



**HAL**  
open science

# Odométrie visuelle en milieu naturel pour les robots mobiles

B. Duperal

► **To cite this version:**

B. Duperal. Odométrie visuelle en milieu naturel pour les robots mobiles. Sciences de l'environnement. 2013. hal-02599528

**HAL Id: hal-02599528**

**<https://hal.inrae.fr/hal-02599528>**

Submitted on 16 May 2020

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Université PAUL Sabatier Toulouse III

Master 2  
Intelligence artificielle, Reconnaissance de formes, Robotique  
(IRR)

## RAPPORT DE STAGE

Effectué à  
Irstea  
Unité TSCF – Equipe TEAM  
Du 18 Mars au 17 Septembre 2013

***Brice DUPERAL***

Sujet

---

# Odométrie visuelle en milieu naturel pour les robots mobiles

---

Tuteurs de stage :

- Irstea, Unité TSCF: *Bernard BENET*
- Université PAUL Sabatier, M2 IRR : *Ariane HERBULOT*

---

Université PAUL Sabatier

LAAS-CNRS  
7 Avenue du Colonel Roche  
31077 Toulouse cedex 4

Irstea de Clermont Ferrand

9 avenue Blaise Pascal  
CS 20085  
63178 Aubière

# Remerciement

Dans un premier temps, je tiens à remercier l'Irstea, et notamment Emmanuel Hugo (directeur régional), pour m'avoir chaleureusement accueilli dans les locaux du centre de Clermont-Ferrand et mis à disposition son matériel, pendant ces six mois. De plus, je remercie Jean-Michel Hénon et Michel Berducat, pour m'avoir intégré, respectivement, à leur unité de recherche et à leur équipe de recherche. Cet institut met tout à disposition pour que ses stagiaires soient dans les meilleures conditions possibles pour réaliser leurs recherches.

Je tiens à exprimer tout mes remerciements à Bernard Benet pour avoir proposé ce sujet de stage et avoir voulu être mon tuteur. Je le remercie pour avoir répondu à toutes mes questions même pendant ses vacances, pour avoir été disponible pour les expérimentations avec le robot.

De plus, je remercie Roland Chapuis (Enseignant/Chercheur à l'Institut Pascal), Thomas Feraud (Post-Doctorant à l'Institut Pascal) et Christophe Debain (Chercheur à l'Irstea) pour avoir apporté leurs avis et leurs expériences sur mes résultats, pour m'avoir conseillé sur les améliorations des algorithmes développés.

Je n'oublie pas de remercier Clément Doullens (stagiaire à l'IUT de Puy-en-Velay), qui aura partagé mon bureau pendant le stage, et qui a apporté son point de vue et ses connaissances lorsque nous devions trouver des solutions à un problème.

# Table des matières

<b>Table des visuels</b> .....	5
<b>1 Introduction</b> .....	6
<b>2 Présentation de l'Irstea</b> .....	7
2.1 A l'origine .....	7
2.2 Un acteur majeur en sciences et technologies pour l'environnement .....	7
2.3 Le centre de Clermont-Ferrand .....	7
2.4 Les trois unités de recherche du centre .....	8
2.5 Les trois équipes de l'unité TSCF .....	9
<b>3 Présentation du projet</b> .....	10
3.1 Le projet .....	10
3.2 Le matériel .....	10
3.2.1 Le robot AROCO .....	10
3.2.2 Le système de stéréovision .....	11
<b>4 Descriptif théorique de la méthode mise en place</b> .....	12
4.1 Pourquoi utiliser un système de stéréovision ? .....	12
4.2 L'appariement automatique .....	12
4.2.1 Détecteur de Harris .....	13
4.2.1.1 Recherche des points .....	14
4.2.1.2 Mise en correspondance .....	16
4.2.2 Méthode SIFT .....	16
4.2.2.1 Recherche des points .....	16
4.2.2.2 Mise en correspondance .....	19
4.3 Calibration du système de stéréovision .....	19
4.3.1 Calibration à partir du modèle sténopé (étalonnage linéaire) .....	19
4.3.2 Prise en compte des distorsions (étalonnage non linéaire) .....	21
4.4 Triangulation .....	23

4.5	Algorithme mis en place pour l'odométrie visuelle .....	25
4.5.1	Précisions sur l'algorithme d'appariement .....	25
4.5.2	Précisions sur le calcul de la distance parcourue .....	25
<b>5</b>	<b>Résultats expérimentaux.</b> .....	<b>27</b>
5.1	Points de Harris ou méthode SIFT ? .....	27
5.1.1	Les paramètres appliqués .....	27
5.1.2	Intérieur. ....	27
5.1.3	Extérieur .....	29
5.1.3.1	Milieu urbain. ....	29
5.1.3.2	Milieu agricole avec tracteur .....	30
5.1.3.3	Milieu naturel. ....	31
5.1.4	Conclusion sur les méthodes .....	32
5.2	Validation de la calibration .....	32
5.3	Mise en place de l'odométrie visuelle .....	34
5.4	Influence de la résolution des caméras sur la précision de la triangulation ...	35
5.5	Qualité caméra .....	35
<b>6</b>	<b>Conclusion</b> .....	<b>36</b>
6.1	Bilan du travail effectué .....	36
6.2	Bilan personnel .....	36
6.3	Perspectives .....	36
<b>Bibliographie</b>	.....	<b>37</b>

# Table des visuels

Figure 1 : Carte des centres de l'Irstea .....	7
Figure 2 : Images des sites du campus des Cézeaux et de Montoldre .....	8
Figure 3 : Le robot AROCO .....	11
Figure 4 : Le système de stéréovision .....	11
Figure 5 : Matrice Hessienne .....	12
Figure 6 : Formule de la gaussienne .....	13
Figure 7 : Moyenne du changement d'intensité E .....	13
Figure 8 : Schéma de l'influence de E selon trois cas .....	14
Figure 9 : Développement de Taylor .....	14
Figure 10 : Schéma simplifié de l'analyse des valeurs propres .....	15
Figure 11 : Calcul du déterminant et de la trace .....	15
Figure 12 : Zero Normalize Cross Corelation .....	16
Figure 13 : Différence de gaussienne (DoG) .....	17
Figure 14 : Exemple de différence de gaussienne .....	17
Figure 15 : Sélection des points d'intérêt suivant sa 26-connexité .....	17
Figure 16 : Calcul de l'orientation .....	18
Figure 17 : Calcul du descripteur .....	19
Figure 18 : Modelé sténopé .....	19
Figure 19 : Représentation des rotations du repère caméra par rapport au repère monde .....	20
Figure 20 : Matrice de transformation caméra/image .....	21
Figure 21 : Matrice de projection perspective .....	21
Figure 22 : Transformation caméra/image avec la distorsion .....	22
Figure 23 : Expression de u et v sans la distorsion en fonction de la position et de l'orientation .....	22
Figure 24 : Expression de u et v avec la distorsion en fonction de la position et de l'orientation .....	22
Figure 25 : Linéarisation de u et v autour d'un point $\Phi_0$ .....	23
Figure 26 : Calcul de la distance pour chaque point .....	25
Figure 27 : Méthode de Harris sur l'image Intérieur .....	28
Figure 28 : Méthode de SIFT sur l'image Intérieur .....	28
Figure 29 : Méthode de Harris sur l'image du milieu urbain avec une voiture .....	29
Figure 30 : Méthode de SIFT sur l'image du milieu urbain avec une voiture .....	29
Figure 31 : Méthode de Harris sur l'image du milieu agricole avec une tracteur .....	30
Figure 32 : Méthode de SIFT sur l'image du milieu agricole avec une tracteur .....	30
Figure 33 : Méthode de Harris sur l'image du milieu naturel .....	31
Figure 34 : Méthode de SIFT sur l'image du milieu naturel .....	31
Figure 35 : Tableau récapitulatif du nombre de points dans chaque milieu .....	32
Figure 36 : Méthode de SIFT sur l'image du milieu naturel T+1 .....	32
Figure 37 : Treize images de la mire pour une caméra .....	33
Figure 38 : Matrices intrinsèques et extrinsèques pour chaque caméra ( (1) gauche, (2) droite) .....	33
Figure 39 : Tableau de comparaison des calculs de distance par rapport à la distance réelle .....	34
Figure 40 : Tableau de comparaison des résolutions .....	35
Figure 41 : Présentation des quatre images couleurs .....	36
Figure 42 : Méthode de SIFT sur les images couleurs transformées en images en niveaux de gris .....	37
Figure 43 : Tableau du nombre de points appariés entre les images couleurs transformées en images en niveaux de gris .....	37

# 1 Introduction

L'Irstea (Institut National de Recherche en Sciences et Technologies pour l'Environnement et l'Agriculture), est un organisme de recherche qui, depuis plus de 30 ans, travaille sur les enjeux majeurs d'une agriculture responsable et de l'aménagement durable des territoires, la gestion de l'eau et les risques associés, sécheresse, crues, inondations, l'étude des écosystèmes complexes et de la biodiversité dans leurs interrelations avec les activités humaines.

L'Institut travaille sur de nombreuses recherches pour automatiser les machines agricoles. Pour réaliser le guidage automatique de ces machines, on utilise un capteur GPS-RTK. Le problème rencontré est que le capteur peut avoir des pertes de signal GPS lorsqu'il se situe près de forêts et de bois, ce qui entraîne des perturbations au niveau du guidage.

L'objectif du stage est de pallier à ces pertes d'informations, et donc d'utiliser une autre méthode qui permettrait de connaître le déplacement, et qui serait en complément du GPS. La méthode qui a été choisie et qui sera développée tout au long de ce stage sera l'odométrie visuelle.

Nous avons effectué une recherche bibliographique sur les différents travaux réalisés sur l'odométrie visuelle avec un système de stéréovision dans un milieu naturel. Nous avons trouvé un rapport qui relatait l'utilisation de l'odométrie visuelle sur les rovers de l'exploration de mars[1]. Mais l'expérience n'avait pas été assez convaincante, puisque l'odométrie visuelle était trop lente, et par conséquent le robot était guidé manuellement.

Après une présentation de l'Irstea, le projet et le matériel utilisé seront décrits précisément. Ensuite, une explication théorique des algorithmes employés pour le calcul de l'odométrie visuelle sera suivie par les résultats expérimentaux obtenus. Finalement, nous concluons et donnerons quelques perspectives de ces travaux.

## 2 Présentation de l'IRSTEA

### 2.1 A l'origine

Le Cemagref, Centre national du Machinisme Agricole, du Génie Rural et des Eaux et Forêts a été créé en 1981 de la fusion du Centre Technique du GREF (CTGREF) et du Centre National d'Études et d'Expérimentations du Machinisme Agricole (CNEEMA). Il devient Irstea en Novembre 2011, afin de faire coïncider son nom avec la réalité actuelle de ses recherches, qui ont évolué, en trente ans, des problèmes de mécanisation et d'aménagement rural vers les questions agro-environnementales.

### 2.2 Un acteur majeur en sciences et technologies pour l'environnement

L'Irstea (Institut National de Recherche en Sciences et Technologies pour l'Environnement et l'Agriculture), est placé sous la double tutelle des ministères en charges de l'agriculture et de la recherche. Le positionnement de l'Institut, exprimé dans son nouveau nom depuis 2011, ses thématiques et ses défis scientifiques, sont parfaitement en phase avec les attentes de la société : eau et qualité environnementale, risques, aménagement des territoires et développement durable à travers les écotechnologies. Ses scientifiques sont hydrologues, géologues, biologistes, chimistes, physiciens, informaticiens, automaticiens, mécaniciens, économistes, sociologues, ... et travaillent en étroite imbrication entre expérimentation, méthodes théoriques et innovation technologiques. Ils interviennent au titre d'experts en appui aux politiques publiques et effectuent des recherches collaboratives avec les acteurs industriels. C'est pourquoi l'Irstea dispose de dix centres régionaux, Aix-en-Provence, Antony, Bordeaux, Clermont-Ferrand, Grenoble, Lyon, Montpellier, Nogent-sur-Vernisson, Rennes et d'une implantation en Martinique. Ses équipes pluridisciplinaires fondent leur recherche à partir des trois thématiques Eaux, Ecotechnologies et Territoires.



Figure 1 : Carte des centres de l'Irstea

### 2.3 Le centre de Clermont-Ferrand

Dirigé par Emmanuel Hugo (Directeur Régional), le centre de Clermont-Ferrand bénéficie d'une localisation bi-site : Aubières, sur le campus universitaire, et Montdore (Figure 2), où les résultats académiques et réalités de terrain peuvent être confrontés. Les recherches conduites concernent les technologies et systèmes d'information pour



les agrosystèmes, la modélisation des systèmes complexes et les dynamiques territoriales des espaces ruraux.

Ses chercheurs sont sollicités pour leurs capacités d'expertise par la sphère publique. Le centre agit également en interconnexion avec le secteur économique : ses recherches sont vectrices d'innovation. Fort d'une réelle reconnaissance régionale, fruit de son implication, il pratique une recherche pluridisciplinaire en faveur du développement des territoires.



Figure 2 : A gauche le site du campus des Cézeaux. A droite, le site de Montoldre

## 2.4 Les trois unités de recherche du centre

- **METAFORT** : Mutations d'Es activités des espAces et des Formes d'ORganisation dans les Territoires ruraux

L'ambition du projet est d'analyser la dynamique des espaces, des activités agricoles et rurales, et la coordination des acteurs dans les territoires ruraux. Les travaux s'inscrivent dans une perspective de développement territorial. Ainsi les chercheurs de l'UMR se donnent le double objectif de contribuer à la compréhension des transformations en cours dans les territoires ruraux et de participer à l'accompagnement des acteurs qui les produisent.

Le projet scientifique de l'UMR est organisé autour de trois thèmes, chacun étant pris en charge par une équipe de recherche :

- SELECT - Systèmes d'ELEVage, Coordination, Territoires ;
- EIDER - Evolution des usages, Intervention publique et Dynamique des Espaces Ruraux et Périurbains ;
- CFORT - Construction de Formes d'ORganisation Territoriale.

- **LISC** : Laboratoire d'Ingénierie pour les Systèmes Complexes

Ils sont ainsi amenés à créer un modèle de chacun des individus d'une population végétale, animale ou humaine, intégré dans un programme informatique qui simule les interactions de chaque individu avec ses pairs ou son milieu. Les simulations révèlent des effets collectifs inattendus, dont la théorie mathématique est parfois très difficile à établir. Ce passage de l'individuel au collectif, qui est au cœur des théories récentes de la complexité, se manifeste dans la plupart des dynamiques environnementales ou sociales.

Ils étudient ces comportements globaux, issus d'interactions individuelles et les modélisent. L'équipe développe un outil logiciel pour mieux conduire des

expérimentations numériques afin d'observer les comportements globaux lorsque les paramètres varient. Puis elle construit des modèles plus simples et plus facilement utilisables en pratique. Ainsi, ils peuvent calculer sur ces modèles robustes des politiques d'action qui maintiennent la viabilité ou la résilience du système.

Leur thème de recherche se nomme :

MOTIVE - Modèles, systèmes d'information et gestion viable de l'environnement.

- **TSCF** : Technologies et Systèmes d'information pour les agrosystèmes

Le projet mobilise les sciences pour l'ingénieur et les sciences et technologies de l'information et de la communication pour conduire des recherches sur les méthodes et outils pour une ingénierie des systèmes agro-environnementaux. Grâce à ses travaux de recherche technologique, elle apporte des réponses concrètes aux besoins d'une agriculture productive écologiquement responsable et de la gestion de l'environnement.

Ses activités qui relèvent du département Écotechnologies d'Irstea sont conduites dans le cadre des thèmes de recherche :

INSPIRE - Innovations technologiques pour l'agriculture durable et l'environnement ;

MOTIVE - Modèles, systèmes d'information et gestion viable de l'environnement.

## 2.5 Les trois équipes de l'unité TSCF

Dirigée par Jean-Michel Henon, l'unité TSCF se divise en trois équipes :

- COPAIN – Systèmes d'information communicants et agri-environnementaux ;
- CARAC'TERRE – Matériaux et milieux ;
- TEAM – Technologies pour l'épandage, les agro-équipements, la mobilité.

L'équipe TEAM, est l'équipe que j'ai rejoint durant mon stage. Elle participe à la conception de systèmes intelligents dans le domaine majoritairement des agroéquipements conciliant :

- Efficience (en termes de précision, vitesse de travail, autonomie...);
- Respect de l'environnement ;
- Sûreté de fonctionnement (machines/opérateurs/environnement).

Elle développe et met en œuvre des méthodologies innovantes (outils de simulation, démonstrateurs, bancs de recherche et d'essais...) permettant de faciliter les développements et de caractériser les performances de dispositifs technologiques en relation avec les acteurs de la recherche et de l'industrie.

Ses axes de recherche portent plus particulièrement sur :

- L'amélioration des appareils de fertilisation qui jouent un rôle très important dans la chaîne de production agricole et le respect de l'environnement ;
- La mobilité des engins évoluant en milieux naturels qui impacte aussi certaines composantes environnementales.

## **3 Présentation du projet**

### **3.1 Le projet**

Le capteur GPS est l'outil utilisé pour permettre aux machines agricoles de se localiser dans un environnement naturel, en mode autonome. On peut ainsi réaliser des opérations de guidage automatique des véhicules, pour accomplir des tâches agricoles. Le problème rencontré est que l'on peut avoir des pertes de signal GPS, dans ce milieu naturel (champs agricoles, bois, forêts), pendant les tâches agricoles. Ces pertes GPS peuvent être dues à différentes raisons, comme par exemple si le véhicule pénètre dans un milieu boisé, contenant beaucoup d'arbres, ou s'il y a un problème technique. Dans ce cas-là, d'autres capteurs doivent être utilisés pour compenser ces pertes GPS et permettre aux véhicules de se localiser dans l'environnement. La centrale inertielle est un capteur qui mesure les accélérations du véhicule entre deux instants, ce qui permet pour des courtes pertes GPS, de déterminer la position du véhicule, à partir de la dernière position GPS connue. Mais pour des pertes de longues durées, les données obtenues avec la centrale inertielle dérivent fortement, ce qui ne permet pas de localiser correctement le véhicule agricole. Dans ce cas, d'autres capteurs de type caméra ou télémètre doivent être utilisés, en complément du capteur GPS et éventuellement aussi de la centrale inertielle, pour localiser les véhicules. En particulier, des techniques de SLAM (Simultaneous Localization And Mapping) sont utilisées pour localiser les véhicules et obtenir aussi une cartographie de l'environnement.

Dans le cadre de ce stage, l'objectif est d'étudier les performances de la vision artificielle, pour permettre à un robot mobile de se localiser, en utilisant soit une caméra, soit un système de stéréovision, dans un environnement naturel (arbres, cultures, champs agricoles, bâtiments,...), suite à des pertes de signal GPS. La localisation de robots par vision artificielle nécessite de réaliser des opérations d'appariement de points invariants (détection d'amers, zones particulières sur les images) sur des images successives acquises à deux instants  $t$  et  $t+1$ , mais aussi sur les images acquises, à l'instant  $t$ , par les deux caméras pour le système de stéréovision. Les travaux de recherche et développement conduits, dans ce projet d'odométrie visuelle, portent sur la recherche et l'application de différentes méthodes d'appariement de points (méthode de Harris, Sift,...), sur différentes banques d'images. Une évaluation de la qualité de ces méthodes sera réalisée. La réussite de l'appariement de points, entre deux instants et aussi à l'instant  $t$  avec les deux images acquises par les deux caméras (pour le système de stéréovision) permettra de connaître le déplacement du robot, entre ces deux instants. Les tests seront réalisés sur Matlab, avant le développement d'algorithmes en langage C++, pour l'application en temps réel sur le terrain.

### **3.2 Le matériel**

#### **3.2.1 Le robot AROCO**

Le robot AROCO est le robot sur lequel nous allons tester et valider les différentes étapes du processus d'odométrie visuelle. Il possède les dimensions suivantes : 2.40 mètres de longueur, 1.40 mètres de largeur et 1.80 mètres de hauteur. Sa distance d'empattement est de 1.20 mètres. Il peut atteindre une vitesse maximale de 4 mètres/seconde.

Il est équipé d'une panoplie de capteurs :

- capteur GPS RTK ;
- centrale inertielle ;
- capteurs d'odométrie sur les roues ;
- velodyn ;
- télémètre laser
- système stéréovision ;
- système de communication wifi.

Ces capteurs sont disposés de la façon suivante sur le robot :

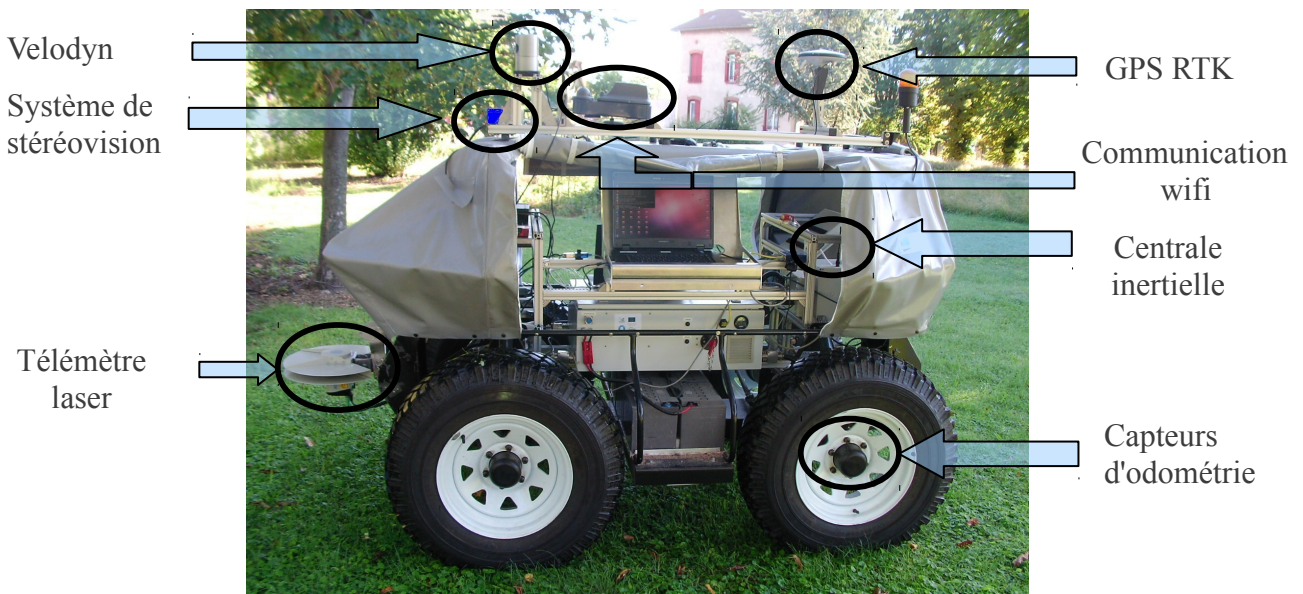


Figure 3 : Le robot AROCO

### 3.2.2 Le système de stéréovision

L'Irstea possède un système de stéréovision expérimental, pour lequel, il voulait évaluer ses performances dans différentes situations, et notamment en milieu naturel.

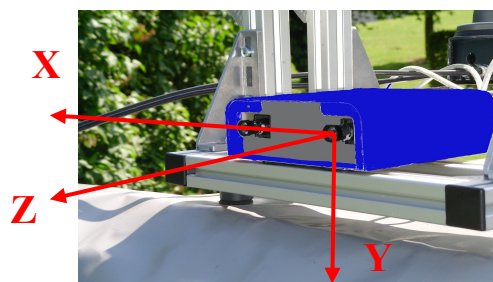


Figure 4 : Le système de stéréovision

Il est donc disposé sur le robot AROCO à une hauteur de 1.50 mètres. C'est grâce aux banques d'images acquises par le système que nous allons pouvoir travailler sur le projet.

Le système de stéréovision possède les particularités techniques suivantes :

- Niveaux de gris ;
- Quatre résolutions différentes : 320x240, 640x480, 800x600 et 1280x1024 ;
- Distance focale réglable pour chacune des caméras ainsi que l'exposition ;
- Distance entre les deux caméras en  $X = 10$  cm,  $Y$  et  $Z = 0$  cm.

## 4 Descriptif théorique de la méthode mise en place

### 4.1 Pourquoi utiliser un système de stéréovision ?

La stéréovision consiste à la localisation dans l'espace d'objets vus simultanément sous des angles différents par au moins deux caméras. Elle permet notamment de reconstruire la profondeur à partir de deux images de la même scène.

Ce principe se réfère à l'être humain, et plus précisément à nos yeux. Nous avons besoin de nos deux yeux pour déterminer un facteur d'échelle de l'environnement qu'ils perçoivent. Le facteur d'échelle est le facteur qui va nous permettre de transformer une distance sans unité en une distance en mètre.

La stéréovision se décompose en trois phases principales :

- L'association (ou appariement) : reconnaissance sur chaque paire d'images des pixels correspondant au même objet réel ;
- La calibration des caméras : détermination des paramètres intrinsèques et extrinsèques des capteurs ;
- Le calcul des coordonnées 3D des points d'un objet à partir des coordonnées 2D de ces mêmes points appariés sur les deux images.

### 4.2 L'appariement automatique

L'appariement permet de trouver le point de l'image de droite qui correspond au point de l'image de gauche, mais cela n'est pas intuitif pour l'ordinateur. Nous allons donc trouver des points d'intérêt qui correspondent généralement à une discontinuité des niveaux de gris. Ils peuvent également apparaître lors d'une modification de la structure, de la texture ou de la géométrie de l'image.

Ces détecteurs de points d'intérêt sont le fruit de longues recherches, puisque le premier à être mis en place, est celui de Beaudet en 1976 [5]. Il se base sur l'utilisation de la matrice hessienne  $H$  (Figure 5) pour déterminer la nature des points critiques d'une fonction.

$$H(f) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 x_2} & \dots & \frac{\partial^2 f}{\partial x_1 x_n} \\ \frac{\partial^2 f}{\partial x_2 x_1} & \frac{\partial^2 f}{\partial x_2^2} & \dots & \frac{\partial^2 f}{\partial x_2 x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n x_1} & \frac{\partial^2 f}{\partial x_n x_2} & \dots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}$$

Figure 5 : Matrice Hessienne

Un point critique est soit dégénéré lorsque le hessien (déterminant de la matrice hessienne) s'annule, soit non dégénéré et dans ce cas, il faut étudier sa nature (point d'extremum local ou point col) à travers le signe des valeurs propres de la matrice :

- si ces dernières sont positives, le point constitue un minimum local ;
- si elles sont négatives, il constitue un maximum local ;
- s'il y a des valeurs propres de chaque signe, le point définit un point selle (point col).

Beaudet propose en 1976, en partant de la matrice hessienne H appliquée à la fonction I(x) de l'image, de calculer une métrique proportionnelle au hessien :

$$k(x) = C \cdot \det(H(g_\sigma * I(x)))$$

où C est une constante positive et où les dérivées secondes partielles sont lissées par une gaussienne  $g_\sigma$  (Figure 6) :

$$g_\sigma(x) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{\|x\|^2}{2\sigma^2}\right)$$

Figure 6 : Formule de la gaussienne

L'auteur se concentre sur la recherche des maxima locaux. Chacun d'entre eux est défini par :

$$x_B = \underset{x}{\operatorname{argmax}} (|k(x)|)$$

Afin de déterminer les coins, deux critères de sélection sont choisis. Tout d'abord les points critiques doivent être non dégénérés, et ensuite les valeurs propres de la matrice H doivent être toutes du même signe.

De nombreux détecteurs ont fait suite à celui de Beaudet, j'en ai retenu deux puisqu'ils sont les plus utilisés actuellement. Il s'agit de Harris et SIFT.

#### 4.2.1 Détecteur de Harris

Ce détecteur a été mis en place par Harris en 1988 [2], et est basé sur l'amélioration du détecteur de Moravec (1980) [4].

La ligne directrice du détecteur de Moravec est de prendre en compte le voisinage d'un pixel (ou d'une fenêtre de pixels) d'une image et de déterminer les variations d'intensité lorsqu'une fenêtre se déplace dans diverses directions.

Nous déterminons la moyenne E du changement d'intensité (Figure 7) lorsque la fenêtre se déplace de (x,y), à l'aide de la fonction suivante :

$$E(x,y) = \sum w(u,v) |I(x+u,y+v) - I(x,y)|^2$$

Figure 7 : Moyenne du changement d'intensité E

où :

- w spécifie la fenêtre/voisinage considéré (valeur 1 à l'intérieur de la fenêtre et 0 à l'extérieur) ;
- I(u,v) est l'intensité au pixel (u,v) .

Nous allons illustrer cette fonction avec un exemple présentant trois cas (Figure 8) :

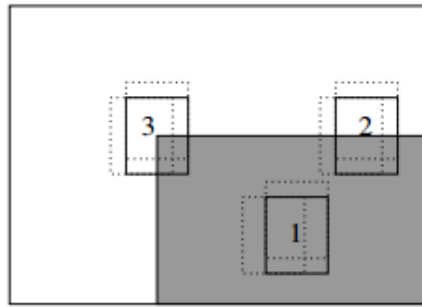


Figure 8 : Schéma de l'influence de E selon trois cas

- Le cas 1, où le déplacement de la fenêtre s'effectue dans une zone uniforme. La fonction E prendra alors de faibles valeurs dans toutes les directions (x,y) ;
- Le cas 2 est situé sur un contour rectiligne : E prendra de faibles valeurs pour les déplacements le long du contour, et de fortes valeurs pour les déplacements perpendiculaires au contour ;
- Et le cas 3 est un coin : la fonction E prendra de fortes valeurs dans toutes les directions.

Le principe du détecteur de Moravec est de rechercher les maxima locaux de la valeur minimale de la fonction E pour chaque pixel. La méthode est soumise à l'utilisation d'un seuil. Le détecteur de Moravec fonctionne donc dans un contexte limité.

#### 4.2.1.1 Recherche des points

En 1988, Harris et Stephen identifient certaines limitations, et en les corrigeant, ils en déduisent un détecteur de coins qui sera très utilisé : le détecteur de Harris.

Ils proposent de modifier la fonction E (Figure 7) afin de palier à la faible discrétisation de cette dernière. Ils utilisent par conséquent un développement de Taylor de la fonction d'intensité I :

$$I(x+u,y+v) \approx I(u,v) + x \frac{\partial I}{\partial x} + y \frac{\partial I}{\partial y} + o(x^2, y^2)$$

Figure 9 : Développement de Taylor

Pour obtenir, après identification :

$$E(x,y) = \sum w \left| x \frac{\partial I}{\partial x} + y \frac{\partial I}{\partial y} + o(x^2, y^2) \right|^2$$

Pour les faibles déplacements,  $o(x^2, y^2)$  peut être négligé et finalement la fonction E devient :

$$E(x,y) = \sum w \left| x^2 \frac{\partial I^2}{\partial x} + 2xy \frac{\partial I}{\partial x} \frac{\partial I}{\partial y} + y^2 \frac{\partial I^2}{\partial y} \right|$$

Ils choisissent par la suite de remplacer le filtre binaire  $w$  par un filtre gaussien  $g_\sigma$  (Figure 6), entraînant un débruitage du signal et donc une amélioration de la réponse du détecteur.

L'équation précédente devient :

$$E(x,y) = x^2 (I_x^2 * g_\sigma)(x,y) + 2xy (I_x I_y * g_\sigma)(x,y) + y^2 (I_y^2 * g_\sigma)(x,y)$$

Harris et Stephens mettent également en place une méthode de sélection de points d'intérêt basée sur l'analyse du comportement général de  $E$ . En effet, il est possible d'en extraire le tenseur de structure noté  $M$  :

$$E(x,y) = (x,y) M (x,y)^t$$

$$\text{avec } M = g_\sigma \begin{bmatrix} I_x^2 & I_x I_y \\ I_y I_x & I_y^2 \end{bmatrix}$$

Les valeurs propres de  $M$  correspondant aux courbures principales de la fonction  $E$ . Le détecteur de Harris se base sur l'hypothèse d'un point est dit d'intérêt si les valeurs des deux courbures sont élevées.

Ceci peut se caractériser par l'analyse des valeurs propres de  $M$ , notées  $\lambda_1$  et  $\lambda_2$  avec  $\lambda_1 \geq \lambda_2$  :

- si  $\lambda_1$  et  $\lambda_2$  sont petits ;  $E$  est presque constant dans toutes les directions : la zone sélectionnée est complètement uniforme (zone homogène) ;
- si  $\lambda_2 \gg \lambda_1$  ou  $\lambda_1 \gg \lambda_2$  : la zone correspond à un contour ;
- si  $\lambda_1$  et  $\lambda_2$  sont grands,  $\lambda_1 \sim \lambda_2$  ;  $E$  augmente dans toutes les directions : la zone caractérise un coin.

La figure suivante représente l'influence des valeurs propres de  $M$  sur le voisinage d'un point :

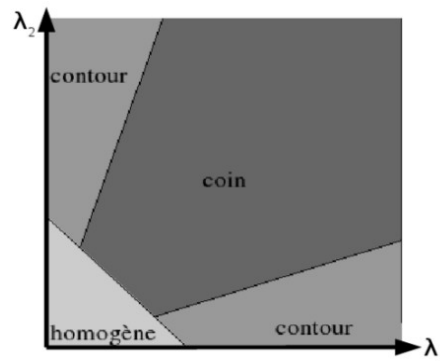


Figure 10 : Schéma simplifié de l'analyse des valeurs propres

Un point est donc considéré comme point d'intérêt si ses valeurs propres  $\lambda_1$  et  $\lambda_2$  sont élevées. Ne souhaitant pas déterminer directement ces dernières, Harris et Stephens utilisent les relations suivantes :

$$\det(M) = \prod \lambda_i \text{ et } \text{trace}(M) = \sum \lambda_i$$

Figure 11 : Calcul du déterminant et de la trace



et proposent cette mesure  $k_h$ :

$$k_h(x,y) = \det(M) - \alpha \text{tr}(M)^2$$

où  $\alpha \in [0.02, 0.06]$  est une constante et  $k_h$  est communément appelé critère de Harris. La dernière étape de leur détecteur est la recherche de maxima locaux de  $k_h$ .

#### 4.2.1.2 Mise en correspondance

Nous appliquons la méthode sur une des deux images, nous trouvons un certain nombre de points. Ces points trouvés ont une correspondance sur la seconde image qu'il faut chercher, et pour cela, nous mettons en place la formule suivante :

$$ZNCC(p_1, p_2) = \frac{\sum [I(u_1 - x, v_1 - y) - \bar{I}] [I(u_2 - x, v_2 - y) - \bar{I}']}{\sqrt{\sum [I(u_1 - x, v_1 - y) - \bar{I}]^2 \sum [I(u_2 - x, v_2 - y) - \bar{I}']^2}}$$

Figure 12 : Zero Normalize Cross Corelation

La formule est celle du *Zero Normalize Cross Corelation*, et on l'applique de la façon suivante :

On prend un point d'intérêt sur l'image 1, on fait la moyenne  $\bar{I}$  de son voisinage (avec le point d'intérêt au centre) d'une fenêtre de taille  $N \times N$ . On définit une zone de recherche dans l'image 2. Pour chaque pixel de la zone, on fait la moyenne  $\bar{I}'$  de son voisinage de même taille que l'image 1. Puis on applique le ZNCC (Figure 12), avec  $(u_1, v_1)$  le pixel central de l'image 1,  $((u_2, v_2)$  pour l'image 2). Nous obtenons un score, on se déplace d'un pixel dans la zone de recherche de l'image 2, et on réapplique le ZNCC, ainsi de suite.

Nous obtenons une matrice de score compris entre 0 et 1, plus le score est grand, plus les pixels se ressemblent.

#### 4.2.2 Méthode SIFT

Comme son nom l'indique, SIFT (Scale Invariant Feature Transform) est un algorithme qui permet la description de points d'intérêt de manière indépendante de l'échelle, ce qui le rend robuste aux variations qui peuvent survenir dans l'image. Il a été développé par David Lowe en 1999 [3].

##### 4.2.2.1 Recherche des points

Dans un premier temps, nous recherchons les extremums locaux de luminosité dans l'image.

L'algorithme utilise un espace 3D, nommé espace des échelles, auxquels les points ont pour coordonnées  $(x, y)$  et un coefficient  $\sigma$  correspondant à un facteur d'échelle. Ce facteur caractérise la taille d'une zone d'intérêt de rayon proportionnel à ce facteur.

L'espace d'échelle d'une image, notée  $L(x,y,\sigma)$  est défini par la convolution d'une fonction gaussienne de paramètre variable et de l'image originelle :

$$L(x,y,\sigma) = g_\sigma * I(x,y)$$

On crée un espace discret en calculant une pyramide gaussienne à partir de l'image. La pyramide est déterminée en calculant  $L(x,y,k\sigma)$  où  $k$  est un facteur multipliant, généralement égal à  $\sqrt{2}$ .

La détection des points d'intérêt se fait en calculant les extremums de la différence de gaussienne (DoG) entre deux niveaux successifs de la pyramide :

$$DoG(x,y,\sigma) = L(x,y,k\sigma) - L(x,y,\sigma)$$

Figure 13 : Différence de gaussienne (DoG)

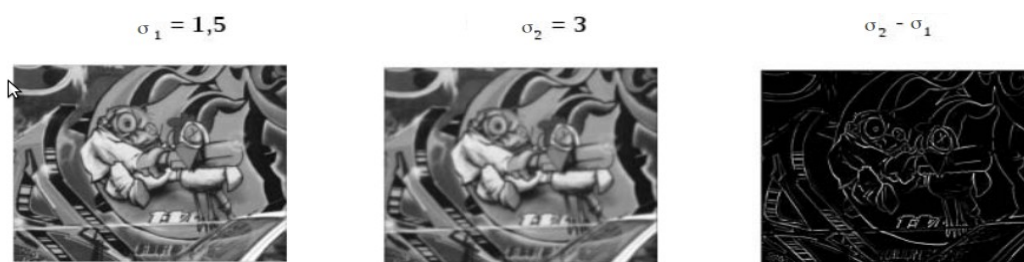


Figure 14 : Exemple de différence de gaussienne

L'avantage de la fonction DoG est sa rapidité de calcul : il suffit de faire une simple différence entre deux images pour l'obtenir.

La recherche des extremums se fait en comparant chaque pixel à ces huit pixels voisins, puis à ces neuf voisins dans les images des échelles supérieures et inférieures, soit 26 connexités (Figure 15). Si sa valeur est maximale, on considère que le pixel est un point d'intérêt. Cette détection permet de trouver un très grand nombre de points, mais ils sont assez mal localisés du fait de l'utilisation de gaussiennes. Ceci est dû au fait que le filtre gaussien rend l'image floue. Plus le coefficient du filtre est grand, plus l'image est floue.

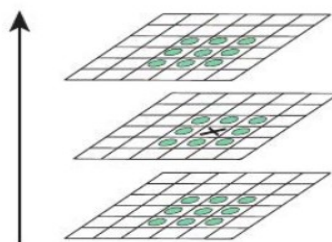


Figure 15 : Sélection des points d'intérêt suivant sa 26-connexité

Par conséquent, il faut augmenter la précision de la localisation de ces points en ayant recours à l'interpolation des coordonnées. Ceci se fait en utilisant le développement de Taylor (Figure 9) du second degré de la fonction  $L$  autour du point considéré, et en calculant le point où s'annule la dérivée de  $L$  par rapport à la position.

Il reste toujours un certain nombre de points qui sont instables, du fait d'un trop faible contraste ou de leur position sur une zone non discriminante (contours).

Pour détecter les points à faible contraste, nous observons leurs dérivées de L aux alentours du maximum. Si leurs valeurs sont trop faibles, les points ne sont pas retenus.

Concernant les points se situant sur une arête ou un contour, ils doivent être éliminés car la fonction DoG y prend des valeurs élevées, ce qui peut donner naissance à des extremums locaux instables. En effet, un contour est défini par le fait que sa courbure principale le long du contour sur lequel il est positionné est très élevée par rapport à sa courbure dans la direction orthogonale. Les courbures se calculent à partir de la fonction hessienne H (Figure 5). Les valeurs propres de H sont proportionnelles aux rayons de courbure recherchés, dont seul le rapport r est intéressant. On calcule la trace et le déterminant (Figure 11). Par conséquent, en adoptant un seuil rth (rth=10 dans la méthode de Lowe) sur le ratio des courbures, on pourra déterminer si le point d'intérêt est considéré comme localisé le long d'une arête ou pas.

$$\frac{tr(H)^2}{det(H)} = \frac{(r+1)^2}{r} < \frac{(r_{th}+1)^2}{r_{th}}$$

Une fois qu'il ne reste que des points d'intérêts stable, nous calculons une orientation pour chacun d'entre eux. Ce calcul est important puisque la direction trouvée sera utilisée pour le calcul des descripteurs et c'est elle qui garantira l'invariance des descripteurs par rotation.

Nous allons procéder de la façon suivante (Figure 16) :

- On estime le gradient de chaque position dans le voisinage du point d'intérêt par différences finies (norme et orientation) ;
- On calcule un histogramme des orientations dans le voisinage. Cet histogramme est pondéré par le rayon de la zone d'intérêt égale à 1.5 fois le facteur d'échelle ainsi que par l'amplitude de chaque point ;
- Les pics de l'histogramme, dont la valeur est au moins égale à 80% de la valeur maximale, sont utilisés pour créer des points supplémentaires qui ne diffèrent que par leur orientation principale.

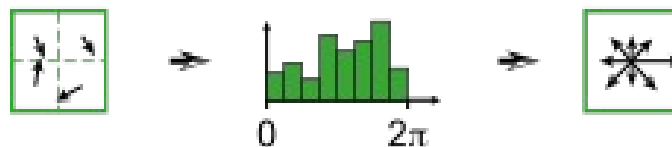


Figure 16 : Calcul de l'orientation

Cette étape achevée, les points d'intérêts sont décrits par quatre coefficients : leurs trois coordonnées dans l'espace des échelles et l'angle définissant leur orientation. On peut alors calculer les descripteurs correspondant à chacun de ces points.

Le principe du calcul des descripteurs est le suivant (Figure 17) :

- On définit pour chaque point d'intérêt, un repère local à partir de l'angle précédemment calculé.
- On considère une zone de 16x16 pixels autour du point d'intérêt subdivisée en 4x4 zones égales. Pour chacune de ces zones, on calcule un histogramme des orientations de huit intervalles compris dans l'intervalle [0 ; 2π]. Le calcul de l'histogramme se fait comme décrit au paragraphe précédent (Figure 16). On applique par ailleurs une double pondération : par l'amplitude, et par une fenêtre gaussienne centrée sur le point d'intérêt.

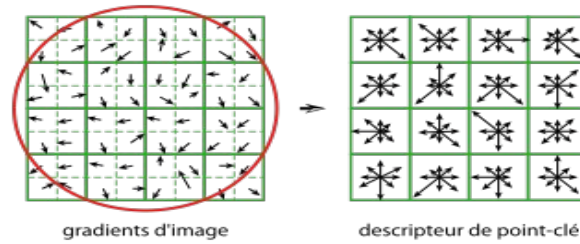


Figure 17 : Calcul du descripteur

Les histogrammes correspondant aux seize zones définies sont alors concaténés et normalisés. Pour diminuer la sensibilité du descripteur aux changements de luminosité, les valeurs supérieures à 0.2 sont mises à 0.2. Ensuite, l'histogramme est de nouveau normalisé, pour obtenir un descripteur du point d'intérêt de 128 valeurs.

#### 4.2.2.2 Mise en correspondance

Une fois que les points d'intérêt sont trouvés dans les deux images, ils sont classés dans une liste grâce à un arbre-kd (ou kd-tree pour k-dimensional tree). Cette arbre est une structure de données de partition de l'espace permettant de stocker des points, et de faire des recherches plus rapidement qu'en parcourant linéairement un tableau de points.

Puis on applique la méthode des plus proches voisins, car elle a une forte probabilité de réussite avec un temps de calcul très faible. Nous cherchons le plus proche voisin et le second plus proche voisin. Lorsque le rapport est supérieur à un seuil, nous décidons de ne pas appairer les points car nous les considérons comme ambigus.

### 4.3 Calibration du système de stéréovision

#### 4.3.1 Calibration à partir du modèle sténopé (étalonnage linéaire)

Afin de modéliser la caméra [6], nous utilisons le modèle sténopé (« pinhole ») (Figure 18), c'est à dire que nous ne conservons qu'un seul rayon.

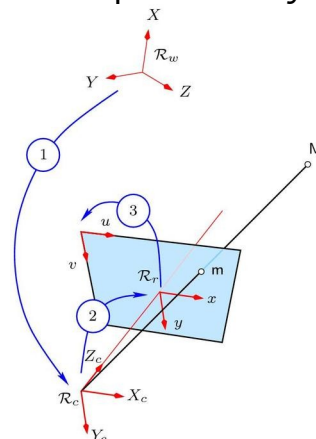


Figure 18 : Modelé sténopé

Ce modèle transforme un point 3D en un point-image, et se décompose en trois transformations successives :

- Transformation entre le repère monde et le repère caméra ((1) Figure 18) ;
- Transformation entre le repère caméra et le repère capteur((2) Figure 18) ;
- Transformation entre le repère capteur et le repère image((3) Figure 18).

La transformation entre le repère monde et le repère caméra symbolise la situation de la caméra dans le repère monde et se compose en une rotation et en une translation.

Ces informations sont stockées dans une matrice nommée matrice extrinsèque :

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$\text{avec : } T = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} \text{ et } R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

La matrice R est en faite la multiplication des trois rotations en  $\alpha, \beta, \gamma$  (Figure 19) :

$$R_\beta = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \beta & -\sin \beta \\ 0 & \sin \beta & \cos \beta \end{bmatrix}, \quad R_\gamma = \begin{bmatrix} \cos \gamma & 0 & \sin \gamma \\ 0 & 1 & 0 \\ -\sin \gamma & 0 & \cos \gamma \end{bmatrix}, \quad R_\alpha = \begin{bmatrix} \cos \alpha & -\sin \alpha & 0 \\ \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

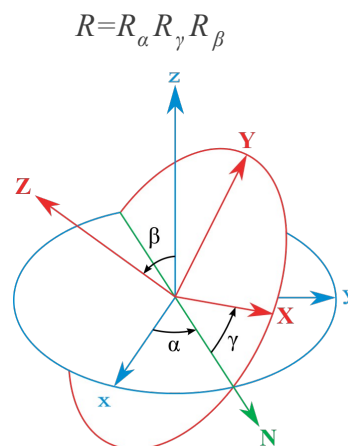


Figure 19 : Représentation des rotations du repère caméra (rouge), par rapport au repère monde (bleu)

Donc nous avons une matrice extrinsèque par caméra :

- la caméra de gauche se compose d'une matrice identité de taille 4x4, puisque nous la considérons comme notre repère ;
- concernant la caméra droite, sa matrice se compose de sa position et de son orientation par rapport à la caméra gauche. Nous devons par conséquent définir ces six inconnues.

Ensuite, la transformation entre le repère caméra et le repère capteur est une projection perspective qui donne les équations non-linéaires suivantes :

$$x = f \cdot \frac{X_c}{Z_c}, \quad y = f \cdot \frac{Y_c}{Z_c}$$

f désigne la longueur focale de l'objectif.

Ces équations transforment un point 3D dans le repère caméra en un point image. Nous pouvons les mettre dans une matrice en utilisant des coordonnées homogènes pour qu'elles soient sous forme linéaires:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix}$$

Pour finir, la transformation entre le repère capteur et le repère image convertit les coordonnées images (en unités métriques) en coordonnées images discrètes (pixels). Dans le cas où les pixels sont carrés, nous obtenons cette matrice simplifiée :

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} k_x & 0 & c_x \\ 0 & k_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Figure 20 : Matrice de transformation caméra/image

Nous multiplions les deux précédentes matrices dans une matrice dite matrice intrinsèque. Elle rassemble toutes les caractéristiques internes à la caméra, notamment :

- la focale de la caméra en u et v en pixel ;
- ainsi que les coordonnées en pixel de l'intersection de l'axe optique avec le plan image.

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha_u & 0 & u_0 & 0 \\ 0 & \alpha_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix}$$

On peut à présent écrire la transformation entre le repère objet et le repère image. Cette matrice M se nomme la matrice de projection perspective. C'est la multiplication de la matrice extrinsèque par la matrice intrinsèque :

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha_u r_{11} + u_0 r_{31} & \alpha_u r_{12} + u_0 r_{32} & \alpha_u r_{13} + u_0 r_{33} & \alpha_u T_x + u_0 T_z \\ \alpha_v r_{21} + v_0 r_{31} & \alpha_v r_{22} + v_0 r_{32} & \alpha_v r_{23} + v_0 r_{33} & \alpha_v T_y + v_0 T_z \\ r_{31} & r_{32} & r_{33} & T_z \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Figure 21 : Matrice de projection perspective

On a donc douze paramètres à définir. On peut les trouver en résolvant le problème par la méthode des moindres carrés. Mais actuellement les distorsions ne sont pas prises en compte.

#### 4.3.2 Prise en compte des distorsions (étalonnage non linéaire)

La distorsion est la déformation qui apparaît sur le bord des images. Elle est divisé en deux types :

- la distorsion radiale est due à l'asymétrie des lentilles ;
- la distorsion tangentielle est due au mauvais alignement des lentilles.

La matrice (Figure 20) est modifiée puisque nous ajoutons les termes des distorsions radiales et tangentielles :

$$\begin{aligned} u_d &= u + \Delta u \quad \text{avec } \Delta u = \Delta u_r + \Delta u_t \\ v_d &= v + \Delta v \quad \text{avec } \Delta v = \Delta v_r + \Delta v_t \end{aligned}$$

Figure 22 : Transformation caméra/image avec la distorsion

Avec l'expression des distorsions suivante :

$$\begin{aligned} \Delta u_r &= (u - u_0) \cdot (k_1 \cdot r^2 + k_2 \cdot r^4 + k_3 \cdot r^6) \\ \Delta v_r &= (v - v_0) \cdot (k_1 \cdot r^2 + k_2 \cdot r^4 + k_3 \cdot r^6) \\ \Delta u_t &= p_1 \left[ r^2 + 2(u - u_0)^2 \right] + 2p_2 (u - u_0)(v - v_0) \quad (3.9) \\ \Delta v_t &= p_2 \left[ r^2 + 2(v - v_0)^2 \right] + 2p_1 (u - u_0)(v - v_0) \\ \text{avec } r &= \sqrt{\left[ \frac{u - u_0}{\alpha_u} \right]^2 + \left[ \frac{v - v_0}{\alpha_v} \right]^2} \end{aligned}$$

r représente la distance de la partie centrale.

Par conséquent, ça rajoute des paramètres à estimer, notamment : k1, k2, k3, p1, p2.

Nous avons le vecteur d'inconnues suivant :

$$\Phi = (u_0, v_0, \alpha_u, \alpha_v, k_1, k_2, k_3, p_1, p_2, \alpha, \beta, \gamma, t_x, t_y, t_z)$$

A partir de la matrice (Figure 21), on peut déterminer les expressions suivantes :

$$\begin{aligned} u &= u_0 + \alpha_u \cdot \frac{r_{11} X + r_{12} Y + r_{13} Z + t_x}{r_{31} X + r_{32} Y + r_{33} Z + t_z} \\ v &= v_0 + \alpha_v \cdot \frac{r_{21} X + r_{22} Y + r_{23} Z + t_y}{r_{31} X + r_{32} Y + r_{33} Z + t_z} \end{aligned}$$

Figure 23 : Expression de u et v sans la distorsion en fonction de la position et de l'orientation

Ces équations (Figure 23) complètent les équations (Figure 22) :

$$\begin{aligned} u_i + \Delta u_i &= u_0 + \Delta u_i + \alpha_u \cdot \frac{r_{11} X_i + r_{12} Y_i + r_{13} Z_i + t_x}{r_{31} X_i + r_{32} Y_i + r_{33} Z_i + t_z} = P(\Phi) \\ v_i + \Delta v_i &= v_0 + \Delta v_i + \alpha_v \cdot \frac{r_{21} X_i + r_{22} Y_i + r_{23} Z_i + t_y}{r_{31} X_i + r_{32} Y_i + r_{33} Z_i + t_z} = Q(\Phi) \end{aligned}$$

Figure 24 : Expression de u et v avec la distorsion en fonction de la position et de l'orientation

On veut linéariser autour de la valeur initiale  $\Phi_0$ :

$$P(\Phi) = P(\Phi_0) + \frac{\partial P}{\partial u_0} \Delta u_0 + \frac{\partial P}{\partial v_0} \Delta v_0 + \dots + \frac{\partial P}{\partial \alpha} \Delta \alpha + \frac{\partial P}{\partial \beta} \Delta \beta + \dots + \frac{\partial P}{\partial k_1} \Delta k_1 + \dots$$

Figure 25 : Linéarisation de u et v autour d'un point  $\Phi_0$

On peut mettre l'équation de P et de Q sous un système à résoudre à partir des équations (Figure 24) et (Figure 25) :

$$u_i + \Delta u_i = P(\Phi_0) + \frac{\partial P}{\partial u_0} \Delta u_0 + \frac{\partial P}{\partial v_0} \Delta v_0 + \dots$$

$$v_i + \Delta v_i = Q(\Phi_0) + \frac{\partial Q}{\partial u_0} \Delta u_0 + \frac{\partial Q}{\partial v_0} \Delta v_0 + \dots$$

$$\Delta u_i = -(u_i - P(\Phi_0)) + \frac{\partial P}{\partial u_0} \Delta u_0 + \frac{\partial P}{\partial v_0} \Delta v_0 + \dots$$

$$\Delta v_i = -(v_i - Q(\Phi_0)) + \frac{\partial Q}{\partial u_0} \Delta u_0 + \frac{\partial Q}{\partial v_0} \Delta v_0 + \dots$$

$$\begin{bmatrix} \Delta u_i \\ \Delta v_i \end{bmatrix} = \begin{bmatrix} \frac{\partial P}{\partial u_0} & \frac{\partial P}{\partial v_0} & \dots & \frac{\partial P}{\partial t_z} \\ \frac{\partial Q}{\partial u_0} & \frac{\partial Q}{\partial v_0} & \dots & \frac{\partial Q}{\partial t_z} \end{bmatrix} \begin{bmatrix} \Delta u_0 \\ \Delta v_0 \\ \Delta t_z \end{bmatrix} - \begin{bmatrix} u_i - P(\Phi_0) \\ v_i - Q(\Phi_0) \end{bmatrix}$$

$$\Leftrightarrow V = A \cdot \Delta \Phi - L$$

On va résoudre ce système par la méthode des moindres carrés :

$$\Delta \Phi = (A^T \cdot A)^{-1} A^T L$$

Nous pouvons déterminer les quinze paramètres, qui étaient inconnus, pour calculer la triangulation.

## 4.4 Triangulation

Grâce à la calibration, nous possédons une matrice intrinsèque pour chacune des caméras, ainsi que la matrice extrinsèque du système de stéréovision. Nous pouvons à présent calculer la triangulation de chacun des points trouvés par les méthodes d'appariement. La triangulation est le fait de calculer les coordonnées 3D d'un point à partir des coordonnées 2D sur l'image gauche et droite de ce même point.

Pour obtenir la relation de passage des coordonnées 2D aux coordonnées 3D, nous commençons par obtenir la matrice M en multipliant la matrice intrinsèque par la matrice extrinsèque, pour chaque caméra ( $M_g$  pour la caméra gauche et  $M_d$  pour la caméra droite) :

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha_u & 0 & u_0 & 0 \\ 0 & \alpha_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & T_x \\ r_{21} & r_{22} & r_{23} & T_y \\ r_{31} & r_{32} & r_{33} & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$



Mais comme notre repère est la caméra gauche, la matrice extrinsèque pour  $M_d$  est celle issue de la calibration, alors que la matrice extrinsèque pour  $M_g$  est une matrice identité de taille 4x4.

Nous gardons la forme générale de la matrice et multiplions les deux matrices pour obtenir une matrice simplifiée :

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

On décompose cette matrice :

$$u = a_{11}X + a_{12}Y + a_{13}Z + a_{14}$$

$$v = a_{21}X + a_{22}Y + a_{23}Z + a_{24}$$

$$1 = a_{31}X + a_{32}Y + a_{33}Z + a_{34}$$

$$u = \frac{a_{11}X + a_{12}Y + a_{13}Z + a_{14}}{a_{31}X + a_{32}Y + a_{33}Z + a_{34}}$$

$$v = \frac{a_{21}X + a_{22}Y + a_{23}Z + a_{24}}{a_{31}X + a_{32}Y + a_{33}Z + a_{34}}$$

$$u(a_{31}X + a_{32}Y + a_{33}Z + a_{34}) = a_{11}X + a_{12}Y + a_{13}Z + a_{14}$$

$$v(a_{31}X + a_{32}Y + a_{33}Z + a_{34}) = a_{21}X + a_{22}Y + a_{23}Z + a_{24}$$

Nous mettons tout en fonction de X, Y et Z, pour la matrice  $M_g$  (u et v), comme pour la matrice  $M_d$  (u' et v') :

$$(a_{31}u - a_{11})X + (a_{32}u - a_{12})Y + (a_{33}u - a_{13})Z = a_{14} - a_{34}u$$

$$(a_{31}v - a_{21})X + (a_{32}v - a_{22})Y + (a_{33}v - a_{23})Z = a_{24} - a_{34}v$$

$$(a'_{31}u' - a'_{11})X + (a'_{32}u' - a'_{12})Y + (a'_{33}u' - a'_{13})Z = a'_{14} - a'_{34}u'$$

$$(a'_{31}v' - a'_{21})X + (a'_{32}v' - a'_{22})Y + (a'_{33}v' - a'_{23})Z = a'_{24} - a'_{34}v'$$

Puis nous rassemblons les formules précédentes sous forme matriciel :

$$\begin{bmatrix} a_{31}u - a_{11} & a_{32}u - a_{12} & a_{33}u - a_{13} \\ a_{31}v - a_{21} & a_{32}v - a_{22} & a_{33}v - a_{23} \\ a'_{31}u' - a'_{11} & a'_{32}u' - a'_{12} & a'_{33}u' - a'_{13} \\ a'_{31}v' - a'_{21} & a'_{32}v' - a'_{22} & a'_{33}v' - a'_{23} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} a_{14} - a_{34}u \\ a_{24} - a_{34}v \\ a'_{14} - a'_{34}u' \\ a'_{24} - a'_{34}v' \end{bmatrix}$$

Pour des raisons de simplicité de lecture, nous définissons des constantes pour chaque partie de la matrice précédente :

$$\varphi = \begin{bmatrix} a_{31}u - a_{11} & a_{32}u - a_{12} & a_{33}u - a_{13} \\ a_{31}v - a_{21} & a_{32}v - a_{22} & a_{33}v - a_{23} \\ a'_{31}u' - a'_{11} & a'_{32}u' - a'_{12} & a'_{33}u' - a'_{13} \\ a'_{31}v' - a'_{21} & a'_{32}v' - a'_{22} & a'_{33}v' - a'_{23} \end{bmatrix}, A = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}, B = \begin{bmatrix} a_{14} - a_{34}u \\ a_{24} - a_{34}v \\ a'_{14} - a'_{34}u' \\ a'_{24} - a'_{34}v' \end{bmatrix}$$

Pour finir nous appliquons la méthode des moindres de carrés, puisque les matrices  $\varphi$  et  $B$  sont composées de variables connues, et que la matrice  $A$  est la matrice à solutionner.

$$\begin{aligned} \varphi A &= B \\ \varphi^T \varphi A &= \varphi^T B \\ A &= (\varphi^T \varphi)^{-1} \varphi^T B \end{aligned}$$

Nous pouvons donc obtenir les coordonnées  $X, Y$  et  $Z$  de tous les points du moment que nous trouvons ses coordonnées  $u$  et  $v$  sur les images gauche et droite.

## 4.5 Algorithme mis en place pour l'odométrie visuelle

Pour résumer les différentes étapes de notre algorithme d'odométrie visuelle, nous commençons par prendre treize images d'une mire pour calibrer le système de stéréovision. Ensuite, nous prenons des images à différents instants ( $T, T+1, T+2, \dots$ ). Sur ces images acquises, nous appliquons soit le détecteur de points de Harris ou la méthode de SIFT pour trouver des points entre les images gauche et droite à l'instant  $T$ , puis trouver ces mêmes points sur les images à l'instant  $T+1$ . Nous connaissons les coordonnées 2D  $(u, v)$  et  $(u', v')$  des points aux instants  $T$  et  $T+1$ , on peut donc connaître leurs coordonnées 3D  $(X_T, Y_T, Z_T)$  et  $(X_{T+1}, Y_{T+1}, Z_{T+1})$ . A partir de ces informations, on peut calculer la distance pour chacun des points entre les instants  $T$  et  $T+1$ . On peut alors effectuer une moyenne des distances obtenues, pour déterminer la distance parcourue par le robot entre deux instants.

### 4.5.1 Précisions sur l'algorithme d'appariement

L'appariement des points n'est pas aussi simple que cité précédemment, puisque nous devons trouver des points d'intérêt à l'instant  $T$ , mais on doit aussi retrouver ces mêmes points à l'instant  $T+1$ . Par conséquent, nous avons décidé d'œuvrer de la façon suivante :

On cherche les points entre l'image gauche et l'image droite à l'instant  $T$ . Puis nous cherchons la correspondance entre les points appariés dans l'image gauche de l'instant  $T$  et l'image gauche à l'instant  $T+1$ . Nous trouvons un certain nombre de points, que nous recherchons dans l'image de droite à l'instant  $T+1$ .

### 4.5.2 Précisions sur le calcul de la distance parcourue

Concernant le calcul de la distance parcourue, nous avons obtenu les coordonnées  $X_T, Y_T, Z_T, X_{T+1}, Y_{T+1}$  et  $Z_{T+1}$  pour un point, nous calculons donc la distance avec la formule suivante :

$$Distance = \sqrt{(X_{T+1} - X_T)^2 + (Y_{T+1} - Y_T)^2 + (Z_{T+1} - Z_T)^2}$$

Figure 26 : Calcul de la distance pour chaque point

Tout au long des expérimentations, nous avons pu déterminer trois formules permettant de calculer la distance parcourue par le robot, à partir de la distance de chaque point :

- Formule 1 : Elle réalise la moyenne des distances calculées pour chaque paire de points appariés.

- Formule 2 : A partir de la moyenne calculée dans la Formule 1, nous définissons un intervalle  $I$  autour de celle-ci ( $I = [moyenne - n; moyenne + n]$ ). Nous recherchons toutes les distances comprises dans cette intervalle, et nous calculons la moyenne de ces distances.

- Formule 3 : Nous attribuons des coefficients en fonction de la profondeur  $Z$  des points. Par exemple, si le point est à une profondeur de moins de 4 mètres, nous lui attribuons un coefficient de 2. Par contre, si la profondeur est à plus de 6 mètres, nous lui attribuons un coefficient de 0.5, pour atténuer son effet négatif sur le calcul. Si la profondeur est comprise entre 4 et 6 mètres, nous lui attribuons un coefficient de 1. Pour obtenir la distance parcourue par le robot, nous calculons la somme des distances multipliées par leur coefficient, le tout divisé par la somme des coefficients.

## 5 Résultats expérimentaux

### 5.1 Points de Harris ou méthode SIFT ?

A présent, nous allons appliquer les deux méthodes sur les mêmes images, et déterminer laquelle est la plus robuste. Les images ont été prises par le système de stéréovision placée sur le robot AROCO :

- dans un bâtiment pour un premier temps ;
- dans un milieu urbain ;
- puis dans un milieu agricole avec tracteur ;
- et finalement dans un milieu naturel.

Toutes les figures suivantes représentent l'appariement des points entre l'image gauche et l'image droite. Les méthodes de Harris et SIFT trouvent un grand nombre de points d'intérêts dans la première phase de leur algorithme. Dans la seconde phase, ces deux algorithmes cherchent à appairer ces points entre les images gauche et droite. Les points effectivement appariés sont représentés sur les images. Par souci de clarté, les points non appariés ne le sont pas.

Chaque paire de points appariés sur les images gauche et droite est visualisé avec la même couleur.

#### 5.1.1 Les paramètres appliqués

L'algorithme de Harris a pour paramètres :

- la taille du patch lors de la corrélation : 11x11 ;
- la taille de la zone de recherche dans la seconde image : 250x30 ;
- le seuil du score de corrélation : 98%.

Pour la taille du patch, nous avons choisi 11x11, car c'est un bon compromis entre le temps de calcul et le nombre d'information suffisante pour calculer un score de corrélation.

Concernant la taille de la zone de recherche dans la seconde image, nous avons remarqué que la différence en  $v$  entre les coordonnées de points appariés est toujours inférieure à 20 pixels, donc nous avons choisi de mettre une marge de sécurité de 10 pixels, soit 30 pixels de recherche. Concernant les 250 pixels de recherche, ça correspond à un tiers de la taille d'une image en résolution 800x600.

L'algorithme de SIFT n'a qu'un seul paramètre :

- le seuil du ratio entre les deux plus proches voisins du point d'intérêt : 75%.

#### 5.1.2 Intérieur

Nous avons choisi de faire des images dans un bureau dans un premier temps, pour pouvoir attester du bon fonctionnement des méthodes de recherche de points, mais aussi d'appariement. Le bureau est un milieu entièrement façonné par l'homme, ce qui signifie qu'il sera composé de nombreuses formes géométriques avec des contours bien définis et des angles. De plus, la luminosité est artificielle et sans changement d'intensité.

Nous commençons par appliquer Harris sur la paire d'images (Figure 27) :

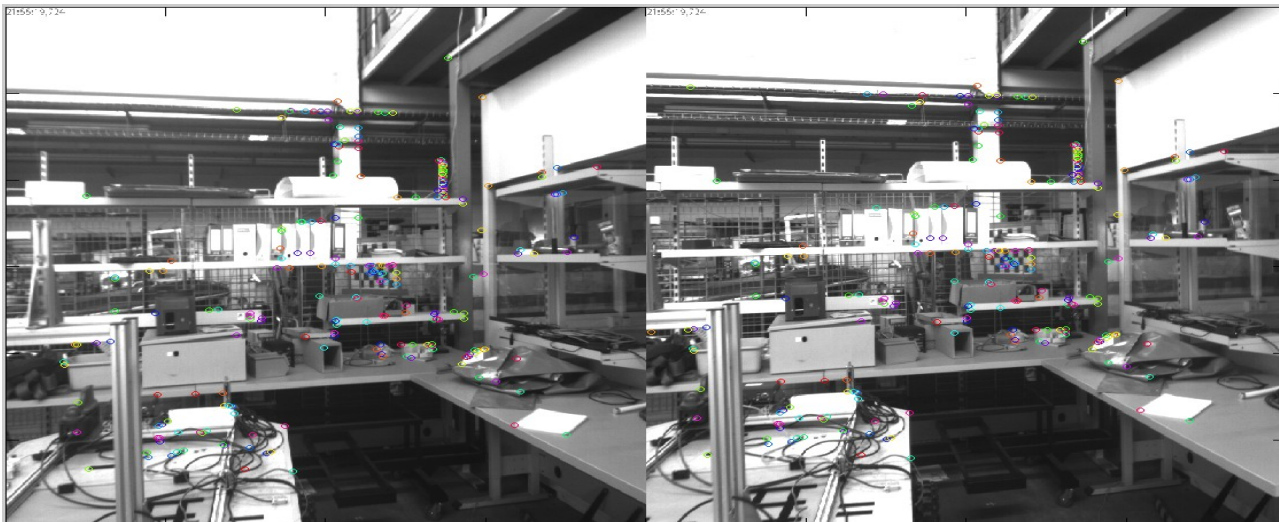


Figure 27 : Méthode de Harris sur l'image Intérieur

Nous pouvons observer que Harris réussit à appairer un certain nombre de point, soit 216 dans ce cas là. Dans l'ensemble, les appariements semblent corrects, sauf en haut de l'image où le motif est répétitif. L'erreur est due au fait que la corrélation prend les intensités des pixels pour comparer avec le points d'intérêt trouvé, et que dans une structure répétitive, toutes les intensités des pixels des motifs se ressemblent, il ne peut donc pas les différencier.

Passons à présent à l'application de la méthode SIFT (Figure 28) :

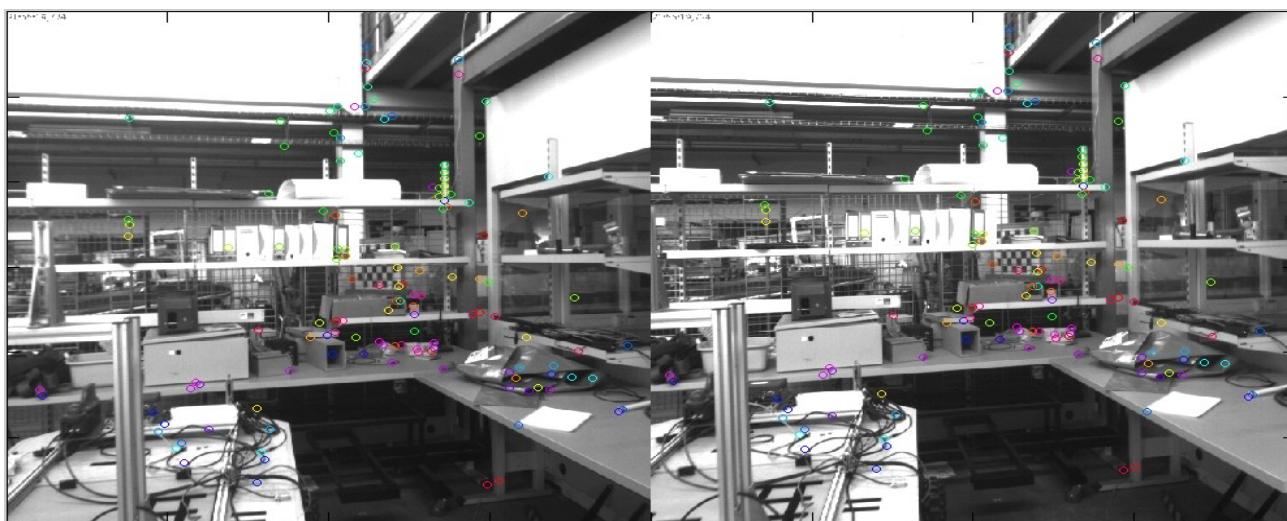


Figure 28 : Méthode de SIFT sur l'image Intérieur

L'algorithme permet de trouver 139 points. Les appariements paraissent corrects avec une précisions de un à deux pixels près. Les motifs répétitifs ne posent pas de problème à cette méthode.

Il n'y a pas de réelle comparaison entre le nombre de points trouvés entre les deux méthodes. Car dans le détecteur de Harris, il y a un seuil qui détermine le pourcentage de ressemblance minimum pour appairer deux points. Dans la méthode de SIFT, le seuil permet d'éviter de faux appariements dus au bruit.

### 5.1.3 Extérieur

Passons à présent au monde extérieur, où la lumière du soleil varie selon les nuages, et où l'ombre provoque de brutal changement d'intensité.

#### 5.1.3.1 Milieu urbain

Le milieu urbain est composé de bâtiments, de véhicules, de routes,... construits par l'homme, donc généralement une grosse quantité de formes géométriques. Ces formes facilitent la recherche de point d'intérêt.

Comme pour l'intérieur d'un bâtiment, nous commençons par appliquer la méthode de Harris sur une image d'une voiture garée dans un bâtiment (Figure 29) :



Figure 29 : Méthode de Harris sur l'image du milieu urbain avec une voiture

Il y a 95 points appariés. Nous pouvons observer que les points sont concentrés dans certaines zones, comme la roue. Une nouvelle fois, le détecteur de Harris est perturbé par la structure répétitive de la grille avant de la voiture, ce qui crée des erreurs d'appariement.

Passons à présent à SIFT (Figure 30) :



Figure 30 : Méthode de SIFT sur l'image du milieu urbain avec une voiture

Nous trouvons 106 points. Les points appariés se situent sur toute l'image, ils ne sont pas concentrés que dans un seul élément de l'image. Là encore, il ne semble pas y avoir de faux appariements. Il y a un motif répétitif en haut de l'image, mais SIFT ne préfère pas appairer les points, pour ne pas commettre d'erreur. Malgré l'intensité très forte de la lumière située au fond, cette méthode n'est pas perturbée.

### 5.1.3.2 Milieu agricole avec tracteur

Nous allons aussi faire un essai sur des images d'un tracteur dans une ferme. Commençons par la méthode de Harris (Figure 31) :



Figure 31 : Méthode de Harris sur l'image du milieu agricole avec une tracteur

Il y a seulement 8 points qui ont été appariés. Nous commençons à voir les limites de Harris. Lorsqu'il y a un trop fort contraste entre le ciel et les arbres, tous les points sont localisés sur cette ligne de contraste. Et la corrélation calcule le score de ces contrastes, qui sont plus ou moins les mêmes, et provoque de faux appariements.

Observons si la méthode SIFT rencontre les mêmes problèmes (Figure 32) :



Figure 32 : Méthode de SIFT sur l'image du milieu agricole avec une tracteur

La méthode de SIFT trouve 229 points appariés. Contrairement à Harris, SIFT permet d'apparier davantage de points, situés sur toute l'image. La précision entre les points de l'image gauche et droite semble très grande (un ou deux pixels près). Il y a des points appariés sur la ligne de contraste ciel/arbre, mais ils sont bien appariés.

### 5.1.3.3 Milieu naturel

Le milieu naturel est composé d'éléments informes. Il n'y a pas d'angle droit dans un arbre ou dans la pelouse, par comparaison avec un objet de type bâtiment. Pour se raccorder à notre projet, nous allons tester les deux algorithmes sur un ensemble d'arbres situé dans un parc.

Nous commençons par tester l'algorithme de Harris (Figure 33) :



Figure 33 : Méthode de Harris sur l'image du milieu naturel

Nous trouvons 50 points appariés avec Harris. Tous les points se situent sur le tronc de l'arbre, il n'y a aucun point dans le feuillage ou dans l'herbe. Les points sont mieux appariés que l'image du tracteur, puisqu'il ne semble pas y avoir de faux appariements.

Testons à présent l'algorithme de SIFT sur cette image du parc (Figure 34) :

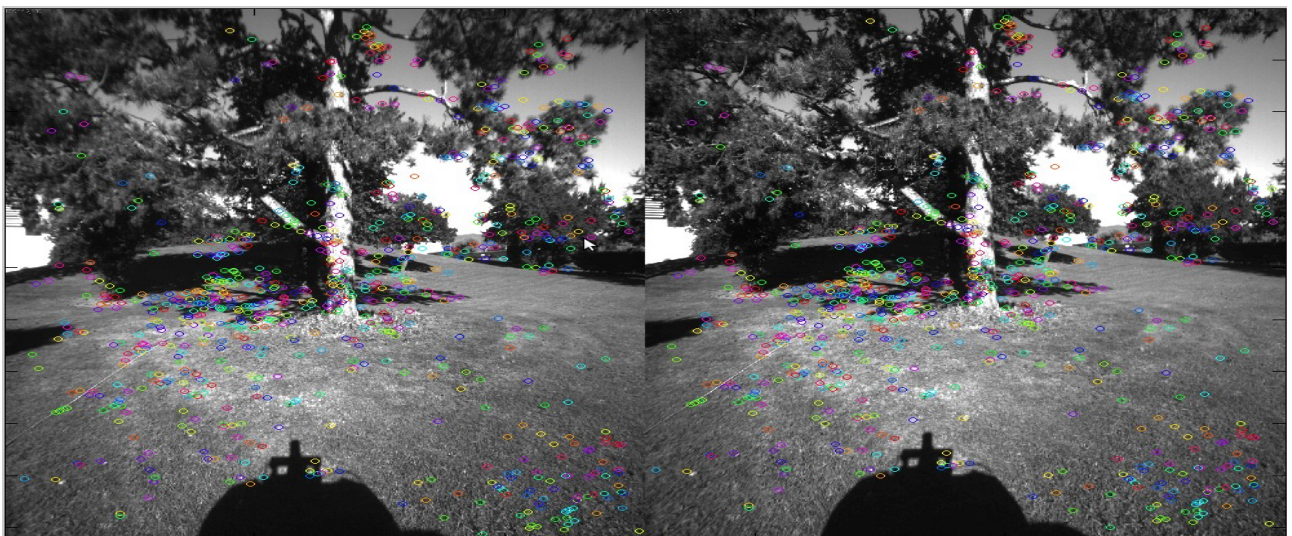


Figure 34 : Méthode de SIFT sur l'image du milieu naturel



L'algorithme de SIFT trouve beaucoup plus de points que Harris car il en trouve 833. Ces points sont dispersés dans toute l'image, et pas seulement sur le tronc. Les appariements paraissent corrects.

### 5.1.4 Conclusion sur les méthodes

Nous avons testé l'algorithme de Harris et de SIFT dans quatre milieux différents. L'impression qui en ressort est que SIFT est beaucoup plus robuste puisque :

- il trouve plus de points que Harris dans les milieux agricole et naturel (Figure 35);
- ses points sont bien appareillés, il n'y a que très peu de faux appariement.

Milieu de l'image/Nombre de points	Méthode de Harris	Méthode de SIFT
Intérieur	216	139
Urbain	95	106
Agricole	8	229
Naturel	50	833

Figure 35 : Tableau récapitulatif du nombre de points dans chaque milieu

Plus il y aura de points appariés à l'instant T, plus nous aurons de chance de pouvoir appairer des points entre les instants T (Figure 34) et T+1 (Figure 36).

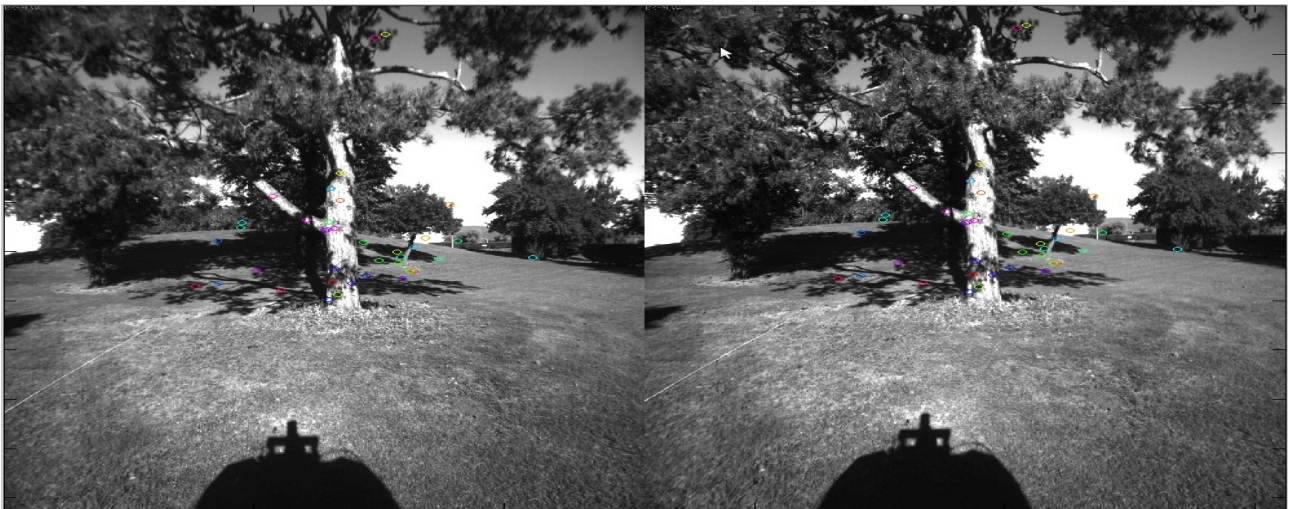


Figure 36 : Méthode de SIFT sur l'image du milieu naturel T+1

Nous pouvons observer que nous trouvons 833 points appariés à l'instant T, et que nous ne trouvons que 39 points appariés à l'instant T+1. Ces points serviront pour l'odométrie visuelle.

Nous choisissons d'utiliser la méthode de SIFT pour la suite du projet.

## 5.2 Validation de la calibration

Pour déterminer les inconnues qui nous permettront de mettre en place la triangulation dans l'étape suivante, nous devons faire une calibration de chaque caméra dans un premier temps, puis du système dans un second temps.

Pour que cela se réalise, nous devons prendre des images d'une mire dans treize positions différentes :

- Une image de la mire dans son ensemble, perpendiculaire à l'axe optique ;
- Quatre positions fronto-parallèles : modifie seulement un angle de la mire, yaw <0, yaw >0, pitch >0 et pitch <0 ;
- Quatre images où la mire est dans les quatre coins ;
- Et les quatre dernières sont très utiles pour vérifier la distorsion avec le respect de la perspective.



Figure 37 : Treize images de la mire pour une caméra

Dans toutes les images, la mire doit être vue entièrement par les deux caméras.

Une fois les images acquises, nous utilisons un plugging de Matlab qui se nomme *calib*, qui réalise les calculs de détermination des inconnues.

On calcule les paramètres intrinsèques de chaque caméra dans un premier temps, en pointant les quatre coins de la mire sur chaque image. Il nous ressort la matrice intrinsèque de chaque caméra.

Puis dans un second temps, la fonction détermine les paramètres extrinsèques à partir des deux matrices intrinsèques et des treize images des deux points de vue (donc vingt-six images). Nous avons donc la position et l'orientation de la caméra gauche par rapport à la caméra droite. (Figure 38)

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} 733 & 0 & 638 & 0 \\ 0 & 735 & 506 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (1)$$

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} 737 & 0 & 645 & 0 \\ 0 & 739 & 488 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0.0015 & -0.0003 & -99.6 \\ -0.0015 & 1 & 0.0017 & 0.46 \\ 0.0004 & -0.0017 & 1 & 0.13 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (2)$$

Figure 38 : Matrices intrinsèques et extrinsèques pour chaque caméra ((1) gauche, (2) droite)

Pour vérifier si la calibration est satisfaisante ou pas, nous savons que la distance entre les deux caméras est de -100mm en X et un écart de moins de 1 centimètre en Y et Z. Nous obtenons 99.6 millimètres en X, 0.46 millimètre en Y et 0.13 millimètre en Z, ce qui est très raisonnable. Concernant les rotations, elles sont très proches de 0 degré. En conclusion, cette calibration est satisfaisante.

### 5.3 Mise en place de l'odométrie visuelle

Pour vérifier si l'algorithme mis en place était valide, nous avons effectué le processus suivant :

- Nous réglons la distance focale des caméras ce qui améliore la netteté de l'image. De plus, nous modifions la variable d'exposition à la lumière pour que l'image soit la plus exploitable possible ;
- Puis, nous prenons les treize images de la mire nécessaires à la calibration. Nous effectuerons la calibration dans un second temps ;
- Ensuite, nous plaçons le robot où nous prenons une paire d'images. Nous obtenons les images de l'instant  $T_0$ . A l'aide du GPS équipé sur le robot, nous prenons la position pour l'instant  $T_0$ . Nous déplaçons le robot, et prenons les images, ainsi que la position GPS, à l'instant  $T_1$ . Nous répétons l'opération une vingtaine de fois. Nous vérifions que toutes les images et les données GPS soient bien sauvegardées, et nous réalisons une nouvelle série. Pour nous assurer d'avoir le déplacement du robot, en cas d'erreur sur la position GPS, nous le mesurons avec un mètre en même temps ;
- Une fois que les banques d'image sont acquises, nous exécutons l'algorithme de SIFT pour déterminer les coordonnées  $((u_1, v_1)_T, (u_2, v_2)_T, (u_1, v_1)_{T+1}$  et  $(u_2, v_2)_{T+1})$  pour tous les points appariés entre les images à un instant  $T$  et les images à un instant  $T+1$ . Nous réalisons la triangulation pour chaque quadruplé  $(u_1, v_1, u_2, v_2)$  à l'instant  $T$  pour obtenir les coordonnées  $X_T, Y_T$  et  $Z_T$ . Nous faisons de même pour obtenir les coordonnées  $X_{T+1}, Y_{T+1}$  et  $Z_{T+1}$  ;
- Nous calculons la distance à partir des coordonnées  $(X_T, Y_T, Z_T, X_{T+1}, Y_{T+1}, Z_{T+1})$  avec la formule (Figure 26), pour tous points appariés ;
- Finalement, nous réalisons les trois calculs (partie 4.5.2), et nous les comparons avec la mesure réelle du déplacement du robot.

Distance réelle	Formule 1	Formule 2	Formule 3
120	216.9	207.4	142.3
90	209	211.3	92.3
80	157.9	155.9	90.7
160	274.9	270.3	154.8
60	153.4	156.8	62.9
100	156.9	158.5	98.9

Figure 39 : Tableau de comparaison des calculs de distance par rapport à la distance réelle

Nous pouvons observer que les formules 1 et 2 ne sont pas du tout exactes, puisque nous trouvons le double de la mesure réelle. On peut voir la faiblesse de la formule 2 qui est basée sur la formule 1. Si la formule 1 est trop éloignée de la distance réelle, l'intervalle autour de la formule 1 sera faux, donc la formule 2 sera fausse. Concernant la formule 3, nous pouvons voir que ses valeurs sont très proches des valeurs réelles.

## 5.4 Influence de la résolution des caméras sur la précision de la triangulation

Nous avons travaillé avec deux résolutions 800x600 et 1280x1024 pour les caméras du système de stéréovision. Nous avons remarqué que la précision de la triangulation était meilleure avec la plus grande résolution.

A titre d'exemple, le tableau (Figure 40) ci-dessous, présente la comparaison entre les deux résolutions pour trois distances mesurées entre un objet et le système de stéréovision.

Distance réelle/Résolution	800x600	1280x1024
268	265	267.5
360	307	357
465	356	469
565	712	578

Figure 40 : Tableau de comparaison des résolutions

A partir de 3.60 mètres, nous remarquons que la précision de la triangulation avec la résolution 800x600 n'est pas correcte. Alors que la précision de la triangulation avec la résolution 1280x1024 commence à ne plus être exacte à partir de 5.65 mètres.

## 5.5 Qualité caméra

Nous avons remarqué que les caméras en niveaux de gris du système de stéréovision utilisées sont très influencées, en milieu extérieur par les conditions ensoleillement (soleil, nuage, ombre ...). Nous avons testé la méthode d'appariement SIFT en utilisant une caméra couleur que nous jugeons de meilleure qualité, moins sensible à la lumière extérieure. La figure (Figure 41), page suivante, présente quatre images acquises par cette caméra couleur à quatre instants différents.



Figure 41 : Présentation des quatre images couleurs

Nous avons transformé les images couleur en images en niveaux de gris, sur lesquelles nous avons appliqué la méthode de SIFT (Figure 42).



Appariement de l'image 1 avec l'image 2



Appariement de l'image 2 avec l'image 3



Appariement de l'image 3 avec l'image 4

Figure 42 : Méthode de SIFT sur les images couleurs transformées en images en niveaux de gris

Appariement image	Nombre de points
Image1-2	27
Image 2-3	61
Image 3-4	80

Figure 43 : Tableau du nombre de points appariés entre les images couleurs transformées en images en niveaux de gris

Ce test a été réalisé avec une caméra couleur, en s'intéressant à l'appariement de points entre différents instants ( $t$ ,  $t+1$ ,  $t+2$ ,...), mais il n'est pas très concluant dans ce cas-là. De nouveaux tests, à grande échelle, seront réalisés ultérieurement, en utilisant un nouveau système de stéréovision couleur, avec des caméras de meilleure qualité que les caméras en niveau de gris utilisés au cours de ce stage.

## 6 Conclusion

### 6.1 Bilan du travail effectué

En conclusion, notre algorithme pour l'odométrie visuelle fonctionne, mais nous sommes contraints par la qualité du système de stéréovision actuel. La triangulation est précise pour tous les objets qui sont situés à moins de 4 mètres de la caméra.

L'algorithme a de grandes chances de fonctionner dans une forêt dense, mais dans un bois éparse, le résultat ne sera pas optimal.

### 6.2 Bilan personnel

Dans un premier temps, ces six mois de stage m'ont permis d'enrichir mes connaissances sur :

- Les systèmes de stéréovision, et notamment la calibration. Cette partie était très floue avant de commencer le stage, car l'application, lors des séances de travaux dirigés, avait été entièrement faite par un industriel, avec le logiciel de son entreprise.

- Les méthodes d'appariement. Nous avons eu un cours sur les méthodes de Harris et de Sift, mais pas de travaux pratiques sur cette partie. Le stage m'a permis d'approfondir ces méthodes, de les coder et de les tester.

Ensuite, j'ai pu découvrir, par l'intermédiaire de l'Irstea, les méthodes de travail dans un laboratoire de recherche. L'idée de la journée d'un chercheur était vague, sans réelle précisions. A présent, j'ai une image plus précise de son travail, grâce à l'ensemble du personnel de l'Irstea.

Finalement, j'ai pu rencontrer de nombreuses personnes avec plus ou moins d'expérience et avec des parcours différents dans les domaines du traitement d'images et de la robotique. Nous avons apporté nos savoirs et nos idées sur nos différents projets.

### 6.3 Perspectives

Pour la suite du projet, il faudrait comparer notre système de stéréovision actuel avec un système de stéréovision de meilleure qualité d'image, avec un système de réglage de l'auto-exposition robuste à la lumière naturelle, et avec une résolution d'image plus grande.

Pour les comparer, il faudrait les monter tous les deux sur le robot, pour qu'ils puissent acquérir les mêmes images en même temps, avec une exposition à la lumière du soleil identique.

Une fois que le système de stéréovision, ainsi que l'algorithme d'odométrie seront validés, il faudra coder le programme en C++ pour pouvoir l'implémenter dans un micro processeur. Ensuite, il faudra tester si l'algorithme est assez rapide pour pouvoir mesurer le déplacement de robot en temps réel. S'il n'est pas assez rapide, il faudra sûrement orienter la méthode d'appariement de points vers la méthode SURF (Speed Up Robust Feature), puisqu'elle est plus rapide que SIFT.

## Bibliographie

- [1] MAIMONE, M., CHENG, Y., MATTHIES, L. (2007), « Two years of visual odometry on the mars exploration rovers », *Journal of Field Robotics* 24(3), p. 169-186.
- [2] HARRIS, C., STEPHENS, M. (1988), « A combined corner and edge detector », *Proceedings of the 4th Alvey Vision Conference*, p 147\_151.
- [3] LOWE, D. G. (1999), « Object recognition from local scale-invariant features », *Proceedings of the International Conference on Computer Vision*, vol. 2,p. 1150–1157.
- [4] MORAVEC, H. (1980), « Obstacle avoidance and navigation in the real world by a seeing robot rover », *Tech Report CMU-RI-TR-3*, Carnegie-Mellon University, Robotics Institute, September 1980.
- [5] BEAUDET, P (1978), « Rotationally invariant image operators », *International Journal of Current Pharmaceutical Research*, pages 579–586.
- [6] Cours M2 IRR(2013), UE Perception, Perception pour la Robotique, Université Toulouse III