Figure III.4 : Schéma d'un système de traitement d'images.

De ce fait, le traitement d'images est l'ensemble des méthodes et techniques opérant sur celles-ci, dans le but de rendre cette opération possible, plus simple, plus efficace et plus agréable, d'améliorer l'aspect visuel de l'image et d'en extraire des informations jugées pertinentes.

III.5.1 Capteurs de vision :

La technologie de vision est devenue ces dernières années un point crucial pour l'intégration de robot ou pour l'automatisation d'un procédé industriel. Les technologies sont de plus en plus fiables, avec une performance élevée et des prix sont devenus beaucoup plus abordables.

La vision est une capacité de haut niveau pour un bras. Etre capable de distinguer des formes et des couleurs dans une scène, voilà le challenge auquel vous allez avoir à faire avec une caméra embarquée sur notre bras manipulateur.

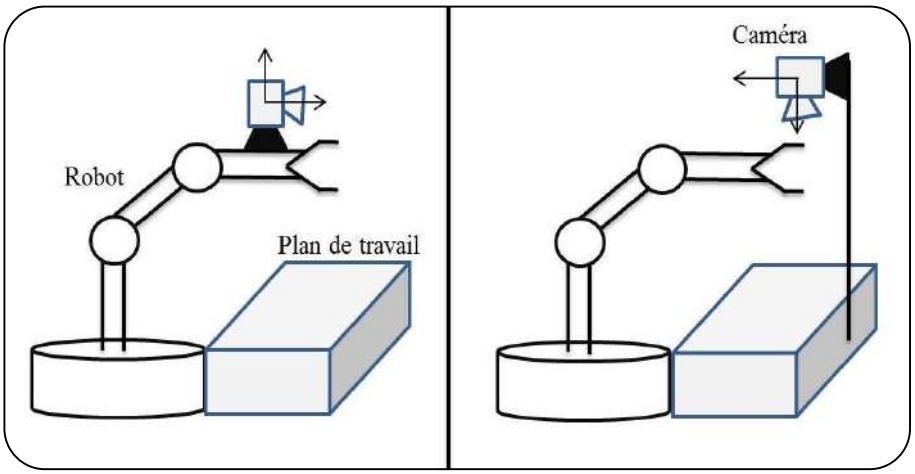
La vision industrielle peut être définie comme l'application de la vision par ordinateur à des problématiques de production. Son principe est de doter les machines de production de la capacité de voir afin d'automatiser les tâches de contrôle qualité ou de contrôle de processus.

Cette automatisation permet d'augmenter les performances et les cadences de production, de rendre la production plus fiable, d'améliorer la qualité des produits, d'assurer leur traçabilité, et de garantir la sécurité.

Les applications de vision industrielle sont aujourd'hui nombreuses et se sont ouvertes à tous les secteurs de l'industrie. En effet, les progrès techniques tant au niveau des caméras et des systèmes d'éclairage que des systèmes informatiques ont permis un élargissement considérable du champ d'application de la vision industrielle.

III.5.2. Le positionnement de la caméra :

Le système de vision (traitement d'image) est généralement intégré sur une ligne de production et nécessite un système mécanique associé à un automatisme permettant de convoyer, détecter et isoler le produit à inspecter. Le système de vision gère également différentes entrées/sorties comme les entrées de déclenchement de l'acquisition (trigger, encodeur) ou les sorties de déclenchement de l'éclairage, il existe deux types configurations de la vision. (Figure I.5)



a) Caméra embarquée **[10]** b) caméra déportée **[11]**

Figure III.5 : Système de composition du la caméra.

Dans notre configuration, la caméra a été déportée et observe le plan constituée de l'organe terminal du bras manipulateur et de l'objet d'intérêt. Ici, la tâche asservie par vision doit être réalisée dans le champ de vision de la caméra qui est fixe. Elle est donc plus adaptée à des tâches locales exécutées dans un petit sous-espace de plan de travail (table ou convoyeur) du bras manipulateur.

De plus dans ce type de configuration, le bras manipulateur sera un système simple, et n'effectuera que très peu de taches.

C'est le système de vision qui est très important dans ce cas de notre projet car c'est lui qui va donner la position de la pièce, les défauts de celle-ci, etc…

III.5.3. Acquisition et numérisation :

L'acquisition d'images constitue un des maillons essentiels de toute chaîne de conception et de production d'images. Pour pouvoir manipuler une image sur un système informatique, il est avant tout nécessaire de lui faire subir une transformation qui la rendra lisible et manipulable par ce système. Le passage de cet objet externe (l'image d'origine) à sa représentation interne (dans l'unité de traitement) se fait grâce à une procédure de numérisation.

Ces systèmes de saisie, dénommés optiques, peuvent être classés en deux catégories principales : les caméras numériques et les scanners.

Il existe deux méthodes principales d'acquisition des images numériques. La première consiste à prendre une diapositive avec un appareil photo classique, puis à la numériser à l'aide d'un scanner, l'inconvénient majeur de cette méthode est nécessité la caractérisation colorimétrique souvent difficile du scanner.

La seconde méthode d'acquisition consiste à numériser le signal produit par une caméra vidéo, cette dernière possède une matrice de détections qui se comportent comme des compteurs de photons, La résolution typique d'une caméra CCD possédant une matrice de diagonale 1/3 inch est de 350000 points, mais peut dépasser le million dans les modèles haut de gamme. La luminance énergétique associée à chaque pixel de l'image est alors représentée par une valeur numérique comprise entre 0 à 255. Cette méthode est présentée en annexe A.

Dans leur grande majorité, les caméras ont une réponse spectrale qui est fonction de la luminance énergétique. Ainsi deux objets de même couleur pourront apparaître différents en fonction du niveau d'éclairement.

L'architecture des systèmes CCD couleur vise à imiter l'organisation de la rétine humaine ; dans une caméra couleur mono-CCD une matrice de filtres rouges, verts et bleus (RVB) est utilisée pour contrôler la fraction de spectre qui atteint chaque détecteur.

III.5.4. Pré-traitements :

De manière générale, les opérations de prétraitement ont pour objectif de faciliter la segmentation en renforçant la ressemblance entre les pixels d'une même région et en accentuant les différences entre pixels de régions voisines. En particulier, les opérations de filtrage permettent de réduire certains bruits et artefacts susceptibles de perturber la segmentation.

III.5.5. Traitement d'images avec MatLab :

L'objectif de cette introduction au traitement d'images sous Matlab est de présenter la notion d'image et d'effectuer des opérations simples d'analyse d'images telles que la détection de contours, le changement d'espace de couleur.

Une image réelle est obtenue à partir d'un signal continu bidimensionnel comme par exemple un appareil photo ou une caméra sur notre bras, on ne peut pas représenter de signaux continus. C'est un signal fini bidimensionnel échantillonné à valeurs quantifiées dans un certain espace de couleurs. Elle est constituée de points (pixels).

Espace de couleur : il existe de nombreuses façon de percevoir les couleurs d'une image, l'espace de représentation le plus connu est l'espace RGB (rouge-vert, bleu).

Une image est une matrice M x N de valeurs entières prises sur un intervalle borné [0,Ng] où : Ng est la valeur maximale du niveau de gris. (Figure III.6)

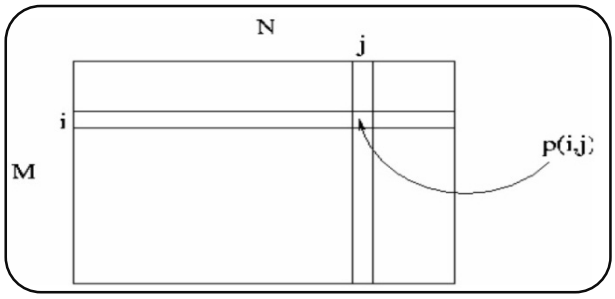


Figure III.6 : Représentation l'image du niveau de gris.

p (i,j) est le niveau de gris du pixel de coordonnées ligne i et colonne j dans l'image.

p (i,j) ϵ [0, Ng]. Les valeurs des niveaux de gris sont des entiers.

III.5.5.a. Types des images :

- Les images binares,
- Les images d'intensité (à niveaux de gris),
- Les images couleur RGB,
- Les images couleur indexées.

- **Image binaire :**

Une image binaire est une image M x N où chaque point peut prendre uniquement la valeur 0 ou 1. Les pixels sont noirs (0) ou blancs (1). Le niveau de gris est codé sur un bit. Dans ce cas, on revient au cas donné en I.1. avec Ng = 2 et la relation sur les niveaux de gris devient : p (i,j) = 0 ou p (i,j) = 1.

- **Image en niveaux de gris :**

Une image ne niveaux de gris autorise un dégradé de gris entre le noir et le blanc. En général, on code le niveau de gris sur un octet (8 bits) soit 256 nuances de dégradé. L'expression de la valeur du niveau de gris avec Ng = 256 devient : p (i,j) [0, 255].

- **Image couleur :**

Une image couleur est la composition de trois images en niveaux de gris sur trois composantes (Figure III.7). On définit donc trois plans de niveaux de gris, un rouge, un vert et un bleu. La couleur finale est obtenue par synthèse additive de ces trois composantes.

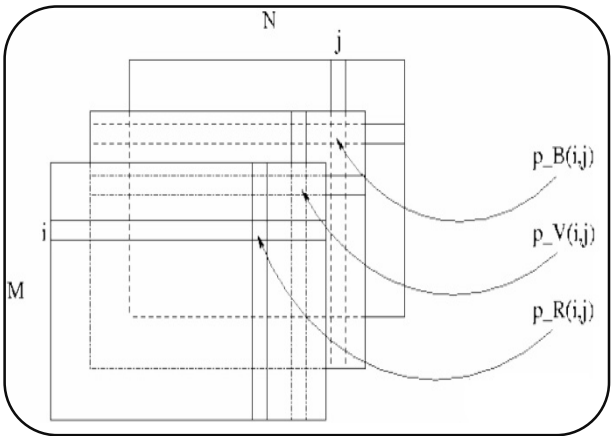


Figure III.7 : Représentation l'image en couleur.

On a les relations sur les niveaux de gris : p_R(i,j) ϵ [0, 255], p_V(i,j) ϵ[0, 255], p_B(i,j) ϵ [0, 255]. On voit bien sur la figure qu'une image couleur est en fait l'association de trois plans de niveau de gris, chacun d'eux étant une couleur de base.

Le passage d'une image couleur à une image intensité correspond à la transformation des composantes RGB en la composante Y des systèmes de transmissions YUV qui séparent l'information de lumineuse et de chrominance.

III.5.6. Visualisation :

Tout système de traitement d'image est doté d'un dispositif de visualisation qui permet l'affichage des images.

L'utilisation de différents types de restituteurs permet de transformer le signal numérique qu'est la matrice image en un signal analogique visible par l'oeil de l'observateur. Pour cela, différents types de supports peuvent être employés : moniteur vidéo, clichés photographiques, impression sur papier. Dans tous les cas et pour chaque échantillon de l'image numérique, on recrée un nouvel élément d'image ou un nouveau pixel dont on choisit la forme de façon à reconstituer une image analogique qui soit la plus proche possible de l'image avant numérisation compte tenu des erreurs introduites lors de l'acquisition, de la numérisation et de la transmission.

III.6. Domaines d'application :

Le traitement d'images possède l'aspect multidisciplinaire. On trouve ses applications dans des domaines très variés tels que les télécommunications (T.V., vidéo, publicité,...), la médecine (radiographie, ultrasons,...), biologie, astronomie, géologie, l'industrie (robotique, sécurité), la météorologie, l'architecture, l'imprimerie, l'armement (application militaire).

De nouvelle applications pratiques sont possibles aujourd'hui et touchent tous les domaines d'activités, tels que : métiers du spectacle, de la radio, créations artistiques.

III.7. Asservissement visuel : (Figure III.8)

Le principe de l'asservissement visuel consiste à prendre en compte des informations visuelles issues d'une ou plusieurs caméras dans la boucle de commande d'un bras afin capable de se suivre une cible de couleur spécifique grâce à un système de vision.

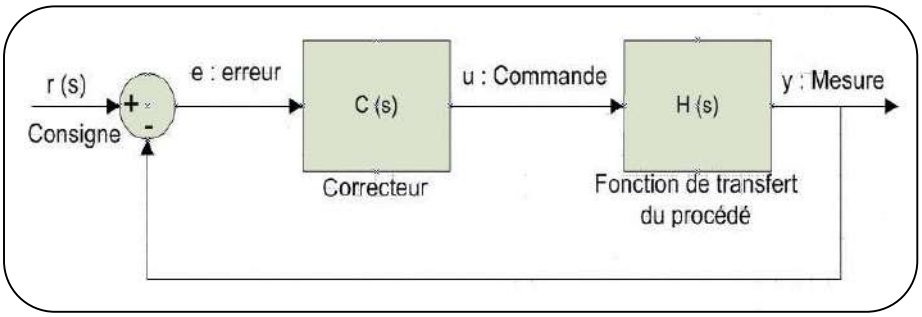


Figure III.8 : Asservissement visuel

La vision industrielle est constituée de deux grands principes d'asservissement qui réponde tous deux à des besoins et critères différents.

III.7.1. Asservissement visuel 2D :

L'asservissement visuel 2D utilise des informations visuelles géométriques extraites des images fournies par une ou plusieurs caméras (Figure III.9). Ces informations visuelles sont rangées dans un vecteur noté s, le vecteur des indices visuels. Les indices visuels peuvent être les coordonnées des points dans l'image, correspondant à des points dans l'espace, ou des caractéristiques géométriques représentant des lignes, des ellipses, des sphères ou des cylindres, sans s'y limiter. Le but de l'asservissement visuel est de minimiser l'erreur entre les vecteurs d'indice visuel s et désirés s^*. Cette erreur a pour expression : **[12]**

$$e(t) = s(q(t), t) - s^*(t) \qquad \text{(III.1)}$$

« q » étant un vecteur décrivant l'état du bras. Le vecteur « $s^*(t)$ » des indices visuels désirés est généralement pris constant. Mais dans certaines applications, comme la planification de chemin dans l'espace d'image, ce vecteur est variable.

Le formalisme des fonctions de tâches, développé dans pour fournir un cadre général et rigoureux à la commande des bras manipulateurs, s'applique à l'asservissement visuel. La fonction d'erreur e est donc définie comme une fonction de tâche. L'on construit généralement une commande en vitesse pour le bras pour réguler l'erreur à zéro. Dans la plupart des cas, on commande le bras de sorte que l'erreur décroisse exponentiellement,

alors la vitesse désirée $\dot{e}^*$ s'écrit, avec un gain γ défini parun réel positif ou une matrice définie positive :

$$\dot{e}^* = - \gamma e \qquad \text{(III.2)}$$

Nous pouvons calculer la dérivée par rapport au temps de la fonction de tâche e. Elle a pour expression :

$$\dot{e}(t) = \dot{s}(t) - \dot{s}^*(t) \qquad \text{(III.3)}$$

La dérivée $\dot{s}^*(t)$ est nulle dans le cas général. Dans le cas contraire elle est imposée, s'il s'agit de planification dans l'image. La dérivée $\dot{s}$ par rapport au temps du vecteur des indices visuels, est liée au mouvement de la caméra ou de la cible. Cette relation peut s'exprimer ainsi :

$$\dot{s} = L_s . \left({}^cT_c - {}^cT_0 \right) \qquad \text{(III.4)}$$

Où :

- L_s est la matrice d'interaction qui relie la variation des indices visuels dans l'imagede la caméra au torseur cinématique de la caméra,
- ${}^cT_c$ ou T_c est le torseur cinématique de la caméra par rapport à un repère fixe, exprimé dans le repère F_c lié à la caméra, en son origine C,
- ${}^cT_o$ est le torseur cinématique qu'aurait un point solidaire de la cible, qui coïnciderait à l'instant avec l'origine du repère F_c. Ce torseur est exprimé dans ce même repère. Généralement, la cible est fixe et le torseur est nul, dans la configuration "embarquée", la plus courante. Dans l'autre configuration "déportée", la cible est mue par un bras et la caméra fixe. Lorsque la caméra et la cible sont tous deux mobiles, il s'agit d'un problème de poursuite de cible et son torseur doit être estimé.

Si la caméra est déportée, c'est le torseur cinématique de l'objet qui est lié à celui de l'organe terminal.

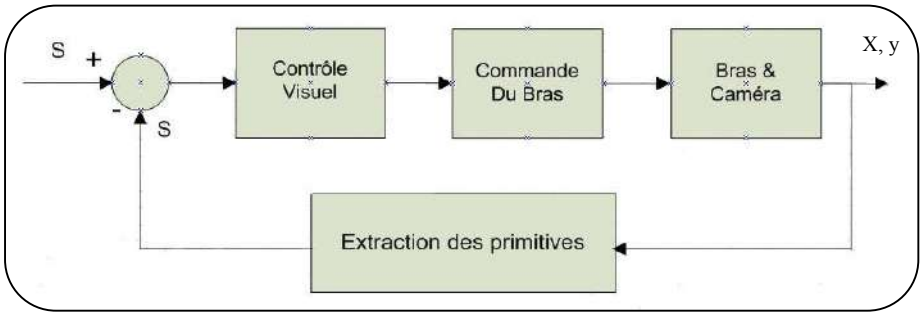


Figure III.9 : Asservissement visuel 2D.

III.7.2. Asservissement visuel 3D :

Pour ce type d'asservissement, la référence est exprimée sous la forme d'une attitude. Dans notre cas, il s'agit de l'attitude « r » d'un repère lié à l'objet vu par la caméra par rapport à un repère lié à l'organe terminal du bras manipulateur (Figure III.10).

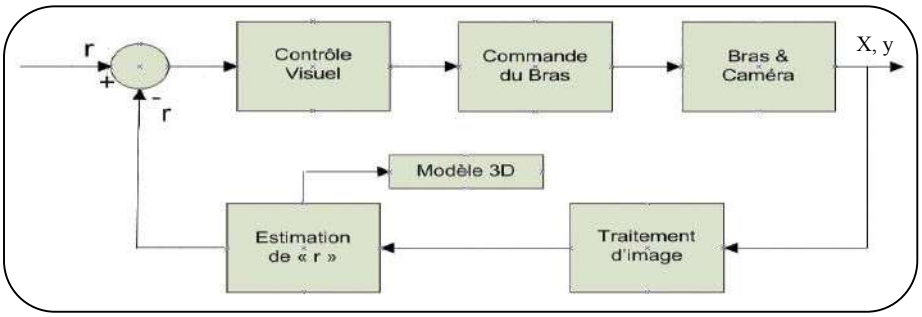


Figure III.10 : Asservissement visuel 3D.

Cette mesure s'obtient grâce aux primitives extraites de l'image et à un modèle géométrique de la cible, mais elle est très sensible aux incertitudes sur ce modèle ainsi qu'aux erreurs de calibrage de la caméra. Néanmoins, ce problème peut être contourné par la boucle fermée, ce qui nous évite les problèmes de convergence du système **[13]**.

La complexité des calculs nécessaires à cette estimation croît avec la complexité de l'objet. C'est pourquoi les objets sont souvent simplifiés au maximum dans les asservissements visuels.

Il a été montré qu'il est nécessaire de connaître la projection d'au moins 4 primitives de type "point" pour pouvoir obtenir cette attitude de manière unique. Si seulement trois projections sont disponibles, l'attitude ne peut être déterminée de manière unique. Le problème de la reconstruction de l'attitude à partir de la projection de 3 ou 4 points au minimum.

Alors que l'asservissement visuel 3D, implique de connaître un modèle tridimensionnel de la cible et de pouvoir la situer dans un repère commun, et de définir une consigne d'asservissement en termes de position et orientation, l'asservissement visuel 2D travaille au niveau de l'image, avec des primitives visuelles (indices) définis au préalable. C'est dans l'image que la reconnaissance de l'objet par ses indices a lieu, et c'est toujours dans l'image que la consigne que doivent atteindre les indices est définie, comme il est indiqué sur la figure suivante.

L'algorithme n'a donc pas besoin de connaître la structure 3D de la scène, les calculs exploitent seulement l'image. Une primitive est une forme géométrique5élémentaire (point, segment de droite, portion d'ellipse,...). Elle sert à modéliser la projection d'un objet dans le

plan image. La plupart des travaux qui traitent de l'asservissement 2D utilisent des primitives constituées de points.

Ces points peuvent par exemple être situés à l'intersection de segments dans l'image ou encore être extraits du centre de gravité de la projection de disques. En fait, les cibles les plus couramment rencontrées sont simplement constituées de plusieurs disques coplanaires. De telles cibles ont l'avantage de nécessiter un temps de traitement d'image faible.

III.8. L'avantage d'un système de vision industrielle :

Est qu'il est possible, avec un seul dispositif, de réaliser systématiquement, sur tous les produits, plusieurs contrôles différents en continu qui nécessiteraient, sans ce dispositif, des appareillages différents :

- le contrôle de conformité d'assemblage permet de vérifier l'absence ou la présence d'éléments constituant le produit à fabriquer, ainsi que leur positionnement et leur orientation,
- le contrôle d'aspect est destiné à examiner les états de surface afin de détecter des défauts d'aspect comme des rayures, des griffures, des trous, des taches, des défauts de nuances de couleurs ou de textures…,
- le contrôle dimensionnel consiste à mesurer les dimensions d'une pièce comme une longueur, un diamètre, une profondeur, un angle ou une géométrie particulière,
- le comptage et le tri de pièces peuvent également être effectués par un système de vision industrielle,
- le pilotage de machines ou de bras est également réalisé grâce aux systèmes de vision. La caméra devient alors l'œil du bras, ce qui permet la détection et la localisation d'une pièce pour la manipuler, l'assembler ou l'aligner. Le suivi d'une trajectoire pour une opération de dépose peut également être ainsi effectué,
- l'identification est destinée à effectuer une reconnaissance ou une vérification de caractères, ainsi que la lecture des codes à barres, des codes matriciels ou des codes couleur. Cette opération permet de référencer un produit et d'assurer sa traçabilité et son traitement statistique, ainsi que de suivre la gestion des stocks ou le flux de production,
- sur site industriel, mais également dans d'autres domaines, la vision permet des opérations de surveillance et de sécurité par du contrôle d'accès (analyse d'empreintes, de visages, de mains, de l'œil, de plaques d'immatriculation) ou du suivi de foules ou de flots de personnes.

III.9. L'architecture de commande 2D du notre bras :

Dans un asservissement 2D visuel indirect, le système de vision transmet les consignes de commande au contrôleur du notre bras. (Figure III.11)

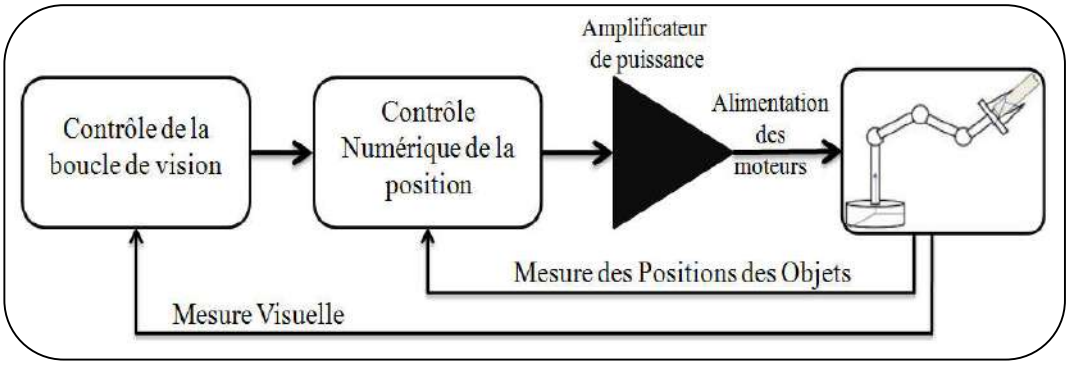


Figure III-11 : Structure d'un asservissement visuel 2D du notre bras.

Le contrôleur effectue à bas niveau l'asservissement en position des coordonnées articulaires. L'avantage de cette architecture réside dans sa simplicité, sa portabilité et sa robustesse, car le bras peut être considère comme un système de positionnement autonome ou les singularités et les sécurités sont gérées par le contrôleur de bas niveau du bras. Cependant les performances du système de vision sont liées à celles du contrôleur de bas niveau qui peut ralentir l'asservissement en introduisant des retards supplémentaires. Il n'est donc pas envisageable d'utiliser cette architecture pour réaliser des asservissements visuels pour lesquels le but recherché est la rapidité.

III.10. Conclusion :

Ce chapitre, nous l'avons voulu à ce qu'il soit une brève introduction aux concepts liés au l'image et domaine du traitement. Les différentes définitions qui y sont développées sont celles des connaissances élémentaires de cette discipline, mais combien même elles sont essentielles pour l'initiation aux traitements approfondis des images.

Dans le prochain chapitre, nous détaillerons l'architecture de notre bras manipulateur. Nos choix quant à l'utilisation de caméra et l'algorithme au travers d'applications de suivi de cible.

Chapitre IV :

Réalisation

IV. Introduction :

La commande en mouvement des bras manipulateurs est essentiellement réalisée par des circuits numériques programmables tels que les microprocesseurs, les microcontrôleurs et les DSP où les lois de commande sont implémentées sous forme d′algorithmes. A noter que les principaux avantages de la commande numérique sont la rapidité de développement, flexibilité et faible espace d′occupation.

Dans ce chapitre le software de commande est présenté en insistant sur deux volets importants qui sont le contrôle des moteurs et le transfert des données.

Avant de passer à la commande du bras série réalisé, pour détection des objets colorés.

Nous trouvons à la fin de ce chapitre des détails sur l'application qui peuvent être réellement réalisé.

IV.1. Chaine d'acquisition :

Le schéma de la (Figure IV.1) montre la chaine d'acquisition des données qui explique comment les informations (le transfert des cinq angles) sont transférées d'un micro-ordinateur ou vers le bras à travers d'un carte Arduino Uno.

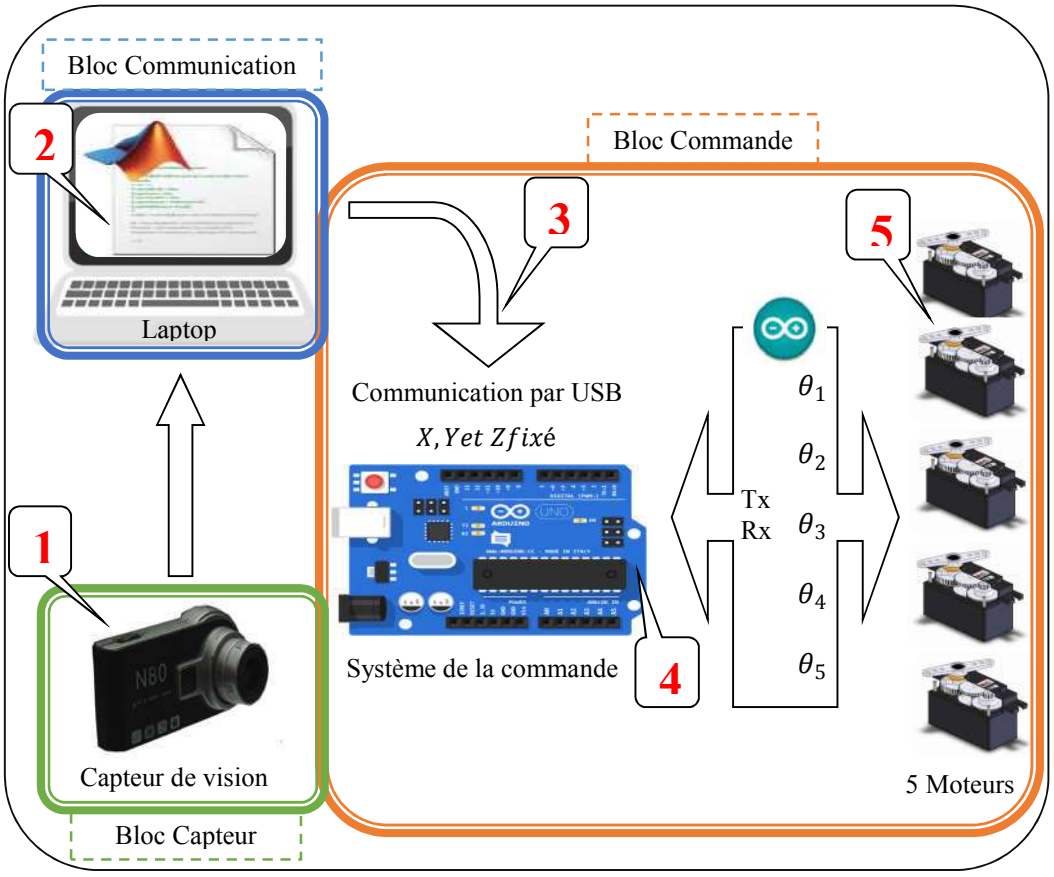


Figure IV.1 : Schéma synoptique de composition du bras.

IV.2. Schéma de bloc du bras :

Notre Bras manipulateur s'articule autour de trois (03) blocs principaux :(Figure IV.2)

- Bloc de commande,
- Bloc de communication,
- Bloc de capteur.

- Les deux premiers blocs représentent l'association du bras (avec ces moteurs électriques à cinq (05) servomoteurs),
- Les deux derniers blocs ils sont regroupés au sein de la carte Arduino et Laptop qui a pour rôle l'acquisition, le traitement des images issus de la webcam et de calculer la loi de commande nécessaire à la poursuite de la détection des objets colorés.

IV.2.1. Capteur de la vision et ses réglages : (Etape 1, Figure IV.3)

La caméra a été choisie ayant un bon rapport signal bruit, Cela limitera l'impact des sources de bruit liées au capteur. (Figure IV.2)

C'est une webcam N80, du 3 Mégapixel, de résolution vidéo «320 x 240» et «640 x 480» pixels, avec une interface de transfert de 2.0 USB.



Figure IV.2 : Caméra.

Dans leur grande majorité, les caméras ont une réponse spectrale qui est fonction de la luminance énergétique. Ainsi deux objets de même couleur pourront apparaître différents en fonction du niveau d'éclairement. L'architecture des systèmes CCD couleur vise à imiter l'organisation de la rétine humaine ; dans une caméra couleur mono-CCD une matrice de filtres rouges, verts et bleus (RVB) est utilisée pour contrôler la fraction de spectre qui atteint chaque détecteur.

Dans notre application, nous avons développé un programme écris en langage MatLab et utilisant les fonctions prédéfinie dans une bibliothèque de traitement d'images. Le but de ce logiciel est l'extraction d'informations visuelles comme le rayon et la position du centre de la

cible. Ces informations permettront par la suite le calcul de la loi de commande pour que le robot puisse suivre la cible.

Un centre de projection unique est une propriété intéressante pour un capteur de vision, appelé point central de projection. Les caméras perspectives conventionnelles sont des capteurs à centre unique.

Un premier travail a consisté à établir un jeu de réglages de la caméra (éclairage, fixation bien de caméra), pour convertir une image en une forme numérisée et y effectuer certaines actions spécialement pour améliorer sa qualité, on se base essentiellement sur la technique de traitement.

L'image sur laquelle nous avons effectué certaines des actions d'amélioration de la qualité est illustrée dans la figure IV.3 pour éviter les bruits, maintenant effectuer certaines des actions majeures pour améliorer sa qualité des objets (sont des pièces bouchon bouteille d'eau).

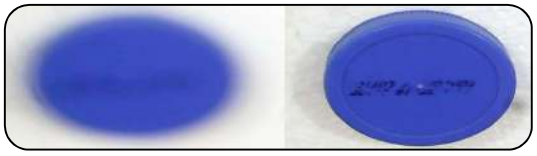


Figure IV.3 : Réglage de la caméra.

Afin de simplifier les notations, nous allons utiliser un module caméra pour la visualisation de notre bras, les caméras perspectives classiques seront incluses dans l'ensemble des caméras, proposent un modèle unifié de projection pour la caméra du point central unique.

D'après ce modèle, toutes les caméras à point centrale unique peuvent être modélisées par une première projection centrale sur un notre plan de travail suivie d'une seconde projection perspective sur le plan image (Figure IV.4).

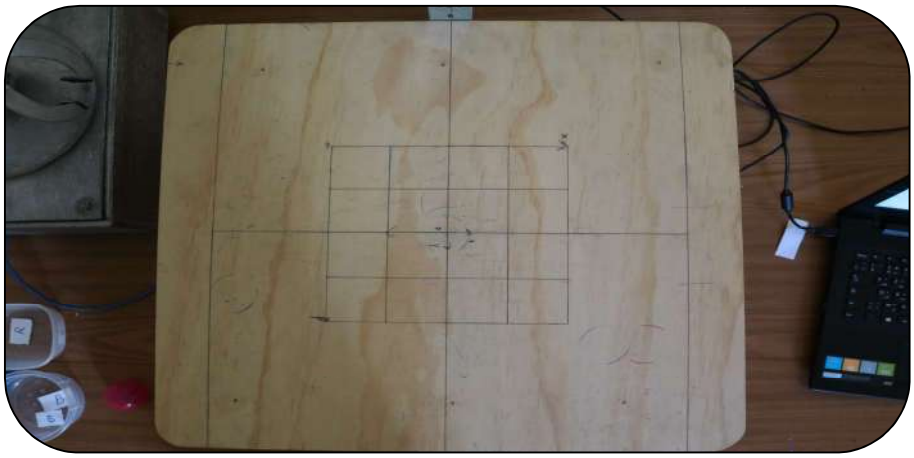


Figure IV.4 : Centralisé le plan de travail (Table).

Dans la suite, (Figure IV.5) on supposera que *Fc*et Fm sont liés par une translation le long de l'axe *Z*. Les centres *C* et *M* de *Fc* et Fm seront appelés centre optique et point principal respectivement. Soit *X* un point 3D de coordonnées **X** = (X Y Z) dans Fm.

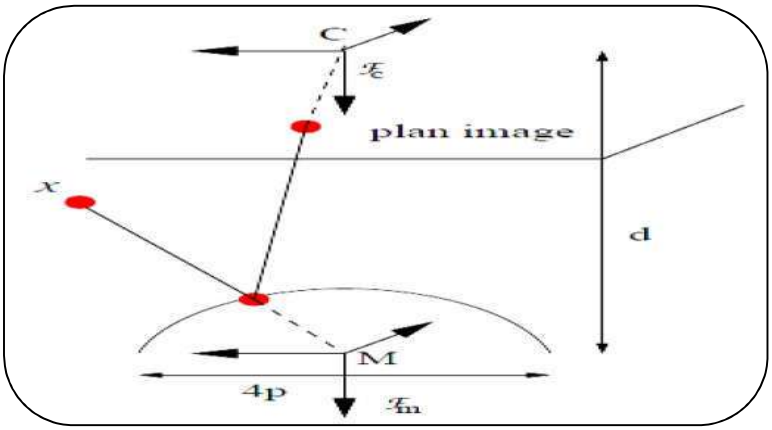


Figure IV.5 : Modèle générique de caméra.

Le plan de travail a été centralisé par la caméra. (Figure IV.6)

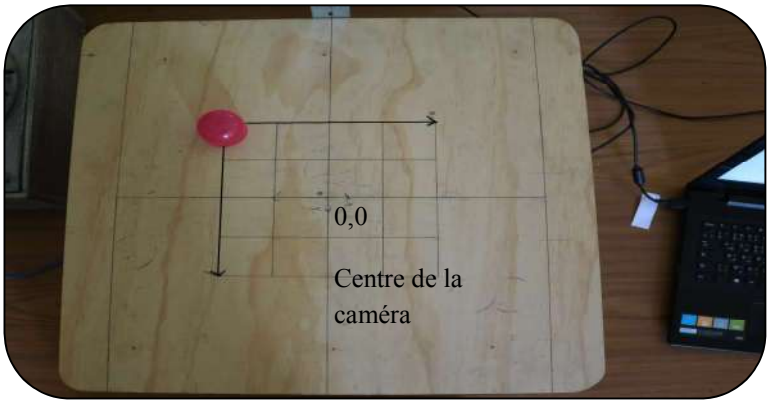


Figure IV.6 : Centralisation de la table.

Cette interface est réalisée à l'aide de MatLab R2014a tout en passant par l'instruction dans bibliothèque. (Figure IV.7).

>>imaqtool % Pour affiché l'image par la Webcam sur MatLab %

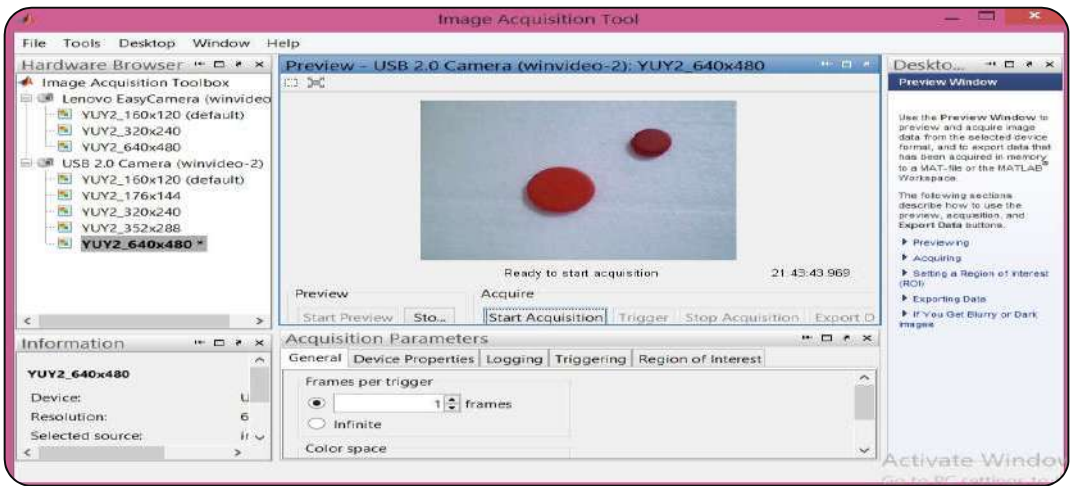


Figure IV.7 : Application d'acquisition d'image sous MatLab.

IV.2.2. Le système de communication : (Etape 2, Figure IV.4)

IV.2.2.a. Rappels sur MatLab :

MatLab est un logiciel de calcul scientifique de développer des solutions à des problèmes technique, il permet de réaliser du calcul numérique et tracer des graphiques pour visualiser et analyser les données, il dispose d'un langage et d'un environnement de programmation interactifs ainsi spécifiques à un domaine d'application comme le traitement d'images par exemple.

La détection des couleurs (couleur des objets) permet de repérer les différents objets qui constituent la scène de l'image qui capter par la caméra. Il existe de nombreuses méthodes pour trouver les contours des objets, la plupart sont basées sur les dérivées premières et secondes de l'image.

La couleur est une donnée importante pour une image, elle modifie la perception que l'on a de l'image. L'espace de représentation standard décompose une image en trois plans de couleur: le rouge, le vert et le bleu - RVB. Les couleurs finales sont obtenues par synthèse additive de ces trois couleurs primaires. Il existe cependant des problèmes qui peuvent nécessité de changer d'espace de couleur pour percevoir différemment l'image.

Une fonction utile à connaître est la fonction de passage d'une image couleur à une image en niveaux de gris. Cette fonction rgb2gray permet d'obtenir une image en niveaux de gris en partant d'une image couleur. Dans certains cas, cette transformation est très utile. Les images seront acquises et analysées à l'aide du logiciel MatLab.

Nous allons voir la détection des couleurs dans les images en utilisant traitement sur MatLab, de façon de détecter les cercles dans les images en utilisant MatLab dans lequel nous avons détecté des objets en fonction de leur figure géométrique,

Grâce à la structure très simple des modules et aux fonctions spécialisées du langage MatLab, de nombreuses opérations d'analyse d'images peuvent être écrites en quelques lignes seulement, soit ils sont en, nous allons distinguer les objets en fonction de leur couleur.

Qu'ils soient de couleur rouge ou de couleur bleu, etc… Nous allons d'abord détecter la couleur, puis créer une frontière autour de cet objet et afficher également ses coordonnées XY. (Figure IV.8 & IV.9)

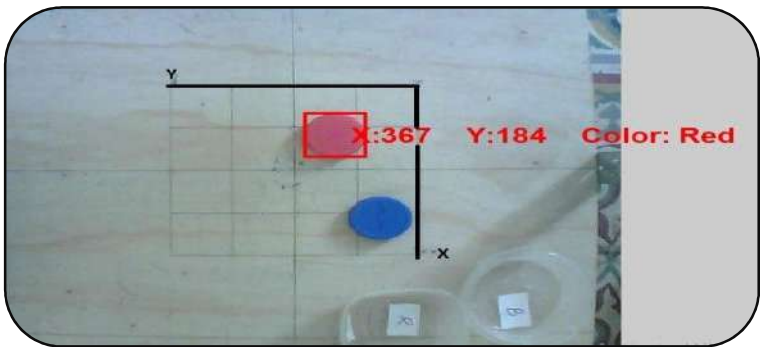


Figure IV.8 : Détection des couleurs rouge.

Nous détectons une couleur rouge dans cette photo (Figure IV.8). Vous pouvez détecter n'importe quelle couleur que vous voulez. Pour ce faire, vous devez modifier très légèrement le code, vous pouvez placer tout ce qui contient de la couleur bleu devant la caméra. L'algorithme détectera et encerclera la couleur bleu dans cette image (Figure IV.9).

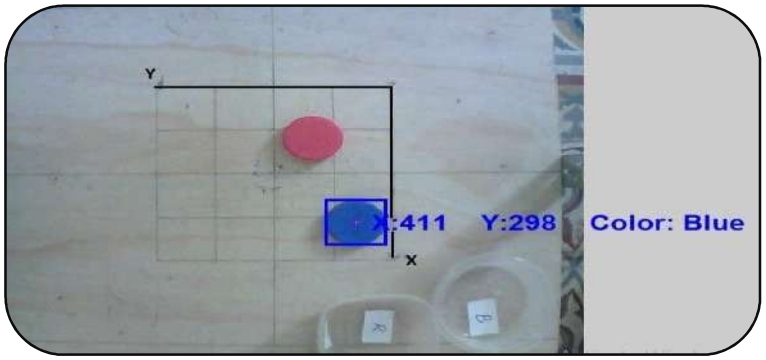


Figure IV.9 : Détection des couleurs Bleu.

Le traitement d'image est un outil important de MatLab. Nous pouvons facilement faire le traitement de l'image à l'aide de la boîte à outils traitement des images, vous devez donc vous assurer que votre MatLab doit avoir une boîte à outils de traitement des images avant d'exécuter ce code. Vous devriez également jeter une caméra de la qualité à ces projets de traitement d'image MatLab.

Comme vous pouvez le voir sur la figure VI.10, il existe différentes formes de couleurs (bleu). Nous allons maintenant détecter ces objets en fonction de leur couleur.

Maintenant, utilisez le code ci-dessous et ajoutez-le dans votre fichier MatLab m.

```
obj=videoinput('winvideo',2,'YUY2_640x480');          % pour ouvrir la caméra%
set(obj,'ReturnedColorSpace','rgb');                  % Code Couleur RVB %
diff_im = imsubtract(data ( : , : , 3), rgb2gray(data));      % Coleur bleu %
```

(:, :, 1) R-values (:, :, 2) G-values (:, :, 3) B-values

Enfin, détectons la couleur bleu, pour ce faire, il vous suffit maintenant de changer le troisième paramètre en 3 et il détectera la couleur bleu comme indiqué dans la figure VI.10.

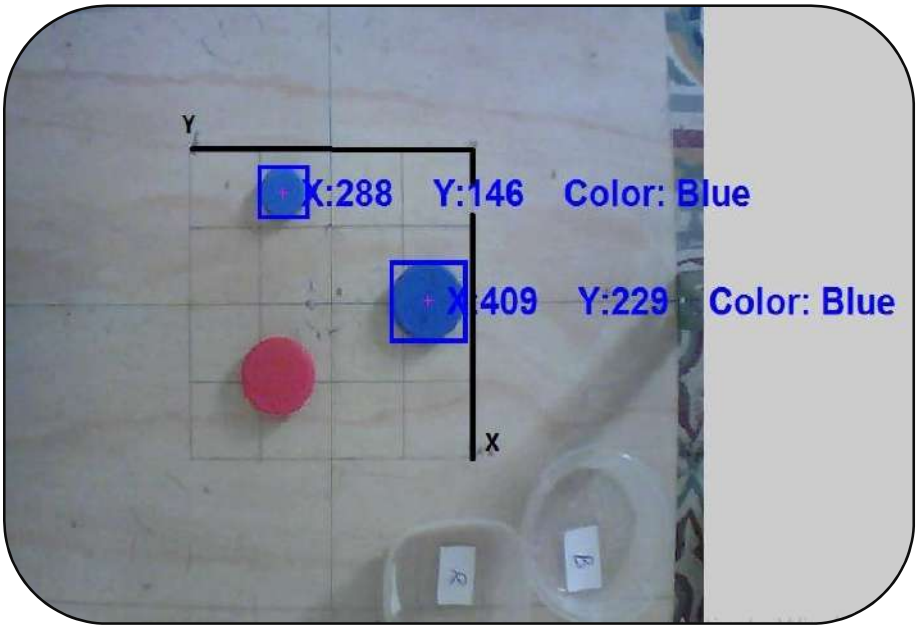
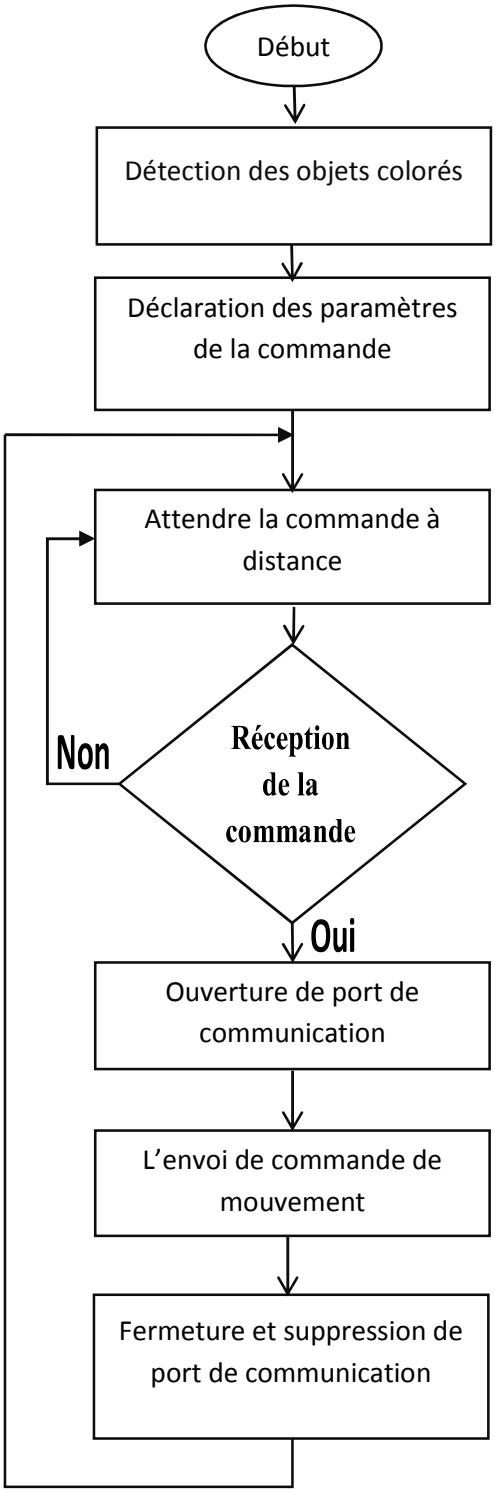


Figure IV.10 : Détection des couleurs bleus.

La figure IV.10 affiche les tâches des détections des objets colorés bleus. C'était donc un peu la technique de traitement de l'image. Maintenant, nous avons combiné tous mes programmes précédents basés sur le traitement d'images MatLab,

Pour simplifier la commande du bras nous avons réalisé un programme par MatLab pour rendre la commande aisée et interactive pour calcul du modèle géométrique inverse (MGI) pour afin d'obtenir les angles de rotation de chaque moteur.

IV.2.2.b. L'organigramme :



IV.2.3. Transfert des données : (Etape 3, Figure IV.5)

Le transfert de données est réalisé à l'aide de la liaison série soit par un adaptateur USB-RS232 pour une communication transparente entre unité de traitement et la carte Arduino. Le transfert des données doit être en code ASCII.

Le circuit de commande va convertir le code ASCII vers Décimal et génère le signal de commande qui correspond à l'information obtenu par le MatLab.

La liaison série à la norme RS 232 est utilisée dans tous les domaines de l'informatique. Elle permet la réalisation d'une liaison simple à mettre en œuvre entre 2 équipements. Elle est de type asynchrone, c'est à dire qu'elle ne transmet pas de signal horloge, les deux équipements doivent être configurés avec une même vitesse de transmission identique. Ils doivent par ailleurs utiliser le même protocole. **[14]**

Afin d'obtenir les angles pour le mouvement désiré, le programme transfère les données obtenue vers le système de commande en utilisant un Protocol d'envoi, il est basé sur les instructions suivantes :

- Serial : pour choisir le port de communication,
- Set ('baudrate') : pour choisir le baudrate de transfère (la vitesse de transfert),
- Fopen : pour ouvrir le port de communication,
- Fprintf : pour envoyer la donnée,
- Fclose : pour fermer le port de communication,
- Delete : pour supprimer le port de communication.

IV.2.3.a. La liaison RS232 :

La carte Arduino Uno est configuré en mode USART, permet une communication série de type asynchrone sous la norme RS232, avec un ordinateur via ses pattes Rx (réception) et Tx (transmission) branchées sur un port série.

IV.2.3.b. Paramètres de la liaison RS232 :

L′utilisation du port série, impose la détermination des paramètres de communications sur le port. Ces paramètres sont : **[12]**

Débit des données : La vitesse de transmission caractérise le nombre de bits transmis par secondes. Elle s'exprime en bauds. Les valeurs courantes sont entre 300 et 9600 bauds.

IV.2.3.c. Format des trames : (Figure IV.11)

- Au repos, la ligne est à 1. Un passage à 0 indique que la transmission va commencer, c'est le bit de début,
- Les bits de données sont ensuite transmis sur un format de 7 ou 8 bits,
- Un bit de parité peut être ajouté pour vérifier la validité des données,
- Un ou deux bits de stop terminent la transmission pour le retour au repos de la ligne.

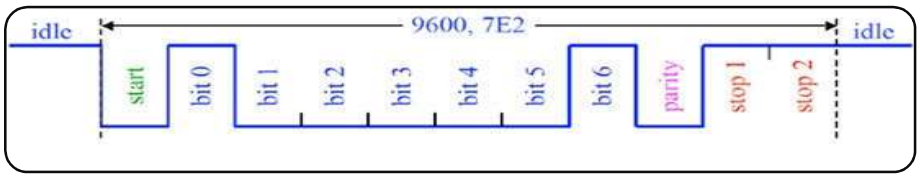


Figure IV.11 : Trame de communication série.

Le bit de parité est tel que la somme des bits à l'état 1 est paire, dans le cas contraire, il y a eu une erreur de transmission. Il est aussi possible de travailler avec un nombre impair de bits.

IV.2.4. Commande par Arduino Uno: (Etape 4, Figure IV.6)

C'est une carte électronique en matériel libre pour la création artistique interactive.

La famille Arduino est apparue en 2006 sous forme d'une carte programmable avec un port série. Puis, de nouvelles versions sont apparues, toujours aussi simples, avec différents formats : petites, grandes, plus puissantes, plus économiques, avec USB, avec Ethernet, Bluetooth, et compatible Androïde, etc... Plus précisément, pour programmer des systèmes électroniques. Le gros avantage de l'électronique programmée c'est qu'elle simplifie grandement les schémas électroniques et par conséquent, le cout de la réalisation, mais aussi la charge de travail à la conception d'une carte électronique.

La carte ARDUINO permet de :

- Contrôler les appareils domestiques,
- Fabriquer votre propre robot,
- Faire un jeu de lumières,
- Communiquer avec l'ordinateur,
- Télécommander un appareil mobile (modélisme),
- Servir de cerveau à un humanoïde,
- Apprendre l'électronique, et encore bien d'autres usages…etc.

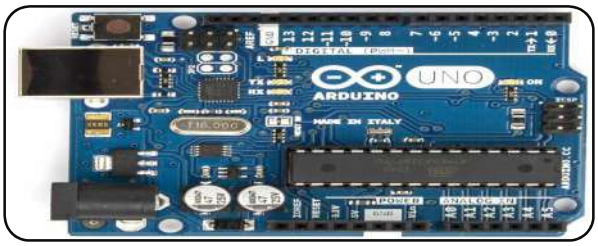


Figure IV.12 : Catre Arduino Uno.

IV.2.4.a. Présentation de la carte ARDUINO :

Le système ARDUINO est composé de deux choses principales : le matériel (la carte) et le logiciel.

- **Le matériel :** Description de la carte Arduino UNO en Annexe B.
  Il s'agit d'une carte électronique basée d'un microcontrôleur ATmega fabricant Atmel,

- **Le logiciel :**

Le logiciel permet de programmer la carte ARDUINO. Il offre une multitude de fonctionnalités. **[14]**

La carte ARDUINO est équipe d'un microcontrôleur. Le microcontrôleur est un composant électronique programmable. On le programme par le biais d'un ordinateur grâce à un langage informatique, souvent propre au type de microcontrôleur utilisé. Un microcontrôleur est constitué par un ensemble d'éléments qui ont chacun une fonction bien déterminée. Il est en fait constitue des mêmes éléments que sur la carte mère d'un ordinateur.

IV.2.4.b. Langage ARDUINO :

Le projet ARDUINO était destinera l'origine principalement a la programmation (Open Source) multimédia interactive en vue de spectacle ou d'animations artistiques. C'est une partie de l'explication de la descendance de son interface de programmation de Processing.

Processing est une librairie java et un environnement de développement libre. Le logiciel fonctionne sur Macintosh, Windows, Linux, BSD et ARDUINO.

Référence :

- Le langage C,
- Le langage Java et MatLab.

Cependant, le projet ARDUINO à développer des fonctions spécifiques à l'utilisation de la carte qui a été listées sur (Table IV-2). Vous obtiendrez la description de chacune d'elles dans le manuel de référence.

Commande	Description
setup ()	Fonction qui ne s'exécute qu'une fois au premier démarrage de l'ARDUINO UNO.
loop ()	Fonction qui s'exécute de façon répétitive âpre que la fonction **setup ()** ait termine et jusqu'à ce que ARDUINO soit éteinte.
pinMode ()	Définit si la broche dont le numéro est fourni en argument est utilisée en sortie d'électricité ou en entrée.
OUTPUT	Mot-clé, second argument de la fonction **pinMode()** qui indique que la broche va délivrer une tension électrique.
digitalWrite ()	Provoque la sortie d'une tension électrique ou sa disparition sur la broche indiquée.
HIGH	Mot-clé utilise pour délivrer une tension électrique dans la fonction **digitalWrite().**
LOW	Mot-clé utilise pour interrompre la sortie d'une tension dans **digitalWrite().**
delay ()	Provoque une pause de l'ARDUINOUNO pendant le nombre de millisecondes spécifie en argument.

Table IV-1 : Différentes commandes et leurs descriptions.

On teste le matériel en chargeant le programme. Avant d'envoyer notre programme vers la carte, il faut dire au logiciel quel est le nom de la carte et sur quel port elle est branchée. Pour cela, allez dans le menu "Tools" (outils) puis dans "Board" (carte). Vérifiez que c'est bien le nom "ARDUINOUNO" qui est coché. Si ce n'est pas le cas, cochez-le.

Allez ensuite dans le menu Tools, puis Serial port (port série). Choisissez le port COMX, X étant le numéro du port qui est affiche. Ne choisissez pas COM1 car il n'est quasiment jamais connecte à la carte, il va falloir envoyer le programme dans la carte. Pour ce faire, il suffit de cliquer sur le bouton Upload (Télécharger).

La carte Arduino reçoit le programme, puis les données X, Y et Z qui fait convertir en θ_n on utilisant MGI qui expriment la situation de l'organe terminal en fonction de la configuration du mécanisme et inversement.

Définir les différentes tâches d'un robot réclame de pouvoir positionner l'organe terminal par rapport à un repère de l'attitude de plan de travail.

La position à atteindre est souvent définie dans un repère lié au socle du bras, l'objet colorés à saisir peut être défini dans un repère de cordonné X, Y et Z indépendant du robot.

Aussi, il faut un référentiel commun afin de "ramener" les diverses informations dans un même référentiel, notamment pour concevoir les consignes des actionneurs.

La carte ARDUINO dispose de six(6) broches qui s'oignent compatibles avec la génération d'une PWM. Elles sont repérées par le symbole tilde (~) .Voici les pins générant une PWM : 3, 5, 6, 9, 10 et 11.

La fonction analogWrite() est une fonction pour utiliser la PWM qui prend deux arguments :

- Le premier est le numéro de la broche ou l'on veut générer la PWM,
- Le second argument représente la valeur du rapport cyclique à appliquer. Cette valeur ne s'exprime pas en pourcentage, mais avec un nombre entier compris entre 0 et 255.

IV.2.5. Partie actionneur : (Etape 5, Figure IV.7)

IV.2.5.a. Définition :

Un actionneur est un dispositif qui transforme l'énergie délivrée par l'interface de puissance, en énergie utilisable par les effecteurs de processus. Les moteurs électriques, les vérins pneumatiques ainsi que les éléments chauffants sont des exemples typiques d'actionneurs utilisés en automatisation industrielle.

IV.2.5.b. Familles d'actionneurs :

Le mouvement est imposé au robot par un ou plusieurs actionneurs par transformation d'une énergie source en énergie de base utilisée peut être.

- Electrique (Asynchrone, Moteur pas-à-pas, Moteur à courant continue),
- Pneumatique (Vérin pneumatique, Ventouses),
- Hydraulique (Vérin eau ou huile, Pompes),
- Mécanique (dispositif à câbles et poulies),
- Combustible fossile.

IV.2.5.c. Les Servomoteurs : (Figure IV.13)

Les servomoteurs sont des actionneurs, très utilisés en modélisme et dans l'industrie. Ils ont comme caractéristique principale leur « couple », C'est-à-dire que l'axe de sortie du servomoteur est capable de s'arrêter sur une position prédéterminée puis de rester sur cette position. Pour la robotique de loisirs, les servomoteurs ont en général peu de couple et sont de taille réduite, bien adaptée à un encombrement minimal et à une énergie disponible limitée.

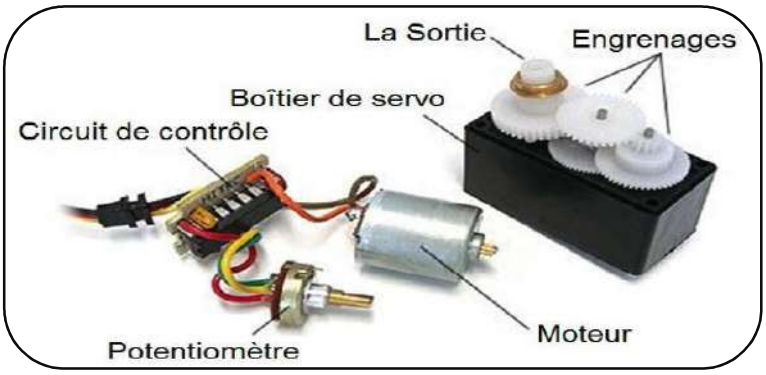


Figure IV.13 : Le constituant d'un servomoteur.

Pour faire fonctionner le servomoteur, une information de positionnement instantané de l'arbre de sortie est renvoyée au circuit de commande au moyen d'un transducteur. Une façon simple de procéder consiste à attacher un potentiomètre à l'arbre de sortie ou quelque part dans le train d'engrenages.

L'électronique de commande compare le signal de contre-réaction (qui contient la position courante de l'arbre) à partir du potentiomètre au signal d'entrée de commande (qui contient les informations de la position désirée de l'arbre), et toute différence entre les valeurs réelles et désirées (connu sous le nom de signal d'erreur) est amplifié et utilisé pour entraîner le moteur à courant continu dans une direction nécessaire pour réduire ou éliminer l'erreur. L'erreur est nulle lorsque l'arbre de sortie arrive à la position désirée.

Le schéma de principe de fonctionnement d'un servomoteur typique est montré dans la Figure IV.14.

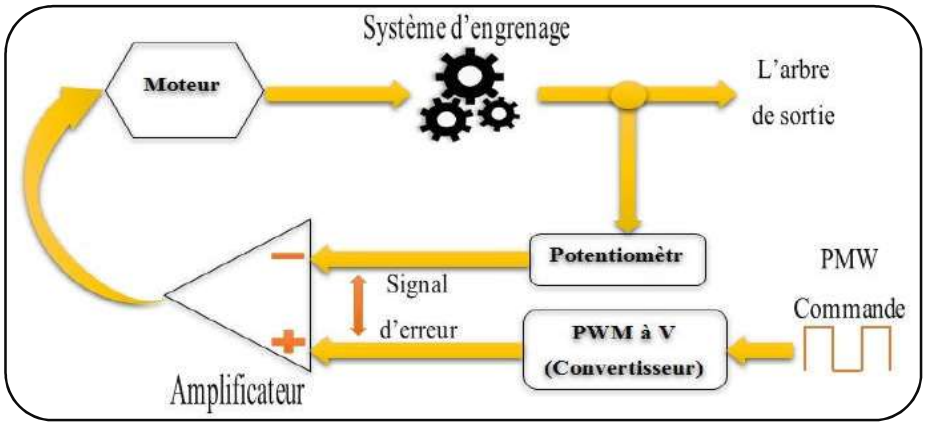


Figure IV.14 : Principe de fonctionnement de servomoteur.

IV.2.5.d. Brochage du servomoteur :

Un servomoteur a trois fils : deux sont désignés pour l'alimentation (Vcc et la masse) et le troisième fil est pour le signal de commande. Le fil Vcc est habituellement rouge et la masse est noire ou brun. Le fil de signal de commande est de couleur blanche, jaune ou orange. Le servomoteur utilisé dans ce travail est de technologies de high-tech et a des couleurs rouge, noir et jaune destiné respectivement pour Vcc, GND, et le signal de commande. Il fonctionne sur une alimentation de 7,4V et fournit une rotation angulaire de 180°.

IV.2.5.e. Les Commandes des servomoteurs :

L'entrée de commande d'un servomoteur est un signal à modulation de largeur d'impulsion (PWM), généralement de fréquence 50 Hz. Cela signifie que l'impulsion doit être répétée toutes les 20ms. La largeur de l'impulsion détermine la position angulaire de l'arbre de sortie. Un circuit électronique à l'intérieur du servomoteur convertit le signal PWM à une tension de sortie proportionnelle qui est comparée à la tension de rétroaction à partir du potentiomètre. Si une différence existe entre les deux, le circuit de commande entraîne le moteur dans une direction appropriée jusqu'à ce que la différence devienne nulle **[3]**.

Une valeur typique de la largeur d'impulsion se situe dans la plage de 1,0 à 2,0 millisecondes (ms). Pour un servo standard, une largeur d'impulsion entre 1,0 ms et 1,5 ms réalise un asservissement à tourner dans le sens horaire, entre 1,5 ms et 2,0 ms permettent de tourner dans le sens antihoraire, et une largeur d'impulsion 1,5 ms transforme le servomoteur à son centre (Figure IV.15). Cependant, ces valeurs peuvent varier en fonction de la marque du moteur.

La plupart des servomoteurs tournent de 0 à 180°. Cependant. Il y a certains qui pourraient tourner à travers un 360 ° ou plus. Les Servomoteurs sont largement utilisés comme les joints mobiles dans les bras robotiques pour leur positionnement angulaire précis.

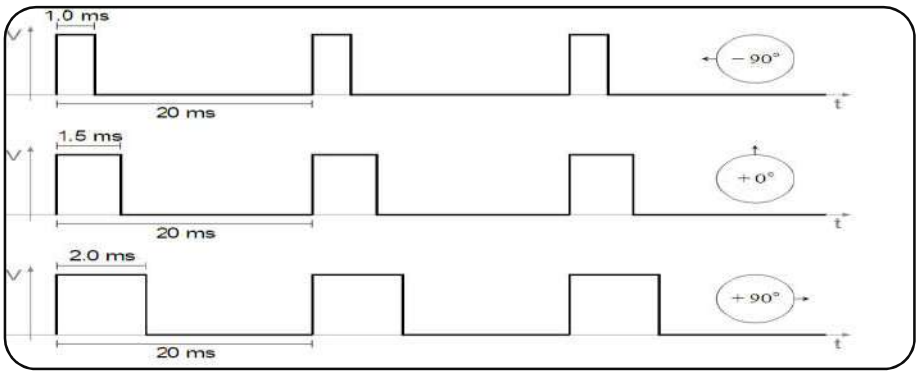


Figure IV.15 Chronogramme du commande de Servomoteur.

VI.3. Dispositif expérimental :

Pour développer notre bras de manipulation par vision :

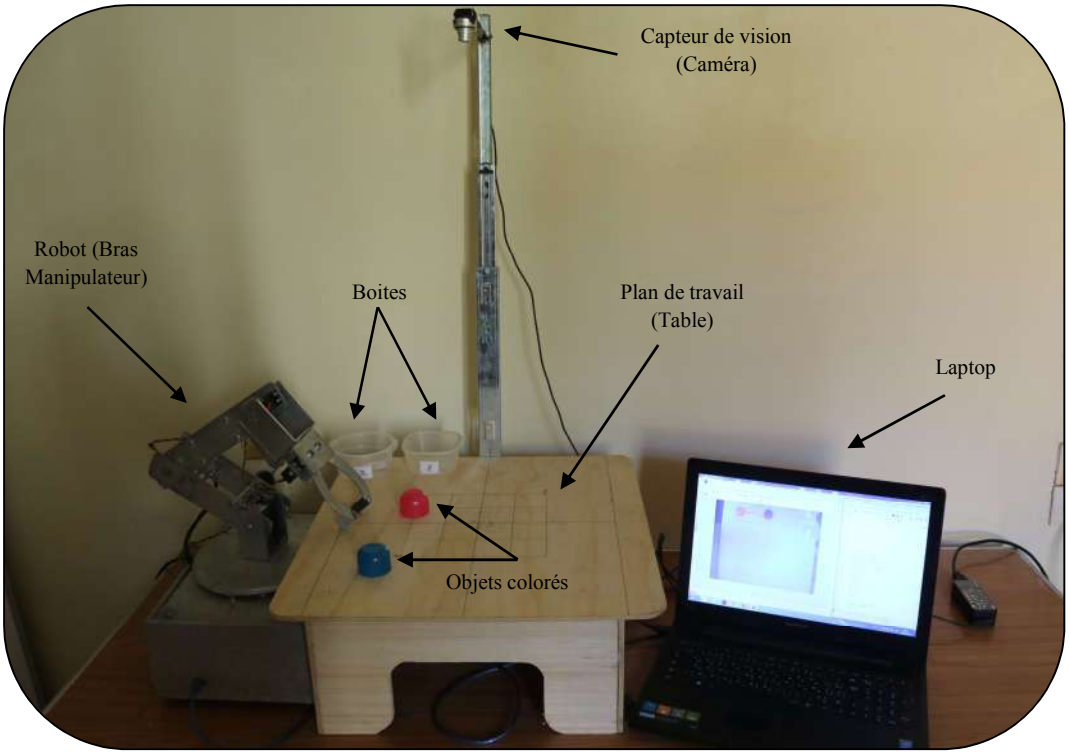


Figure IV.16 : Photo de la réalisation du projet.

La figure IV.16 illustre la composition du projet complet. Nous utilisons un ordinateur de type Laptop, Il permet d'effectuer le traitement d'image nécessaire à l'extraction des informations visuelles et de calculer la vitesse de commande articulaire à appliquer au manipulateur.

La caméra est déporté fournit un signal vidéo pour détecter des objets coloré sur le plan de travail qui est numérisé les angles de chaque actionneur est calculé par MatLab. Cette synchronisation permet d'extraire de manière robuste les informations visuelles nécessaires au système de commande par vision. Etant donné que les informations visuelles sont obtenues à partir d'une image complète (coordonnées XY des objets), le système de commande par vision est de temps du 3 seconde.

Le transfert de données entre Laptop et la carte Arduino Uno du bras est réalisé à l'aide de la liaison série soit par un adaptateur USB- RS232, pour une communication transparente de connexion série câblée, le transfert des données doit être en code ASCII, il transfert des cinq angles à la carte du bras qui lui renvoie les valeurs courantes des coordonnées articulaires du bras.

Le circuit de commande va convertir le code ASCII vers Décimal et génère le signal de commande qui correspond à l'information obtenu par le MatLab.

Finalement les actionneurs transforme l'information électrique vers une rotation mécanique de l'arbre du moteur l'ensemble de la structure mécanique du bras se met en mouvement.

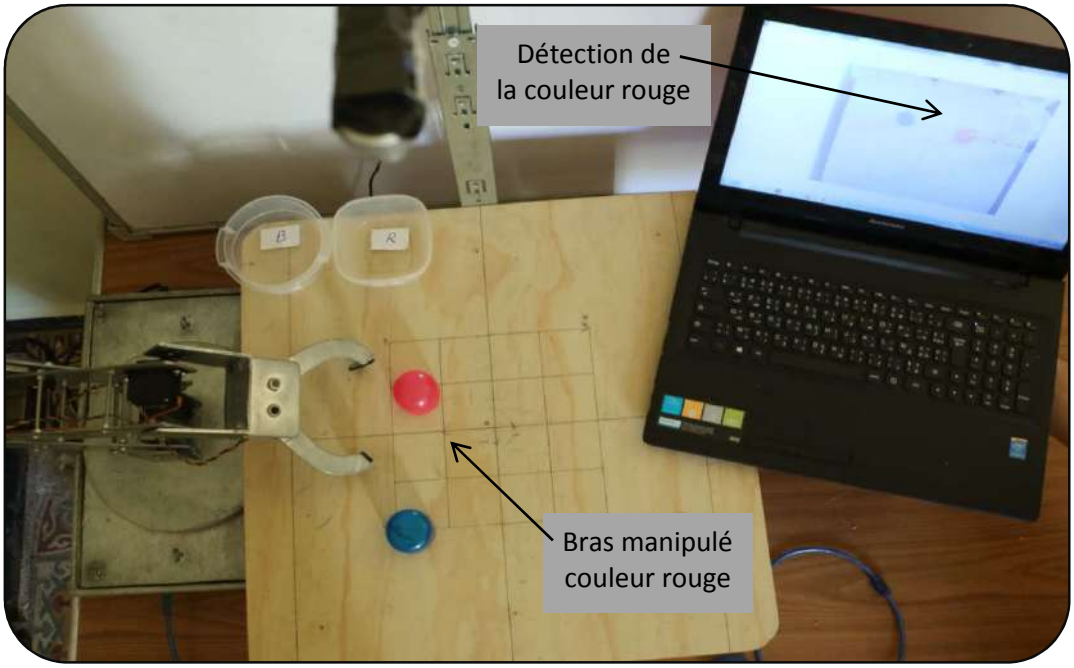


Figure IV.17 : Test de la détection des objets rouge.

VI.4. Conclusion :

Ce chapitre représenté par un schéma global qui résume la chaine d'acquisition des données par les servomoteurs à travers la carte Arduino, et la caméra déportée traiter le traitement d'image du notre bras manipulateur.

On a finalement passé de la réalisation a la réalité ce qui nous a permet de détailler les différentes manipulations réalisées par notre bras manipulateur sur l'espace de travail.

Conclusion Générale

Conclusion Générale

Ce mémoire est le résultat d'un travail d'amélioration d'un bras manipulateurs série en l'industrie assistée par robot. Dans ce travail, ont été réalisés l'étude, le développement et la validation expérimentale d'un système de vision permettant de déplacer, détecter, automatiquement par asservissement visuel, un bras en vue d'assister humain dans l'industrie un système robotique.

Deux tâches automatiques sont proposées : la première consiste à récupérer le bras manipulateur pour l'amener au plan du travail s'il n'est pas présent dans le champ visuel de la caméra, la deuxième tâche permet au humain de placer automatiquement le bras manipulateur à une position choisie dans l'espace de travail.

Dans le premier chapitre et après une note introductive sur le thème des robots industriels et ses notions de base, nous avons exposé les différents constituants technologiques d'un robot manipulateur ainsi que les différents types d'articulations, ensuite nous avons présenté les défirent types de robots les plus couramment utilisés dans l'industrie.

Dans le deuxième chapitre, la modélisation géométrique, tout en soulignant son grand intérêt dans la robotique.

Ce qui concerne le troisième chapitre, nous décrivons les travaux existants sur la technique du traitement des images, à l'aide de caméra dans lesquels la commande par retour visuel est utilisée. Des méthodes robustes de traitement d'image permettant d'extraire les informations visuelles malgré de nombreuses difficultés liées à la complexité de plan de travail.

Pour terminer ce mémoire, nous présentons dans le quatrième chapitre, différentes commandes par vision permettant d'orienter le bras manipulateur sur une cible sont d´entaillées. Les résultats du positionnement complet de bras manipulateur obtenus lors d'essais sur le plan de travail et également d'expérimentations.

En effet, la mise en place du protocole de communication entre le LapTop et le robot, la commande cinématique du robot, la programmation des algorithmes de commande et du traitement d'image.

Cette mise en place du système expérimental m'a permis d'augmenter mes compétences dans différents domaines, complémentaires les uns aux autres, qui sont la robotique, le traitement d'image, la commande par vision, l'automatique, l'informatique.

Enfin, j'ai pu remarquer grâce aux expérimentations qu'un fossé important existe entre l'étude théorique, qui est validée par diverses simulations, et la réalité de concrète et pratique, cette difficulté est accentuée également par les mauvaises conditions de positionnement du bras, plan du travail, réglages de la caméra et l'éclairage.

Perspectives :

Dans le prolongement direct de ces travaux de ce projet, plusieurs études sont à envisager ;

Pour ce faire, il serait nécessaire de mesurer la position relative entre la caméra et le plan de travail.

Cependant, une étude de sensibilité nous a montré que la précision de cette mesure est dépendante de la configuration géométrique entre la caméra, bras, l'organe, et les objets, ce qui peut réduire la précision du positionnement de pour détection des objets.

La conception d'une tète bras motorisée capable de supporter un système de vision plus performant (caméra embarqué).

Remplacer le LapTop par une carte Raspberry (Cerveau du bras manipulateur) plus performante encore (Pour un projet autonome).

Grâce aux recherches en cours, il est clair que le bras manipulateur tient une place de plus en plus importante dans la recherche actuellement menée en robotique industriels. Par les progrès qu'elle apporte ou promet d'apporter, et la passion de tous, des scientifiques et du grand public. Le défi est énorme et passionnant !

Références

Référence :

[1] F. GUERRACHE, «Modelisation d'un Robot a Structure Parallele», Option Construction Mécanique, Université de Boumerdes, 2005.

[2] A. ALLOUI, A et HAJ Brahim. «Proposition d'une solution multi-agent pour la commande et la coopération multi -robot mobile». Mémoire d'ingénieur d'état en automatique, Université Biskra, Juin 2007.

[3] R. EULDJ et Y. SELAMI, « Étude, Conception et Réalisation d'un Robot Parallèle type DELTA», Mémoire de Master, Filière : Automatique des Systèmes, Université Dr. Yahia farés, Médéa, 2015.

[4] K. Ait Dahmane et M. Ait Ziane «Conception et Réalisation d'un Bras Manipulateur Commandé par API», Université de Khemis Miliana, Option Automatique des Systèmes et Production, 2014.

[5] W. Khalil et E. Dombre, «Bases de la Modélisation et de la Commande des Robots-Manipulateurs de type série», 2012.

[6] J. Denavit et R.S. Hartenberg, «A Kinematic Notation For Lower Pair Mechanism Based On Matrices». Trans, of ASM E, J. of Applied Mechanics, Vol. 22, Juin 1955

[7] M. BOUHARAOUA, M. SAIFI et Med El Hachemi, « Etude, Conception et Réalisation d'un Robot Serie 5 Axes ». Mémoire de Master, Filière : Automatique des Systèmes, Université Dr. Yahia Farés, Médéa, 2015.

[8] S. BENFRIHA et A. HAMEL « Segmentation d'image par Coopération région-contours », Université Kasdi Merbah-Ouargla, Domaine: Informatique et Technologie de l'Information, 2016.

[9] C.Houassine, « Segmentation d'images par une approche biomimétique hybride ». Université M'Hamed bougara- boumerdes. 2012.

[10] GUANGLONG D. et PING Z., Online robot calibration based on vision measurement, Robotics and computer-Integrated Manufacturing, Vol. 29.

[11] P. RENAUD, « Apport de la Vision Pour L'identification Géométrique de Mécanismes Paralleles », Thèse de Doctorat, Université Blaise Pascal, 2003.

[12] R. Fleurmond « Asservissement Visuel Coordonne de Deux Bras Manipulateurs », Thèse de Doctorat, EDSYS : Automatique, Signal, Productique, Robotique 4200046, Université Toulouse, 2015.

[13] Oldrik CEUGNART, Rodolphe Wronski « Rapport Robotique et Vison » Licence Professionnelle Gestion de Production Industrielle, Spécialité : Vision Industrielle, Université Lille 1, 2012.

[14] Ying Bai, livre « The Windows Serial Port Programming Handbook », Première Edition CRC Press LLC, USA 2005.

Annexes

ANNEXE A

Acquisition d'images sous Matlab :

Un objet d'entrée vidéo est un objet de type structure sur laquelle il est possible de régler plusieurs propriétés :

- les paramètres liés au périphérique comme :
  - le format de l'image,
  - l'espace de codage de l'image (RGB, YCbCr, niveaux de gris, ...)...
- Les paramètres liés à l'acquisition comme :
  - la luminosité (Brightness),
  - le contraste (Contrast),
  - le temps d'exposition (Exposure),
  - la correction gamma (Gamma),
  - la netteté (Sharpness)...

La fonction propinfo (ou les fonctions get et set) ainsi que la fonction inspect permettent d'accéder et de connaître les caractéristiques détaillées de chaque propriété. Une propriété peut également être un objet de type structure avec ses propres propriétés. L'accès à cet objet s'adresse de la façon suivante : nom: propriete. Attention, certaines propriétés ne sont accessibles qu'en lecture seule selon qu'une acquisition est en cours ou non.

La fonction preview permet de créer une fenêtre d'aperçu afin de visualiser la scène observée et la fonction closepreview permet de fermer cette fenêtre. C'est ainsi qu'il est possible de régler certain paramètre d'acquisition.

Réglages des paramètres intrinsèques et extrinsèques :

Le programme suivant utilise les fonctions précédentes afin de configurer l'acquisition d'une image et effectuer les réglages nécessaires avant d'acquérir l'image :

```
clearall;       % efface toutes les variables en mémoire
closeall;       % ferme toutes les fenêtres ou vertes
clc ;           % efface le contenu de la fenêtre de commande

%% Affichage des informations logicielles et matérielles
    info_id = imaqhwinfo ( ' winvideo ' )
    info_dev = imaqhwinfo ( ' winvideo ' , 1 )
    formats = info_dev. Supported Formats % affichele formats d'images possible

%% Création d'un objet d'entrée vidéo
    vid =video input ( ' winvideo ' , 1 , 'RGB8_1280x1024 ' ) ; % associe un objet d'entrée vidéo
    % au périphérique matériel de la caméra en transmettant le numéro
    % d'identification de ce périphérique avec un format d'image 1280 x 1024.

%  Informations sur les propriétés de " Source "
    Information = get ( vid )

% Réglage du périphérique
    propinfo ( vid , ' Video Format ' ) % affiche les informations sur le format vidéo
    propinfo ( vid , ' Returned Color Space ' ) % affichele informations sur l'espace couleur
    set ( vid , ' Returned Color Space ' ) % permet de visualiser les réglages possibles
    % ou : get ( v id , ' Returned Color Spac e ' )
    set ( vid , ' Returned Color Space ' , ' grayscale ' ) % sélectionnel espace de représentation
    % ( ici monochrome )
    % get ( vid , ' Returned Color Space ' ) pour affichage

% Affichage des informations sur les propriétés
    propinfo ( vid , ' Source ' )

%% Accès à la source vidéo
    Source = getselected source ( v i d ) ; % <=> Source = vid . Source ;
    % Informations sur les propriétés de la source vidéo
    Information = get ( Source ) % <=>  set ( Source )
    % Affichage des valeurs minimales et maximales de réglage des principales propriétés
    info = propinfo ( Source , 'Gain ' ) ; disp ( 'Gain ' ) ; disp ( info . Constraint Value ) ;
    info = propinfo ( Source , ' Contrast ' ) ; disp ( ' Contrast ' ) ; disp ( info . Constraint Value ) ;
    info = propinfo ( Source , ' Exposure ' ) ; disp ( ' Exposure ' ) ; disp ( info . Constraint Value ) ;
    info = propinfo ( Source , 'Gamma ' ) ; disp ( 'Gamma ' ) ; disp ( info . Constraint Value ) ;
    info = propinfo ( Source , ' Sharpness ' ) ; disp ( ' Sharpness ' ) ; disp ( info . Constraint Value ) ;


% Affichage et réglage de spropriétés de la source vidéo
    inspect ( Source ) ; % ouvre une fenêtre avec les différents paramètres à régler
    % et permet le réglage de ces paramètres de manière interactive

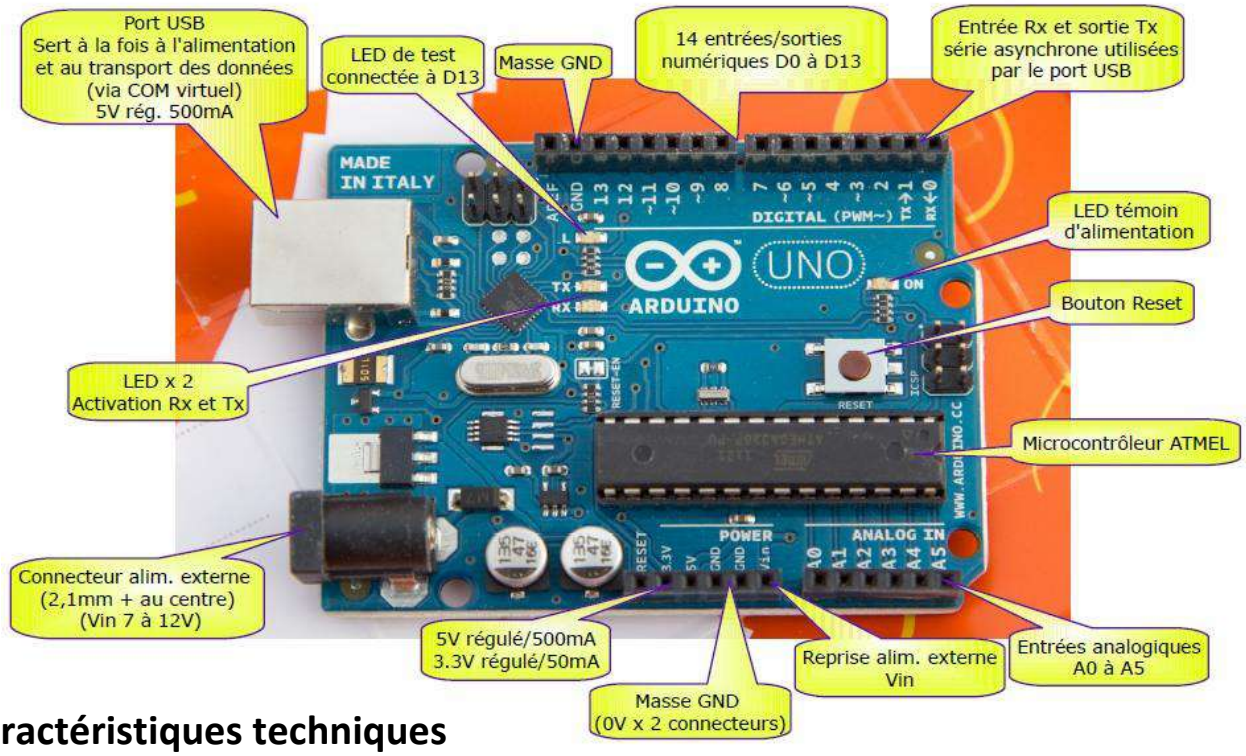
%% Réglage du système
    preview ( vid ) ; % ouvre la fenêtre de prévisualisation de l ' image


    pause ; % attend que l'utilisateur appui sur une touche de puis l'éditeur de commande
    % avant de continue afin d'effecuer les réglages
            close preview ; % ferme la fenêtre de prévisualisation
```

TECHNOLOGIE au Collège

Initiation aux sciences de l'ingénieur
Programmation – Câblage - Simulation

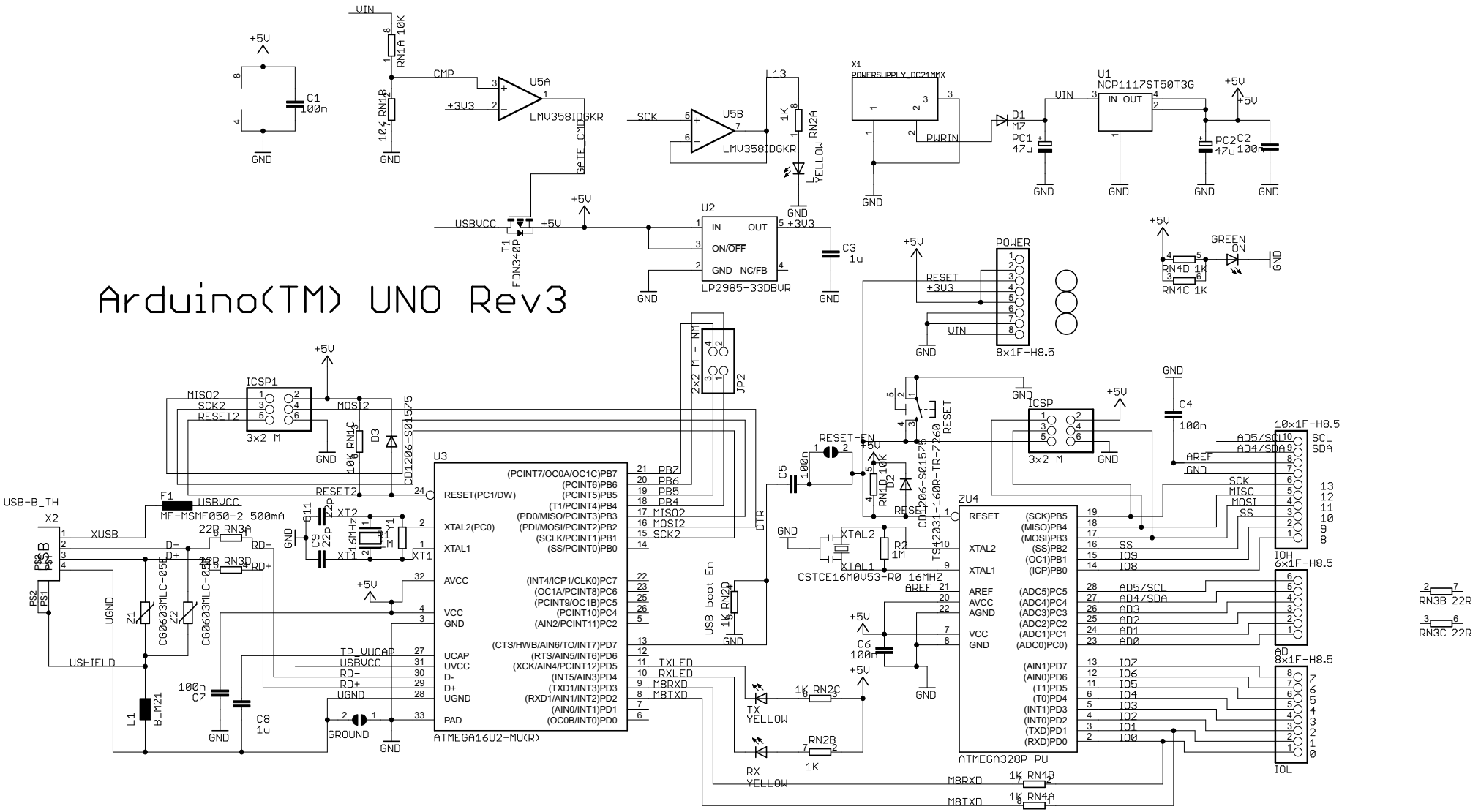
Description de la Carte Arduino UNO



Caractéristiques techniques

Microcontrôleur	ATmega328
Fréquences Horloge	16 Mhz
Tension de fonctionnement	5V
Tension d'alimentation (recommandée)	7-12V
Tension d'alimentation (limites)	6-20V
Broches E/S numériques	14 (dont 6 disposent d'une sortie PWM représenter par le symbole tilde)
Broches d'entrées analogiques	6 (utilisables en broches E/S numériques)
Intensité maxi disponible par broche E/S (5V)	40 mA (Attention 200mA max total entrée/sortie cumulées)
Intensité maxi disponible pour la sortie 3.3V	50 mA
Intensité maxi disponible pour la sortie 5V	500 mA max Sur port USB
Mémoire Programme Flash	32 KB (ATmega328) dont **0.5 KB** sont utilisés par le bootloader (chargeur d'amorçage)
Mémoire SRAM (mémoire volatile)	2 KB (ATmega328)
Mémoire EEPROM (mémoire non volatile)	1 KB (ATmega328)
Interface	USB (permet d'alimenté la carte et de transféré les programmes vers la carte)
Alimentation externe	Jack
Dimensions	6,86 cm x 5,3 cm

Un module Arduino est généralement construit autour d'un microcontrôleur Atmel (ici AT mega 328) et de composants complémentaires qui facilitent la programmation et l'interfaçage avec d'autres circuits. Le microcontrôleur peut être programmé de manière à effectuer des tâches très diverses comme la domotique (le contrôle des appareils domestiques - éclairage, chauffage…), le pilotage d'un robot, de l'informatique embarquée, etc… Le logiciel de programmation des modules Arduino est une application Java, libre et servant d'éditeur de code et de compilateur, et qui peut transférer le programme au travers de la liaison série (RS-232, Bluetooth ou USB selon le module). Le langage de programmation utilisé est le C++, compilé avec avr-g++ , et lié à la bibliothèque de développement Arduino, permettant l'utilisation de la carte et de ses entrées/sorties.



Reference Designs ARE PROVIDED "AS IS" AND "WITH ALL FAULTS. Arduino DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, REGARDING PRODUCTS, INCLUDING BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE Arduino may make changes to specifications and product descriptions at any time, without notice. The Customer must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Arduino reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The product information on the Web Site or Materials is subject to change without notice. Do not finalize a design with this information.

ARDUINO is a registered trademark.

Use of the ARDUINO name must be compliant with http://www.arduino.cc/en/Main/Policy