

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE

Ecole Nationale Polytechnique

Département d'Automatique



Projet de Master

Mémoire en vue de l'obtention du diplôme de Master en Automatique

Thème :

**Traitement d'image appliqué à l'asservissement
visuel**

Réalisé par :

Mlle Salima IMOUDACHE
Mlle Lyna KHERBOUCHE

Encadré par :

Mr. O. STIHI

Juin 2013

E.N.P- 10, Avenue Hassen Badi, 16200 El Harrach, Alger

Remerciements

Nous tenons à exprimer nos remerciements à Mr Omar Stihi pour nous avoir encadré, pour sa disponibilité et pour nous avoir facilité l'accès au LCP.

Nous tenons aussi à exprimer notre gratitude à Mr Nouredine Ouadah du CDTA, pour sa précieuse aide, sa disponibilité et sa gentillesse. Nous lui adressons nos vifs remerciements pour nous avoir fait l'honneur de faire partie de notre jury.

Nos remerciements vont aux membres de jury qui nous font l'honneur d'évaluer notre travail.

Nous remercions également vivement Mme Latifa Hamami enseignante à l'ENP pour avoir reçu nos questions, pour nous avoir consacré de son temps et avoir fait de son mieux pour nous aider. Sans oublier Mr Rachid Illoul enseignant à l'ENP, pour sa perpétuelle bonne humeur et ses précieux conseils.

Nous tenons à exprimer notre reconnaissance à toute personne qui a contribué par son aide à ce travail, particulièrement, Mr Boussif Abderraouf ingénieur automatique de l'ENP, qui a toujours été serviable et qui n'a jamais manqué de répondre à nos questions.

Nous ne pouvons terminer ces remerciements sans exprimer notre gratitude à deux personnes qui nous ont marqué par leur gentillesse, leur grandeur d'âme et leur sens du service ; Ami Saleh de l'ENP sans qui, l'école polytechnique, et en particulier l'ancienne bibliothèque ne seraient pas les mêmes, ainsi que Mme Fahsil du service formation du CDTA.

Enfin ; nous rendons à travers ce mémoire un hommage aux personnes à qui revient tout le mérite ; nos parents qui nous ont soutenu et qui ont tant sacrifié pour nous.

Kherbouche-Imoudache

Dédicaces

Aux personnes qui ont fait de moi ce que je suis aujourd'hui : mes anges gardiens ;Jeddi Bouhou et Yemma Touma ; et mes chers parents, Vava et Yemma ;A ma chère soeur, Ilhem ;A mes merveilleux frères, Merzouk et Mustafa ;A mon cher oncle, Idir et sa femme,A mes chères tantes et leurs maris,en particulier Hakima et Azdine qui etaient mon deuxième foyer tout au long de ces 5 années.A ma merveilleuse tribu de cousins : Yanis, Hakim, Iles, Aksil, Milissa, Lydia, Nassim, Yacine, Salima, Massi, Katia et Salim,ainsi qu'a Dania et Sonia.A ceux de ma famille qui ne sont plus de ce monde,A la terre qui porte leurs tombes,A Akbou, la ville qui m'a vu grandir ;A mes deux meilleures amies, Dyhia et Nesrine, sans qui ma vie d'étudiante ne serait pas la meme,A ma binome, Salima avec qui j'ai passé tant de moments,A tous mes amis de polytech, surtout Amina, Lamia et Nejla.A tous ceux qui ont peuplé mes 5 années d'études ger ;Aux souvenirs de nos meilleures années...

Lyna

Dédicaces

A ceux, qui par leurs amour, leurs sacrifices et leurs soutient ont fait de moi la personne que je suis aujourd'hui ; a ce que je possède de plus cher au monde ; mes parents ;

A mes chers grands-parents, Jeddi Seddik , Jidda et Ayyi.

A la mémoire de mon ancle, Khali Mouhammed ;

A mes chers frères, mes chers cousins et cousines, particulièrement Karima ;

A mon amie et binome, Lyna qui m'a tant supporté ;

A tous mes amis pour tous les bons moments ;

A Zami Saleh pour sa gentillesse et sa grandeur d'ame ;

A Mr Tidjini ;

Salima

Table des matières

1	Généralités sur l'imagerie	1
1.1	Introduction	1
1.2	Notions de l'image	1
1.2.1	Caractéristique de l'image	2
1.2.2	Les capteurs d'image	2
1.3	Bruit dans les images	3
1.4	Techniques de traitement d'image	3
1.4.1	Les Prétraitements	4
1.4.2	Les techniques de segmentations	
	5	
1.4.3	c. Les techniques de haut niveau	6
1.5	Conclusion	6
2	Algorithmes de traitements d'image et d'extraction de primitives	8
2.1	Introduction	8
2.2	L'asservissement visuel	8
2.3	Traitement d'image et extraction de primitives	9
2.3.1	La transformée de Hough	10
2.3.2	L'extraction de primitives	11
2.4	Résultats expérimentaux du traitement d'images	19
2.4.1	La transformée de Hough	20
2.5	Conclusion	21
3	Application des algorithmes de traitement d'image	23
3.1	Introduction	23
3.2	Calcul des indices visuels	23
3.3	Applications de SURF,SIFT et ORB	24
3.4	Conclusion	29

Table des figures

1.1	Les niveaux de gris.	2
1.2	Histogramme d'une image.	3
1.3	L'image originale.	5
1.4	L'image segmentée.	5
1.5	Détection de contour avec Canny :(a) image originale, (b) image après dérivation suivant X, (c) image après dérivation suivant Y, (d) image après application du gradient, (e) image après seuillage, (f) image après hystérésis.	7
2.1	Schéma de principe des stratégies de commande référencées vision[1]. . .	9
2.2	Sinusoïde représentant la famille de ligne passant par un point donné du contour d'une image.	10
2.3	Mise en évidence des paramètres de la ligne commune à 3 points de contour par l'intersection de leurs sinusoides.	11
2.4	L'image d'origine	11
2.5	Résultats de l'application de la transformée de Hough.	11
2.6	Résultats de l'application du Hessian detector.	13
2.7	goss.	14
2.8	Sinusoïde représentant la famille de ligne p	15
2.9	Détection de droite après seuillage à 0.5.	21
2.10	Détection de droite après seuillage à 0.2.	21
2.11	Détection de droite avec un faible seuil de PIC.	21
2.12	Détection de droite avec le seuil de PIC choisi.	21
3.1	Image de l'objet	24
3.2	Extraction du centre de gravité de l'objet.	24
3.3	L'image référence.	25
3.4	Detectin des points d'intérêt par l'algorithme SURF.	25
3.5	Matching de L'image de référence avec une image différente	26
3.6	Matching de L'image de référence avec la même image décalée.	26
3.7	Matching de l'image de référence avec elle même	26
3.8	Detectin des points d'intérêt par l'algorithme SIFT.	27
3.9	Matching de l'image de référence avec une autre image différente	27
3.10	Matching de l'image de référence et de la référence	27

3.11 avec l'image référence	27
3.12 Detectin des points d'intérêt par l'algorithme ORB.	28
3.13 Matching de l'image de référence avec une image différente	28
3.14 Matching de L'image de référence avec la même référence decalée.	28
3.15 Matching de l'image de référence avec elle même	29

Introduction générale

le traitement d'image est un domaine indissociable de l'asservissement visuel. en effet la robustesse même des lois de commande par retour visuel dépend de l'extraction des primitives de l'image.

les premiers travaux en traitement d'image appliqué à l'asservissement visuel utilisaient l'intensité lumineuse des pixels pour détecter des points particuliers de l'image ; contours, centre de gravité...etc, mais rapidement ces méthodes ont montrées leurs limites, car une telle approche est très sensible aux variations lumineuses de la scène, à l'échelle et à la rotation des images, ce qui limite fortement leurs applications. par la suite, d'autres algorithmes, basées aussi sur l'intensité lumineuse sont développés, mais ils se sont basés non plus sur des primitives géométriques mais sur des primitives globale à l'image qui assurent l'invariance à la rotation ,à la translation et à l'échelle.

les applications de tel algorithmes sont très variées, elles vont de la détection d'objets au tracking en passant par les systèmes de sécurité biométriques.

Dans ce travail, nous nous sommes proposé d'étudier trois algorithmes d'extraction de primitives, le SURF,le SIFT et l'ORB ainsi que leurs applications en robotique.

Organisation de la thèse :

1. Dans le premier chapitre,nous présenterons quelques notions de base sur l'imagerie et les méthodes de traitement d'image.
2. Le deuxième chapitre ,nous allons le consacrer à la théorie d'extraction de primitives ,nous allons détailler dans ce chapitre le principe de chaque algorithme utilisé à savoir le SURF,le SIFT et l'ORB.
3. Dans ce dernier chapitre ,nous exposons les résultats obtenus par application des algorithmes d'extraction de primitives.s

Chapitre 1

Généralités sur l'imagerie

1.1 Introduction

Le traitement d'images est un thème de recherche situé entre l'informatique et le traitement du signal [2], né de l'idée et de la nécessité de remplacer l'observateur humain par la machine [3].

Dans ce chapitre on commencera par présenter quelques définitions de l'imagerie. Ensuite, on parlera des capteurs d'images utilisés dans la vision par ordinateur. Enfin, on terminera par exposer les différentes opérations d'analyse d'images telles que la détection de contours, la segmentation et le traitement haut niveau.

1.2 Notions de l'image

- **L'image**

une image est une représentation planaire d'une scène ou d'un objet situé en général dans un espace tridimensionnel. Son élaboration résulte de la volonté de proposer une entité observable par l'œil humain. Ceci explique d'une part son aspect planaire et d'autre part le fait que l'information élémentaire associée à chaque point de l'image soit transcrite en niveau de gris ou en couleur [3].

- **L'image numérique**

Une image numérique est définie comme un signal fini bidimensionnel échantillonné à valeurs quantifiées dans un certain espace de couleurs. Elle est constituée de points (pixels) [2]. Par Pixel on désigne le plus petit élément d'une surface d'affichage auquel on peut associer individuellement une couleur et une intensité. Dans le cas d'un écran monochrome, le pixel s'identifie à un point et dans le cas d'un écran couleur, il est constitué de trois points de couleur différente (rouge, vert, bleu) [4].

- **Image en niveau de gris**

Le niveau de gris est la valeur de l'intensité lumineuse en un point. Une image en niveau de gris est une image composée de points gris plus ou moins foncés[5], donc 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[2].

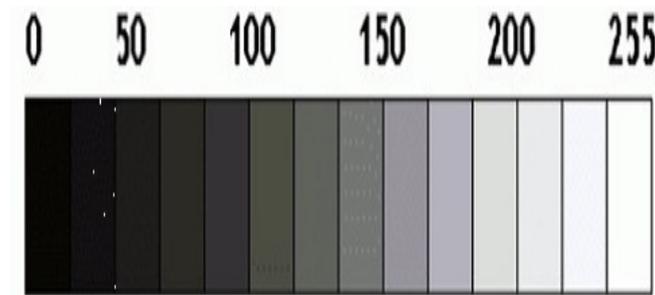


FIGURE 1.1: Les niveaux de gris.

1.2.1 Caractéristique de l'image

1. Définition et Résolution :

On appelle définition le nombre de points (pixel) constituant l'image, c'est-à dire le nombre de colonnes de l'image que multiplie son nombre de lignes.

La résolution est le nombre de points par unité de surface, exprimé en points par pouce PPP, un pouce représentant $2.54cm$ [6].

2. L'histogramme :

L'histogramme représente la répartition des pixels en fonction de leur niveau de gris. Il fournit diverses informations comme les statistiques d'ordre et permet d'isoler des objets [7]. Par convention un histogramme représente le niveau d'intensité en abscisse en allant du plus foncé au plus clair 1.2. L'histogramme est un outil privilégié en analyse d'images car il représente un résumé simple[6]. Ainsi l'histogramme d'une image en 256 niveaux de gris sera représenté par un graphique possédant 256 valeurs en abscisses, et le nombre de pixels de l'image en ordonnées.

1.2.2 Les capteurs d'image

Les capteurs d'images sont formés de capteurs optiques sur lesquels est projetée l'image. Un capteur optique délivre un signal électrique proportionnel à l'éclairement moyen de sa surface photosensible ou photosite [8]. Il existe deux types de capteurs d'images :

1. Capteurs à tube

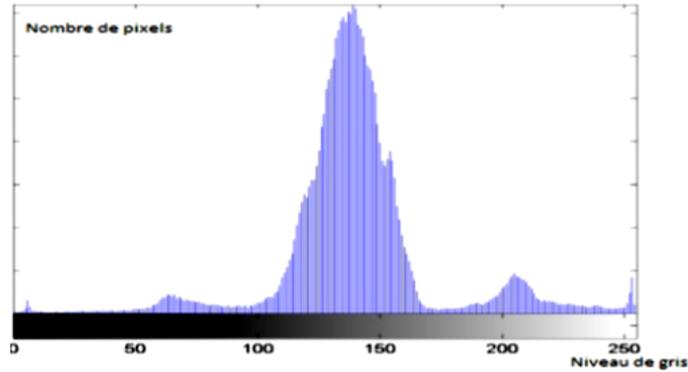


FIGURE 1.2: Histogramme d'une image.

Le principe est d'effectuer un balayage d'un faisceau d'électrons issu d'une cathode, accéléré et concentré au moyen d'électrodes portées à des potentiels élevés. Ce faisceau est dévié par des champs magnétiques créés par des bobines. Néanmoins, ce type de capteurs présente les inconvénients suivants : encombrement, durée de vie limitée, puissance consommée importante [9].

2. Capteurs intégrés

Ce sont des puces de silicium comprenant des circuits nécessaires à la conversion photoélectrique et à l'analyse de l'image, à savoir les photoéléments ou capteurs optiques organisés en ligne ou en matrice. Divers capteurs intégrés existent : les capteurs à transfert de charges CCD (*Charge Coupled Device*) ou à injection de charges CID (*Charge Injection Device*) [9].

1.3 Bruit dans les images

Un système d'acquisition et d'enregistrement d'image ne restitue pas l'image d'une manière parfaite, des informations parasites s'ajoutent aléatoirement aux détails de la scène d'origine constituant ce qu'on appelle le bruit. Le bruit n'a pas les mêmes origines, mais provoque des effets assez semblables comme la perte de netteté sur les détails ou l'apparition de grains.

Le bruit est d'autant plus présent que l'intensité du signal baisse, il se distingue mieux sur les zones sombres. C'est pour cette raison que l'on mesure l'effet du bruit par le rapport *Signal/Bruit*. Quand cette valeur augmente, cela signifie que la proportion du signal parasite dans le signal d'origine diminue.

1.4 Techniques de traitement d'image

Selon le niveau d'abstraction, les techniques de traitement d'images peuvent être classées en trois catégories :

- Les processus de bas niveau, qui nécessitent très peu d'informations sur le contenu des images.

Il s'agit de processus qu'on peut regrouper sous l'appellation pré traitement d'images et extractions d'indices ;

- Les processus de niveau intermédiaire, qui englobent les techniques de segmentation d'images en régions ;

- Les processus de haut-niveau, qui peuvent nécessiter des informations sur le contenu des images, et dans lesquels on trouve, la reconstruction tridimensionnelle (modélisation de l'environnement), la reconnaissance de formes, l'analyse de mouvements,... etc[6].

1.4.1 Les Prétraitements

1. Seuillage

Une image numérique, pour pouvoir être exploitée a généralement besoin d'être simplifiée, c'est le but du seuillage. Ce dernier consiste à transformer l'image codée sur 6, 8 ou 16 bits en une image binaire où les pixels à 1 correspondent aux objets et les pixels à 0 au fond de l'image [10].

Le seuillage permet de sélectionner les parties de l'image qui intéressent l'opérateur, par exemple 2 types de grains (noirs et blancs) dans un mélange, pour cela il suffit de fixer un seuil ou plus selon lequel ou lesquels, on affecte au pixel en cours la couleur noire ou blanche [8]. Deux types de seuillage existent :

- **Seuillage simple**

Consiste à fixer un seuil entre 0 et 255, et décider par la suite d'affecter la couleur noire ou blanche à un pixel de l'image selon que le niveau de gris est inférieur ou supérieur au seuil.

- **Seuillage par Hystérésis**

Dans cette méthode, on choisit deux seuils A et B tels que $A < B$. En parcourant l'image initiale, on crée une nouvelle en affectant la couleur noire au pixel donc le niveau de gris est soit inférieur à A soit supérieur à B et la couleur blanche ailleurs. Ce type de seuillage permet de mettre en évidence un ensemble précis de pixels où leur niveau de gris est entre A et B [8].

2. Filtrage

Il consiste à appliquer une transformation à l'ensemble ou une partie d'une image numérique en utilisant des filtres[8] afin d'améliorer le contraste entre les zones de l'image à conserver après seuillage, et celles que l'opérateur juge inutiles [10]. Son principe est de modifier la valeur des pixels d'une image, généralement dans un but d'améliorer son aspect, il s'agit de créer une nouvelle image en se servant des pixels

de l'image d'origine. Sa qualité sera améliorée [6].

f On distingue généralement les types de filtres suivants [8] :

- **Filtre passe-haut**

Atténue les composantes de basse fréquence de l'image et permet d'accentuer les détails et le contraste. il est représenté par le *gradient* et le *Chapeau haut de forme*.

- **Filtre passe-bas**

Consiste à atténuer les composantes de l'image ayant une fréquence haute ainsi que le bruit de l'image. On l'appelle aussi *filtre de lissage*. Ces filtres peuvent être linéaires ou non linéaire, on cite par exemple le filtre *Gaussien* qui est un opérateur de lissage utilisé pour estomper les détails et le bruit.

1.4.2 Les techniques de segmentations

La segmentation d'une image binaire a la même finalité que le seuillage ; d'ailleurs, certains auteurs classent le seuillage parmi les méthodes de segmentation. Elle consiste à séparer l'image en plusieurs zones. La segmentation la plus simple consiste à séparer les particules convexes apparues comme connexes à la suite des opérations précédentes[10].



FIGURE 1.3: L'image originale.



FIGURE 1.4: L'image segmentée.

Il existe de nombreuses méthodes de segmentation que l'on peut regrouper en quatre principales classes d'algorithmes [8] :

- Segmentation basée sur les régions (region-based-segmentation) ;
- Segmentation basée sur une approche globale de l'image (seuillage, histogramme, approches basées sur le nuage couleur...etc) ;
- Segmentation basée sur les contours (edge-based-segmentation) ;

- Segmentation basée sur la coopération entre les trois premières segmentations.

- Détection de contours

Plusieurs techniques de détection de contour existent dans la littérature. Cependant, comme un contour est défini comme étant une zone de forte discontinuité des pixels d'une image, ces méthodes se basent toutes sur le même principe, qui est celui du gradient.

Parmi les filtre les plus utilisés ,on peut citer :

- Masque de Prewitt,
- Masque de Sobel,
- Masque de Kirsch,
- Filtre de SHEN et CASTAN,
- Le Filtre de Canny.

La détection de contour par la méthode *Canny* commence par la convolution de l'image avec une gaussienne d'un sigma donné. Après convolution, les dérivées suivant X et suivant Y sont calculées, ce qui permet d'avoir le gradient de l'image.

Une fois le gradient obtenu, un seuillage est appliqué à l'image. Ceci a pour but d'éliminer les pixels qui ne constituent pas un maximum local. Finalement, un opérateur hystérésis est appliqué à l'image, ce qui a pour effet de marquer chaque pixel comme appartenant ou non au contour. La figure 1.5 montre les différents résultats obtenus par application de l'algorithme Canny à une image donnée.

1.4.3 c. Les techniques de haut niveau

Une fois les informations extraites sous forme d'un ensemble d'entités, la dernière étape consiste à prendre une décision en fonction de ces informations. Selon l'application visée, cette décision peut être une simple valeur booléenne (l'image fait partie de la catégorie visée ou non). Ou bien une commande moteur (le cas de l'asservissement et de l'évitement d'obstacles).ou un ensemble plus complexe de traitements (mise à jour d'une carte par exemple)[6].

1.5 Conclusion

Ce chapitre nous a permis de traiter la théorie de traitement d'image, on a abordé la notion d'image en donnant ces caractéristiques ainsi que les méthode d'analyse et les techniques de prétraitement qui permettent par la suite l'extraction des primitives pour une application d'asservissement visuel.

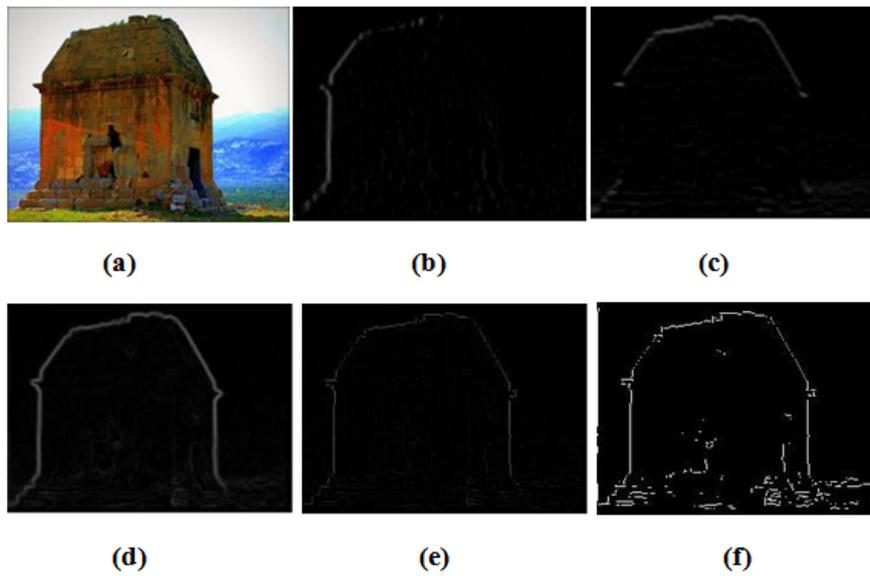


FIGURE 1.5: Détection de contour avec Canny :(a) image originale, (b) image après dérivation suivant X, (c) image après dérivation suivant Y, (d) image après application du gradient, (e) image après seuillage, (f) image après hystérésis.

Chapitre 2

Algorithmes de traitements d'image et d'extraction de primitives

2.1 Introduction

Dans ce chapitre, on commencera tout d'abord par donner quelques généralités sur l'asservissement visuel. Ensuite, on donnera les techniques de traitement d'image appliquées à l'asservissement visuel. Après présentation de la transformée de *hough*, on détaillera trois algorithmes d'extraction de primitives SURF, SIFT et ORB dont on présentera leurs applications dans l'asservissement visuel à la fin de ce chapitre.

2.2 L'asservissement visuel

L'intégration de la vision dans la boucle de commande a donné naissance à une nouvelle discipline en robotique, l'asservissement visuel. Il consiste à contrôler les mouvements d'un système robotique en utilisant des informations visuelles, notées s , issues d'un ou plusieurs capteurs de vision, embarqués ou non sur le système [11]. L'asservissement visuel est défini comme étant le domaine qui permet d'apporter des solutions à deux problèmes :

- La possibilité d'extraire les informations suffisantes de l'image donnée par le capteur de vision,
- L'implémentation d'algorithmes de contrôle assez simples pour pouvoir travailler à une fréquence compatible avec celle du capteur de vision ; mais aussi une robustesse acceptable vis-à-vis des erreurs inévitables entre l'environnement et l'image fournie par la vision.

La première étape de l'asservissement visuel consiste à extraire de l'image fournie par le capteur de vision, des formes caractéristiques qui constituent l'objectif à

atteindre appelées primitives, la deuxième étape est le design d'une commande qui assurera la convergence vers la configuration correspondante à l'image de l'objectif, et ceci en débutant d'une configuration initiale quelconque.

De nombreux chercheurs se sont penchés sur ce problème dès le début des années 70, mais leurs travaux ont été limités par les contraintes technologiques à savoir la puissance de calcul des processeurs de l'époque, ce n'est qu'en 1982 que des travaux plus formels ont été faits, ils ont exposé deux types de problèmes : le premier étant le choix et l'extraction des primitives à utiliser, et le deuxième étant l'analyse et la synthèse des commandes[12].

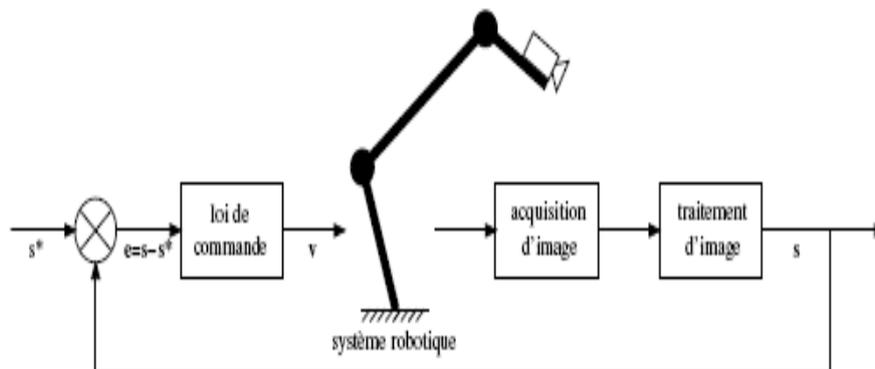


FIGURE 2.1: Schéma de principe des stratégies de commande référencées vision[1].

2.3 Traitement d'image et extraction de primitives

L'extraction des informations visuelles S des images fournies par la caméra qui constituent le retour de la boucle de commande, représente l'un des problèmes majeurs de l'asservissement visuel.

En effet, la robustesse même de la tâche dépend fortement de l'étape d'extraction des primitives des images fournies par le capteur de vision. De plus, le calcul de l'erreur sur le positionnement se fait en utilisant ces informations visuelles qui sont sujettes aux erreurs, ce qui pourrait fausser la tâche de positionnement [13].

Cette section présentera les principales techniques de traitement d'image, d'extraction et d'appariement de primitives utilisées dans la vision par ordinateur et dans l'asservissement visuel en particulier.

2.3.1 La transformée de Hough

Dans la transformée de Hough, les lignes sont exprimées en coordonnées polaires :

$$y = -\left(\frac{\cos(\theta)}{\sin(\theta)}\right)x + \frac{r}{\sin(\theta)} \quad (2.1)$$

Pour chaque point (x_0, y_0) appartenant à un contour, on peut trouver une infinité de ligne (r, θ) passant par ce point. Et en traçant cette famille de ligne passant par ce point on obtient une sinusoïde comme montrée par la figure 2.2.

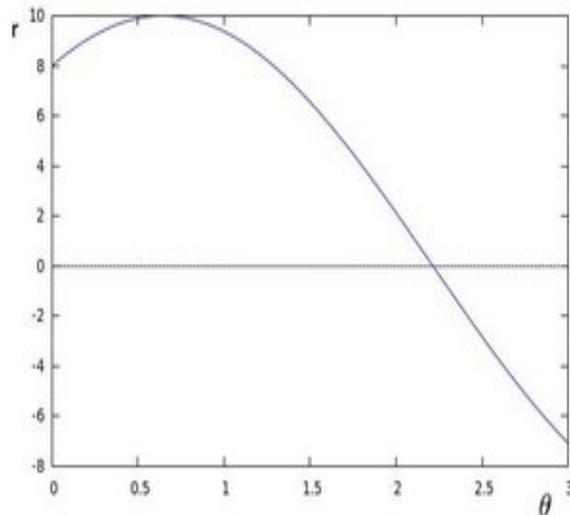


FIGURE 2.2: Sinusoïde représentant la famille de ligne passant par un point donné du contour d'une image.

Et si maintenant on trace les sinusoïdes correspondantes à chaque point du contour comme le montre la figure 2.7, on peut détecter les points du contour qui appartiennent à la même ligne. Ceci est mis en évidence par l'intersection des sinusoïdes propres à chaque point. Cette intersection fournit les paramètres (r, θ) de la ligne commune[14].

Comme la transformée de Hough utilise les point du contour d'une image, elle n'est applicable qu'après détection de contour. Un exemple d'application de l'algorithme de Hough est donné par la figure 2.5.

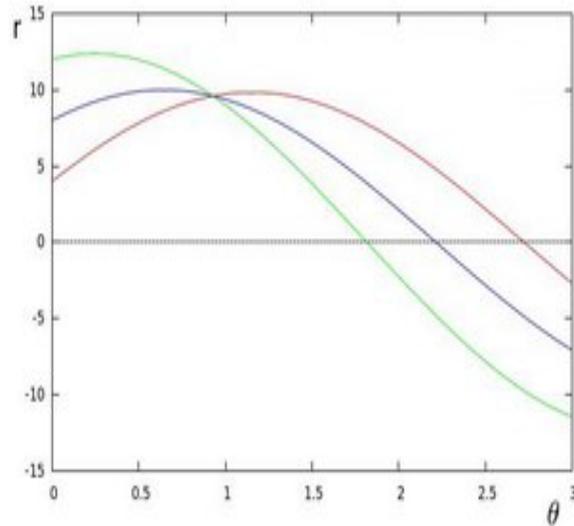


FIGURE 2.3: Mise en évidence des paramètres de la ligne commune à 3 points de contour par l'intersection de leurs sinusoides.



FIGURE 2.4: L'image d'origine

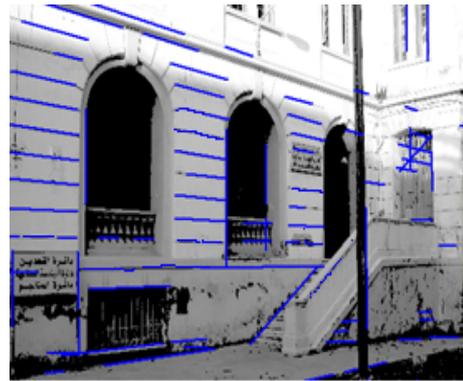


FIGURE 2.5: Résultats de l'application de la transformée de Hough.

2.3.2 L'extraction de primitives

Plusieurs algorithmes d'extraction et d'appariement de primitives ont été proposés ces dernières années. On peut citer les travaux de *Kallem* [15] qui a utilisé dans sa méthode, l'intensité lumineuse des pixels de l'image. Cette méthode présente l'inconvénient d'être sensible aux conditions de luminance utilisée. Ce problème a été, cependant en partie résolu par *Collée* dans [16] en intégrant les conditions de luminance dans la boucle de commande. Cette méthode a d'ailleurs été appliquée avec succès par *Marchand* dans [17] sur le positionnement par rapport à 4 points.

D'autres algorithmes, qui étaient développés à l'origine pour des fins de reconnaissance ou de pistage d'objets ou de personnes, sont utilisés désormais dans les boucles de retour visuel, parmi lesquelles les algorithmes *SURF*, *SIFT* et *ORB*.

Dans ce qui suit on donnera une idée sur le principe de ces algorithmes et on renverra le lecteur à [18], [19] et [20] pour des explications plus approfondies.

Généralement, une primitive dans une image peut être :

- Un pixel,
- Un contour,
- Un segment issu d'une détection de contour.

Pour le cas des algorithmes étudiés, l'extraction de primitives se base sur la sélection de *points d'intérêts* dans l'image. Un point d'intérêt est un point qui doit avoir des caractéristiques plus significatives que les autres. Ils correspondent à des changements bidimensionnels du signal et peuvent être des coins en L, des jonctions en T ou des blobs (structures)[21]. Cette sélection est effectuée grâce à un détecteur qui devra avoir pour propriétés :

- La répétabilité(Repeatability) ; i.e., la capacité à détecter les points d'intérêt sous différentes conditions de vue.
- L'invariance : que ce soit vis-à-vis de la translation, de la rotation ou du changement d'illumination.
- L'exhaustivité : il faut couvrir suffisamment les zones à décrire.

Le détecteur le plus communément utilisé est le détecteur de *Harris* proposé en 1988 basé sur les valeurs propres de la matrice des moments du deuxième ordre. Plusieurs travaux se sont cependant, portés sur le *Hessian detector* [22]. D'autres détecteurs ont été proposés, on peut citer les détecteurs basés sur la notion de région saillante dans l'image et ceux utilisant les contours.

(a) **Le Hessian detector**

Le Hessian d'une image est donné par :

$$Hessian(I) = \begin{pmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{pmatrix} L_{xy} \quad (2.2)$$

Ceci correspond à la recherche de variations importantes dans deux directions orthogonales. La détection des points d'intérêt se fait en imposant un seuil au déterminant du Hessian donné par :

$$\det(\text{Hessian}(I)) = I_{xx}I_{yy} - I_{xy}^2 \quad (2.3)$$

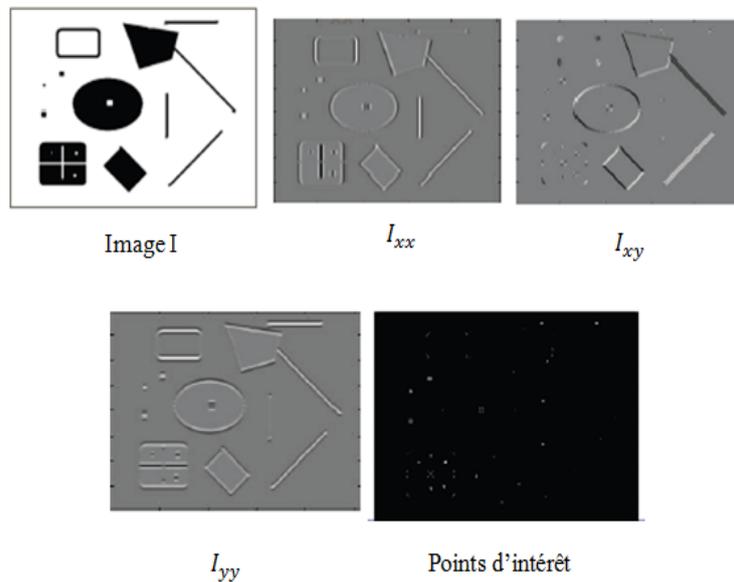


FIGURE 2.6: Résultats de l'application du Hessian detector.

(b) **Le Harris detector**

Comme déjà mentionnée, le détecteur de Harris se base sur la matrice des moments d'ordre deux :

$$\mu(\sigma_I, \sigma_D) = g(\sigma_I) \begin{pmatrix} I_{xx}^2(\sigma_D) & I_x I_y(\sigma_D) \\ I_x I_y(\sigma_D) & I_{yy}^2(\sigma_D) \end{pmatrix} L_{xy} \quad (2.4)$$

Avec,

$g(\sigma_I)$: Filtre gaussien.

Une fois les variations de l'image selon deux directions sont détectées, un filtre gaussien est appliqué pour trouver les points d'intérêt.

(c) **Le SIFT**

Cet algorithme est nommé *Scale Invariant Feature Transform (SIFT)* du fait qu'il transforme les données image en des coordonnées invariantes à l'échelle relatives aux primitives locales. L'extraction de primitive par cet algorithme passe par trois étapes :

- Détection des maxima du scale-space,
- Détection des maxima locaux,
- Localisation des points d'intérêt.

a. Détection des maxima du scale-space

La détection de points d'intérêt dans une image est faite par un filtrage en cascade utilisant des algorithmes efficaces pour identifier les candidats qui seront après examinés en détails. La première étape de la détection de points d'intérêt est l'identification des positions et des échelles qui peuvent être assignées de manière répétée sous différentes vues du même objet.

Détecter des positions qui sont invariantes au changement d'échelle de l'image peut être accompli en recherchant des primitives stables dans toutes les échelles en utilisant une fonction d'échelle continue notée *Scale Space*.

Il a été démontré que la seule scale space possible est la fonction gaussienne. De ce fait, le scale space d'une image est défini comme une fonction $L(x, y, \sigma)$, produite par convolution d'une gaussienne à échelle variable $g(x, y, \sigma)$ et d'une image $I(x, y)$:

$$L(x, y, \sigma) = g(x, y, \sigma) * I(x, y) \quad (2.5)$$

Où ;

$$g(x, y, \sigma) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \quad (2.6)$$

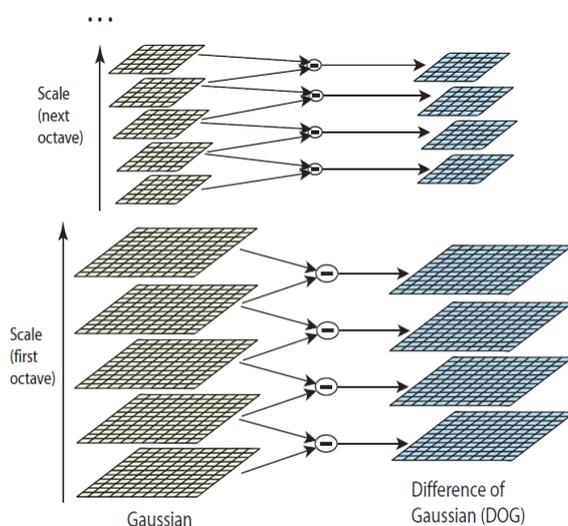


FIGURE 2.7: goss.

Afin de permettre une détection plus efficace des points d'intérêt, la recherche des extrema se fait sur la fonction *DoG* (Diference Of Gaussian) convoluée avec l'image $D(x, y, \sigma)$, qui peut être calculée à partir de la différence de deux échelles :

$$D(x, y, \sigma) = (g(x, y, \sigma) - g(x, y, k\sigma)) * I(x, y) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (2.7)$$

La raison de ce choix vient du fait que la fonction D peut désormais être calculée par simple soustraction d'image .

b. Détection des extrema locaux

Dans le but de détecter les extrema de $D(x, y, \sigma)$, chaque pixel est comparé à ses huit voisins dans l'image courante et à neuf voisins dans les échelles supérieures et inférieures (voire figure 2.8). Il est sélectionné seulement s'il est le plus grand de tous ses voisins ou le plus petit d'entre eux. Un tel point est dit point échantillon.

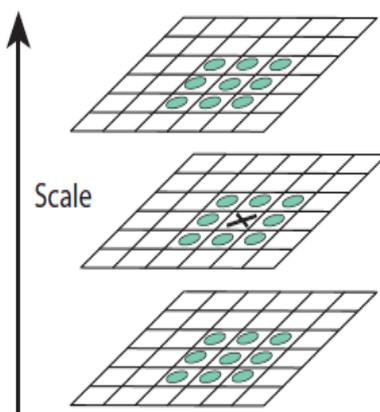


FIGURE 2.8: Sinusoïde représentant la famille de ligne p

c. Localisation des points d'intérêt

Une fois un point échantillon est détecté par comparaison de chaque pixel à ses voisins, la prochaine étape est d'effectuer un accès détaillé au voisinage de ces points pour la localisation et l'échelle. Cette information permet de rejeter les points avec un faible contraste (qui sont de ce fait sensibles aux bruits), ou ceux qui sont mal localisés selon un contour.

Le point d'intérêt peut être localisé tout simplement à la position et à l'échelle du point échantillon, cependant, une amélioration de la stabilité des points détectés peut être obtenue en utilisant le développement en série de Taylor de la fonction D au point considéré :

$$D(X) = X + \frac{\partial D^T}{\partial X} X + \frac{1}{2} X^T \frac{\partial D^2}{\partial X^2} X \quad (2.8)$$

Où ;

D est ses dérivées sont calculées au point échantillon et $X = (x, y, \sigma)^T$ représente la compensation de ce point.

L'extremum de \widehat{X} est déterminée en annulant la dérivée de $D(X)$ par rapport à X ce qui donne :

$$\widehat{X} = -\frac{\partial D^{-1}}{\partial X^2} \frac{\partial D}{\partial X} \quad (2.9)$$

Si \widehat{X} est supérieur à 0.5 selon au moins une direction, cela veut dire que l'extremum se trouve plus près d'un autre point échantillon. Dans ce cas le point échantillon est changé et l'interpolation refaite. la valeur finale de \widehat{X} est rajoutée à la position du point échantillon correspondant pour obtenir une estimation de la position de l'extremum.

d. Calcul du descripteur SIFT

Un descripteur permet d'assigner à chaque point d'intérêt une orientation se basant sur les propriétés internes de l'image. Le descripteur *SIFT* a été élaboré dans le souci de fournir les résultats les plus stables. Dans cette méthode, l'échelle du point d'intérêt est utilisée pour sélectionner l'image lissée L, avec la plus proche échelle, de sorte que toutes les opérations soit effectuée a échelle-invariante. Pour chaque image lissée, L à une échelle donnée, la magnitude du gradient $m(x, y)$ et l'orientation, $\theta(x, y)$ sont calculée par les équations :

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2} \quad (2.10)$$

$$\theta(x, y) = \tan^{-1}\left(\frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)}\right) \quad (2.11)$$

(d) Le SURF

Le SURF (*Speed Up Robust Feature*), est un algorithme d'extraction et d'appariement de primitives développé dans le souci d'être invariant vis-à-vis de la rotation, de la translation et de l'échelle de l'image. L'algorithme SURF utilise comme détecteur le *Fast Hessian detector* intégrant dans son expression la position et l'échelle d'un point donné de l'image. Si on se donne un point $X(x, y)$

de l'image, le Hessian de l'image en X à une échelle σ est défini comme suit :

$$H(X, \sigma) = \begin{pmatrix} L_{xx}(X, \sigma) & L_{xy}(X, \sigma) \\ L_{xy}(X, \sigma) & L_{yy}(X, \sigma) \end{pmatrix} \quad (2.12)$$

$L_{xx}(X, \sigma)$: Représente la convolution de la dérivée du second ordre d'une gaussienne $\frac{\partial^2}{\partial x^2} g(\sigma)$ avec l'image I au point X . De même pour $L_{xy}(X, \sigma)$ et $L_{yy}(X, \sigma)$ [?].

La détection par la méthode SURF se fait en trois étapes :

a. Détection des maxima du Hessian

Le détecteur SURF est basé sur le déterminant de la matrice Hessian 2.12. Le signe du déterminant ainsi que les signes des deux valeurs propres de cette matrice nous informent sur la nature des points détectés. la variation de σ va nous permettre de parcourir plusieurs échelles. Bay [18] a fait une approximation du déterminant de la matrice H en utilisant LoG (Laplacian of Gaussian). Elle est donnée par la formule 2.13 :

$$\det(H_{approx}(x, y, \sigma)) = D_{xx}(x, y, \sigma)D_{yy}(x, y, \sigma) - (0.9 * D_{xy}(x, y, \sigma))^2 \quad (2.13)$$

b. Localisation des points d'intérêt

Dans cette étape est effectué un filtrage des points instables pour ne garder que les points stables de manière plus précise et robuste à la rotation dans l'image et à travers toutes les échelles :

- D'abord, les valeurs des déterminants de la matrice $H(DoH)$ subissent un seuillage supérieur, le seuil est choisi selon l'application.
- Ensuite, une suppression des non maxima est effectuée (comparaison avec les 18 voisins).
- Enfin, la précision de détection est améliorée par une interpolation polynomiale quadratique des extrema locaux du DoH dans le scale space.

Ceci est effectuée comme pour la méthode *SIFT* , par le développement de Taylor à l'ordre deux de DoH centré autour des points d'intérêt candidats exprimé par :

$$H(X) = H + \frac{\partial H^T}{\partial X} X + \frac{1}{2} X^T \frac{\partial H^2}{\partial X^2} X \quad (2.14)$$

c. Calcul du descripteur SURF

Les performances du descripteur SIFT [19] ont conduit *H. Bay* [18] à élaborer un descripteur SURF qui maintient des propriétés similaires en étant plus robuste et plus rapide.

Ce descripteur est extrait en deux étapes :

- **Calcul de l'orientation principale :**

L'orientation principale est recherchée dans les pixels voisins dans le cercle du rayon 6σ . Le gradient de l'image peut être défini en tout point par les deux paramètres magnitude et orientation donnés par les équations (2.15) :

$$\begin{cases} m(x, y) = \sqrt{I_x^2(x, y) + I_y^2(x, y)} \\ \theta(x, y) = \tan^{-1}\left(\frac{I_x(x, y)}{I_y(x, y)}\right) \end{cases} \quad (2.15)$$

Avec,

$I_x(x, y)$, $I_y(x, y)$ représentent la dérivée de la matrice intensité de l'image par rapport à x (resp. par rapport à y).

Cependant, pour le calcul de la direction, le descripteur SURF remplace les dérivées de l'image par les différences successives des intensités en utilisant la décomposition en ondelettes de *Haar* de taille 4σ sur les séries de pixels échantillonnés à une cadence σ dans chaque direction x et y des régions d'intérêt.

Les réponses du filtre de *Haar* sont pondérées par une gaussienne circulaire d'écart type 2.5σ centrée sur le point d'intérêt. Ces réponses sont représentées à l'intérieur d'un cercle ; ce cercle est décomposé en sz portions de soixante degrés. L'orientation principale est alors obtenue par la direction du vecteur inscrivant le plus grand score parmi ceux déterminés à partir des 6 portions du cercle et sera donc associé au point d'intérêt.

- *Calcul des composantes du descripteur*

Pour calculer le descripteur SURF, d'abord on divise la fenêtre de taille 20σ qui entoure le point en $4X4$, chaque sous région est divisée de sa part en $5X5$ zones de taille σ . On applique le filtre de *Haar* à nouveau avec un écart

type 3.3σ . Pour chaque sous-région, on calcule le vecteur de l'équation (celle qui suit) où d_x et d_y sont les réponses du filtre de *Haar*.

$$V_{(sous-région)} = [\sum d_x \sum d_y \sum |d_x| \sum |d_y|] \quad (2.16)$$

Le descripteur est obtenu en concaténant les vecteurs de 16 sous-régions, il est alors de taille 16×4 [21].

(e) **L'ORB**

L'algorithme ORB était proposé par *E. Rublee*[?] comme une alternative au *SUFT* et *SIFT*, il est basé sur le détecteur *FAST* et le descripteur *BRIEF*, d'où son nom : *Oriented Fast and Rotated BRIEF (ORB)*.

• **Le détecteur FAST**

Comme déjà vu, la détection de points d'intérêt requiert pour *SIFT*, le calcul du gradient, et pour *SURF* une approximation du *DoH*. Ces deux façons de faire sont soit, trop lourdes en temps de calcul soit, pour le cas du *SURF*, pauvre en approximation. Le détecteur *FAST* constitue alors une solution aux deux problèmes précédent, il se base sur la technique dite *centroïde d'intensité*.

La technique de la *Centroïde d'intensité* stipule que l'intensité d'un coin est compensée en son centre. Ceci définit un vecteur orienté du centre du coin vers la centroïde dont l'orientation fournit une information caractérisant le point d'intérêt.

• **Le descripteur BRIEF**

BRIEF est un descripteur récent qui utilise de simples tests binaires entre pixels dans une image lissée. Ses performances sont similaires à celles du *SIFT*, notamment en ce qui concerne la robustesse à l'illumination et aux distorsions perspectives.

2.4 Résultats expérimentaux du traitement d'images

Cette section détaillera les résultats expérimentaux des techniques de traitements d'image présentés plus haut. L'acquisition des images pour le traitement a été fait avec une *Webcam*. L'acquisition et le traitement d'image est réalisé grâce à la bibliothèque *OPENCV* utilisé sous *Matlab*.

Le but de ces expérimentations est d'étudier les performances de chaque méthode afin de sélectionner la meilleure pour une application de vision sur le système réel.

2.4.1 La transformée de Hough

L'application de la transformée de *hough* permet la détection de lignes dans une image. L'algorithme renvoie, après traitement, les valeurs de ρ et θ de chaque ligne détectée. Pour le cas d'un suivi de ligne, ces valeurs constituent directement le vecteur des informations visuelles S du retour de vision.

La précision des informations visuelles calculée par la transformée de *Hough* est étroitement liée à la qualité de l'image traitée, en effet une image de mauvaise qualité conduit à la détection de fausses droites. Cette qualité est améliorée lors de la phase de prétraitement et qui consiste en :

- Le seuillage,
- L'élargissement et l'égalisation d'histogramme.

Un autre paramètre influent sur l'exactitude de la détection, de droite par *Hough* est le choix du seuil des pics détectés.

Plusieurs tests ont été effectués afin de déterminer les valeurs optimales de ces paramètres, les figures 2.9, 2.10 montrent les résultats obtenus pour deux seuillages différents, quand à la figure (2.11, ??) elle montre les résultats obtenus pour différentes valeurs du seuil des pics.

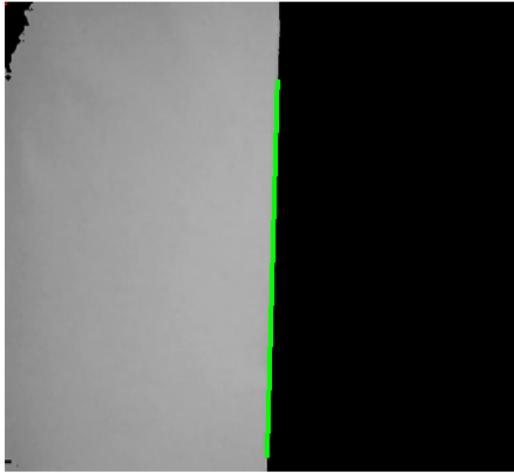


FIGURE 2.9: Détection de droite après seuillage à 0.5.

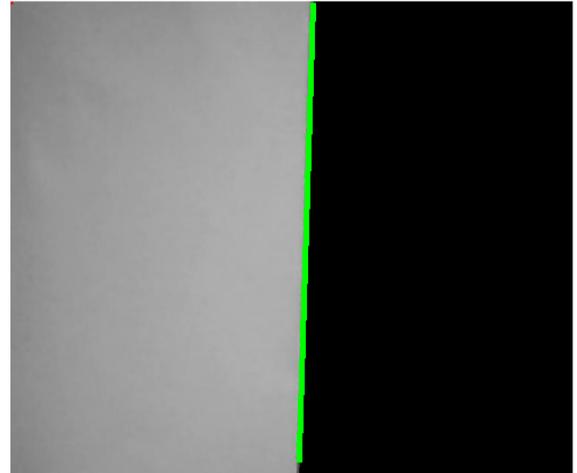


FIGURE 2.10: Détection de droite après seuillage à 0.2.

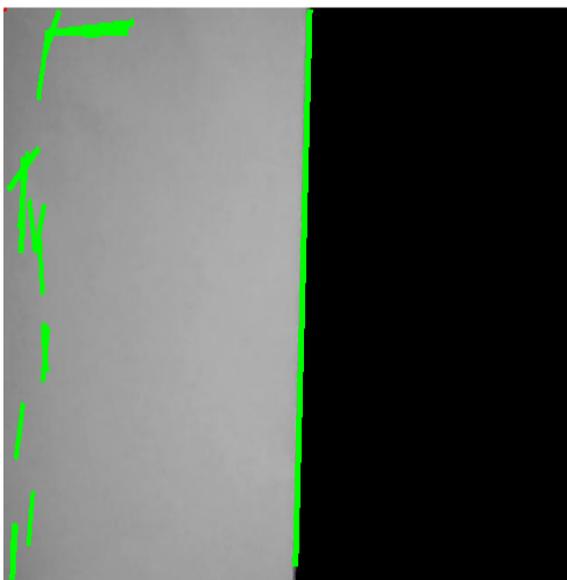


FIGURE 2.11: Détection de droite avec un faible seuil de PIC.

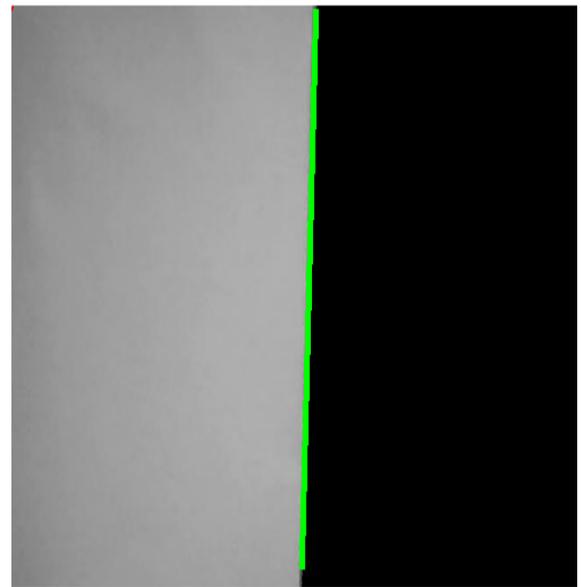


FIGURE 2.12: Détection de droite avec le seuil de PIC choisi.

2.5 Conclusion

Dans ce chapitre, on a exposé quelques généralités sur l'asservissement visuel. Ensuite, on a donné les techniques de traitement d'image appliquées à l'asservissement visuel à savoir la technique de Hough utilisée pour des applications de suivi de ligne dans

des systèmes robotiques ,on a détaillé trois algorithmes d'extraction de primitives, à savoir le SURF,le SIFT et l'ORB dont on présentera leurs application dans le chapitre trois de ce mémoire

Chapitre 3

Application des algorithmes de traitement d'image

3.1 Introduction

Dans ce chapitre, nous exposerons en premier lieu, les résultats du traitement d'images et d'extraction de primitives, puis une comparaison entre les différentes méthodes utilisées à savoir le SURF, le SIFT et l'ORB afin de designer les plus adéquates à utiliser pour l'asservissement visuel.

3.2 Calcul des indices visuels

Plusieurs tâches de positionnement peuvent être envisagées en asservissement visuel, en plus du positionnement par rapport à quatre points étudiés en simulation, on peut citer celui par rapport à une croix ou le cas le plus simple, le positionnement par rapport à un point[23].

Le cas du positionnement par rapport à un point consiste à positionner la camera de telle sorte que le centre de gravité de l'image d'un objet soit au centre de l'image. Pour ce cas, les informations visuelles sont réduites à $(u.v.)$, les coordonnées de ce centre de gravité dans le plan image.

C'est ce cas que nous avons choisit d'utiliser pour la comparaison des algorithmes d'extraction de primitives. La première méthode, qui est la plus basique, consiste à calculer le centre de gravité dans l'image de l'objet de la figure 3.2

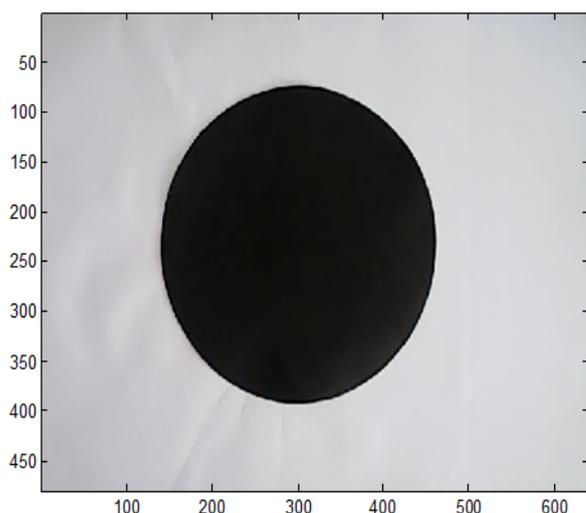


FIGURE 3.1: Image de l'objet

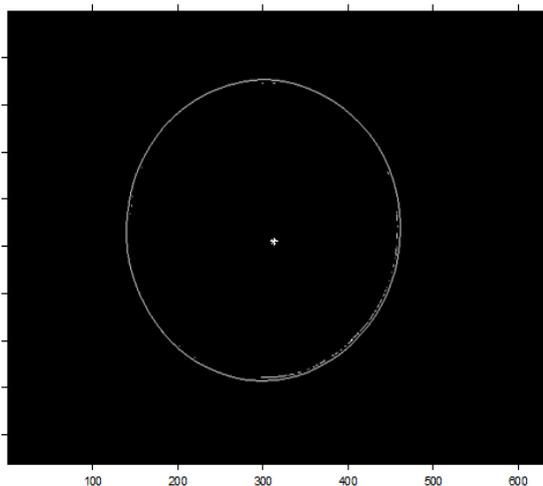


FIGURE 3.2: Extraction du centre de gravité de l'objet.

3.3 Applications de SURF,SIFT et ORB

La plupart des applications de l'asservissement visuel en robotique se basent sur une tâche de positionnement, qui doit être robuste vis à vis de l'apparence de la scène. L'approche déjà développée d'un asservissement visuel est basée sur des primitives géométriques (contours, centre de gravité, . . . etc), mais cette approche présente l'inconvénient d'être dépendante de primitives particulières qui doivent être choisies pour chaque objet et être extraites de l'image.

Pour remédier à ce problème, des primitives globales peuvent être envisagées. En effet, la position désirée peut tout simplement être définie comme étant l'image prise à la position finale de la camera, l'asservissement visuel consiste alors à faire bouger la camera afin d'aligner l'image courante avec l'image de référence. Dans ce cas, la commande de la camera est effectuée en minimisant une fonction qui n'est rien d'autre que la SSD (Sum-Squared Différence) entre l'image courante et l'image désirée. La SSD est donnée par :

$$SSD(I, I^*) = \sum (I(x) - I(x)^*)^2 \quad (3.1)$$

Avec ;

$I(x)$: l'intensité du point x de l'image courante, $I^*(x)$: l'intensité du point x dans l'image de référence.

De tel asservissements visuels peuvent être développés en se basant sur les algorithmes d'extraction de primitives. Dans notre cas nous avons utilisé les algorithmes SURF, SIFT et ORB pour effectuer l'appariement entre des images courantes acquises à partir d'une caméra et une image de référence que nous avons fixée au préalable et qui correspond à l'image désirée. Pour cela nous avons exploité la distance renvoyée après le matching des images.

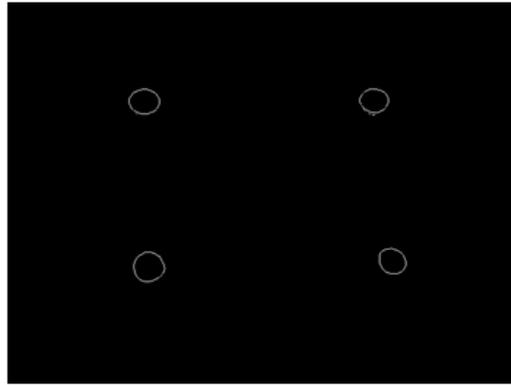


FIGURE 3.3: L'image référence.

- Résultats obtenus avec le Surf

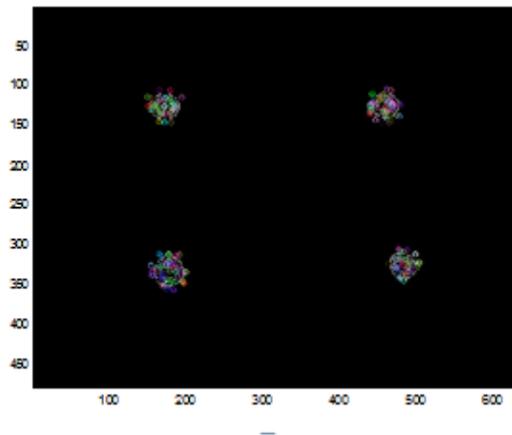


FIGURE 3.4: Détection des points d'intérêt par l'algorithme SURF.

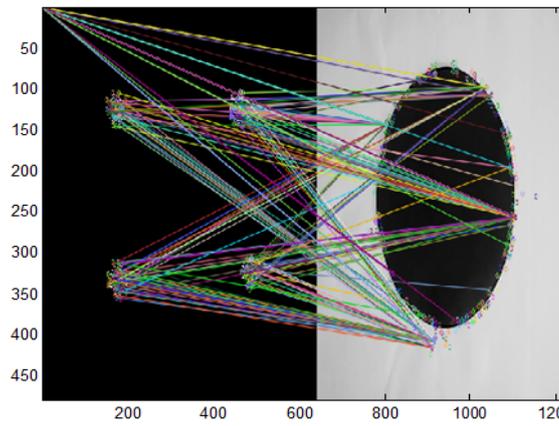


FIGURE 3.5: Matching de L'image de référence avec une image différente

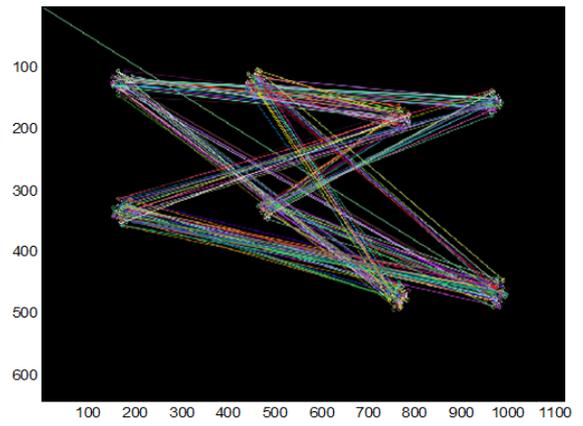


FIGURE 3.6: Matching de L'image de référence avec la même image décalée.

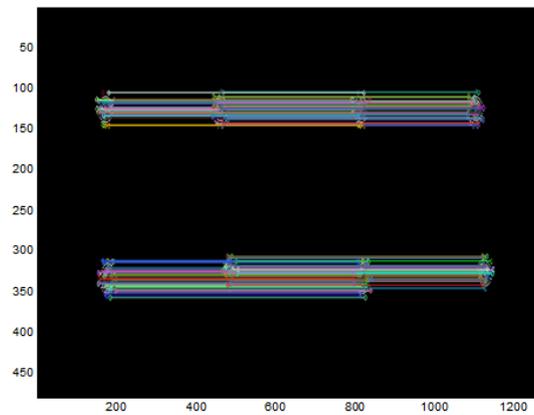


FIGURE 3.7: Matching de l'image de référence avec elle même

L'application de l'algorithme SURF a renvoyé une distance de 0.772 pour le cas 3.5 et une distance de matching de 0.0761 dans le cas 3.6 et de zero dans le cas 3.7, le temps de calcul est de 0.67s par image.

- **Résultats obtenus avec le SIFT**

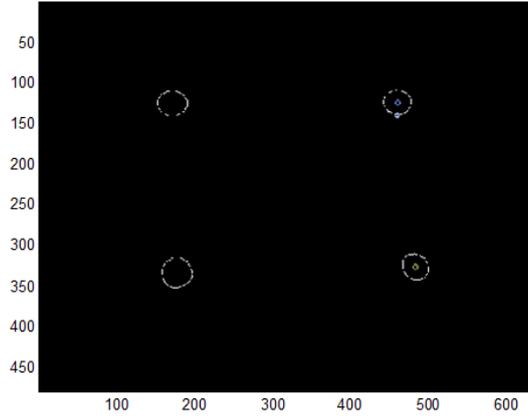


FIGURE 3.8: Detectin des points d'intérêt par l'algorithme SIFT.

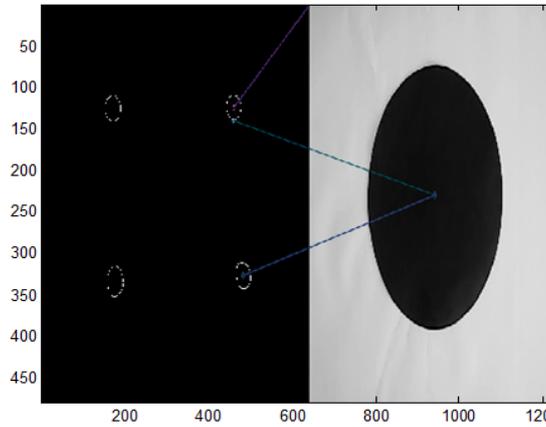


FIGURE 3.9: Matching de l'image de référence avec une autre image différente

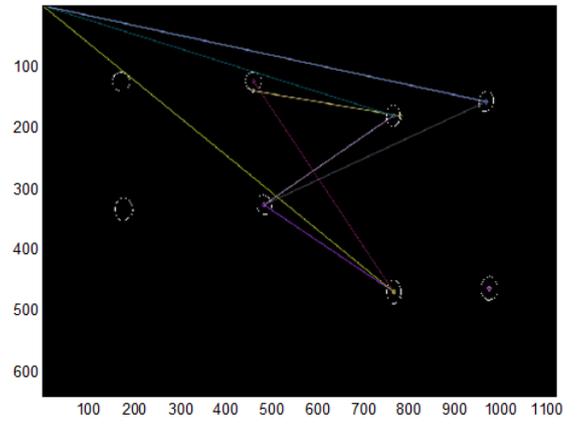


FIGURE 3.10: Matching de l'image de référence et de la référence .

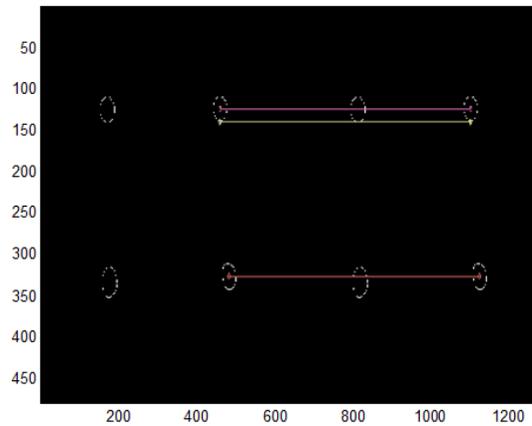


FIGURE 3.11: avec l'image référence

L'application de l'algorithme SIFT a renvoyé une distance de 561.7589 pour le cas 3.9 et une distance de matching de 17.3205 dans le cas 3.10 et de zero dans

le cas 3.11, le temps de calcul est de 0.39s par image.

- Résultats obtenus avec l'ORB

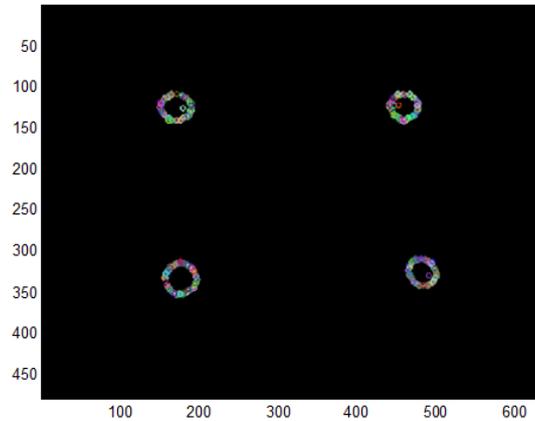


FIGURE 3.12: Detectin des points d'intérêt par l'algorithme ORB.

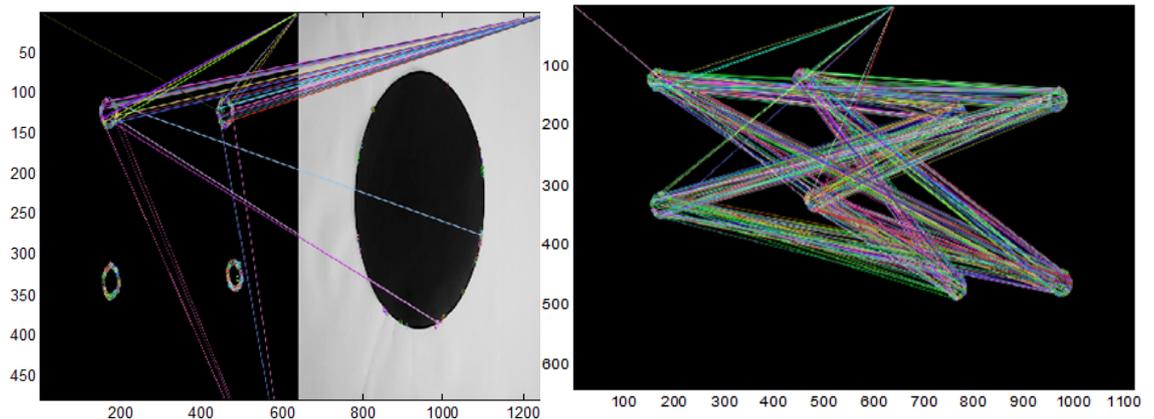


FIGURE 3.13: Matching de l'image de référence avec une image différente

FIGURE 3.14: Matching de L'image de référence avec la même référence décalée.

L'application de l'algorithme ORB a renvoyé une distance de 593.8442 pour le cas 3.13 et une distance de matching de 71.8123 dans le cas 3.14 et de zero dans le cas 3.15, le temps de calcul est de 0.24s par image.

Ces résultats montrent la puissance des algorithmes SURF SIFT et ORB. Par comparaison du temps de calcul, nous retrouvant bien la rapidité de traitement connue de l'ORB, en effet son temps de calcul est le plus réduit. Afin de mieux tester les performances de chaque algorithme, des applications sur des images prises en temps réel a partir d'une camera ont étaient effectués afin de permettre une détection d'objet. les meilleurs performances obtenues étaient avec l'algorithme SURF, en ce qui concerne le matching en temps réel. notre application a était

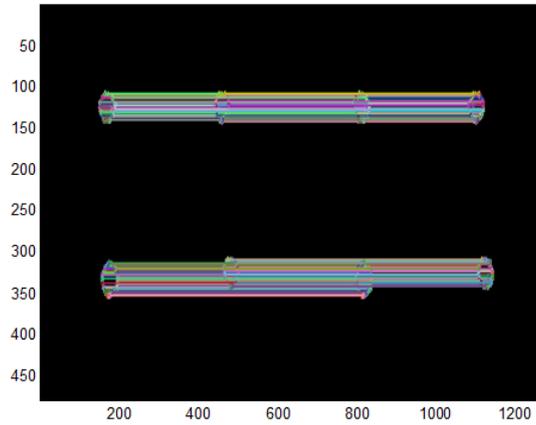


FIGURE 3.15: Matching de l'image de référence avec elle même

limité par la puissance du calculateur utilisé et par la fréquence d'acquisition des images de la camera.

3.4 Conclusion

Ce chapitre résume les différentes techniques de traitement d'image appliquées à l'asservissement visuel. on a présenter la transformée de *hough*, ainsi que trois algorithmes d'extraction de primitives SURF, SIFT et ORB, à la fin on a cité leurs applications dans l'asservissement visuel.

Conclusion

Afin de compléter notre étude sur l'asservissement visuel, nous avons traité une étape dissociée de la commande mais non moins importante : le traitement d'image.

Nous avons commencé notre étude par présenter des notions de base sur l'imagerie, on a aussi expliqué les différentes méthodes utilisées lors d'un traitement d'image ; le seuillage, la détection de contour et la segmentation.

Après ce préambule sur l'imagerie, nous avons expliqué le principe des algorithmes d'extraction de primitives utilisés, nous avons introduit les notions de détecteur de point d'intérêt et nous avons donné les méthodes d'extraction et de mise en correspondance pour chaque algorithme SURF, SIFT et ORB.

A la fin nous avons donné les résultats obtenus par application des algorithmes sur des images réelles, le dispatching de l'image courante et celle de référence. Les performances de chaque algorithme sont comparées en terme de nombre de points d'intérêt considérés, du temps de calcul et d'invariance.

Bibliographie

- [1] Marco Antonio Perez Cisneros. Intelligent models structures in visual servoing. *Thèse de Doctorat, Institut of Science and Technology WMIST*, 2004.
- [2] IS. Initiation au traitement d'images avec matlab. *Élément d'un cours d'informatique scientifique, ENSBANA*, 2003.
- [3] S Philipp J.P.Cocquerez Massons. Analyse d'images : filtrage et segmentation. *Élément d'un cours*, 1995.
- [4] J Laumand. La robotique mobile. *Edition Hermés*, 2001.
- [5] V Rouilly. Détection d'ellipse par transformation de hough. <http://www.tsi.enst.fr/>, 2000.
- [6] Rabah Ammour-Abderaouf Boussif. Commande référencée vision du robot mobile "robucar". *Thèse d'ingénieur, ENP, Algérie*, 2012.
- [7] Antoine Manzanera. Traitement d'image et vision artificielle. *Élément de cours, ENSTA, Unité d'électrotechnique*.
- [8] R K. Réalisation d'une application logiciel de commande pour saisir des objets dans un système robotique. *thèse d'ingénieur, institut nationale d'informatique, Algérie, OPTkey = RYME*.
- [9] Stéphane Breton. Une approche neuronale du contrôle robotique utilisant la vision binoculaire par reconstruction tridimensionnelle. *Thèse de Doctorat, Université de Haute-Alsace U.F.R des Sciences et Techniques, pages 21-28, 1999*, 1999.
- [10] ecole des mines. Analyse d'image. *Élément d'un cours, Génie des Procédés, centre SPIN, Ecole des Mines de Saint-Etienne*, 2001.
- [11] Nouredine Ouadah. Le control visuel appliqué dans la robotique mobile. *Thèse Doctorat, ENP*, 2011.
- [12] François Chaumette-Bernard Espiau. A new approach to visual servoing in robotics. *IEEE*, 1992.
- [13] Dame Amaury. A unified direct approach for visual servoing and visual tracking using mutual information. *Thèse de doctorat, Université Rennes*, 2010.
- [14] OpenCV. The opencv tutorials. *Release 2.4.3*, 2012.

- [15] V Kallem-M Dewan-G.D-J.P.Swenssen-Hager and N.J Cowan. based visual servoing. *IEEE-Rsj Int.Conf. on intelligent Robots and systems,IROS n'7,pages 1975-1980, San Diego, USA*, 2007.
- [16] C Collewet-E Marchard. Modeling complex luminance variations for target tracking. *IEEE.Int.Conf.on computer vision and pattern recognition CVPR n'08,Anchorage, Alaska*, 2008.
- [17] E Marchard. positionnement relatif d'une camera et d'une source lumineuse en utilisant les gradients d'intensité de l'image. *INRIA,IRISA,CRI,RENNES*, 2008.
- [18] Herbert Bay-Luc Van Gool-Tinne Tuytelaars. Surf :speed up robust features. 2006.
- [19] D.G lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 2004.
- [20] Ethan Rublee-Vincent Rabaud-Kurt-Konolige-Gary Bradsle. Orb :an efficient alternative to sift or surf. *Willow garge,Menlo Park,California*.
- [21] Abdelghani Boucheloukh. Odométrie visuelle 3d pour la localisation d'un robot mobile dans un environnement dynamique. *thèse de Magister,Ecole Militaire Polytechnique*, 2013.
- [22] K Mikolajczyk-C Schmid. Scale and affine invariant interest point detectors. *International journal of computer vision 60,63,86*, 2004.
- [23] F Chaumette. La relation commande-vision : théorie et application à des taches robotiques. *Thèse de doctorat,Université Rennes I*, 1990.