



AVERTISSEMENT

Ce document est le fruit d'un long travail approuvé par le jury de soutenance et mis à disposition de l'ensemble de la communauté universitaire élargie.

Il est soumis à la propriété intellectuelle de l'auteur. Ceci implique une obligation de citation et de référencement lors de l'utilisation de ce document.

D'autre part, toute contrefaçon, plagiat, reproduction illicite encourt une poursuite pénale.

Contact : ddoc-theses-contact@univ-lorraine.fr

LIENS

Code de la Propriété Intellectuelle. articles L 122. 4

Code de la Propriété Intellectuelle. articles L 335.2- L 335.10

http://www.cfcopies.com/V2/leg/leg_droi.php

<http://www.culture.gouv.fr/culture/infos-pratiques/droits/protection.htm>



Institut National Polytechnique de Lorraine
Ecole Doctorale IAE+M
DFD Automatique et Production Automatisée



CMY 1999 LALLEMENT, A.

THESE

présentée pour l'obtention du titre de

Docteur de l'Institut National Polytechnique de Lorraine

Spécialité : Automatique et Traitement Numérique du Signal

par

Alex LALLEMENT

LOCALISATION D'UN ROBOT MOBILE PAR COOPERATION ENTRE VISION MONOCULAIRE ET TELEMETRIE LASER

Service Commun de la Documentation
INPL
Nancy-Brabois

Soutenue publiquement le 06 juillet 1999 devant le jury :

Président	: D. WOLF	Professeur à l'ENIM, (Metz)
Rapporteurs	: J. GRESSER	Professeur à l'UHA, (Mulhouse)
	: P.L. WENDEL	Professeur à l'ENSPS, (Strasbourg)
Examineurs	: M. DUFAUT	Maître de conférence à l'ENSEM, (Nancy)
Directeur de thèse	: R. HUSSON	Professeur à l'ENSEM, (Nancy)

À mon épouse,

à mes parents,

Avant-propos

Ce mémoire de thèse présente la synthèse des travaux de recherche que j'ai effectué au CRAN¹-INPL², au sein de l'équipe VRI³. Je souhaite exprimer mes remerciements à toutes les personnes sans lesquelles ce travail n'aurait jamais pu voir le jour.

En tout premier lieu je voudrais remercier ma famille qui m'a toujours soutenu et encouragé dans mes projets. Je remercie également mon épouse Sophie pour son amour et sa patience, ainsi que ses parents pour leur soutien inestimable.

Je remercie vivement les membres de mon jury de leur aide précieuse durant ces années de thèse. Je leur suis très reconnaissant pour leur grande disponibilité et leur non moins grande patience lors de la correction de ce mémoire.

J'exprime ma profonde reconnaissance à mon directeur de thèse, Monsieur René HUSSON, Professeur à l'ENSEM⁴ et Directeur de l'équipe VRI, pour m'avoir accueilli dans son équipe ainsi que pour la confiance qu'il a bien voulu me témoigner pendant ces années de recherche.

J'adresse mes sincères remerciements à mon codirecteur de thèse, Monsieur Michel DUFAUT, Maître de conférence à l'ENSEM et Directeur de Recherche au CRAN, pour le soutien et les conseils qu'il m'a prodigué tout au long de cette étude.

Je remercie également Monsieur Didier WOLF, Professeur à l'ENIM⁵ et Directeur de Recherche au CRAN, pour m'avoir fait l'honneur de présider mon jury de thèse.

¹ Centre de Recherche en Automatique de Nancy, CNRS UPRESA 7039

² Institut National Polytechnique de Lorraine

³ Vision pour la Robotique et l'Inspection

⁴ Ecole Nationale Supérieure d'Electricité et de Mécanique

J'exprime ma sincère reconnaissance à Monsieur Julien GRESSER, Professeur à l'UHA⁶, et à Monsieur Pierre Louis WENDEL, Professeur à l'ENSPS⁷, pour l'honneur qu'ils m'ont fait en acceptant d'être rapporteurs de ma thèse et pour l'intérêt qu'ils ont porté à mon travail.

Il m'est très difficile de remercier toutes les personnes qui ont contribué de près ou de loin à ce mémoire. Un grand merci à Monsieur Jean-Marie MUNIER, technicien du laboratoire, à Mademoiselle Christine PIERSON, secrétaire de l'équipe, à Monsieur Benoît MARCHAL et Madame Marie-Claude CARDOT, ingénieurs système de l'ENSEM, pour leur disponibilité et leur aide permanente.

J'adresse mes profonds remerciements à Messieurs Régis GRESSON et Ali SIADAT pour l'amitié qu'ils m'ont témoigné au quotidien tout au long de ces années de travail.

L'expression de ma sympathie la plus vive va également à l'ensemble des collègues du laboratoire pour leurs témoignages d'amitié et tout particulièrement à Messieurs Wilfrid COUROUX, Kokou DJATH, Axel KASKE, Arnaud SCHEFFLER, Hans-Joachim VON DER HARDT, et Mademoiselle Gisèle HILT.

⁵ Ecole Nationale d'Ingénieur de Metz

⁶ Université de Haute Alsace

⁷ Ecole Nationale Supérieure de Physique de Strasbourg

Table des matières

INTRODUCTION GÉNÉRALE.....	1
 CHAPITRE 1	
LOCALISATION PAR PERCEPTION ARTIFICIELLE EN ROBOTIQUE MOBILE.....	5
1.1 INTRODUCTION ET PROBLEMATIQUE	5
1.2 LES METHODES DE LOCALISATION.....	8
1.2.1 Localisation relative.....	9
1.2.1.1 L'odométrie	9
1.2.1.2 Les méthodes inertielles	10
1.2.1.3 Le magnétomètre.....	11
1.2.1.4 Critiques.....	11
1.2.2 Localisation absolue.....	12
1.2.2.1 Amers artificiels.....	12
1.2.2.1.1 Les balises passives.....	13
1.2.2.1.2 Les balises actives	14
1.2.2.1.3 Détermination de la localisation du robot	15
1.2.2.2 Amers environnementaux.....	16
1.2.2.2.1 Choix des amers environnementaux.....	17
1.2.2.2.2 Carte globale de l'environnement.....	18
1.2.2.2.3 Détection et modélisation des structures locales	18
1.2.2.2.4 Mise en correspondance et détermination de la localisation.....	19
1.2.2.3 Critiques.....	21
1.3 LA PERCEPTION ARTIFICIELLE	22
1.3.1 Les capteurs extéroceptifs embarqués	22
1.3.1.1 Capteurs télémétriques.....	22
1.3.1.1.1 Télémétrie ultrasonique.....	23
1.3.1.1.2 Télémétrie laser 2D	23
1.3.1.1.3 Télémétrie laser 3D.....	24
1.3.1.2 Capteurs de vision	24
1.3.1.2.1 Vision monoculaire	24
1.3.1.2.2 Stéréovision.....	25
1.3.2 Les systèmes mono-capteur.....	26
1.3.2.1 Télémétrie	26
1.3.2.2 Vision	28
1.3.2.3 Critiques.....	29
1.3.3 Combinaison de capteurs	30
1.3.3.1 Systèmes multi-capteurs.....	31

1.3.3.2	Intégration de capteurs	32
1.3.3.3	Fusion de primitives	33
1.3.3.4	Association.....	34
1.3.3.5	Critiques.....	34
1.4	SYSTEME DE LOCALISATION ABSOLUE MIS EN ŒUVRE.....	35
1.4.1	Choix d'une méthode de localisation	35
1.4.1.1	Sélection des amers environnementaux	36
1.4.1.2	Sélection d'une méthode de mise en correspondance	37
1.4.2	Choix d'un système de perception.....	38
1.4.2.1	Choix des capteurs	39
1.4.2.2	Stratégie de perception	40
 CHAPITRE 2		
	SYSTÈME DE PERCEPTION VISION-LASER	43
2.1	INTRODUCTION	43
2.2	LE MODULE DE VISION.....	45
2.2.1	Présentation matérielle.....	45
2.2.1.1	La caméra et l'objectif	46
2.2.1.2	La tourelle porte caméra	47
2.2.1.3	La transmission vidéo	47
2.2.1.4	La carte d'acquisition d'image vidéo	47
2.2.1.5	La transmission des paramètres de réglage.....	48
2.2.2	Modélisation de la caméra et de l'objectif	48
2.2.2.1	Modélisation d'un système de vision monoculaire	48
2.2.2.2	Expression du modèle sténopé étendu direct.....	52
2.2.2.3	Identification des paramètres intrinsèques du modèle sténopé étendu	54
2.2.2.3.1	Coordonnées (u_0, v_0) du centre de l'image numérisée.....	56
2.2.2.3.2	Facteurs d'échelle de l'image d_x et d_y	59
2.2.2.3.3	Longueur focale f et distance inter plan d_p	60
2.2.3	Caractéristiques de la tourelle porte caméra	65
2.3	LE TELEMETRE LASER	66
2.3.1	Présentation matérielle.....	66
2.3.1.1	Principe de fonctionnement	67
2.3.1.2	Acquisition d'une profilométrie	68
2.3.2	Caractéristiques du télémètre laser	68
2.3.2.1	Réponse du télémètre pour des cibles de natures différentes.....	69
2.3.2.2	Divergence du faisceau	70
2.3.2.3	Résolution angulaire	71
2.3.2.4	Résolution de profondeur.....	72
2.4	EXPRESSION DES PRIMITIVES SENSORIELLES DANS UN REPERE COMMUN.....	74
2.4.1	Les repères du robot	74
2.4.1.1	Transformation du repère R_{robot} vers $R_{caméra}$ et R_{robot} vers R_{laser}	76
2.4.1.2	Procédure de calibration	77

2.4.2	Exploitation dans le cadre de notre application	81
2.4.2.1	Réglage des paramètres du module de vision.....	81
2.4.2.2	Transformation du repère de l'image numérisée vers R_{robot}	83
2.5	CONCLUSION.....	87

CHAPITRE 3

CONSTRUCTION DU MODÈLE LOCAL DE L'ENVIRONNEMENT..... 89

3.1	INTRODUCTION	89
3.2	MODELE LOCAL TELEMETRIQUE	91
3.2.1	L'approximation polygonale	91
3.2.2	Mise en évidence des structures formées par des plans verticaux	92
3.2.2.1	Acquisition de la profilométrie laser	94
3.2.2.2	Filtrage de la chaîne de mesure	94
3.2.2.3	Fractionnement de la chaîne	95
3.2.2.4	Partitionnement des chaînes partielles	97
3.2.2.5	Segmentation des portions	101
3.3	MISE EN ÉVIDENCE DE CARACTÉRISTIQUES POTENTIELLES.....	104
3.3.1	Espace accessible par le système de vision.....	104
3.3.1.1	Points extrémités p accessibles au module de vision	104
3.3.1.2	Primitives segments accessibles au module de vision	106
3.3.1.3	Modèle local accessible au module de vision	108
3.3.2	Génération de primitives coin.....	109
3.3.3	Sélection des structures potentielles	112
3.4	EXTRACTION DES AMERS	116
3.4.1	Stratégie de recherche d'amers.....	116
3.4.1.1	Réglage des paramètres du module de vision.....	118
3.4.1.2	Analyse de la fenêtre centrale	118
3.4.1.3	Analyse des fenêtres latérales.....	121
3.4.1.4	Discussion de la stratégie de perception mise en œuvre	123
3.4.2	Extraction des primitives vision.....	126
3.4.2.1	Choix d'une approche « dérivative ».....	126
3.4.2.2	Détection des contours verticaux.....	128
3.4.2.2.1	Gradient de l'image et recherche des maxima locaux.....	128
3.4.2.2.2	Seuillage et chaînage des maxima locaux	131
3.4.2.2.3	Segmentation des chaînes.....	134
3.5	CONCLUSION.....	137

CHAPITRE 4

DÉTERMINATION DE LA LOCALISATION..... 139

4.1	INTRODUCTION	139
4.2	REPRESENTATION DES DONNEES.....	141

4.2.1	Descripteurs invariants	141
4.2.2	Le modèle global de l'environnement.....	144
4.2.2.1	Les amers environnementaux.....	144
4.2.2.2	Formation de triplets consistants	146
4.2.2.3	Le modèle global.....	150
4.2.3	Modèle local de l'environnement.....	152
4.2.3.1	Constitution des triplets d'arêtes.....	152
4.2.3.2	Fiabilité du modèle local	155
4.3	MISE EN CORRESPONDANCE	156
4.3.1	Mise en correspondance de triplets.....	156
4.3.1.1	Tables de hachage	157
4.3.1.2	Listes ordonnées.....	161
4.3.1.3	Performances de la mise en correspondance	166
4.3.2	Appariements de primitives	168
4.4	DETERMINATION DE LA POSITION ET DE L'ORIENTATION DU ROBOT.....	170
4.4.1	Estimation par les moindres carrés	170
4.4.2	Recours à l'incertitude des mesures	172
4.4.3	Performance de la localisation	176
4.4.3.1	Exploitation du modèle télémétrique.....	176
4.4.3.2	Apport de la vision	180
4.4.3.3	Incetitude du modèle global.....	182
4.4.3.4	Mises en échec	183
4.5	CONCLUSION.....	185
 CONCLUSION GÉNÉRALE		187
 RÉFÉRENCES BIBLIOGRAPHIQUES		191

Table des figures

Figure 1.1 : Schéma de la fonction mobilité autonome.....	5
Figure 1.2 : Localisation (x_e, y_e, Θ_e) du système R_{robot} dans le système absolu $R_{env.}$	7
Figure 1.3 : Les différentes méthodes de localisation.....	8
Figure 1.4 : Localisation relative.....	9
Figure 1.5 : Localisation par amers artificiels.....	12
Figure 1.6 : Quelques exemples de balises passives codées.....	14
Figure 1.7 : Triangulation par télémétrie (figure de gauche) et goniométrie (figure de droite).....	15
Figure 1.8 : Localisation par amers environnementaux.....	16
Figure 1.9 : Modélisation des données sensorielles dans le cadre de la télémétrie.....	26
Figure 1.10 : Amers environnementaux typiques de la vision.....	28
Figure 1.11 : Configuration des systèmes combinant différents capteurs.....	30
Figure 1.12 : Lumière structurée (figure de gauche), Laser Eye (figure de droite).....	32
Figure 1.13 : Amer environnemental A_i	37
Figure 1.14 : Analyse des discontinuités d'une profilométrie laser.....	40
Figure 2.1 : Architecture du projet ROMANE.....	43
Figure 2.2 : Le module de vision.....	45
Figure 2.3 : Chaîne d'acquisition et de traitement de l'image.....	46
Figure 2.4 : Orientation en azimut et en site de la tourelle porte caméra.....	47
Figure 2.5 : Les quatre groupes optiques formant un objectif.....	49
Figure 2.6 : Déplacement des lentilles lors de la variation de focale avec un focus constant.....	49
Figure 2.7 : Modèle sténopé.....	50
Figure 2.8 : Modèle de lentille mince.....	50
Figure 2.9 : Modèle de lentille épaisse.....	51
Figure 2.10 : Modèle sténopé étendu.....	51
Figure 2.11 : Transformation par le modèle sténopé étendu.....	52
Figure 2.12 : Optimisation du plan image dans le cas d'amers de hauteur maximale H_{max} fixe.....	55
Figure 2.13 : Couples $C_{i, i=1 \text{ à } 16}$ de consignes « focale _i (o), focus _i (*) ».....	55
Figure 2.14 : Recherche des coordonnées (u_o, v_o) du centre de l'image numérisée.....	57
Figure 2.15 : Coordonnées moyennes (u_o, v_o) du centre de l'image numérisée en fonction des couples de consignes $C_{i, i=1 \text{ à } 16}$ « focale _i , focus _i ».....	58
Figure 2.16 : Déformation d'échelle de l'image numérisée.....	60
Figure 2.17 : Détermination de la longueur focale f et de la distance inter plan d_{ip}	60
Figure 2.18 : Sensibilité de la projection d'un pixel de l'image aux paramètres f et d_{ip}	64
Figure 2.19 : Télémètre laser Sick PLS200.....	66
Figure 2.20 : Structure de la transmission des données laser.....	67
Figure 2.21 : Principe de fonctionnement du télémètre laser SICK PLS200.....	67
Figure 2.22 : Coordonnées d'un point de mesure p_i dans le repère laser R_{laser}	68
Figure 2.23 : Réponse du télémètre pour des cibles de natures différentes.....	69
Figure 2.24 : Divergence du faisceau laser.....	70

Figure 2.25 : Point de moyennage.	70
Figure 2.26 : Chevauchement des impacts laser.	71
Figure 2.27 : Largeur $L_{\min}$ d'un objet pouvant être détecté en fonction de la distance.	71
Figure 2.28 : Courbe B, résolution dynamique R_{dyn} en fonction de la distance à la cible (échelles logarithmiques).	72
Figure 2.29 : Ecart type δd_i (o) et écart maximal Δd_i (*) classés en fonction des mesures de profondeur moyenne $d_{i \text{ moyen}}$	73
Figure 2.30 : Positions respectives des repères R_{robot} , R_{azimut} , $R_{\text{caméra}}$ et R_{laser}	75
Figure 2.31 : Mire mixte vision/télémétrie laser.	77
Figure 2.32 : Calibration du système de perception.	78
Figure 2.33 : Recherche des coordonnées d'une arête de référence P.	80
Figure 2.34 : Projection d'un point u_a de l'image numérisée dans le repère R_{robot}	82
Figure 2.35 : Incertitudes de projection δ_x et δ_y	84
Figure 2.36 : Incertitudes δ_x et δ_y sur la projection dans le repère R_{robot} d'un pixel de l'image pour le couple de consigne C_5 ($\epsilon d = 10 \text{ mm}$).	85
Figure 2.37 : Incertitudes δ_x et δ_y sur la projection dans le repère R_{robot} d'un pixel de l'image pour le couple de consigne C_5 ($\epsilon d = 15 \text{ mm}$).	85
Figure 2.38 : Projection du pixel d'indice 512 engendrant des incertitudes δ_x et δ_y inférieures à un seuil de 25 mm.	86
Figure 3.1 : Construction du modèle local de l'environnement.	90
Figure 3.2 : Mauvaise modélisation par une primitive coin $_k$ d'un angle de mur entraînant une erreur de localisation des arêtes a_s , a_n et a_m dans le repère R_{robot}	93
Figure 3.3 : Exemple de point de mesure P_k entraînant une erreur de segmentation au niveau d'une porte.	94
Figure 3.4 : Fractionnement de la chaîne initiale par une méthode de fusion.	95
Figure 3.5 : Chaînes obtenues après filtrage et fractionnement de la profilométrie A.	96
Figure 3.6 : Partitionnement d'une chaîne (1 ^{ère} passe).	97
Figure 3.7 : Partitionnement d'une chaîne (2 ^{nde} passe).	98
Figure 3.8 : Résultat du partitionnement et de la segmentation pour différentes valeurs de seuil $l^i_{\text{découpage}}$	100
Figure 3.9 : Partitionnement de la chaîne de mesure décrivant la profilométrie A.	101
Figure 3.10 : Recherche des points extrêmes des segments.	103
Figure 3.11 : Profilométrie A segmentée.	103
Figure 3.12 : Occlusion du point p_i du modèle local télémétrique due au segment $_k$	105
Figure 3.13 : Construction du modèle local accessible par le module de vision.	107
Figure 3.14 : Portion du modèle local télémétrique A accessible par le module de vision.	108
Figure 3.15 : Primitive coin $_k$	109
Figure 3.16 : Projection dans le plan de l'image numérisée d'une primitive coin.	111
Figure 3.17 : Primitive coin de type (a).	112
Figure 3.18 : Primitive coin de type (b).	113
Figure 3.19 : Primitive coin de type (c).	113
Figure 3.20 : Poids $_{\text{fenêtres}}$ lié à la projection de la primitive coin dans le plan image.	114
Figure 3.21 : Poids $_{\Omega}$ lié à l'angle Ω	114
Figure 3.22 : Primitives coins classées pour l'analyse par la vision.	115
Figure 3.23 : Organigramme de l'inspection d'une primitive coin par la vision.	117
Figure 3.24 : Projection des primitives vision dans le plan (X_n, Y_r) du repère R_{robot}	119

Figure 3.25 : Inspection d'une primitive coin causée par un objet non référencé (une caisse).....	121
Figure 3.26 : Inspection d'une primitive coin modélisant un angle de mur référencé.....	122
Figure 3.27 : Taux d'observation de M d'arêtes verticales en fonction des N primitives coin inspectées.....	125
Figure 3.28 : Arêtes verticales $a_i, i=1...6$ extraites du modèle télémétrique local A	126
Figure 3.29 : Impact du paramètre de réglage ξ sur la forme du filtre de Deriche.....	129
Figure 3.30 : Les maxima locaux correspondant aux contours obtenus pour différentes valeurs du paramètre ξ sont représentés en superposition de l'image initiale.....	130
Figure 3.31 : Procédure de chaînage des maxima locaux.....	133
Figure 3.32 : Maxima locaux seuillés par s_{bas}	135
Figure 3.33 : Maxima locaux chaînés après le seuillage s_{haut}	135
Figure 3.34 : Arêtes verticales extraites de l'image numérisée.....	136
Figure 4.1 : Procédure de mise en correspondance.....	140
Figure 4.2 : Descripteurs invariants associés à tout triplet d'amers $\{A_a A_b A_c\}$	143
Figure 4.3 : Plage de variation de l'angle Ω d'un amer.....	145
Figure 4.4: Amers appartenant à un même plan vertical.....	145
Figure 4.5 : Triplet d'arêtes observées.....	146
Figure 4.6 : Constellations de triplets d'amers délicates à percevoir.....	147
Figure 4.7 : Triplets d'amers perceptibles selon la configuration du site.....	148
Figure 4.8 : Liste $\mathcal{L}$ décrivant le modèle global.....	150
Figure 4.9 : Nombre moyen de triplets d'amers caractérisés par des valeurs d'invariants L et α proches.....	151
Figure 4.10 : Nombre moyen de triplets d'amers caractérisés par des valeurs d'invariants L , α , Ω et statut proches.....	151
Figure 4.11 : Les triplets formés à partir des arêtes observées $a_i, i=1...6$	152
Figure 4.12 : Incertitude sur les invariants L et α	154
Figure 4.13 : Définition des listes $\mathcal{L}^p$ relatives aux descripteurs invariants L et α	157
Figure 4.14 : Distribution de la table de hachage $\mathcal{T}^1$	158
Figure 4.15 : Distribution de la table de hachage $\mathcal{T}^4$	159
Figure 4.16 : Table de hachage relative à un descripteur invariant.....	160
Figure 4.17 : Algorithme de mise en correspondance de triplets par tables de hachage.....	160
Figure 4.18 : Recherche binaire dans une liste ordonnée.....	162
Figure 4.19 : Mise en correspondance de triplets par listes ordonnées.....	163
Figure 4.20 : Pouvoir discriminant du premier critère testé.....	164
Figure 4.21 : Restriction de la liste $\mathcal{L}$ courante au fur et à mesure de l'analyse.....	164
Figure 4.22 : Algorithme de mise en correspondance de triplets par listes ordonnées.....	165
Figure 4.23 : Temps de recherche moyen lié au descripteur invariant L_{ij}	166
Figure 4.24 : Temps de recherche des triplets d'amers pour les 6 premiers invariants.....	167
Figure 4.25 : Recherche des appariements.....	169
Figure 4.26 : Localisation estimée à partir de deux appariements.....	171
Figure 4.27 : Détermination d'une orientation Θ_e possible du robot.....	173
Figure 4.28 : Détermination d'un ensemble de positions (x_e^*, y_e^*) possibles du robot.....	174
Figure 4.29 : Intersection de plusieurs polygones.....	175
Figure 4.30 : Approximation de l'ensemble des localisations du robot.....	175
Figure 4.31 : Carte globale de l'environnement.....	176
Figure 4.32 : Traitement parallèle des informations.....	178

<i>Figure 4.33 : Modèle local A et localisation correspondante.</i>	<i>179</i>
<i>Figure 4.34 : Modèle local B et localisation correspondante.</i>	<i>179</i>
<i>Figure 4.35 : Modèle local Bbis et localisation correspondante.</i>	<i>180</i>
<i>Figure 4.36 : Modèle local C et localisation correspondante.</i>	<i>181</i>
<i>Figure 4.37 : Modèle local D et localisation correspondante.</i>	<i>181</i>
<i>Figure 4.38 : Modèle local E et localisation correspondante.</i>	<i>182</i>
<i>Figure 4.39 : Mise en échec causée par un manque de primitive coin.</i>	<i>183</i>
<i>Figure 4.40 : Mise en échec causée par une faible représentativité du modèle local.</i>	<i>184</i>

Introduction générale

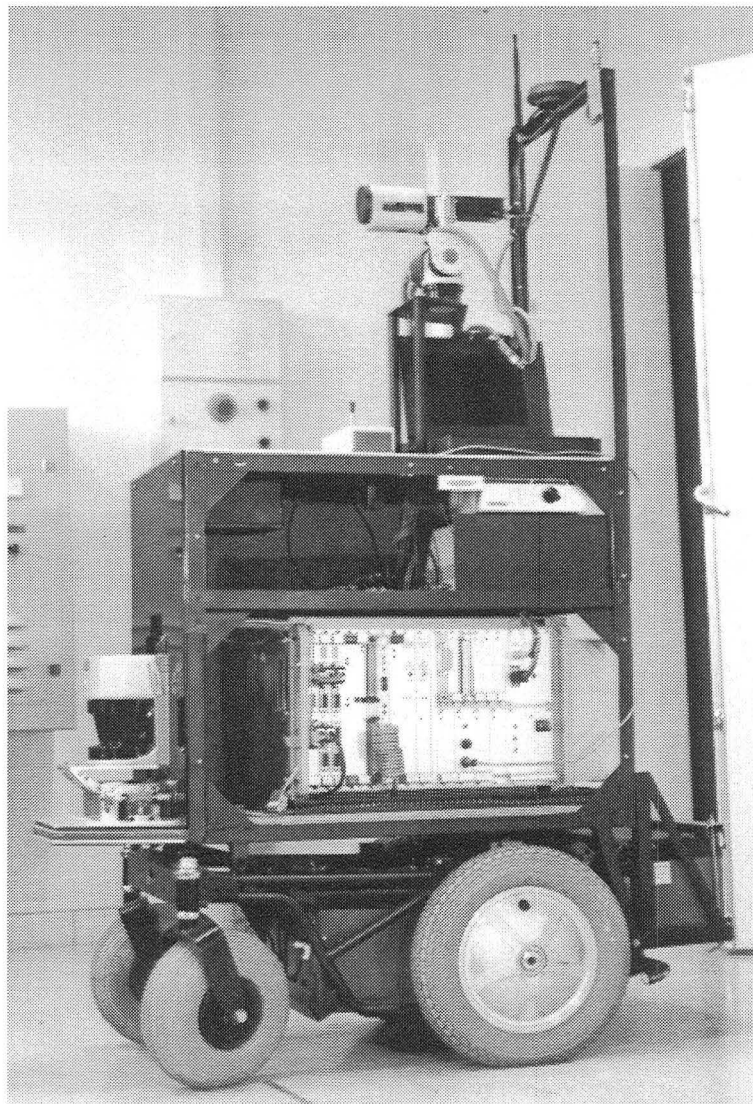
La détermination de la position d'un observateur à partir de données sensorielles est une fonction indispensable pour de nombreuses applications de robotique mobile. Elle est notamment un pré-requis nécessaire permettant à un véhicule de planifier sa trajectoire. Or l'exécution par une machine autonome d'une tâche pourtant relativement simple à énoncer : **"Repère toi par rapport à ton environnement"**, soulève de nombreux problèmes du fait de la complexité des scènes à analyser et de la diversité des ressources nécessaires à sa réalisation.

La recherche de la localisation d'un robot mobile nécessite en effet la mise en œuvre de mécanismes de raisonnement et de décision opérant sur des modèles et des représentations symboliques, images imparfaites de la réalité. La problématique se rattache ainsi à celle plus générale de la perception artificielle, qui à elle seule englobe de nombreux domaines de recherche tels que la fusion multisensorielle, la commande référencée capteur, la modélisation de l'environnement, l'analyse et la prédiction de scène, ...

Si l'intégration des moyens de perception est de plus en plus poussée, en revanche, l'interprétation des données sensorielles et la prise de décision par rapport à cette interprétation constituent des problèmes difficiles à résoudre. Ainsi, le degré de complexité des algorithmes augmente considérablement en fonction de la sophistication des capteurs utilisés : plus la restitution du milieu d'évolution est précise, plus le niveau d'abstraction des algorithmes d'interprétation doit être élevé.

La thématique de recherche développée au sein de l'équipe Vision pour la Robotique et l'Inspection du Centre de Recherche en Automatique de Nancy consiste à concevoir un robot mobile capable d'exécuter des tâches prédéfinies et possédant suffisamment d'autonomie de réaction pour s'adapter aux contraintes de son environnement.

Ce projet se concrétise sous la forme du démonstrateur ROMANE (*RObot Mobile Autonome pour la Navigation en Extérieur*) qui a fait l'objet de travaux récents.



La plate-forme expérimentale ROMANE.

Le robot est ainsi équipé d'un système de localisation relative autorisant des déplacements sur une surface plane sans faire appel à des références extérieures [Hardt 97]. L'estimation de la position et de l'orientation est réalisée par un filtrage de Kalman sur les données sensorielles issues de différents capteurs proprioceptifs : deux codeurs incrémentaux, un gyromètre ainsi qu'un magnétomètre. Le robot peut se déplacer sur des distances de quelques dizaines de mètres avant que les erreurs de position ne deviennent trop importantes.

Afin de permettre au robot de se déplacer sur un chemin de type rural, un système de détection de bords de route imprécis et flous a été développé [Kaske 97]. Il repose sur la mise en correspondance des informations sensorielles provenant d'un système de vision monoculaire, avec un modèle de bord de route simple. Combiné avec le module de localisation relative, il permet au robot de naviguer tout en restant sur le chemin.

Une procédure de localisation absolue par vision monoculaire, dans le cadre d'un environnement extérieur de type parking, permet de recalibrer périodiquement le robot lorsque l'information de localisation relative devient trop imprécise [Couroux 97]. A partir de la localisation a priori du robot et de l'orientation de la caméra, des amers environnementaux sont prédits puis mis en correspondance avec les primitives réellement perçues par le système de vision. La position et l'orientation du robot sont alors déterminées par triangulation.

Un système de navigation basé sur l'utilisation d'un télémètre laser 2D assure enfin une certaine autonomie au robot lors de ses déplacements à l'intérieur de bâtiments [Siadat 99]. Une carte de profondeur générée en temps réel permet ainsi au robot d'atteindre un point donné tout en évitant les obstacles. Il peut donc réagir par rapport à l'environnement et effectuer des tâches simples de type : suivi de mur, passage de porte, ...

Le travail présenté dans ce mémoire porte sur la détermination de la localisation absolue d'un robot mobile dans le cadre d'un environnement intérieur dynamique. Comme nous l'avons souligné précédemment, il s'agit là d'une étape préalable indispensable à toute prise de décision fiable. Mais, alors que dans le cas de la robotique dite industrielle, l'environnement est aménagé en fonction des tâches que devra effectuer le robot ; dans le cas de la robotique mobile, c'est le robot qui doit s'adapter à l'environnement. Cette contrainte génère un degré de complexité beaucoup plus grand puisqu'elle implique de définir une méthode de localisation robuste face aux perturbations liées à la présence d'objets mobiles. En contrepartie, il est possible d'obtenir une meilleure souplesse d'utilisation du robot si la méthode est suffisamment évolutive pour être exploitée dans un environnement de type similaire sans nécessiter de modifications fondamentales.

Nous nous sommes pour cela penchés sur l'aspect perception et modélisation de l'environnement, qui constituent des domaines où les problèmes ne sont à l'heure actuelle que partiellement résolus. Nous avons notamment développé un système de perception basé sur la coopération de deux capteurs extéroceptifs complémentaires visant à limiter les phénomènes de sur-détection et de sous-détection, ainsi qu'une procédure de modélisation de l'environnement ne nécessitant qu'un nombre réduit de primitives géométriques simples.

Il nous a également paru intéressant de développer l'autonomie décisionnelle et opérationnelle du robot en déterminant sa position et son orientation à l'aide d'un minimum d'informations de localisation a priori. Il s'agit notamment de permettre l'initialisation automatique du système de localisation, ainsi que la vérification de la validité de l'information de position et d'orientation délivrée par le système de localisation relative.

Les travaux présentés dans les quatre chapitres de ce mémoire contribuent donc à la recherche de solutions particulières au problème général de la localisation en robotique mobile.

Le premier chapitre pose la problématique abordée ainsi que le cadre de notre travail. Les techniques de localisation ainsi que les capteurs extéroceptifs couramment utilisés en robotique mobile sont présentés et analysés. Le choix des arêtes verticales en tant qu'amers environnementaux est discuté, puis le principe de la stratégie de perception par coopération de capteurs mis en œuvre est exposé.

Le deuxième chapitre est consacré au système de perception vision/télémétrie du robot mobile ROMANE. Les caractéristiques principales ainsi que la méthode de calibration de chaque capteur sont détaillées, puis les transformations permettant d'exprimer les primitives sensorielles dans un repère commun sont présentées.

Le troisième chapitre porte sur la construction du modèle local de l'environnement à l'aide des informations sensorielles. La génération d'un modèle télémétrique constitué de primitives *coin* est tout d'abord détaillée. L'analyse de ces primitives télémétriques par le module de vision ainsi que la validation d'arêtes verticales en temps qu'amers potentiels est alors discutée.

Le quatrième chapitre concerne la procédure de mise en correspondance et de localisation mise en œuvre. La transformation des primitives sensorielles en une représentation faisant apparaître des invariants géométriques est détaillée, puis l'algorithme réalisant les appariements entre amers et arêtes observées est décrit. L'ensemble des paramètres de localisation compatibles avec les incertitudes de mesure des arêtes observées est alors déduit.

En conclusion générale, les résultats obtenus sont discutés et les perspectives ouvertes par ce travail sont présentées.

Chapitre 1

Localisation par perception artificielle en robotique mobile

1.1 INTRODUCTION ET PROBLEMATIQUE

La robotique mobile couvre un domaine d'activités aussi variées que le nettoyage, la manipulation et le transport de charges, la surveillance et le diagnostic, le secteur militaire, ... Il apparaît que pour ce vaste champ d'application, la fonction « *mobilité autonome* » des robots revêt une importance primordiale.

L'analyse du problème de la mobilité pour un engin autonome peut se faire à l'aide du schéma *perception / décision / action* présenté à la Figure 1.1. C'est à partir de la connaissance de son environnement et des buts qu'il a à atteindre, qu'un robot mobile élabore des lois de commande et prend des décisions d'action.

La pertinence de la modélisation de son environnement conditionne donc fortement la réussite de la mission dont le robot est chargé.

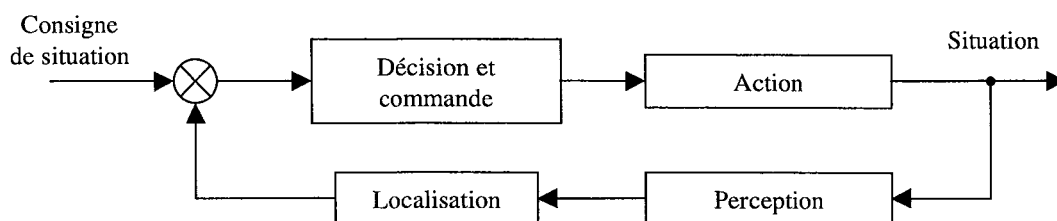


Figure 1.1 : Schéma de la fonction mobilité autonome.

Dans le cadre de la robotique mobile, une tâche de navigation ne peut ainsi être entreprise qu'une fois résolus les problèmes cruciaux de la perception de l'environnement et de la détermination de la position du robot au sein de cet environnement.

L'étape de perception consiste en un ensemble d'activités visant à saisir un certain nombre d'informations dans le but d'acquérir une connaissance et une compréhension du milieu d'évolution du robot. Il faut distinguer la perception à horizon très proche, qui est directement liée aux aspects « sécurité », de la perception à horizon plus lointain qui est reliée aux fonctions de modélisation, localisation et de navigation.

Le choix d'un système de perception repose non seulement sur des critères de résolution spatiale, de temps de réponse et de coût, mais il est également fortement conditionné par les contraintes inhérentes à l'environnement du robot.

Dans le cadre d'un environnement intérieur semi-structuré, les hypothèses suivantes peuvent être retenues :

- espace fini, notamment en hauteur,
- sol plan et horizontal,
- les délimitations de l'espace peuvent être approchées par des formes géométriques simples telles que des plans ou des cylindres.

Une perception de l'espace situé dans un plan parallèle au sol est ainsi généralement suffisante. A contrario, dans le cas d'un environnement extérieur semi-structuré ou naturel, il devient nécessaire de disposer d'un ensemble de capteurs permettant une perception tridimensionnelle ainsi qu'un repérage d'amers à grandes distances. Cette contrainte est due notamment aux problèmes de non-planéité du sol, de grande dispersion des objets de la scène et de l'espace.

Pour chaque capteur est défini un système de coordonnées local R_{capteur} dans lequel sont exprimées les mesures réalisées. Une phase de modélisation visant à conditionner les données sensorielles acquises en une forme interprétable permet de générer des attributs représentatifs de l'état du robot ou de son environnement. L'interprétation des données sensorielles est alors réalisée afin de sélectionner parmi les attributs générés ceux qui seront utilisés pour la détermination de la localisation du robot mobile. L'ensemble des primitives sensorielles retenues sont alors exprimées dans un système de coordonnées commun R_{robot} propre au robot.

L'étape de localisation consiste à exploiter les attributs sensoriels afin de déterminer la situation du robot mobile au sein de son environnement. Il est pour cela nécessaire de définir un système de coordonnées absolu $R_{env.}$ lié à l'environnement dans lequel évolue le robot. Le choix de ce repère n'est pas forcément simple car il faut définir son emplacement avec précision, son sens et s'assurer de son adéquation avec l'environnement et la mission du robot.

Dans un espace à trois dimensions, définir la position et l'orientation d'un objet demande en général 6 degrés de liberté (3 translations et 3 rotations). Or, dans le cadre de la robotique mobile en milieu intérieur, il est fait l'hypothèse de déplacement sur sol plan, nous nous trouvons donc dans le cadre d'un espace à deux dimensions. La position et l'orientation d'un objet peuvent ainsi être représentées par le triplet (x_e, y_e, Θ_e) , où x_e et y_e représentent la translation du système de coordonnées R_{robot} par rapport au système de référence absolu $R_{env.}$, et Θ_e représente sa rotation autour d'un axe vertical (Figure 1.2).

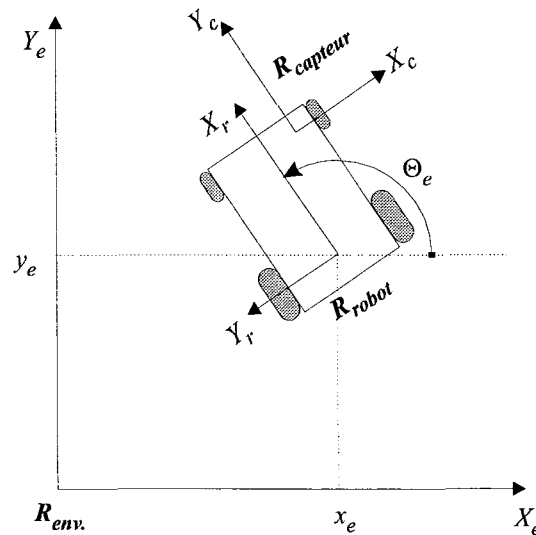


Figure 1.2 : Localisation (x_e, y_e, Θ_e) du système R_{robot} dans le système absolu $R_{env.}$

Nos travaux portent sur la localisation d'un robot mobile en environnement intérieur semi-structuré. Il s'agit là d'un problème qui est encore très ouvert et pour lequel de nombreuses approches sont envisageables. Le choix d'une méthode de localisation conditionnant le type d'informations sensorielles nécessaires ainsi que leur traitement, nous étudions donc tout d'abord les différents principes de localisation existants (paragraphe 1.2). Nous portons par la suite une attention particulière sur les systèmes de perception extéroceptifs qui sont les plus adaptés à nos besoins et exposons les techniques de modélisation qui leur sont associées (paragraphe 1.3). Nous présentons enfin le système de perception retenu ainsi que la stratégie de localisation mise au point dans le cadre du projet ROMANE (paragraphe 1.4).

1.2 LES METHODES DE LOCALISATION

La détermination de la position et de l'orientation d'un observateur à partir de données sensorielles est un problème important pour de nombreuses applications de robotique. Il est particulièrement délicat pour des systèmes devant estimer leur localisation dans des environnements complexes en utilisant des données imprécises, incomplètes ou incertaines.

L'un des points importants qui ressort de l'étude de la littérature traitant du positionnement des robots mobiles est qu'il n'existe pas actuellement de solution générale satisfaisante à ce problème [Borenstein 96]. On peut toutefois distinguer deux grandes familles de méthodes qui dépendent directement de la classification des capteurs embarqués.

La **localisation relative**, ou localisation à l'estime, consiste à évaluer la position, l'orientation et éventuellement la vitesse du robot mobile par rapport aux dernières mesures effectuées, ce qui revient à réaliser une intégration des informations fournies par divers capteurs proprioceptifs (odomètres, centrales inertielles, magnétomètre), depuis une position de départ connue dans le repère absolu R_{env} . (point de départ ou de recalage).

La **localisation absolue**, ou localisation par rapport à l'environnement, permet de se repérer directement dans le domaine d'évolution du robot mobile à l'aide des informations fournies par divers capteurs extéroceptifs. Le robot mobile doit disposer d'un modèle global de l'environnement regroupant les amers artificiels ou environnementaux, susceptibles d'être détectés et dont la position est connue avec précision. La détermination de la localisation du robot se fait alors par la mise en correspondance des primitives du modèle sensoriel avec les amers de la carte théorique.

La Figure 1.3 détaille les principales méthodes de localisation utilisées dans le cadre de la robotique mobile.

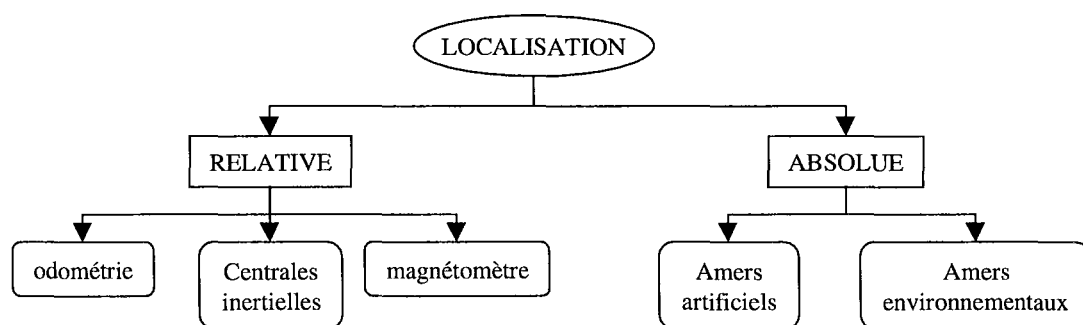


Figure 1.3 : Les différentes méthodes de localisation.

Dans les paragraphes suivants, les différentes méthodes de localisation couramment utilisées sont décrites afin d'en analyser les performances et les limites.

1.2.1 Localisation relative

Dans le cas des mesures de position relative, des capteurs proprioceptifs fournissent des informations propres au comportement interne du robot, c'est à dire sur son état à un instant donné. Les capteurs mesurent en permanence les paramètres cinématiques du robot par rapport à des références fixes. On obtient ainsi des informations sur des déplacements élémentaires, des vitesses, des angles de giration ou encore des angles d'attitude. La Figure 1.4 illustre la détermination de la localisation d'un robot dans le repère de référence $R_{env.}$ par intégration de ces informations.

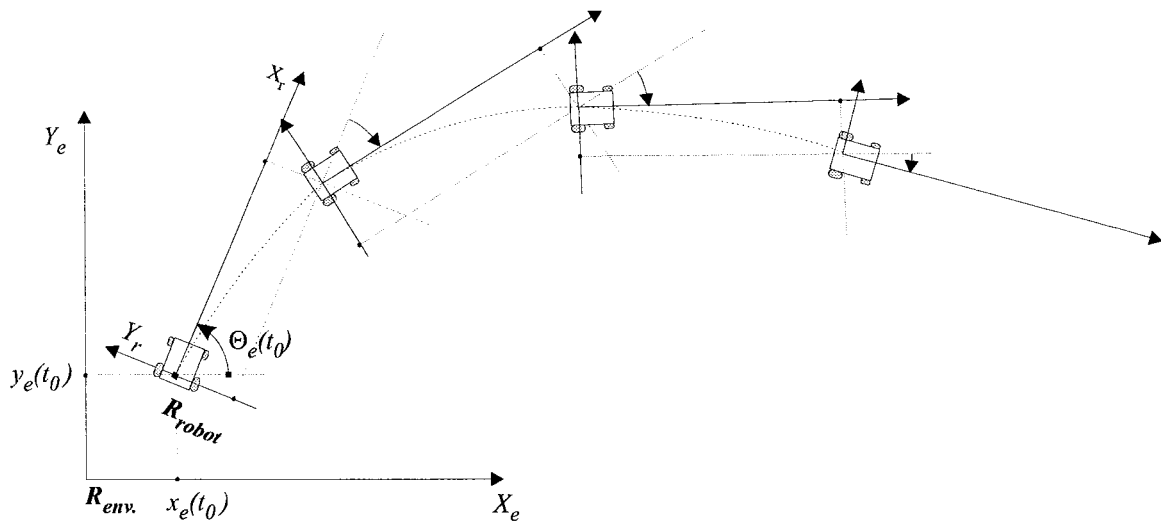


Figure 1.4 : Localisation relative.

Nous présentons ci-après les trois méthodes de localisation relative les plus couramment utilisées : l'odométrie, les méthodes inertielles et la magnétométrie.

1.2.1.1 L'odométrie

Elle permet de déterminer la position et le cap d'un mobile par intégration de ses déplacements élémentaires par rapport à un repère fixe. La rotation des roues est mesurée par des codeurs angulaires incrémentaux, puis, connaissant le rayon des roues et la distance entre chaque roue, il est possible d'exprimer de manière récursive la position et le cap du robot.

Les sources d'erreurs proviennent de l'incertitude sur la mesure des paramètres géométriques du robot, de la perte d'adhérence des roues avec le sol et des décalages introduits dans les mesures odométriques lorsque le robot évolue sur des plans inclinés. L'imprécision sur le déplacement du robot est principalement due à l'erreur cumulative engendrée lors de l'intégration des déplacements élémentaires.

Cette méthode est facile à mettre en œuvre, fournit une estimation de la position et du cap à une cadence relativement élevée et reste d'un coût modique. Ces caractéristiques en font le système de localisation le plus couramment utilisé en robotique mobile [Cox 91]. Cependant, lorsque les distances à parcourir sont importantes et que la trajectoire comporte de nombreux changements de direction, l'imprécision sur la position et le cap augmente considérablement de façon non bornée avec la distance parcourue [Arnould 93]. En pratique, l'odométrie n'est utilisable que sur des distances inférieures à une dizaine de mètres. Au delà, des recalages consistant en la détermination d'une nouvelle origine des mesures s'avèrent indispensables.

1.2.1.2 Les méthodes inertielles

Elles utilisent des informations d'accélération suivant des axes prédéterminés ou bien l'effet gyroscopique par rapport à des axes de rotation prédéfinis. Le calcul du déplacement élémentaire du mobile est alors effectué par double intégration de ces informations.

L'accéléromètre est un capteur qui mesure la composante de l'accélération linéaire en un point donné par rapport à l'axe de sensibilité du capteur. En pratique, la mesure de l'accélération est effectuée à l'aide d'une masse d'épreuve M reliée au boîtier du capteur : il s'agit de mesurer l'effort massique non gravitationnel que l'on doit appliquer à M pour la maintenir en place dans le boîtier lorsqu'une accélération est donnée au boîtier. Les microaccéléromètres peuvent mesurer des accélérations de l'ordre de 0,1 g avec un biais de quelques μg . Toutefois, ils s'avèrent être très sensibles aux chocs [Barshan 95].

Le gyroscope est un capteur constitué d'une toupie entraînée à grande vitesse de rotation autour de son axe principal permettant la détermination du cap d'un mobile [Vaganay 93]. Cette toupie est libre par rapport au boîtier. Une phase d'initialisation consistant à donner une direction de référence est nécessaire avant d'effectuer des mesures.

L'inconvénient majeur des méthodes inertielles réside dans la double intégration des erreurs de mesure qui peut conduire à des erreurs non bornées sur l'information de position et/ou de cap.

1.2.1.3 Le magnétomètre

Il délivre des informations proportionnelles aux 3 composantes du champ magnétique terrestre et permet de déterminer l'orientation du mobile par rapport à celui-ci.

L'utilisation d'un magnétomètre suppose la stationnarité du champ magnétique de référence, or ce dernier peut être localement et temporairement perturbé par la présence de masses métalliques. La présence de champs magnétiques parasites causés par les cages d'ascenseur, câbles électriques, ... rend ainsi délicate son utilisation dans un environnement intérieur. De plus, si la précision moyenne est généralement bonne, la précision instantanée est quant à elle faible [Byrne 92].

1.2.1.4 Critiques

Ces systèmes permettent d'obtenir un flux d'informations important en position et en orientation. Leur mise en œuvre est relativement aisée et le traitement informatique nécessaire pour l'exploitation des données sensorielles peut être relativement sommaire. Il est à signaler également que les erreurs engendrées par l'utilisation de capteurs proprioceptifs sont pour la plupart modélisables.

Le problème propre à toute méthode de localisation relative provient de l'erreur cumulative générée à chaque intégration et qui augmente rapidement avec la distance parcourue, donc avec le temps. Pour pallier à ce problème, on peut fusionner les informations de ces différents types de capteurs [Larsson 98]. La précision de ces systèmes reste toutefois fonction de la distance parcourue ainsi que du type de trajectoire suivie [Hardt 97]. Un recalage de l'information de position par un système de localisation absolue est donc nécessaire dès que l'estimation de l'erreur de localisation devient trop importante.

Il apparaît également qu'un système de localisation relative restant par définition aveugle à l'environnement, la localisation calculée ne peut être exploitée immédiatement par une tâche de type « navigation ». Le robot n'ayant aucune information concernant son environnement, il ne peut que suivre des ordres du type : « avancer tout droit sur 10 mètres puis tourner à gauche de 45°, ... ». Toute modification de l'environnement ou apparition d'obstacles sur le trajet du robot risque alors de mettre en péril son déplacement.

1.2.2 Localisation absolue

Dans le cas des mesures de position absolue, des capteurs extéroceptifs fournissent des informations de position et d'orientation relatives à l'environnement du robot. On cherche donc à déterminer « *Qu'observe-t-on ?* », puis « *d'où observe-t-on ?* ». La localisation absolue nécessite donc toujours une représentation de l'environnement permettant au robot de disposer d'une banque de données regroupant les caractéristiques de références externes connues.

La sélection de ces références spatiales externes est extrêmement délicate et conditionne l'ensemble de la chaîne de localisation. Afin d'assurer l'autonomie du robot, elles doivent être perceptibles par le robot tout au long de son trajet. De plus, les informations de perception correspondantes doivent pouvoir être modélisées avec précision à l'aide de primitives géométriques afin d'être appariées aisément avec les références externes.

Il existe deux familles de références externes : les amers artificiels et les amers environnementaux. Les amers artificiels sont des balises passives ou actives devant être placées et référencées avec précision au sein de l'environnement dans le seul but de permettre la navigation d'un robot mobile. Les amers environnementaux sont des objets ou des éléments caractéristiques existant déjà au sein de l'environnement et ayant une fonction initiale autre que celle d'aider à la navigation du robot.

Nous présentons ci-après les méthodes de localisation absolue exploitant ces différents types d'amers afin d'en analyser les performances et les limites.

1.2.2.1 Amers artificiels

La localisation utilisant des amers artificiels consiste à détecter des balises placées à des positions connues dans le milieu d'évolution du robot. Le robot disposant d'une liste de ces balises, la principale tâche permettant de déterminer la localisation du robot consiste à identifier ces balises de façon robuste (Figure 1.5).

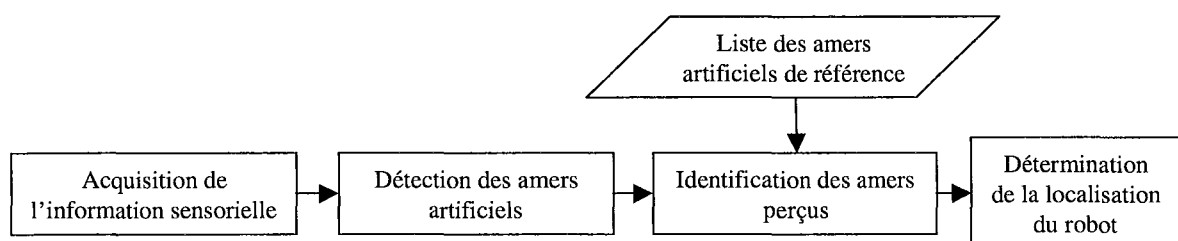


Figure 1.5 : Localisation par amers artificiels.

Deux familles d'amers artificiels sont couramment utilisées en robotique mobile. Il peut s'agir soit de balises passives telles que des réflecteurs optiques, soit de balises actives possédant leur source d'énergie propre telles que des émetteurs de sources lumineuses infrarouges ou bien encore d'antennes émettrices hyperfréquences. Nous ne considérons pas les solutions utilisant des câbles électriques enfouis dans le sol ou des peintures magnétiques [Tsumura 86] ; en effet, ces techniques ne permettent pas au robot de se déplacer librement dans un espace, mais seulement de suivre des chemins prédéfinis.

1.2.2.1.1 Les balises passives

Les balises passives ne sont pas interactives avec le système de perception du robot et ne peuvent donc que réfléchir une onde provenant du système de mesure. Elles sont conçues pour offrir un contraste maximal avec l'environnement. Elles sont d'un coût de revient peu élevé et d'une mise en œuvre facile.

Il s'agit généralement de balises fixes que le système de perception embarqué doit détecter. La télémétrie est alors particulièrement bien adaptée à ce type de problème. Le système MDARS (*Mobile Detection Assessment and Response System*) utilise des réflecteurs passifs en conjonction avec un couple de capteurs embarqués. Les balises sont constituées par de petites bandes verticales (2,5 cm) rétro réfléchies placées à hauteur fixe sur les objets immobiles de l'environnement de chaque côté de la trajectoire pouvant être suivie par le robot. Les deux capteurs sont placés de chaque côté du robot [Everett 94].

Dans le cas d'une balise embarquée et de capteurs fixes, la vision est également bien adaptée et peut constituer un système de perception performant. Le Laboratoire d'Automatique et d'Analyse des Systèmes de Toulouse présente ainsi un système de *vision globale* utilisant plusieurs caméras fixes réparties au sein de l'environnement afin de couvrir la totalité de l'espace de manœuvre du robot (3 caméras pour une zone de 5 m sur 10 m). Une balise géométrique composée de plusieurs points coplanaires est fixée sur le robot [Fleury 92]. Ces amers sont extraits des images en temps réel et permettent de réaliser une prédiction sur la localisation du robot et donc de limiter l'analyse des images à une zone d'intérêt réduite.

Chaque balise peut également être codée afin de permettre son identification individuelle dans le milieu d'évolution du robot. Leur forme et leur taille sont alors définies afin de fournir un maximum d'informations quelque soit l'angle sous lequel elles sont perçues. La vision est couramment utilisée car elle offre un large éventail de possibilités. La Figure 1.6 présente quelques exemples de balises codées.

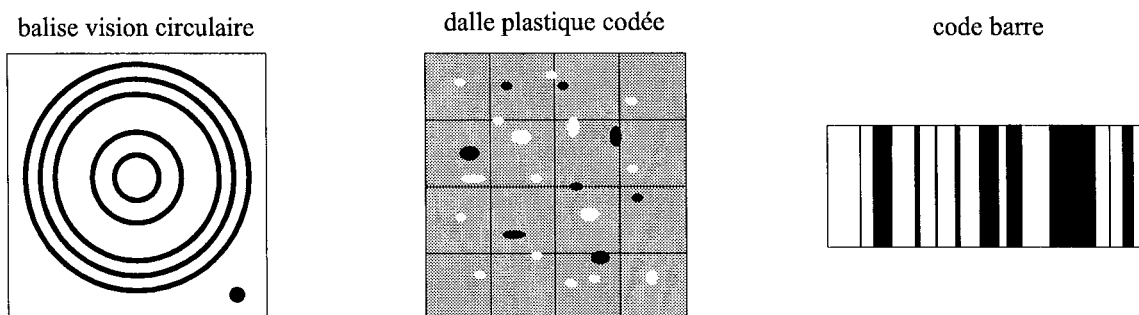


Figure 1.6 : Quelques exemples de balises passives codées.

Ainsi, le robot mobile développé à l'université de Western Australia [Ellepola 96] utilise une caméra CCD embarquée sur le robot et pointant vers le haut afin de reconnaître des balises circulaires situées au plafond de couloirs. Les balises sont placées aux intersections des couloirs et n'autorisent donc le calcul de localisation que dans certaines zones de l'environnement.

A l'opposé, il est possible de recourir à l'usage d'une caméra CCD couleur pointant vers le sol afin d'identifier des dalles situées à divers endroits stratégiques des couloirs d'un bâtiment [Cord 98]. Ces dalles sont caractérisées par des taches de couleur formant un motif aléatoire unique. Ce type de solution est toutefois fortement contraignant envers l'environnement.

1.2.2.1.2 Les balises actives

Les balises actives sont munies d'une source d'énergie propre et sont interactives avec le système de perception de l'engin mobile puisque ce sont elles qui émettent l'onde qui est captée par l'équipement de mesure. Les balises peuvent émettre en continu ou être déclenchées par ordre provenant de l'équipement embarqué sur le robot mobile.

Le système SIREM (*Système Intégré de Repérage d'Engins Mobiles*) développé à l'Ecole Centrale de Nantes se compose d'une caméra linéaire rotative et de balises émettrices lumineuses permettant au véhicule de se repérer dans l'espace en mesurant des angles d'azimut et de site [Bonnifait 96].

L'Université Kanazawa du Japon propose un système de localisation très peu coûteux basé sur deux émetteurs ultrasonores fixés au plafond [Seki 98]. L'émission des impulsions ultrasonores est commandée par le robot mobile qui envoie un code d'identification par liaison radio. Les deux récepteurs ultrasonores du robot mesurent alors le temps de vol des deux

impulsions afin de déterminer sa position et son orientation. La surface de déplacement autorisée avec une configuration de deux émetteurs est de $4 m^2$.

On peut également citer le système GPS (*Global Positioning System*) qui fournit une position précise dans les trois dimensions en utilisant la triangulation entre au moins trois satellites spécifiques d'éphémérides connues qui émettent en radiofréquence des codes d'identification. Un quatrième satellite est également nécessaire afin de pouvoir calculer les décalages d'horloges sur les satellites. L'intérêt du système GPS réside dans sa robustesse liée à la possibilité de se localiser en tout point d'un environnement extérieur avec une précision pouvant être de l'ordre du mètre [Botton 96]. Toutefois, son utilisation au sein de bâtiments est interdite du fait de l'impossibilité d'observer les satellites.

1.2.2.1.3 Détermination de la localisation du robot

Une fois les balises identifiées et connaissant leur position dans le repère de référence, le mobile calcule par triangulation sa position par rapport aux balises (Figure 1.7). Il est nécessaire de pouvoir observer au moins trois balises simultanément afin de déterminer la localisation du robot. Si plus de trois balises sont observées, alors la précision de la localisation peut être augmentée grâce à la redondance d'informations.

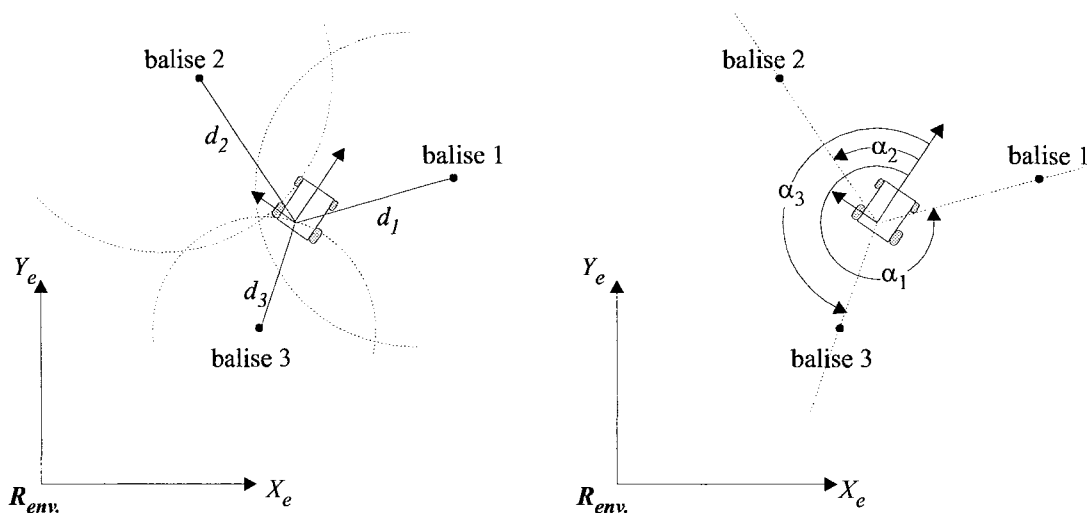


Figure 1.7 : Triangulation par télémétrie (figure de gauche) et goniométrie (figure de droite).

La triangulation peut faire appel à des mesures de distance du mobile aux balises considérées (système télémétrique), ou à des mesures d'angle entre les balises par rapport au mobile (système goniométrique), voire aux deux types conjointement [Sugihara 88]. Les systèmes télémétriques sont les mieux adaptés aux grandes étendues car leur précision varie peu avec la

distance. Les systèmes goniométriques quant à eux sont plus précis sur de courtes distances, l'erreur sur la position augmentant considérablement avec la distance. Les algorithmes les plus couramment utilisés sont la *recherche itérative*, la *triangulation géométrique*, l'*intersection de cercles géométriques* ainsi que la *méthode de Newton-Raphson* [Cohen 92]. Globalement, ces méthodes s'avèrent être très performantes pour une majorité de configurations de balises, mais chacune possède une ou plusieurs configurations la mettant en défaut.

La précision atteinte par ces algorithmes dépend principalement de la position relative et de l'angle entre le robot et les balises [Madsen 97]. En général, la précision diminue lorsque la distance augmente. De plus, il existe une gamme d'angles pour lesquels une bonne précision peut être atteinte, alors que la précision chute si l'on s'en écarte. La répartition des balises au sein de l'environnement conditionne donc fortement l'ensemble de la chaîne de localisation.

1.2.2.2 Amers environnementaux

La localisation utilisant des amers environnementaux consiste à exploiter des éléments caractéristiques du milieu d'évolution ayant une fonction autre que celle d'aider à la localisation du robot. Les phases de détection et d'identification de ces amers environnementaux sont par conséquent plus délicates à réaliser que dans le cas d'amers artificiels.

Le robot doit tout d'abord modéliser les structures de son environnement local à partir des informations sensorielles obtenues. Les éléments caractéristiques du modèle local sont alors mis en correspondance avec les amers de la carte théorique de l'environnement, puis la localisation du robot est déduite des appariements réalisés. Une fois la localisation déterminée, le robot peut éventuellement générer une mise à jour de la carte des amers environnementaux et ainsi enrichir ses connaissances au fur et à mesure de ses déplacements. Le schéma fonctionnel d'une méthode de localisation absolue par amers environnementaux est détaillé dans la Figure 1.8.

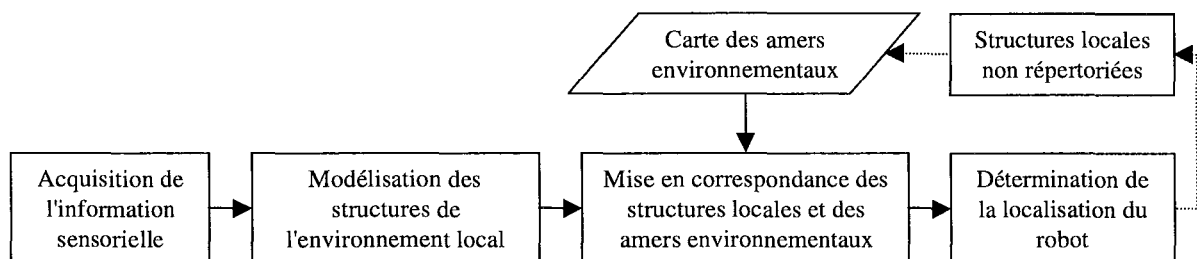


Figure 1.8 : Localisation par amers environnementaux.

1.2.2.2.1 Choix des amers environnementaux

Un milieu de type intérieur fortement structuré présente de nombreux éléments caractéristiques susceptibles d'être utilisés comme amers environnementaux. Il peut s'agir soit d'objets complexes tels que des éléments de mobilier, soit d'éléments simples tels que des pans ou des arêtes de mur.

Dans le cas *d'objets complexes*, des attributs de forme, de texture ou de couleurs sont généralement combinés afin de fournir une description complète de l'amer environnemental. Le choix des attributs décrivant l'amer est capital car il détermine la pertinence des opérations de modélisation et par conséquent la fiabilité de la phase de mise en correspondance.

Ces amers sont identifiables individuellement et sont généralement utilisés dans des routines de poursuite d'amers au fil des déplacements du robot. Il est ainsi possible de corriger l'orientation et la position d'un robot mobile par l'observation d'un seul amer à l'aide d'un système de vision monoculaire [Chen 91, Huttenlocher 94].

De tels amers nécessitent en contrepartie une modélisation tridimensionnelle complexe permettant de prendre en compte des facteurs d'échelle, de rotation et de translation.

Les méthodes mises en œuvre dans le cas d'objets complexes s'avèrent être spécifiques à un type d'environnement particulier et sont donc peu généralisables. De plus, elles sont peu tolérantes à l'encombrement du milieu d'évolution car sensibles aux phénomènes d'occultation.

Dans le cas *d'éléments simples de l'environnement*, une ou plusieurs primitives géométriques de type segment ou courbe sont alors utilisées pour décrire les amers [Kosaka 92, Vandorpe 96].

Les amers couramment utilisés possédant sensiblement les mêmes caractéristiques, il en résulte que leur description ne permet pas de les identifier individuellement. Il est par conséquent nécessaire de pouvoir en observer simultanément plusieurs afin de réaliser les mises en correspondance.

Le recours à des éléments simples tels que arêtes verticales, horizontales ou obliques de l'environnement est justifié par le fait que ces éléments sont généralement présents uniformément au sein de l'espace de manœuvre du robot, et que des phénomènes d'occultation ne posent que peu de problèmes si le nombre d'amers est suffisamment élevé.

1.2.2.2.2 Carte globale de l'environnement

Les amers environnementaux sont recensés dans une carte théorique décrivant l'environnement au sein duquel va évoluer le robot. Ce modèle global doit permettre un accès rapide aux informations utiles, mais également être très précis.

Le modèle global peut être décrit suivant des **relations topologiques** exprimées par un réseau de nœuds et d'arcs [Kortenkamp 94]. Les nœuds du réseau correspondent à des emplacements distincts de l'environnement. Les amers appartenant à un emplacement donné sont décrits dans un repère local propre à l'emplacement considéré. Les arcs représentent les chemins entre ces repères locaux. Ce type de représentation permet non seulement une représentation compacte de l'environnement puisque seules les zones importantes sont décrites, mais également de limiter le cumul des erreurs de cartographie.

Le modèle le plus utilisé décrit l'environnement suivant les **relations géométriques** liant chaque amer de l'environnement à un repère de référence global. Ce type de modèle peut être représenté par un quadrillage de cellules auxquelles sont accordées des certitudes d'occupation [Crowley 89] ou bien par une carte plus abstraite composée de primitives segments ou polygones [Castellanos 98]. Il s'avère délicat de générer un tel modèle pour de grands espaces car les erreurs de mesure s'accumulent et limitent la précision de la représentation. De nombreux efforts ont été réalisés en vue de diminuer les erreurs de cartographie lors de la génération du modèle global. Il apparaît que cette méthodologie atteint une limite et par conséquent les recherches actuelles s'orientent davantage vers des méthodes qui tiennent compte des erreurs liées à la détection.

1.2.2.2.3 Détection et modélisation des structures locales

Il s'agit d'extraire des informations sensorielles les structures susceptibles de correspondre à des amers environnementaux. Ces structures locales sont modélisées à l'aide de primitives géométriques (points, segments, courbes, plans, ...) devant représenter au mieux les données sensorielles tout en minimisant l'espace mémoire nécessaire pour les décrire. Cette étape doit viser à minimiser les phénomènes de sur-détection (*faux positif* : détecter un amer alors qu'il n'y en a pas) et de sous-détection (*vrai négatif* : ne pas détecter un amer alors qu'il y en a un). Ce sont en effet des artefacts susceptibles de causer des ambiguïtés lors de la phase de mise en correspondance. L'aspect perception et modélisation est traité au paragraphe 1.3 car il est étroitement lié au type de capteur mis en œuvre.

1.2.2.2.4 Mise en correspondance et détermination de la localisation

Les appariements entre les primitives locales et les amers environnementaux sont généralement réalisés selon des critères de longueur et/ou d'orientation des primitives. On peut distinguer trois grandes familles de méthodes de mise en correspondance : l'alignement, l'accumulation, et l'indexation.

Les méthodes par **alignement** ou « **prédiction-vérification** ». Un alignement est une transformation du repère des données sensorielles vers le repère du modèle global [Ayache 91]

La méthode consiste à calculer l'ensemble des alignements possibles en utilisant un minimum de correspondances entre les amers du modèle global et les primitives sensorielles. Chaque alignement hypothétique est ensuite vérifié en transformant les primitives observées dans le repère du modèle global et en les comparant avec les amers du modèle. Le processus de vérification consiste à examiner si les primitives observées transformées par l'alignement hypothétique coïncident avec des amers théoriques. Le recours à une assez bonne localisation a priori permet de limiter le nombre d'alignements hypothétiques devant être générés.

Une solution plus fine consiste à réaliser une mise en correspondance récursive des primitives, chaque nouvel appariement permettant d'améliorer la localisation du robot. Le filtre de Kalman Etendu est couramment utilisé dans ce but [Forsberg 95, Curan 95]. Afin que l'algorithme converge, il est nécessaire d'initialiser le filtre avec les appariements hypothétiques les plus fiables, ce qui suppose une information de localisation a priori. La localisation peut ainsi être calculée et affinée très rapidement mais il est nécessaire de disposer d'un modèle du bruit associé aux mesures. De plus, des minimums locaux peuvent fausser la localisation.

Les méthodes par **accumulation** consistent à accumuler dans l'espace des transformations les paramètres d'une transformation géométrique de type rotation-translation visant à superposer les primitives sensorielles et les amers.

On crée pour cela un accumulateur correspondant à l'espace des transformations (X, Y et Θ) que l'on discrétise suivant les pas de quantification $\delta_{x,y}$ et δ_{Θ} . [Vandorpe 96]. Des appariements de primitives et d'amers sont ensuite formés selon un critère d'orientation similaire (chaque primitive du modèle local peut être appariée avec une ou plusieurs primitives du modèle global). Pour chaque appariement réalisé, les cellules de l'accumulateur correspondantes aux transformations qui sont compatibles avec l'appariement sont incrémentées. La cellule correspondant au pic de l'accumulateur fournit alors les paramètres de localisation du robot.

La transformation de Hough est également utilisée pour détecter des primitives géométriques simples en accumulant des évidences dans leur espace paramétrique [Vergnet 94].

Les méthodes d'accumulation sont robustes face aux erreurs de segmentation des primitives et par la même aux occlusions partielles. Par contre, les dimensions de l'accumulateur ainsi que le temps de calcul augmentent rapidement si l'on désire une bonne précision ou si l'on n'a pas accès à une localisation a priori permettant de réduire l'espace des transformations.

L'indexation ou **hachage géométrique** consiste à calculer des invariants géométriques caractérisant des groupements particuliers d'amers. Ces invariants sont ensuite accumulés dans un espace d'apprentissage commun. La phase de reconnaissance consiste à calculer les mêmes types d'invariants à partir des groupements de primitives locales, puis à identifier les modèles les plus représentés dans l'espace d'apprentissage [Weiss 93].

Considérons par exemple l'appariement d'amers représentés par des points (projection d'arêtes verticales sur le plan) qui sont soumis à une similitude (rotation, translation et un facteur d'échelle) [Yang 96]. Le hachage géométrique utilise le fait qu'apparier une paire de points du modèle et une paire de points des données définit de manière unique une similitude, et le fait que les points préservent leurs coordonnées sous une similitude quelconque si ils sont représentés dans un repère local attaché aux deux mêmes points. Dans l'étape de pré-traitement qui est réalisée hors ligne, chaque paire de points ordonnés est utilisée pour définir un repère orthogonal, et la paire est appelée *paire de base*. Tous les autres points du modèle sont représentés par des coordonnées dans ce repère. Chaque coordonnée discrétisée est alors utilisée comme une entrée d'un tableau qui stocke toutes les paires « *modèle, paire de base* » pour chaque coordonnée. Dans l'étape de reconnaissance, une paire de points ordonnés arbitrairement est choisie dans les données et les coordonnées des autres points sont calculées en ligne dans cette base. Pour chaque coordonnée, l'entrée appropriée du tableau est examinée et le compteur correspondant aux paires « *modèle, paire de base* » est incrémenté. Si une certaine paire obtient le meilleur score, alors elle est associée à celle choisie dans la scène. La transformation de similarité entre les repères de coordonnées correspondant à ces paires de base est ensuite calculée et peut éventuellement être affinée en utilisant des appariements supplémentaires.

De manière générale, les méthodes à base d'indexation ou de hachage géométrique reprennent le principe de l'accumulation. Leur originalité réside dans la complexité supérieure des primitives utilisées ainsi que dans les méthodes de stockage et d'accès rapide à ces primitives. Ces méthodes permettent de rendre plus fiables les mises en correspondance puisqu'elles

portent simultanément sur plusieurs primitives. La contrepartie en est un temps accru nécessaire pour générer *hors ligne* les tables d'indexation pour de grands espaces.

1.2.2.3 Critiques

Les méthodes de localisation absolue basées sur l'utilisation d'amers artificiels s'avèrent être les plus fiables puisque les balises sont conçues pour être facilement détectables et identifiables. La détermination d'une estimation de position et d'orientation étant très rapide, le robot peut se déplacer sans interruptions. La précision est généralement très bonne si les balises sont uniformément réparties au sein de l'environnement.

En revanche, cette solution nécessite d'aménager la totalité du milieu d'évolution du robot ainsi que de prévoir une maintenance des balises. Il est donc délicat d'utiliser ces balises dans un environnement ouvert au public (masquage ou déplacement fortuit des balises, détériorations, etc.).

Les méthodes de localisation absolue exploitant des amers environnementaux ne nécessitent pas d'aménagement particulier du milieu d'évolution du robot. Elles peuvent donc être mises en œuvre dans le cadre d'un environnement différent mais de même type sans modifications lourdes. De plus, si l'environnement s'y prête, le robot mobile peut construire le modèle global de l'environnement au fur et à mesure de ses déplacements.

Le succès de ces méthodes repose essentiellement sur l'étape de mise en correspondance entre les caractéristiques observées et les amers environnementaux. Aussi, afin de s'assurer de la fiabilité des appariements réalisés, il est généralement fait appel à une information de localisation *a priori*. Une solution couramment utilisée consiste donc à recourir à un système de localisation relative permettant d'avoir à tout moment une estimation de la localisation du robot ainsi que de l'erreur de localisation. Lorsque l'erreur de localisation dépasse un seuil, le processus de localisation absolue est enclenché en prenant la localisation courante comme information *a priori*. Il est important de noter que ce type de solution ne permet qu'un recalage de l'information de position et ne peut donc par conséquent pas être utilisé lors des phases d'initialisation et de ré-initialisation du robot.

Il apparaît que la qualité des étapes de détection et de modélisation des amers environnementaux conditionnent fortement l'ensemble de la chaîne de traitement. Un système de perception capable de générer des primitives locales précises et fiables est donc indispensable.

1.3 LA PERCEPTION ARTIFICIELLE

La perception est une fonction fondamentale et incontournable vis-à-vis du concept d'autonomie. Elle consiste en un ensemble d'activités visant à saisir un certain nombre d'informations dans le but d'acquérir une connaissance et une compréhension du milieu d'évolution. Elle est nécessaire pour la sécurité du robot mobile, la modélisation de son environnement ainsi que pour la détermination de sa localisation.

Le choix d'un système de perception est donc dépendant du milieu d'évolution du robot mobile ainsi que des méthodes mises en œuvre pour remplir la mission dont est chargé le robot.

Les contraintes d'utilisation et les limites du système de perception utilisé (portée, résolution, ...) doivent ainsi être prises en compte et mises en adéquation avec la fonction du véhicule mobile autonome. Le critère du coût du capteur s'avère également être une contrainte très importante en pratique.

Nous présentons dans les paragraphes suivants les principaux capteurs extéroceptifs embarqués, puis nous étudions les principaux systèmes de perception pouvant être mis en œuvre (mono-capteur et combinaison de capteurs).

1.3.1 Les capteurs extéroceptifs embarqués

Les capteurs extéroceptifs embarqués sont principalement de deux natures : les capteurs télémétriques, basés sur l'émission et la réception d'une onde, et les capteurs de vision, qui fournissent des informations de luminance. Nous présentons ci-après ces deux grandes familles de capteurs.

1.3.1.1 Capteurs télémétriques

Un capteur télémétrique fournit une information de profondeur directement accessible sans aucun pré-traitement. L'acquisition des données télémétriques ainsi que leur modélisation sous forme de primitives géométriques est ainsi réalisable en temps réel [Siadat 97].

La baisse du coût des capteurs télémétriques fait qu'ils sont devenus applicables dans le cadre de la robotique mobile.

1.3.1.1.1 Télémétrie ultrasonique

La méthode de mesure consiste à exciter une céramique piezzo-électrique par des impulsions de fréquence égale à la fréquence propre de la pastille qui génère alors une onde ultrasonore. La mesure de distance est calculée à partir du temps de vol de l'onde. Cette onde ultrasonore a généralement une fréquence de 44 KHz et permet de mesurer des distances comprises entre 30 cm et 10 m.

Leur faible prix de revient et leur simplicité d'implantation en font les capteurs télémétriques les plus courants [Hoppenot 97, Blanes 98]. En revanche, ils possèdent une faible résolution angulaire, sont sensibles aux courants d'air et le faisceau peut se réfléchir de façons multiples. L'énergie réfléchi par le récepteur peut devenir rapidement indécélable si la surface est inclinée de plus de 30° par rapport à la direction de l'onde, d'où l'existence d'une zone aveugle rendant ce type de capteur difficilement utilisable pour des approches et des déplacements précis en environnement contraint.

Les capteurs ultrasoniques sont généralement présents sous la forme d'une ceinture d'une vingtaine de capteurs autour du corps du robot [Kortenkamp 94].

1.3.1.1.2 Télémétrie laser 2D

Le télémètre laser est basé sur l'émission d'une onde électromagnétique structurée qui permet d'obtenir un faisceau d'ondes lumineuses en phase. Le principe de la mesure repose soit sur la mesure du temps de vol d'une impulsion courte par un compteur rapide déclenché par l'émission et arrêté par la réception de l'impulsion, soit sur la mesure du déphasage entre l'onde incidente et l'onde réfléchi.

La résolution angulaire est excellente et l'angle d'incidence du faisceau sur la cible peut atteindre 70° selon la longueur d'onde et la nature de la surface. De plus une précision de l'ordre du centimètre peut être obtenue sur des distances de l'ordre de 50 mètres.

La télémétrie 2D est obtenue à l'aide d'un système de déviation en azimut du faisceau et permet de recueillir des informations en temps réel dans un plan généralement parallèle au sol [Gonzalez 95, Wallner 97]. Un pas de balayage de 0,5° est couramment utilisé et permet d'obtenir de 360 à 720 points de mesure pour un balayage de 180° à 360°.

1.3.1.1.3 Télémétrie laser 3D

La télémétrie 3D est obtenue à l'aide d'un système de balayage du faisceau à la fois en site et en azimuth. La déviation du faisceau peut être obtenue à l'aide d'une tourelle supportant le télémètre ou à l'aide de miroirs rotatifs.

Ce type de capteur permet d'obtenir des informations de profondeur sur une région de l'espace en particulier [Garcia 98, Mayora 98]. Ces systèmes permettent soit d'avoir une vue d'ensemble de la scène (pas de balayage élevé, donc large fenêtre possédant un maillage lâche) soit de focaliser l'attention sur un point de la scène (pas de balayage faible, donc petite fenêtre possédant un maillage serré). L'inconvénient majeur réside dans la lenteur des mouvements (équilibrage mécanique) et dans le coût d'un tel capteur qui reste assez élevé.

1.3.1.2 Capteurs de vision

Un capteur de vision fournit des informations de luminance et permet de détecter des informations de contraste par des techniques de segmentation [Fennema 90].

L'image sensorielle de l'environnement fournie par un capteur de vision est très proche de celle perçue par l'être humain. Si les amers susceptibles d'être détectés sont extrêmement variés, ils doivent toutefois être soigneusement choisis afin d'offrir un contraste suffisant avec le reste de l'environnement.

La vision est ainsi très performante en terme de portée, précision et quantité d'informations exploitables, mais le flux très important de données rend délicat et coûteux en temps de calcul, le traitement d'une image.

Les systèmes de vision utilisés en robotique mobile peuvent avoir une orientation fixe, ou être placés sur une tourelle possédant un degré de liberté en site et/ou azimuth permettant ainsi de couvrir l'ensemble de l'espace.

1.3.1.2.1 Vision monoculaire

La majorité des systèmes de vision mis en œuvre sont basés sur des capteurs CCD noir et blanc, mais des applications particulières peuvent tirer profit de l'utilisation de caméras CCD couleur [Crisman 93].

La plupart des caméras sont équipées d'un objectif à focale fixe dont la distance focale est déterminée en fonction du milieu d'évolution du robot. La caméra peut toutefois être équipée d'un objectif permettant d'élargir le champ de vision et donc d'augmenter le nombre d'amas accessibles [Couroux 97], ou au contraire de focaliser l'attention sur un point particulier et ainsi améliorer son observation.

Le recours à une lentille de type *fish eye* [Kurata 98], de même que l'utilisation d'un miroir conique [Delahoche 96], permettent d'obtenir une vue panoramique de la scène.

1.3.1.2.2 Stéréovision

Le principe de la stéréovision consiste à observer une même scène avec deux caméras (ou plus) légèrement décalées l'une de l'autre. Connaissant la géométrie exacte du système stéréoscopique, la méthode consiste à mettre en correspondance les informations bi-dimensionnelles homologues des deux images et de reconstruire ensuite l'information 3D de la scène par triangulation [Asensio 98].

Les méthodes d'appariement proposées diffèrent selon les primitives utilisées (points ou segments de droite) et suivant la structure et la géométrie du système d'acquisition (nombre de caméras, axes optiques parallèles ou non). Cette phase de mise en correspondance des informations engendre des temps de calcul importants. En effet, il est nécessaire d'appliquer des contraintes locales (contrainte épipolaire, unicité de l'appariement, ordre d'apparition de gauche à droite, ...) puis de lever les ambiguïtés résiduelles sur les appariements en examinant la consistance globale.

La stéréovision constitue l'un des moyens de perception les plus performants, en revanche son principal handicap réside dans l'aspect combinatoire de la mise en correspondance qui peut conduire à un nombre d'opérations qui tend vers l'infini (explosion combinatoire).

1.3.2 Les systèmes mono-capteur

La mise en œuvre d'un capteur extéroceptif sur un robot mobile nécessite de prendre en compte l'encombrement du capteur embarqué ainsi que sa consommation en énergie qui peut limiter l'autonomie du robot. De plus, le coût du système de traitement de l'information qui lui est associé doit aussi être pris en considération si des développements industriels sont envisagés. Compte tenu de ces contraintes, de nombreuses équipes ont développé des systèmes de perception reposant sur l'utilisation d'un seul capteur extéroceptif. Cette solution permet également de mettre en œuvre des robots mobiles de petits gabarits.

Nous présentons ci-après les différents systèmes mono-capteur et étudions l'aspect modélisation des structures locales de l'environnement.

1.3.2.1 Télémétrie

Les amers environnementaux détectables par télémétrie sont principalement des plans verticaux tels que des pans de mur, des armoires, ... Compte tenu des hypothèses généralement admises dans le cadre d'un environnement intérieur structuré (sol plan notamment), une perception bi-dimensionnelle horizontale de l'environnement suffit pour définir des plans verticaux.

La Figure 1.9 illustre la modélisation des structures locales de l'environnement qui peut être fondée soit sur l'exploitation des données brutes (figure de gauche), soit sur la génération de primitives géométriques (figure de droite).

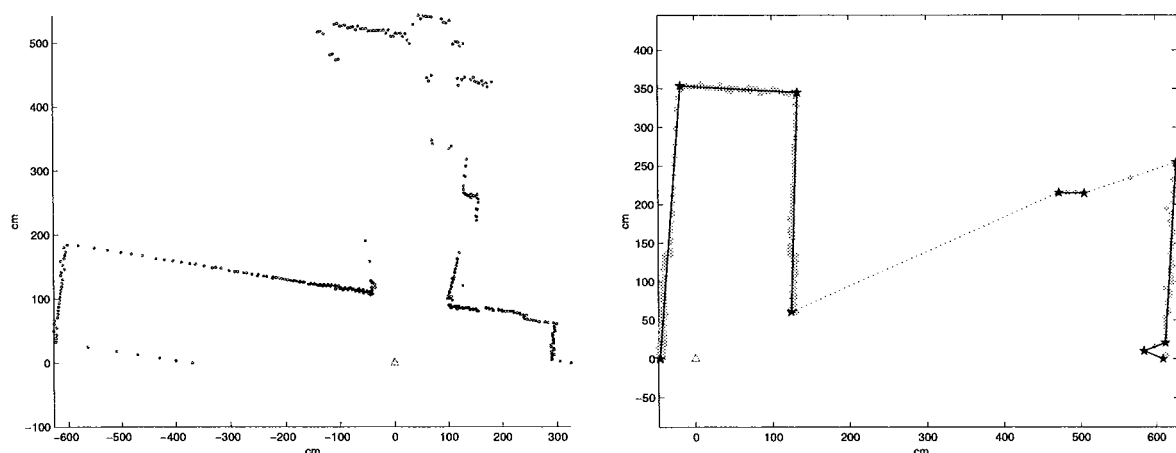


Figure 1.9 : Modélisation des données sensorielles dans le cadre de la télémétrie.

L'exploitation directe des données brutes de la profilométrie assure un gain de temps dans le traitement de l'information sensorielle et donne la possibilité d'élaborer facilement une planification du chemin du robot dans des environnements peu contraints.

Wallner du groupe GRAVIR-PRIMA réalise ainsi l'analyse en composantes principales d'une profilométrie sur 180° [Wallner 97]. Les caractéristiques générées fournissent une représentation canonique de l'environnement occupant très peu de place mémoire. Il est ainsi possible de calculer une représentation de l'environnement propre à chaque localisation possible du robot et de comparer ces modèles globaux avec le modèle local afin de déterminer la position et l'orientation du robot.

Les relations géométriques entre chaque point de mesure et son voisin peuvent être analysées. Weiss de l'Université de Kaiserslautern propose ainsi un système basé sur l'analyse d'une profilométrie couvrant un champ de 360°. La distribution statistique de l'orientation des segments formés par deux points de mesure successifs constitue un histogramme d'angle permettant de mettre en évidence l'orientation des structures principales de l'environnement. La détermination de la localisation est alors réalisée par la corrélation des histogrammes d'angles de la profilométrie acquise et d'une profilométrie de référence [Weiss 94].

La génération de primitives géométriques simples permet de décrire l'environnement en occupant un minimum d'espace mémoire. Elle ne peut toutefois être mise en œuvre que dans des environnements fortement contraints de type polygonaux. Différents types de primitives sont alors susceptibles d'être employées.

Les primitives de type segments de droites sont les plus couramment utilisées du fait de leur géométrie simple et de l'invariance de leur direction par projection suivant certains axes du repère sensoriel. Le système de localisation développé pour le robot LIAS de l'université de Leuven est ainsi fondé sur la génération de primitives de type segment et de type agglomérat de points issues de la profilométrie [Vandorpe 96].

Des primitives de niveau supérieur constituées à partir de primitives simples peuvent également être définies ; par exemple, des primitives coins formées par deux primitives segments connexes. A chaque primitive coin correspond une arête verticale située au point de jonction des primitives segments [Skrzypczynski 98].

1.3.2.2 Vision

Bien que les amers environnementaux détectables par la vision au sein d'un environnement intérieur structuré soient très variés, l'aspect modélisation est généralement limité à la génération de primitives de type segment horizontaux, verticaux ou obliques, comme l'illustre la Figure 1.10.

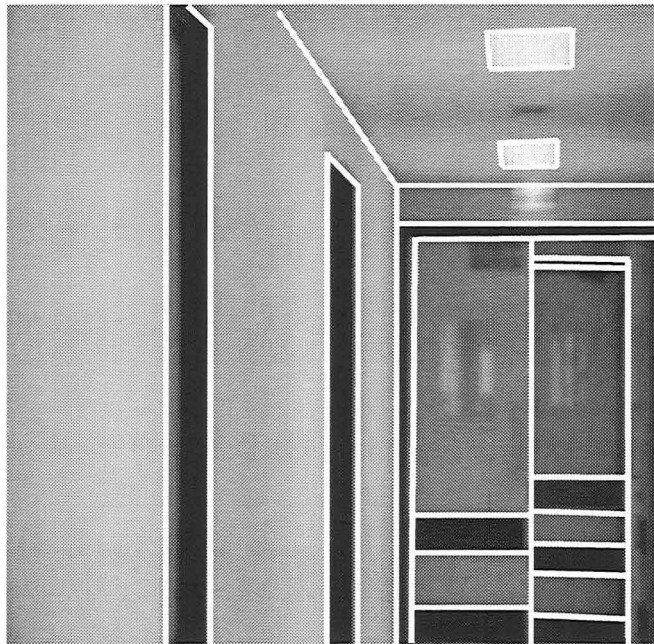


Figure 1.10 : Amers environnementaux typiques de la vision.

Une approche intéressante pour l'extraction de primitives vision est d'avoir un modèle détaillé de l'environnement permettant d'émettre des hypothèses à la fois en terme de *localisation spatiale* d'entités de la scène, mais également sur les *opérateurs* capables de détecter ces objets dans une image. Ce modèle global de l'environnement est le plus souvent un modèle 3D « fil de fer ».

Kosaka propose ainsi un système de localisation fondé sur la comparaison d'un modèle prédit composé d'amers de type segments verticaux avec les primitives extraites de l'image acquise [Kosaka 92]. Le système de navigation possède à tout moment une estimation de l'incertitude liée à la position du robot au cours de ses mouvements. Cette estimation de l'incertitude est alors utilisée afin de prédire les limites sur la position et l'orientation des amers devant être observés pour la position supposée du robot. La génération d'amers prédits réduit considérablement le temps de recherche des appariements entre les caractéristiques observées par le système de vision et les amers théoriques. Un des problèmes majeurs pouvant mettre cette technique en échec est celui des occlusions des amers.

Une alternative à l'utilisation d'un modèle prédit consiste à poursuivre des éléments caractéristiques de l'environnement au fil des déplacements du robot [Guerrero 98]. Guerrero corrige ainsi l'orientation d'un robot mobile à l'aide d'une caméra monoculaire non calibrée. Les lignes non verticales extraites du plan image permettent tout d'abord d'obtenir le point de fuite de l'image. L'orientation du robot est ensuite déduite de la mise en correspondance des lignes verticales de deux images successives.

La stéréovision permet quant à elle de générer des primitives très riches puisque celles-ci allient des informations de contraste avec des informations de profondeur [Eberst 98]. De deux à trois caméras peuvent alors être mises en œuvre afin d'opérer la perception tridimensionnelle de l'environnement d'un robot mobile. Les principaux handicaps de cette technique, outre le nombre de capteurs, sont l'alourdissement des traitements (dépendant des processeurs utilisés) ainsi qu'une fenêtre d'étude réduite du fait des contraintes d'épipolarité.

1.3.2.3 Critiques

La télémétrie permet d'obtenir des informations de profondeur précises sur l'environnement proche du robot. La génération en temps réel de primitives immédiatement exploitables est également un des atouts de ce type de capteur. Toutefois, le type d'amers environnementaux utilisables est limité et la résolution angulaire fait qu'il est délicat de générer des primitives fiables sur des distances de plus de 20 mètres.

Les systèmes de vision sont à même de percevoir un grand nombre d'amers environnementaux de différents types et peuvent fournir une information sensorielle très riche. Le principal handicap de ces capteurs réside dans le temps de calcul très élevé dès lors que l'image entière est analysée.

Bien que les systèmes de perception mono-capteurs soient très performants, il est généralement nécessaire de disposer d'une localisation a priori permettant d'initialiser les recherches d'appariement. En effet, les primitives n'étant décrites que par un nombre limité d'attributs, leur mise en correspondance peut déboucher sur une explosion combinatoire. Ces systèmes ne peuvent donc pas être utilisés en cas de perte totale de référence. Il s'avère également difficile de développer un système muni d'un capteur unique capable de détecter différents types d'amers environnementaux. En effet, en cas de présence d'obstacles, un amer peut ne pas être reconnu, ou bien des objets de l'environnement peuvent être confondus avec des amers. Ceci induit de sérieux risques de fausse interprétation de l'environnement pouvant déboucher sur une mauvaise localisation.

1.3.3 Combinaison de capteurs

Au vu des limites des systèmes de perception mono-capteurs, de nombreuses équipes de recherche se sont orientées vers l'utilisation conjointe de plusieurs capteurs extéroceptifs. Il est alors nécessaire de réaliser un calibrage entre les différents capteurs afin de pouvoir exprimer les coordonnées des différentes primitives dans un repère commun.

L'exploitation des données provenant de différentes sources d'information à des instants donnés communs ou différents permet non seulement de diversifier les types d'amers susceptibles d'être utilisés mais également de faciliter leur détection par une redondance ou une complémentarité d'informations.

Les systèmes faisant appel à différents capteurs extéroceptifs en robotique mobile sont constitués essentiellement par des binômes « télémétrie-vision ». La complémentarité de ces deux sources permet en effet d'obtenir des informations de différentes natures concernant un même type d'objet.

On obtient ainsi une information plus riche, plus précise et plus fiable puisque l'information résultante reflète non seulement les données fournies par chaque capteur mais également l'information qui n'aurait pas pu être inférée par aucune des sources prises séparément.

L'exploitation des données sensorielles provenant de différents capteurs A et B peut se faire de plusieurs manières [Ishikawa 92] :

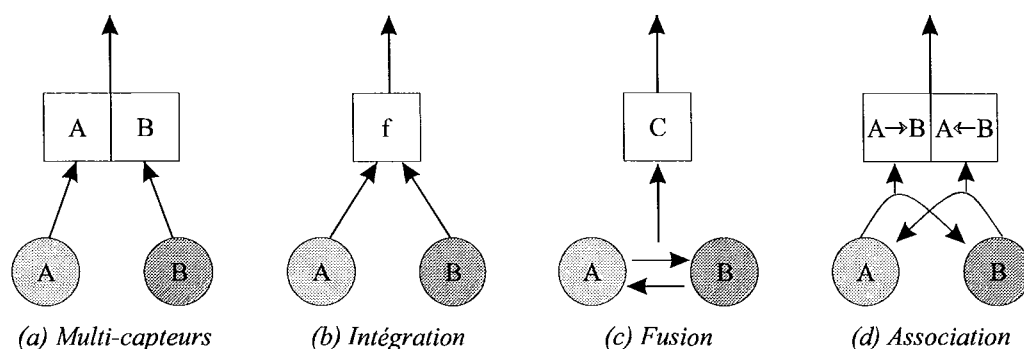


Figure 1.11 : Configuration des systèmes combinant différents capteurs.

Système multi-capteurs (Figure 1.11a) : $A, B \rightarrow A + B$ ou (A, B) . Le processus générant l'information de sortie consiste en un traitement parallèle et complémentaire des informations sensorielles. Il n'y a pas de traitement entre A et B . Cette configuration permet la standardisation du signal de sortie, l'intégration hardware des capteurs ainsi qu'un contrôle sélectif des capteurs. C'est un cumul d'informations.

Intégration (Figure 1.11b) : $A, B \rightarrow f(A, B)$. L'information de sortie est définie de façon cohérente à l'aide d'une certaine fonction de calcul f faisant appel à des entrées de différents types.

Fusion (Figure 1.11c) : $A, B \rightarrow C$. La détermination d'une nouvelle représentation sensorielle est obtenue à partir des relations mutuelles entre les différentes entrées.

Association (Figure 1.11d) : $A, B \rightarrow (A \rightarrow B, B \rightarrow A)$. Il s'agit d'un processus dans lequel les entrées coopèrent directement entre elles.

Nous détaillons ci-après les différentes approches en présentant pour chacune d'elle quelques travaux significatifs déjà réalisés. Nous étudierons particulièrement l'aspect modélisation des structures locales de l'environnement en vue de l'exploitation pour une tâche de localisation.

1.3.3.1 Systèmes multi-capteurs

Les informations issues de chaque capteur sont traitées sans interaction entre elles.

Les informations de contraste et de profondeur peuvent être utilisées en fonction de la tâche à réaliser. Le système de perception du robot CARMEL développé à l'université du Michigan est composé d'une ceinture de 24 capteurs ultrasons ainsi que d'un système de vision monoculaire monté sur une tourelle un axe [Kortenkamp 94]. CARMEL utilise une carte globale topologique dont chaque nœud correspond à une région de l'environnement caractérisée par une signature sonar typique. A chaque nœud correspond plusieurs images de la scène contenant des amers environnementaux perçus sous différents angles. Les informations télémétriques sont ainsi utilisées pour la navigation et pour déterminer à quels endroits le robot peut extraire des informations vision lui permettant de se localiser.

Les informations issues de chaque capteur peuvent être traitées séparément puis mises en commun afin de fournir une information de haut niveau. Courtney propose de générer une carte de télémétrie ultrasonore ainsi qu'une carte de vision basées sur les grilles d'occupation [Courtney 94]. Des descripteurs géométriques sont alors utilisés pour représenter les informations spatiales contenues dans chaque grille d'occupation. La totalité de ces caractéristiques est ensuite exploitée afin de mettre en correspondance la carte locale multisensorielle avec une des cartes globales décrivant l'environnement.

1.3.3.2 Intégration de capteurs

Les informations issues de chaque capteur sont directement utilisées pour exprimer les informations sensorielles dans les trois dimensions. Les deux approches envisageables sont illustrées par la Figure 1.12.

Le robot VEGA développé au LIRMM utilise la lumière structurée pour détecter les éléments caractéristiques de l'environnement [Aldon 94]. Cette technique consiste à projeter un plan de lumière structurée et à observer grâce à une caméra la ligne d'intersection de ce plan avec l'environnement. Après un filtrage infrarouge de l'image, seule la ligne laser est perçue et l'information de relief est alors obtenue par triangulation.

Le même principe est utilisé dans le cadre du projet ROBTET, mais l'adjonction d'un miroir constituant un système de déflexion en site du faisceau laser permet d'obtenir des informations de profondeur tridimensionnelles [Lamela 98]. La résolution de ce système est un maillage carré de 160 points de mesure de coté à une profondeur de 7 mètres. La précision est excellente mais des problèmes d'occlusion peuvent se poser puisque les deux capteurs sont éloignés l'un de l'autre.

Une solution intégrant un télémètre laser et une caméra CCD dans un seul capteur peut également être envisagée. C'est le cas du système appelé *Laser Eye* [Jasiobedzki 93] équipant le robot mobile ARK (*Autonomous Robot for a Known Environment*). Les deux capteurs sont couplés de telle sorte que leurs axes optiques soient parallèles. Un miroir possédant une réflectivité différente pour la lumière visible et la lumière infrarouge est utilisé afin d'obtenir la colinéarité des axes optiques. La mesure de distance réalisée par le télémètre est alors référencée par rapport au centre de l'image de la caméra.

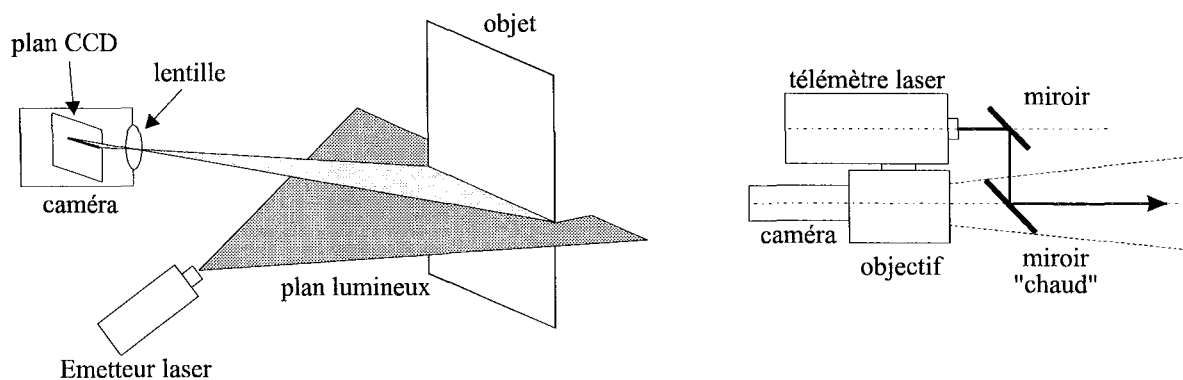


Figure 1.12 : Lumière structurée (figure de gauche), Laser Eye (figure de droite).

1.3.3.3 Fusion de primitives

Un capteur peut être utilisé simplement en vue d'accroître la pertinence des informations issues de l'autre capteur, comme l'illustrent les deux méthodes présentées ci-dessous.

Les informations laser peuvent être utilisées comme source d'information supplémentaire permettant de compléter la description des informations vision. Yang propose ainsi d'utiliser un télémètre laser 2D réalisant une profilométrie sur 270° et deux caméras pointant dans des directions différentes [Yang 96]. Les deux caméras sont calibrées séparément par rapport au télémètre laser. Les arêtes verticales extraites de chacun des plans image sont projetées dans le plan laser afin de déterminer leurs coordonnées dans le repère robot.

Les informations vision peuvent également être utilisées afin de valider des primitives laser. Le système de perception équipant le robot SARAH développé à l'Université Jules Verne de Picardie est ainsi composé d'un télémètre laser rotatif 2D réalisant une profilométrie sur 360° ainsi que d'un système de vision monoculaire panoramique permettant un champ de vision de 360° [Delahoche 96]. Les segments verticaux extraits des informations vision sont utilisés pour valider les primitives coins formées par deux segments laser successifs. La localisation se fait alors par triangulation sur les sommets des primitives coins.

Une autre approche consiste à générer des primitives à partir des données des capteurs puis à les analyser en vue d'obtenir des informations de plus haut niveau.

Castellanos propose ainsi de générer des primitives de type demi-plan vertical à partir des segments horizontaux extraits des informations laser, et des segments verticaux extraits des informations vision [Castellanos 98].

Le système de perception monté sur le robot PRIAMOS développé à l'Université de Karlsruhe est composé d'un télémètre laser 2D réalisant une profilométrie sur 180° , associé au système de vision trinoculaire KASTOR [Weckesser 97]. Les informations 3D fournies par la stéréovision ainsi que les informations laser 2D sont tout d'abord utilisées séparément pour construire deux cartes géométriques de l'environnement local. Les reconstructions 3D et 2D sont ensuite fusionnées en une seule et même représentation 3D dont les primitives sont plus fiables car observées de plusieurs sources.

1.3.3.4 Association

Les informations issues de chaque capteur sont utilisées à différents niveaux.

Le robot mobile développé par l'Université de Munich est un exemple typique de ce type d'architecture complexe [Eberst 98]. Un télémètre laser 2D panoramique permet de générer des primitives segment pouvant être interprétées comme des structures de l'environnement. Ces primitives sont utilisées pour former une carte locale dynamique. Les primitives vision correspondantes à ces structures sont ensuite prédites. Un système de stéréovision binoculaire est alors utilisé pour rechercher des primitives vision de la scène. Si une primitive prédite est appariée à une primitive observée, alors elles sont fusionnées pour tenir compte des incertitudes de mesure. Si une primitive observée n'est pas appariée, alors elle intègre la carte locale dynamique en tant que nouvel attribut. Si une primitive prédite n'est pas appariée, alors son taux de confiance diminue jusqu'à ce qu'elle soit éliminée de la carte locale dynamique.

1.3.3.5 Critiques

Les résultats obtenus par les systèmes multi-capteurs sont généralement satisfaisants dans le cadre d'environnements simples (très peu d'objets non répertoriés dans le modèle global de l'environnement sont présents dans le monde réel). En effet, les informations sensorielles étant utilisées sans qu'il y ait interaction, leur complémentarité n'est donc pas exploitée.

L'intégration directe permet d'obtenir des informations tridimensionnelles en raisonnant directement sur les mesures. Cependant, l'existence de données bruitées ainsi que des points de mesure aberrants peuvent conduire rapidement à des interprétations erronées.

La fusion de primitives propres à chaque capteur permet de générer des primitives fiables et riches en information contextuelle.

L'association des primitives issues de chaque capteur s'intègre dans une structure permettant une représentation hiérarchisée des informations sensorielles avec un retour possible à différents niveaux intermédiaires. Cette méthode permet de générer de façon incrémentale une carte fiable de l'environnement. Elle est par contre délicate à mettre en œuvre du point de vue algorithmique.

1.4 SYSTEME DE LOCALISATION ABSOLUE MIS EN ŒUVRE

Le but de nos travaux est de développer une méthode de localisation pouvant répondre aux problèmes posés par la navigation autonome des engins mobiles dans un environnement de type intérieur dynamique. Le site expérimental choisit est constitué d'une suite de couloirs sur un même niveau donnant accès à des pièces (bureaux).

Nous présentons dans ce paragraphe les solutions retenues au vu des systèmes étudiés dans les paragraphes précédents.

1.4.1 Choix d'une méthode de localisation

Le milieu d'évolution du robot mobile est connu, mais non coopératif avec le robot. Ainsi, l'environnement peut partiellement évoluer au cours du temps de façon imprévisible à cause de la présence aléatoire d'agents extérieurs (présence de piétons, déplacement ou dépôt d'objets tels que chaises, cartons, meubles, éléments de décoration ou d'information, ...) et l'espace de travail n'est pas entièrement agencé ou conditionné pour favoriser les déplacements du robot.

Ce manque de coopération entre l'espace de travail et le robot implique que celui-ci puisse s'adapter à l'environnement et lui impose donc de posséder une certaine autonomie décisionnelle et opérationnelle.

- Le recours à une méthode de localisation uniquement relative ne peut être envisagé compte tenu de la présence d'obstacles mobiles pouvant gêner le robot dans ses déplacements.
- Une méthode de localisation absolue faisant appel à des amers artificiels ne peut également être retenue compte tenu des modifications de l'environnement qu'elle implique.
- Afin de garantir la portabilité de la solution, nous ne pouvons pas utiliser d'amers environnementaux complexes propres à notre site expérimental.

Il est donc nécessaire d'avoir recours à une **méthode de localisation absolue** faisant appel à **des amers environnementaux simples** pouvant être présents dans tout environnement intérieur de type polygonal.

1.4.1.1 Sélection des amers environnementaux

Les structures de l'environnement susceptibles d'être retenues comme amers doivent satisfaire à plusieurs contraintes.

- Elles doivent être aisément détectables par le système de perception afin de limiter le nombre de *vrais négatifs* (ne pas détecter un amer alors qu'il y en avait un).
- Il doit s'agir de structures pouvant être dissociées des autres éléments présents dans l'environnement afin de limiter le nombre de *faux positifs* (détecter un amer alors qu'il n'y en a pas).
- Elles doivent être réparties de façon relativement homogène au sein de l'environnement afin de s'assurer la possibilité d'en percevoir suffisamment en tout point de l'espace de travail du robot. L'impact des phénomènes d'occultation sera ainsi limité.
- Leur localisation dans le repère de référence R_{env} doit être connue avec précision. Le recours à des amers immobiles est le meilleur garant de cette précision.
- Leur modélisation doit permettre de mettre en évidence le plus grand nombre possible de traits caractéristiques.

Compte tenu de ces contraintes, notre choix s'est porté sur des amers environnementaux de type **arête verticale** causées par des objets fixes de l'environnement. Toutefois, afin de limiter le nombre d'amers environnementaux devant être modélisés, nous ne retenons que les arêtes d'une longueur supérieure à un seuil (150 cm). Il peut donc s'agir d'angles de mur ou de mobilier, d'encadrements de porte ou encore de différences de contraste entre portions de mur.

Chaque amer A_i peut ainsi être modélisé sous la forme d'une primitive géométrique simple de type segment vertical caractérisé par les attributs suivants :

$$A_i : \{(x_i, y_i), \Omega_i, V_i\} \quad (1.1)$$

Avec :	(x_i, y_i), coordonnées de la projection de l'arête A_i dans le plan (X_e, Y_e) du repère de référence R_{env}. Ω, l'angle formé par les plans verticaux servant de support à A_i. V_i, liste des amers A appartenant également aux plans verticaux support de A_i.
--------	--

La longueur exacte de l'arête n'a pas été retenue comme critère descriptif car des phénomènes d'occultation ou de bruit de mesure peuvent tronquer la primitive segment vertical correspondant à l'arête.

La Figure 1.13 illustre la description d'un amer typique de notre environnement, à savoir une arête verticale causée par un angle de mur.

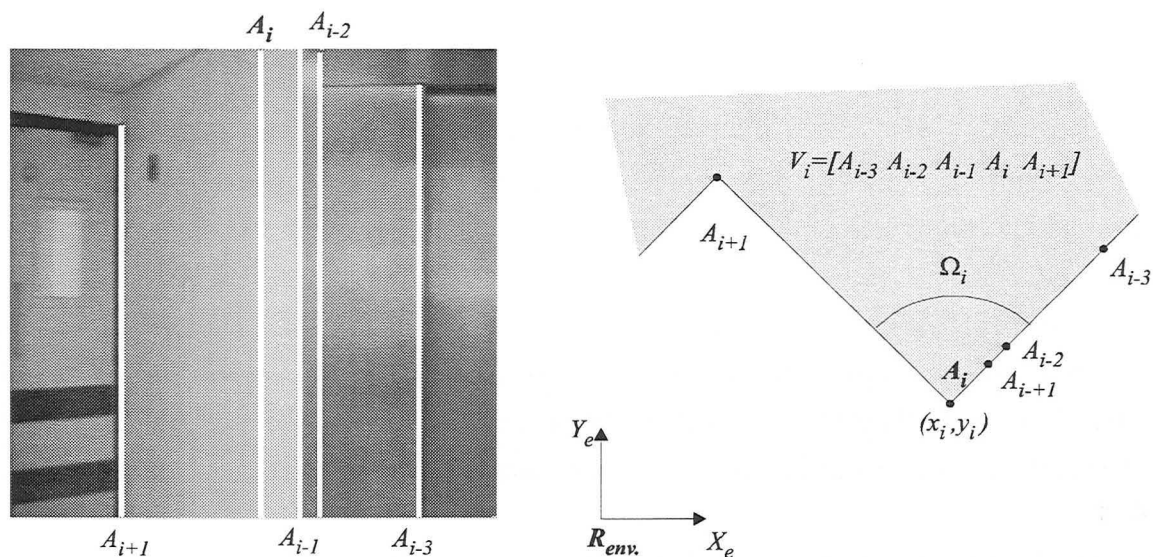


Figure 1.13 : Amer environnemental A_i .

1.4.1.2 Sélection d'une méthode de mise en correspondance

Les systèmes de localisation absolue couramment utilisés font appel à une information de localisation a priori fournie par un système de localisation relative. Cette information permet de prédire la position des amers dans l'environnement et ainsi de limiter l'ensemble des appariements possibles entre les primitives observées et les amers du modèle global. Il s'agit alors d'un recalage de l'information de position et d'orientation du robot mobile.

Or le problème de la localisation sans utiliser d'informations de localisation a priori est crucial pour assurer l'autonomie du robot, et notamment permettre:

- l'initialisation automatique du système,
- de vérifier de façon indépendante la viabilité des mises en correspondances au fil du temps,
- de ré-initialiser le système si l'information de localisation se dégrade fortement.

Afin d'assurer l'autonomie de ROMANE, nous avons donc choisit de développer une méthode de mise en correspondance pouvant être mise en œuvre avec un minimum d'informations concernant la localisation du robot. Nous considérons ainsi que le robot sait dans quelle zone de l'environnement il se trouve mais qu'il ne dispose pas d'informations suffisamment précises pour émettre des hypothèses concernant la position des amers. Nous nous plaçons donc dans le cas de figure où l'information de localisation a priori est très fortement perturbée (glissement des roues, perturbation magnétique importante, appariements erronés précédemment réalisés, ...).

- Les techniques d'appariement à base d'accumulation ne sont dans ce cas pas exploitables du fait de la taille prohibitive de l'accumulateur nécessaire pour couvrir l'ensemble de l'espace de travail du robot.
- Une méthode par alignement n'est pas non plus satisfaisante car les amers environnementaux disponibles ne sont pas suffisamment distincts les uns des autres pour être identifiés facilement. Les combinaisons d'appariements possibles entre les primitives observées et les amers de la carte globale sont alors trop nombreuses.

Il en découle que seule une **méthode d'appariement par indexation** semble convenir à ce type de problème ainsi qu'aux amers retenus. Nous orienterons plus particulièrement notre recherche vers l'expression de relations liant plusieurs amers entre eux car ce type de représentation permet de générer une description très riche en informations. Nous réaliserons ainsi des appariements entre des groupements d'amers et de primitives observées, puis entre amers et primitives observées. Nous pourrions notamment mettre à profit les informations de voisinage liant les amers entre eux.

1.4.2 Choix d'un système de perception

Afin de déterminer la localisation d'un robot mobile à l'aide d'amers environnementaux, il est vital d'obtenir des données sensorielles les plus riches possibles en information. Les primitives observées doivent également être très fiables et correspondre à des structures réelles de l'environnement.

Le choix du système de perception ainsi que du traitement de l'information sensorielle est donc crucial.

1.4.2.1 Choix des capteurs

Il est difficile de développer un système de perception doté d'un capteur unique dans le cadre d'un environnement dynamique. En effet, en cas de présence d'obstacles, un amer peut ne pas être reconnu ou bien des objets de l'environnement peuvent être confondus avec des amers. Bien que généralement assez lourd à mettre en œuvre, tant du point de vue matériel que du point de vue algorithmique, un système faisant appel à une combinaison de capteurs extéroceptifs est le plus adapté à nos besoins.

C'est pourquoi nous avons recours à un *système de perception mixte télémétrie-vision*. La complémentarité de cet équipement rendant possible l'obtention d'informations de différentes natures concernant un même type d'objet.

Nous avons choisi un *capteur télémétrique laser 2D* permettant d'obtenir une perception de l'environnement situé sur un plan parallèle au sol. Compte tenu des hypothèses concernant l'espace de travail, cela suffit pour détecter les plans verticaux et également délimiter l'espace navigable du robot. L'utilisation des ultrasons a été rejetée en raison du manque de précision des mesures.

L'analyse des données de profondeur permet de mettre en évidence des discontinuités susceptibles de correspondre à des arêtes verticales de type angle de mur. Toutefois, comme l'illustre la Figure 1.14, la résolution limitée du télémètre fait qu'un encadrement de porte ne sera pas détecté si la porte est fermée, engendrant ainsi une perte d'information utile. De plus de nombreuses discontinuités parasites peuvent être causées par des objets non répertoriés comme arête de mur, engendrant ainsi une détection de faux amers.

Le recours à un *système de vision monoculaire noir et blanc* permet d'obtenir des informations liées au contraste des objets de la scène. Une solution à base de stéréovision, coûteuse en temps de calcul, ne s'impose pas puisque des informations de profondeur pourront être tirées de la source télémétrique. Une caméra couleur aurait permis d'obtenir un supplément d'informations mais elle engendre une plus grande complexité dans le traitement des données. L'utilisation d'une *tourelle possédant deux degrés de liberté en site et en azimuth* autorise une perception panoramique de l'ensemble de l'espace entourant le robot. Compte tenu des hypothèses faites sur l'environnement du robot, nous n'utiliserons que l'angle d'azimut. Enfin, l'adjonction d'un *objectif réglable en focale et mise au point* va permettre de réaliser une inspection fine de certaines zones de l'environnement tout en assurant la netteté de l'image.

Le module de vision est bien adapté à la détection d'arêtes verticales car ce sont des primitives peu sensibles à la déformation due aux problèmes de projection perspective. Il se pose toutefois le problème de la sur-détection ainsi que de la très grande richesse en information. L'analyse d'une image ne peut en effet se faire que si l'on sait ce que l'on cherche et où on doit le chercher.

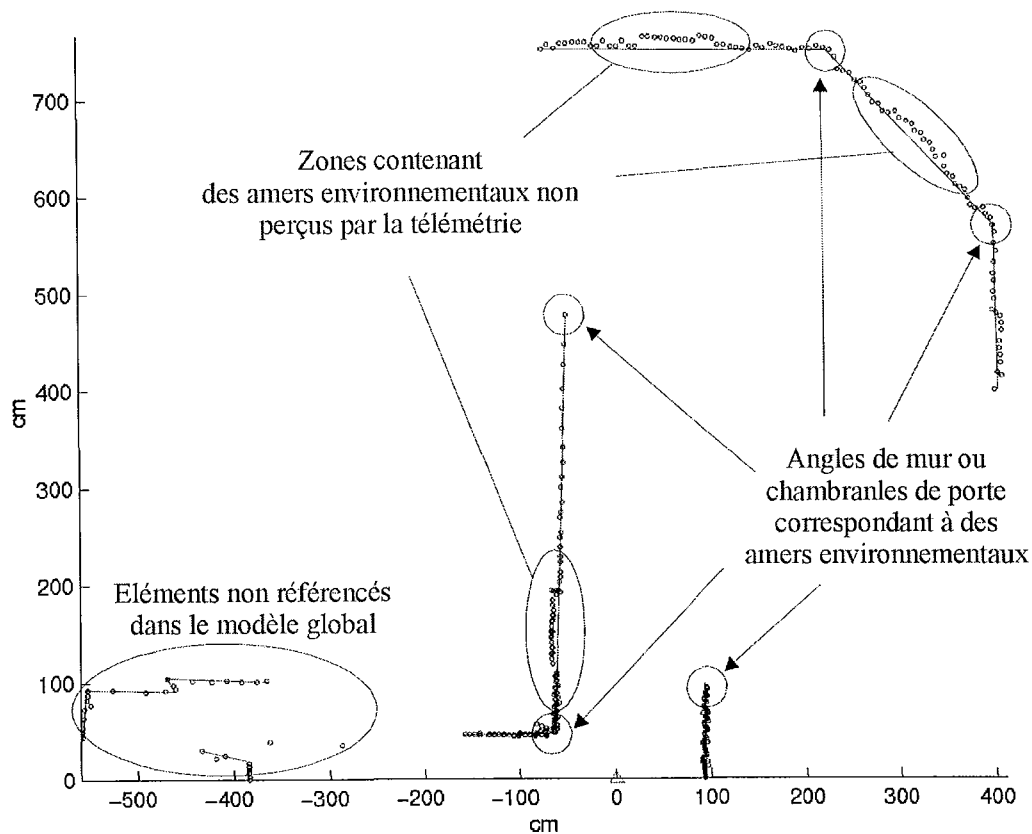


Figure 1.14 : Analyse des discontinuités d'une profilométrie laser.

1.4.2.2 Stratégie de perception

Afin de pallier aux handicaps inhérents à chaque capteur, nous devons définir une méthode permettant d'exploiter au mieux cet équipement sensoriel :

- Un système multi-capteurs n'offre qu'une redondance limitée et ne permet donc pas de s'assurer de la fiabilité des amers détectés.
- L'intégration directe des mesures s'avère être sensible au bruit de mesure.

- L'association des primitives issues de chaque capteur n'est envisageable que dans le cas de capteurs pouvant générer des primitives présentant des caractéristiques communes.

Une approche de type fusion de primitives est la plus adaptée à nos besoins ainsi qu'aux capteurs utilisés.

Nous avons donc mis au point une stratégie de perception basée sur la confirmation et l'enrichissement des informations laser par les informations vision. Il s'agit d'une approche de type ***fusion de primitives*** avec "***focalisation d'attention***" au niveau d'une partie d'un signal ou d'une décision locale : la vision se mettant au service de la télémétrie pour enrichir une information ponctuelle, valider un choix et améliorer une précision. Le but étant de générer des primitives de type arêtes verticales les plus fiables possibles en exploitant au mieux les informations sensorielles disponibles.

Compte tenu des hypothèses liées à l'environnement, nous pouvons considérer que les segments laser définissent des plans verticaux. Chaque paire de segments forme une primitive *coin* dont le sommet correspond à une arête verticale susceptible d'être répertoriée comme amer environnemental. Le module de vision est alors utilisé afin d'inspecter ces arêtes potentielles. Si l'analyse valide la présence d'une arête verticale suffisamment longue pour être considérée comme étant un amer, alors le module de vision réalise l'analyse des plans verticaux définis par les deux segments laser afin d'y rechercher de nouvelles arêtes verticales. La vision peut ainsi extraire des amers que la télémétrie n'a pas su mettre en évidence.

Note travail se rattache ainsi au concept de vision active dans lequel on suppose que le système de vision ne perçoit pas passivement l'environnement mais réagit à cette perception. Ceci se traduit notamment par un déclenchement du mouvement de la caméra avec un réglage des paramètres internes tels que focale et mise au point, afin de focaliser l'attention du système de perception sur une partie déterminée de la scène.

Chapitre 2

Système de perception vision-laser

2.1 INTRODUCTION

La plate-forme expérimentale utilisée est celle du démonstrateur ROMANE développée au sein de l'équipe Vision pour la Robotique et l'Inspection du Centre de Recherche en Automatique de Nancy. L'idée initiale du projet ROMANE était de concevoir un robot mobile pour la surveillance automatique d'un parc municipal [Elarbi 92]. Aujourd'hui, le projet s'oriente vers des applications liées à la navigation d'un véhicule autonome à l'extérieur [Arnould 93, Couroux 97, Hardt 97, Kaske 97] et à l'intérieur de bâtiments [Siadat 99].

La Figure 2.1 illustre le système global composé d'un poste fixe constitué par un réseau de stations de travail communiquant par liaison hertzienne avec le robot mobile ROMANE.

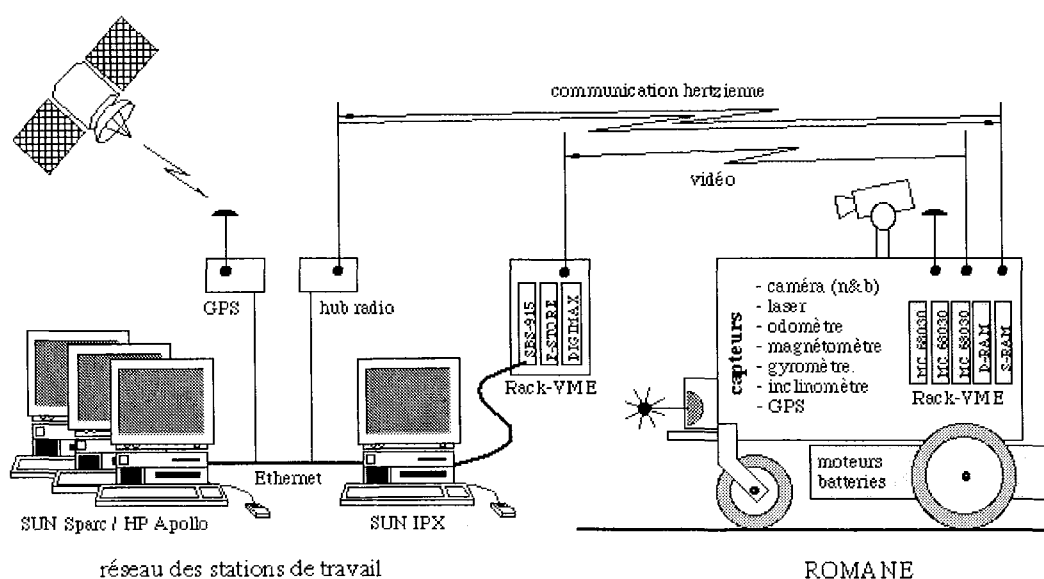


Figure 2.1 : Architecture du projet ROMANE.

Le poste fixe assure la majeure partie des traitements informatiques. Le traitement des données en provenance du robot ainsi que l'élaboration des lois de contrôle sont réalisés de façon déportée afin de limiter la puissance de calcul devant être embarquée.

Le robot mobile ROMANE est construit autour d'une base de fauteuil roulant électrique pour handicapé. Il mesure 200 *cm* de haut et pèse 140 *kg*. Deux roues folles indépendantes montées sur l'arbre avant permettent de diriger le robot. Il est propulsé par deux moteurs électriques identiques mais indépendants. Chaque moteur actionne une roue motrice de l'arbre arrière assurant une vitesse maximale de l'ordre de 50 *cm/s*. Une alimentation à base de batteries sèches d'une puissance de 4 x 15 *kWh* assure une autonomie d'environ 5 heures.

L'architecture informatique embarquée est bâtie autour d'un bus VME contenant 3 cartes CPU, deux cartes de RAM externes et plusieurs cartes d'entrées-sorties. Un système d'exploitation *VxWorks* temps réel et multi-tâches est utilisé pour la gestion informatique du robot ROMANE.

Les capteurs embarqués sont de différents types : odomètres, magnétomètre, gyroscope, laser 2D, caméra CCD noir et blanc, inclinomètres, GPS. Une grande partie des pré-traitements liés aux données issues des capteurs proprioceptifs est réalisée sur ROMANE même, afin d'obtenir des temps de réponse cohérents avec la notion d'autonomie.

Les images vidéo sont transmises par un émetteur hertzien unidirectionnel du robot vers le poste fixe. Alors que les consignes de commande ainsi que les retours des informations sensorielles provenant des autres capteurs utilisent une liaison numérique.

Nous présentons dans ce chapitre les caractéristiques ainsi que la calibration individuelle de chaque capteur mis en œuvre : à savoir le module de vision (paragraphe 2.2) puis le télémètre laser (paragraphe 2.3). Enfin, nous détaillons les équations permettant la transformation des coordonnées des primitives entre les repères capteurs R_{laser} et $R_{caméra}$, et le repère lié au robot R_{robot} (paragraphe 2.4).

2.2 LE MODULE DE VISION

Le système de vision est composé d'une caméra munie d'un objectif à focale, ouverture et mise au point (appelée par la suite « focus ») réglables. L'ensemble est monté sur une tourelle porte caméra à deux axes : angles de site et d'azimut, comme l'illustre la Figure 2.2.

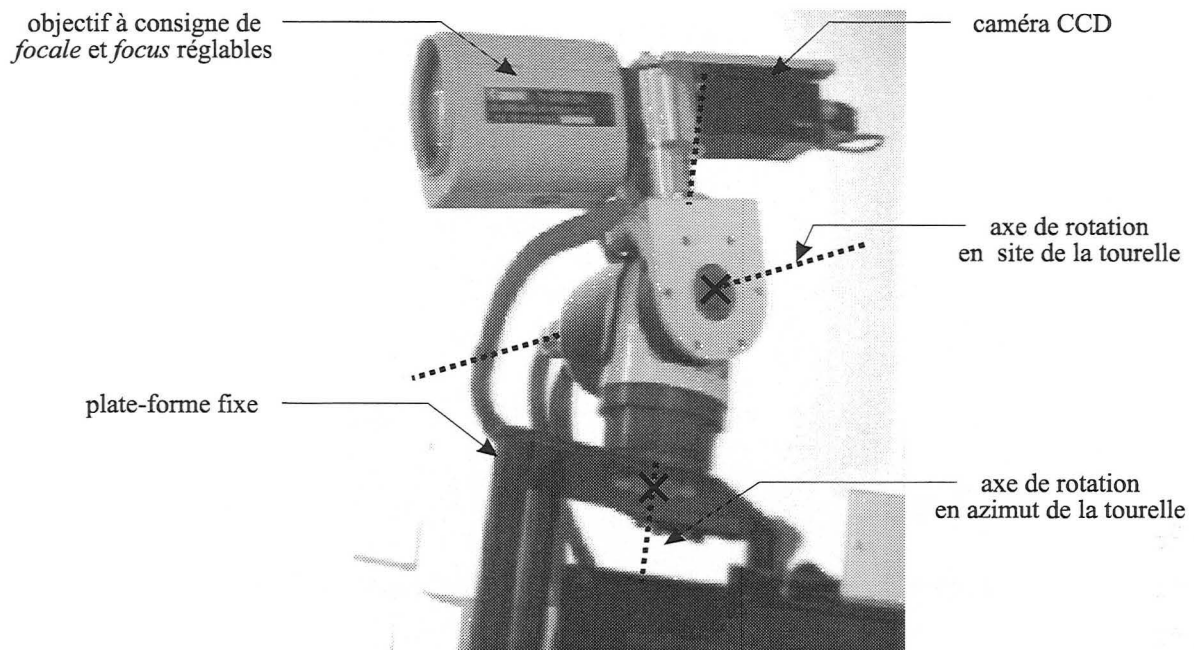


Figure 2.2 : Le module de vision.

Nous présentons dans les paragraphes suivants les aspects matériels des différents éléments composant le module de vision, puis nous détaillons les modèles utilisés pour les représenter.

2.2.1 Présentation matérielle

Le module de vision a été conçu en ayant comme objectif de limiter la puissance informatique devant être embarquée sur le robot. Pour cette raison, les images acquises par le robot sont transmises vers le poste fixe afin d'être analysées par des algorithmes de traitement d'image. Cette solution technique se traduit par la chaîne d'acquisition et de traitement de l'image détaillée Figure 2.3.

Les différents paramètres de réglage du module de vision (*focale, focus, angles d'azimut α et de site β , vitesse d'obturation*) sont tout d'abord transmis du poste fixe vers ROMANE par la

liaison numérique via le système ARLAN. L'acquisition de l'image est ensuite déclenchée par le poste fixe. Le signal vidéo composite correspondant est alors transmis au poste fixe à l'aide d'un émetteur vidéo HF dédié. La numérisation de l'image correspondante est réalisée par une carte d'acquisition implantée dans un rack VME. L'image numérisée est ensuite disponible sur le réseau Ethernet du laboratoire et peut être traitée sur n'importe quelle station de travail.

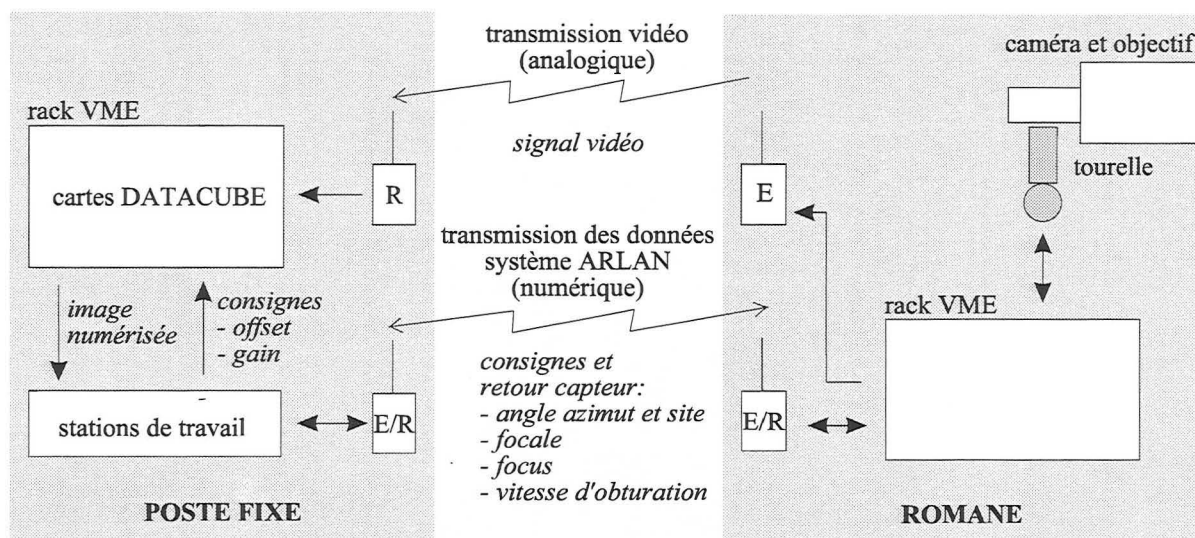


Figure 2.3 : Chaîne d'acquisition et de traitement de l'image.

Nous présentons ci-après les caractéristiques principales des différents éléments composant le module de vision.

2.2.1.1 La caméra et l'objectif

La caméra utilisée est une caméra CCD noir et blanc de type industriel. Elle est équipée d'une cellule CCD à pixels carrés de $11 \mu m$ de côté et offre une résolution de 581 lignes sur 756 colonnes. Le signal vidéo délivré en sortie de caméra correspond à une image d'une résolution de 575 lignes par 560 colonnes.

La caméra dispose d'un obturateur électronique réglable de $1/60^{\text{ème}}$ à $1/10000^{\text{ème}}$ de secondes. ROMANE opérant dans des couloirs éclairés par des tubes néons (période d'éclairage de 10 ms), il s'est révélé nécessaire de régler un temps d'obturation le plus long possible (c'est-à-dire 16 ms) afin de s'assurer d'un éclairage de la scène à peu près constant.

L'objectif zoom retenu possède une focale variable de 10 à 100 mm pour une ouverture de 1 : 1,4. La profondeur de champ est réglable sur une plage variant de 1 à 35 mètres.

2.2.1.2 La tourelle porte caméra

L'ensemble caméra-objectif est placé sur une tourelle possédant deux axes de rotation. La vitesse maximale de rotation est de $26,2^\circ$ par seconde. Comme l'illustre la Figure 2.4, l'amplitude est de -159° à $+159^\circ$ en azimut, et de -60° à $+54^\circ$ en site.

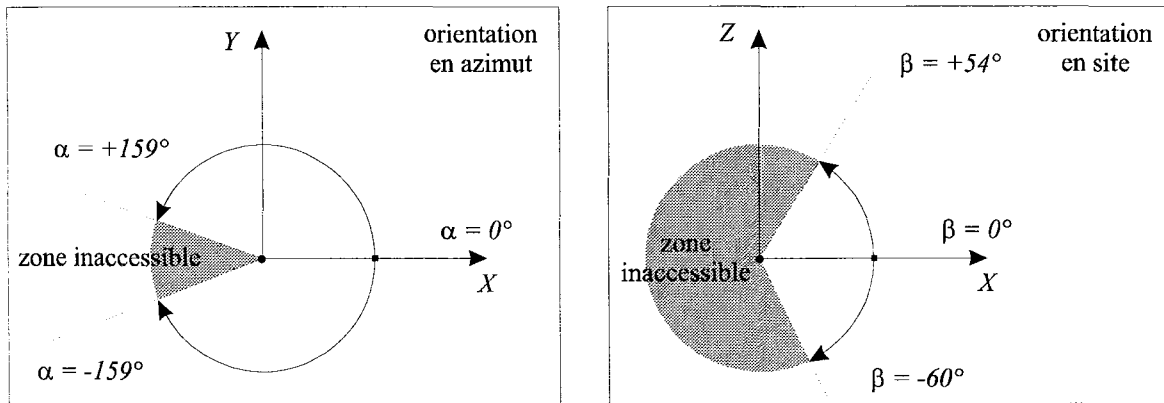


Figure 2.4 : Orientation en azimut et en site de la tourelle porte caméra.

2.2.1.3 La transmission vidéo

Lors des déplacements du robot, l'image transmise subit des déformations aléatoires (piquetage ou distorsion) liées à la proximité de structures métalliques importantes (ex : ascenseurs) qui perturbent la transmission radio. Ce phénomène est en partie imputable à la faible puissance d'émission, limitée par la législation à 10 mW pour une largeur de bande de 450 MHz . Pour améliorer la qualité de la transmission, il serait souhaitable de numériser les images sur le robot, de les compresser, puis de réaliser une transmission numérique via le système ARLAN.

2.2.1.4 La carte d'acquisition d'image vidéo

La carte d'acquisition d'images vidéo Datacube est connectée à une station de travail Sun IPX par l'intermédiaire d'une carte S-Bus. Elle numérise les images vidéo avec 256 valeurs de gris en $512\text{ par }512\text{ pixels}$. Cette carte présente un défaut important : elle ajoute toujours une bande noire de 14 pixels sur le bord gauche de l'image. La chaîne vidéo modifie ainsi la taille de l'image de $756\text{ par }581\text{ pixels}$ à l'acquisition en $498\text{ par }512\text{ pixels}$ une fois numérisée. L'image résultante en bout de chaîne est donc fortement comprimée dans le sens horizontal. Les consignes de *gain* et d'*offset* de la carte d'acquisition ont été fixées expérimentalement de façon à obtenir un bon contraste pour des images prises à l'intérieur de bâtiments.

2.2.1.5 La transmission des paramètres de réglage

La transmission des données correspondantes aux paramètres de réglage du module de vision est effectuée par une liaison numérique. Des hubs radio de type ARLAN 640 assurent la transmission des données des capteurs par le protocole TCP/IP et fournissent un débit de transmission de 2 *Mbauds* en réception idéale. Ils réalisent une connexion Ethernet par radio et sont totalement transparents pour chaque protocole (FTP, Telnet, etc.). Le système possède un module de correction d'erreurs : en cas de perte temporaire de la porteuse, il est capable de la retrouver et de réinstaller la transmission sans perte de données. Ceci est un point indispensable à tout système mobile autonome puisque l'on ne peut pas assurer a priori que la transmission ne sera jamais interrompue.

2.2.2 Modélisation de la caméra et de l'objectif

Le problème de la calibration d'un système de vision peut se poser en ces termes : étant donné un point P de coordonnées (x_e, y_e, z_e) dans le repère lié à l'environnement R_{env} , quelle transformation permet de connaître la position discrète (u, v) de sa projection dans l'image numérisée ?

Nous présentons ci-après les différents principes de modélisation couramment utilisés, puis nous en sélectionnons un convenant à notre application.

2.2.2.1 Modélisation d'un système de vision monoculaire

Un objectif est formé de quatre groupes optiques : une lentille frontale, un variateur, un compensateur et un relais optique (Figure 2.5). Le réglage du focus ainsi que de la focale est obtenu en déplaçant les uns par rapport aux autres ces groupes optiques (les déplacements mécaniques sont des translations le long de l'axe optique assurées par des rails de guidage).

Chacun de ces quatre groupes assure une fonction privilégiée. La lentille frontale bouge seule pour assurer la mise au point. Le variateur permet d'obtenir de grandes variations de la longueur frontale. Le compensateur est l'élément qui maintient constant le plan de focalisation. Enfin, le relais optique assure la convergence des rayons lumineux dans le plan de l'image. La Figure 2.6 présente le déplacement des groupes de lentilles lors de variation de la longueur focale mais en conservant un focus constant.

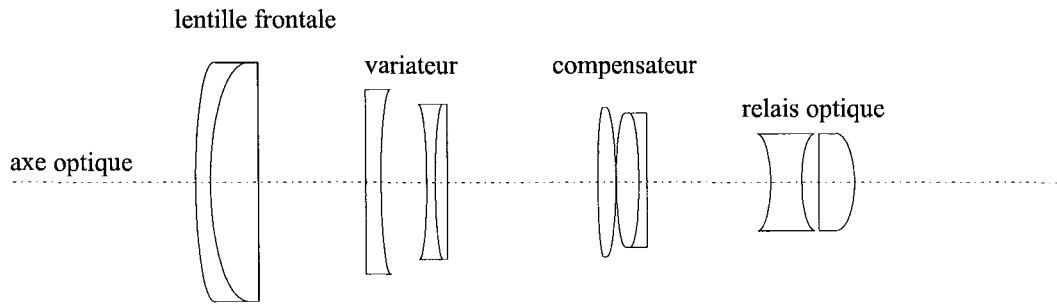


Figure 2.5 : Les quatre groupes optiques formant un objectif.

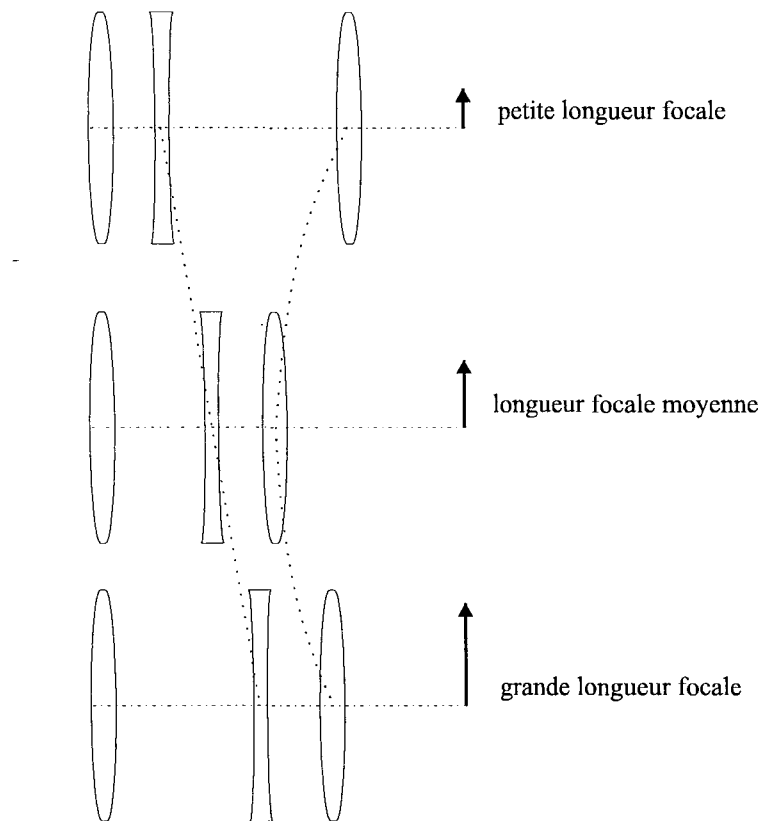


Figure 2.6 : Déplacement des lentilles lors de la variation de focale avec un focus constant.

Un modèle basé sur une description géométrique complète de tous les groupes optiques peut être défini [Lavest 93]. C'est le plus complet et le plus précis des modèles ; il permet notamment de prendre en compte les aberrations optiques dues aux défauts des lentilles.

Toutefois, les relations entre les paramètres de contrôle «*focale, focus* » et la configuration physique des lentilles sont trop complexes pour être exprimées sous forme d'équations utilisables. Ce modèle s'avère donc extrêmement complexe à identifier et à implémenter.

Afin d'exprimer de façon efficace les différents paramètres du processus permettant la formation de l'image, il est nécessaire d'utiliser des modèles basés sur des simplifications du véritable processus de formation de l'image au travers des blocs optiques. La connexion entre la configuration physique des lentilles et les paramètres du modèle n'est donc pas directe.

Le modèle *sténopé* schématisé Figure 2.7 présume que tous les rayons lumineux provenant d'un objet convergent vers le plan image au travers d'un seul point appelé le point focal. Ce modèle n'est précis que si la distance séparant le plan image du point focal est négligeable par rapport à celle qui le sépare de l'objet.

C'est le modèle le plus simple et le plus couramment utilisé dans le cas d'objectifs à focale fixe [Watt 89], mais il n'est pas adapté dans le cas d'objectifs à focale et focus variables [Lavest 93].

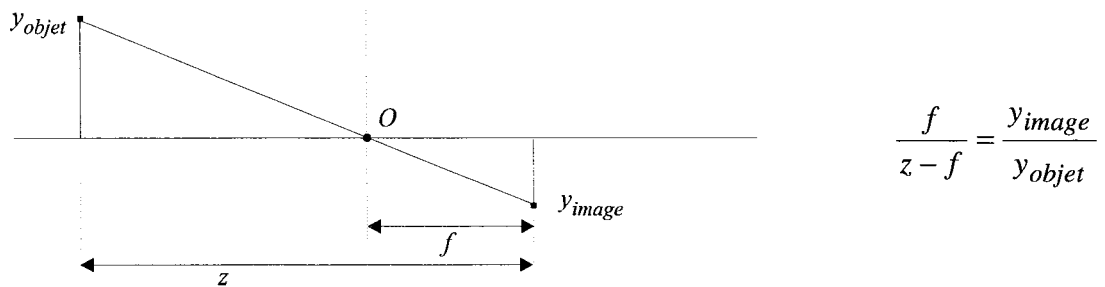


Figure 2.7 : Modèle sténopé.

Le modèle de *lentille mince* schématisé Figure 2.8 suppose une lentille infiniment mince, modélisée par un plan. Les rayons lumineux provenant de l'objet atteignent le plan sous un angle particulier et poursuivent leur trajectoire sous un angle différent après l'avoir traversé.

Ce modèle permet d'exprimer les phénomènes liés à la mise au point de l'image [Born 93]. La longueur focale f_i et le focus z_i sont définis par la loi des lentilles gaussiennes.

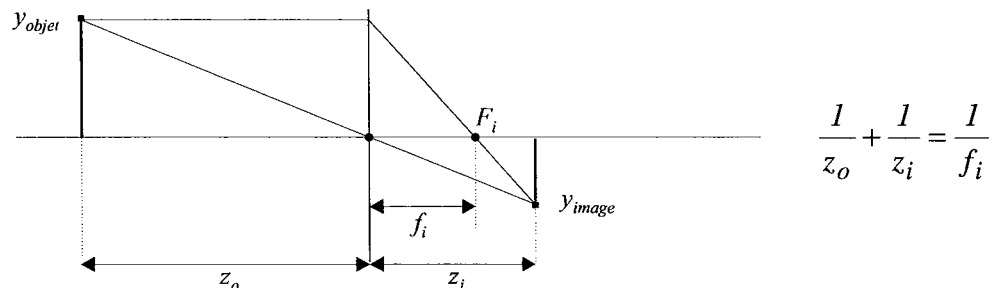


Figure 2.8 : Modèle de lentille mince.

Le modèle de **lentille épaisse** représente la lentille par deux plans H_o et H_i appelés les plans principaux. Les rayons lumineux atteignent le premier plan H_o sous un angle donné, traversent l'espace inter plan en restant parallèles à l'axe optique, puis poursuivent leur trajectoire sous l'angle initial après avoir traversé le deuxième plan H_i , comme l'illustre la Figure 2.9. Le modèle de lentille épaisse est celui qui convient le mieux à un système optique à focale et focus variables [Subbarao 88]. Ainsi, lorsque la longueur focale des lentilles modélisées change, la distance entre les plans principaux H_o et H_i varie d'une manière dépendant de la construction des lentilles (La distance inter plan $H_i H_o$ est positive pour les petites focales et négatives pour les grandes focales). Cette relation peut être déterminée par calibration à l'aide d'un équipement optique sophistiqué.

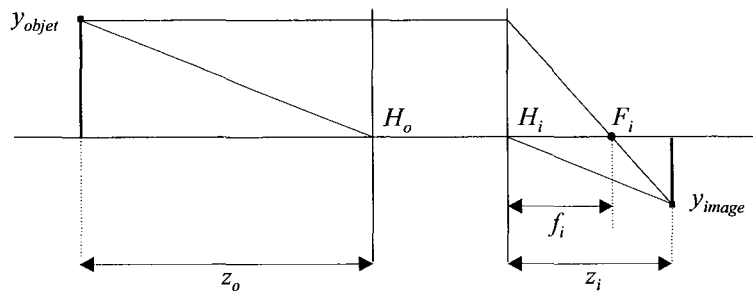


Figure 2.9 : Modèle de lentille épaisse.

Afin de simplifier la procédure de calibration propre au modèle de lentille épaisse, il est possible de définir un modèle **sténopé étendu** comme étant une lentille épaisse possédant un point focal infiniment petit et à mise au point parfaite (Figure 2.10). Le modèle sténopé étendu prend en compte la distance entre les plans principaux et permet ainsi d'expliquer le changement apparent de la distance objet à l'image lorsque la longueur focale change [Cahn von Seelen 97].

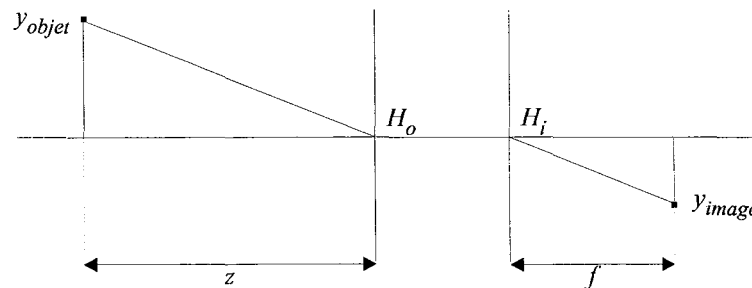


Figure 2.10 : Modèle sténopé étendu.

C'est ce modèle de lentille que nous allons utiliser pour notre ensemble caméra et objectif.

2.2.2.2 Expression du modèle sténopé étendu direct

Le modèle direct permet d'exprimer les coordonnées 3D d'un point P de l'espace dans le repère 2D de l'image numérique. Le modèle inverse permet quant à lui de déterminer la ligne de visée dans l'espace 3D correspondant à un point de l'image numérique 2D.

La Figure 2.11 décrit les systèmes de coordonnées cartésiennes utilisés comme repères :

$R_{env.} (O_e, X_e, Y_e, Z_e)$, repère propre à l'environnement.

$R_{caméra} (O_c, X_c, Y_c, Z_c)$, repère lié à la caméra. L'axe optique Z_c , défini comme étant la ligne passant par les centres des composants optiques de la lentille, est perpendiculaire au plan du capteur CCD. L'axe X_c est colinéaire à l'axe X_i du capteur CCD (respectivement Y_c et Y_i).

$R_{image} (O_i, X_i, Y_i)$, repère lié au capteur CCD de la caméra. L'origine O_i est confondue avec l'origine O_c du repère $R_{caméra}$. On suppose que l'orthogonalité des pixels de la matrice CCD est parfaite et l'on définit l'axe X_i comme étant parallèle aux lignes V de la matrice CCD (respectivement Y_i aux colonnes U).

$R_{numérisée} (O_n, U, V)$, repère de l'image numérisée. Le centre (u_o, v_o) de l'image numérisée est confondu avec l'origine O_i du repère R_{image} .

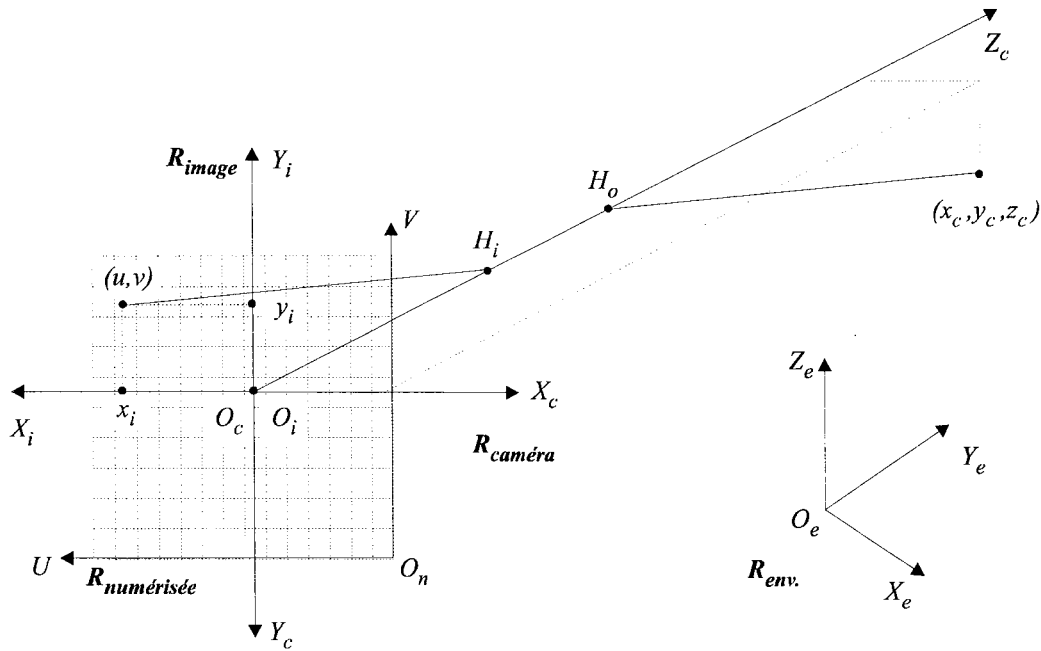


Figure 2.11 : Transformation par le modèle sténopé étendu.

Etant donné un point P dont les coordonnées (x_e, y_e, z_e) sont exprimées dans le repère R_{env} , la transformation par le modèle sténopé étendu direct peut être décomposée en trois phases :

- Tout d'abord l'expression des coordonnées (x_c, y_c, z_c) du point P dans le repère $R_{caméra}$. Il s'agit d'une transformation rigide correspondant à trois rotations Rot_ω, Rot_β et Rot_α autour des axes X_e, Y_e et Z_e du repère R_{env} , et à trois translations T_x, T_y et T_z .

$$\begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} = \begin{bmatrix} Rot_{1,1} & Rot_{1,2} & Rot_{1,3} & T_x \\ Rot_{2,1} & Rot_{2,2} & Rot_{2,3} & T_y \\ Rot_{3,1} & Rot_{3,2} & Rot_{3,3} & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_e \\ y_e \\ z_e \\ 1 \end{bmatrix} \quad (2.1)$$

La matrice de rotation Rot est obtenue par multiplication des rotations élémentaires :

$$Rot = Rot_\alpha Rot_\beta Rot_\omega = \begin{bmatrix} \cos\alpha \cos\beta & \cos\alpha \sin\beta \sin\omega - \sin\alpha \cos\omega & \cos\alpha \sin\beta \cos\omega + \sin\alpha \sin\omega \\ \sin\alpha \cos\beta & \sin\alpha \sin\beta \sin\omega + \cos\alpha \cos\omega & \sin\alpha \sin\beta \cos\omega - \cos\alpha \sin\omega \\ -\sin\omega & \cos\beta \sin\omega & \cos\beta \cos\omega \end{bmatrix}$$

- Puis le calcul des coordonnées (x_i, y_i) de la projection de ce point dans le repère R_{image} .

$$x_i = x_c \left(\frac{f}{z_c - d_{ip} - f} \right) \quad y_i = y_c \left(\frac{f}{z_c - d_{ip} - f} \right) \quad (2.2)$$

Avec :

$$\left\{ \begin{array}{l} f = \overline{O_c H_i}, \text{ longueur focale de la caméra (mm),} \\ d_{ip} = \overline{H_i H_o}, \text{ distance inter plans (mm).} \end{array} \right.$$

- Et enfin, la détermination des coordonnées (u, v) du point P dans l'image numérisée.

$$u = \frac{x_i}{d_x} + u_o \quad v = \frac{y_i}{d_y} + v_o \quad (2.3)$$

Avec :

$$\left\{ \begin{array}{l} u_o \text{ et } v_o, \text{ coordonnées du centre de l'image numérisée (pixels),} \\ d_x \text{ et } d_y, \text{ facteurs d'échelle des éléments sensoriels du capteur CCD et de la chaîne} \\ \text{d'acquisition de l'image dans les directions } X_i \text{ et } Y_i \text{ (mm/pixel).} \end{array} \right.$$

Bien que la distorsion radiale existe, nous l'avons négligé car l'erreur qu'elle provoque est relativement faible pour notre application. En effet, nous recherchons des segments verticaux présents sur toute la hauteur de l'image et pour lesquels nous calculons une abscisse moyenne u_a . Les défauts de l'optique, à savoir déformation convexe à petite focale et concave à grande focale, se compensent donc.

Les paramètres permettant la transformation des coordonnées (x_e, y_e, z_e) d'un point P exprimées dans le repère $R_{env.}$ aux coordonnées (u, v) dans l'image numérisée peuvent être divisés en deux catégories :

Les **paramètres extrinsèques** $(\omega, \beta, \alpha, T_x, T_y, T_z)$ qui dépendent de la position et de l'orientation de la caméra par rapport au repère $R_{env.}$. Lorsque la caméra est déplacée, ces 6 paramètres doivent être recalculés.

Les **paramètres intrinsèques** $(f, d_{ip}, u_o, v_o, d_x, d_y)$ qui sont des coefficients dépendant uniquement de la géométrie de la caméra utilisée, donc du réglage de la focale et du focus, ainsi que du mode de numérisation de l'image.

2.2.2.3 Identification des paramètres intrinsèques du modèle sténopé étendu

Afin de réaliser une calibration complète de la caméra, il est nécessaire de déterminer pour tous les couples possibles de consignes « *focale, focus* » la valeur des paramètres intrinsèques $(f, d_{ip}, u_o, v_o, d_x, d_y)$ correspondants. Si l'on considère une discrétisation des consignes de focale selon un pas de 5 mm, et de focus selon un pas de 100 cm, il serait nécessaire de réaliser $20 \times 35 = 700$ campagnes de calibration !

Or, dans le cadre de la stratégie de perception mise en œuvre, nous recherchons à l'aide du module de vision des arêtes verticales de l'environnement d'une hauteur comprises entre 150 et 210 cm.

Afin d'optimiser la projection de ces arêtes dans l'image numérisée, nous choisissons donc de régler la consigne de focale de façon à placer le sommet d'une arête verticale située à une profondeur z_c et de hauteur théorique $H_{max} = 210$ cm dans le haut de l'image (ligne pixel 1), comme l'illustre la Figure 2.12. La consigne de focus est alors réglée de façon à obtenir le plan net sur l'arête verticale pour le réglage de la focale réalisé, nous assurant ainsi d'obtenir des contours de type marche d'escalier pour les amers environnementaux retenus.

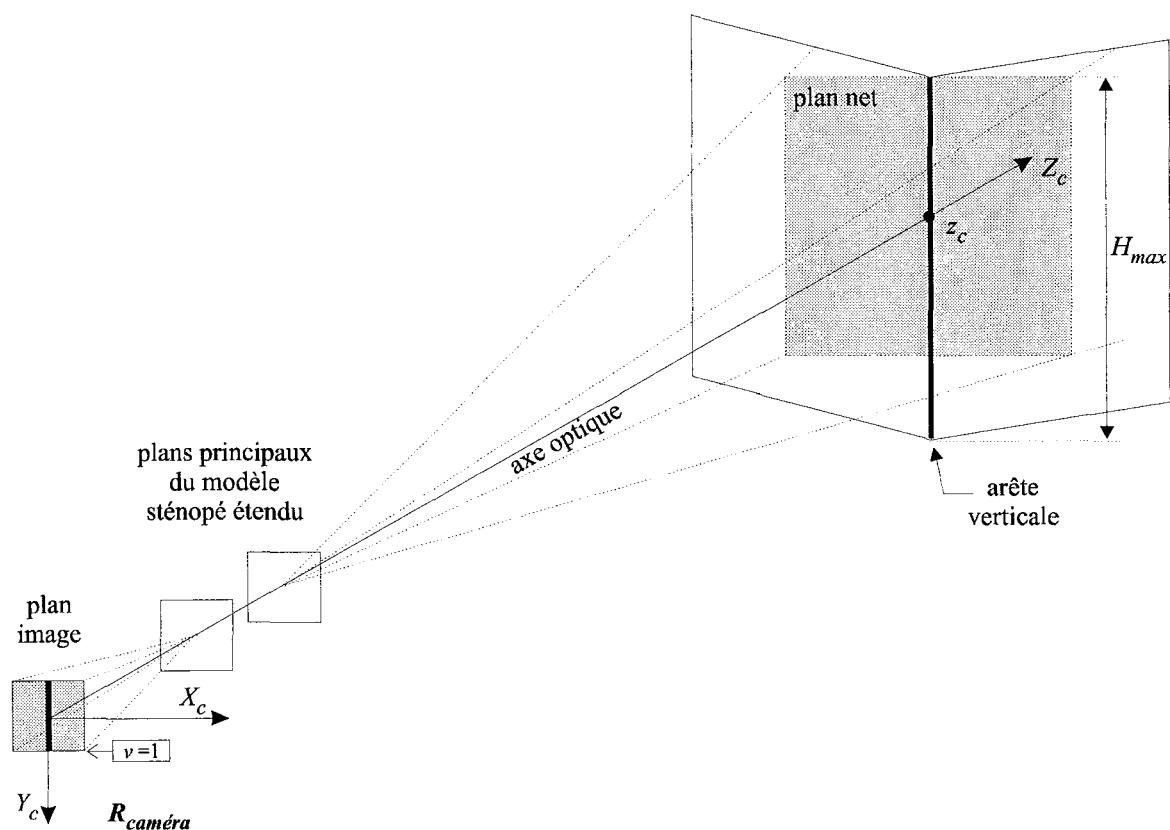


Figure 2.12 : Optimisation du plan image dans le cas d'amers de hauteur maximale H_{max} fixe.

Une campagne de mesure a permis de déterminer expérimentalement les couples C_i , $i=1...16$ de consignes «*focale_i*, *focus_i*» optimaux en fonction de la coordonnée de profondeur z_c de l'arête dans le repère $R_{caméra}$ (Figure 2.13).

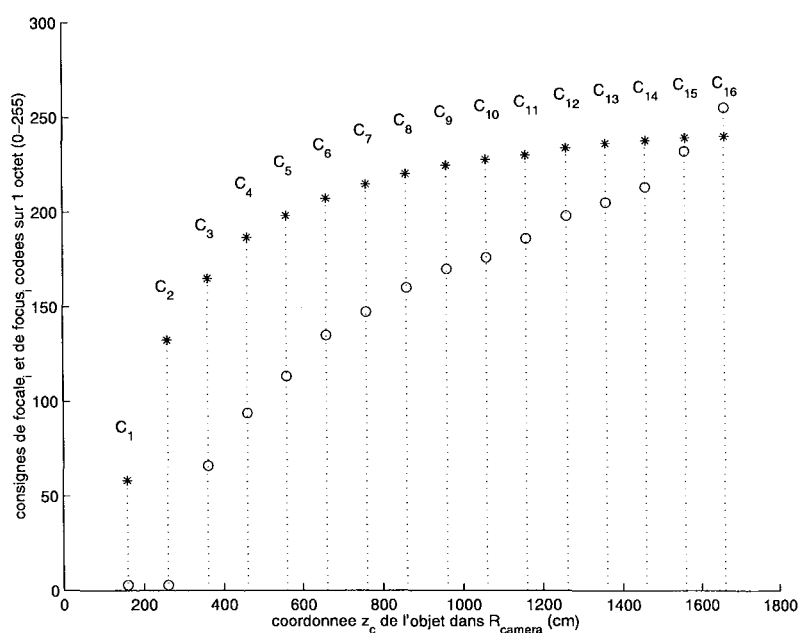


Figure 2.13 : Couples C_i , $i=1$ à 16 de consignes «*focale_i* (o), *focus_i* (*)».

2.2.2.3.1 Coordonnées (u_o, v_o) du centre de l'image numérisée

Dans le cas d'une caméra munie d'un objectif à focale et focus variables, des groupes de lentilles sont alignés et montés de façon à former une lentille complète. Lors des changements de focale, les lentilles de notre objectif ont un mouvement de translation le long de l'axe optique Z_c , ce qui induit un déplacement de type translation du centre de l'image numérisée. On peut toutefois poser comme hypothèse que la précision de la mécanique assurant le déplacement des lentilles de l'objectif est suffisante pour négliger tout déplacement significatif du centre de l'image lors de variation de la distance focale [Couroux 97, Kaske 97]. Le changement de focus est quand à lui obtenu à l'aide d'un mouvement de rotation de la lentille frontale, ce qui conduit à supposer un déplacement de type rotation du centre de l'image numérisée [Willson 94].

La détermination expérimentale de la position (u_o, v_o) du centre de l'image numérisée repose sur une propriété des points de fuite dans le cadre de la projection perspective ; à savoir que toute translation d'un objet le long de l'axe optique de la caméra induit à l'image des points qui convergent le long de lignes de fuite vers le centre optique. Afin de limiter les erreurs de mesures pouvant être provoquées par un déplacement de la mire, et compte tenu des remarques précédentes, nous choisissons de réaliser une translation virtuelle de la mire en conservant fixe la distance z_c séparant la mire du capteur CCD, et en faisant varier la consigne de focale. Pour chaque couple de consignes $C_i, i=1 \dots 16$, nous déterminerons donc une position moyenne (u_o, v_o) du centre de l'image numérisée, à consigne de focus constante et à consigne de focale variable.

Pour les expérimentations, nous utilisons une mire plane de type damier composée de carreaux noirs et blancs dont les coins constituent des points de référence P_n . Cette mire est placée perpendiculairement au plan du capteur CCD.

Afin d'obtenir une bonne précision sur les coordonnées (u_o, v_o) du centre de l'image numérisée, il faut qu'un nombre N suffisamment élevé de points de référence soient répartis de façon homogène dans l'image et que leur extraction soit la plus précise possible ($\pm 0,5$ pixels). Pour chaque couple de consigne $C_i, i=1 \text{ à } 16$, nous avons donc placé la mire à la distance z_c et réglé la consigne $focus_i$ correspondante. Nous avons ensuite fait varier la consigne de focale de 10 à 100 mm et acquis pour chaque consigne $focale_m, m=1 \text{ à } M$ une image $img_{i,m}$. Pour chaque série de M images prises à consigne de focus constante, nous avons extrait les coordonnées $(u_{n,m}, v_{n,m})$ de chaque point de référence $P_n, n=1 \text{ à } N$.

La Figure 2.14 illustre la procédure de recherche des coordonnées du centre de l'image numérisée.

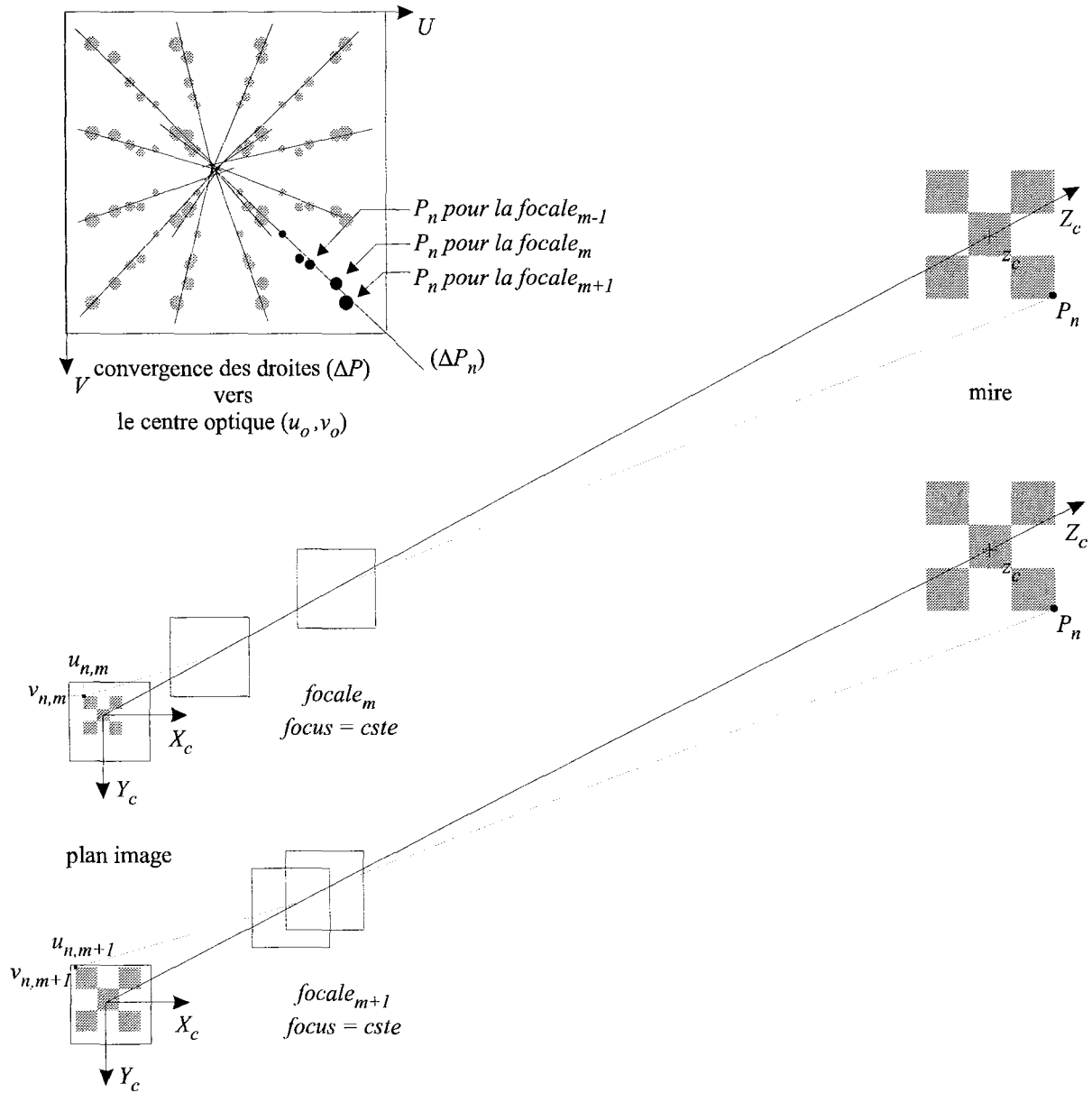


Figure 2.14 : Recherche des coordonnées (u_o, v_o) du centre de l'image numérisée.

Pour chaque point de référence P_n , nous déterminons ensuite la droite (ΔP_n) qui approche au mieux les M projections au sens des moindres carrés linéaires

$$(\Delta P_n) \quad v = a_n \times u + b_n \quad (2.4)$$

$$a_n = \frac{M \sum_{m=1}^M u_{n,m} v_{n,m} - \sum_{m=1}^M u_{n,m} \sum_{m=1}^M v_{n,m}}{M \sum_{m=1}^M u_{n,m}^2 - \left(\sum_{m=1}^M u_{n,m} \right)^2} \quad b_n = \frac{\sum_{m=1}^M u_{n,m}^2 \sum_{m=1}^M v_{n,m} - \sum_{m=1}^M u_{n,m} \sum_{m=1}^M u_{n,m} v_{n,m}}{M \sum_{m=1}^M u_{n,m}^2 - \left(\sum_{m=1}^M u_{n,m} \right)^2}$$

Le point de convergence des N droites (ΔP) correspond au centre optique de la caméra. Les coordonnées moyennes (u_o, v_o) du centre de l'image numérisée, pour la consigne de focus réglée, sont alors calculées par les moindres carrés linéaires.

$$u_o = \frac{\sum_{n=1}^N a_n \sum_{n=1}^N b_n - N \sum_{n=1}^N a_n b_n}{N \sum_{n=1}^N a_n^2 - \left(\sum_{n=1}^N a_n \right)^2} \quad v_o = \frac{\sum_{n=1}^N a_n^2 \sum_{n=1}^N b_n - \sum_{n=1}^N a_n \sum_{n=1}^N a_n b_n}{N \sum_{n=1}^N a_n^2 - \left(\sum_{n=1}^N a_n \right)^2} \quad (2.5)$$

La Figure 2.15 présente les coordonnées moyennes (u_o, v_o) du centre de l'image numérisée exploitées par le système de perception.

On peut noter d'une part, un déplacement de type rotation du centre optique, et d'autre part une plus forte sensibilité pour les couples de faibles consignes de focale et de focus.

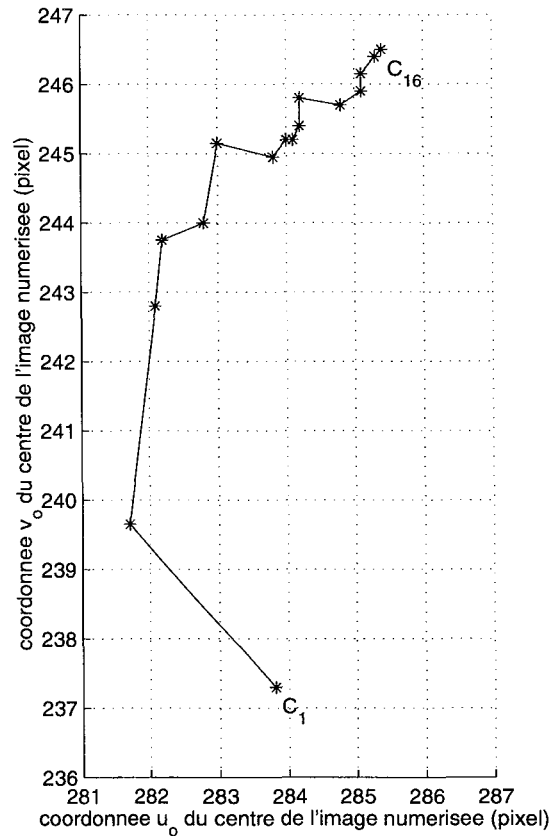


Figure 2.15 : Coordonnées moyennes (u_o, v_o) du centre de l'image numérisée en fonction des couples de consignes $C_i, i=1$ à 16 « focale_i, focus_i ».

2.2.2.3.2 Facteurs d'échelle de l'image d_x et d_y

Les facteurs d'échelle d_x et d_y sont les rapports entre la taille réelle d'un pixel dans le repère R_{image} et sa taille dans l'image numérisée disponible pour les traitements informatiques. Ces facteurs d'échelle ne dépendent que de la géométrie du capteur CCD et de l'étape de numérisation, nous avons recours à une lentille de focale fixe f_{fixe} connue de 25 mm et ne présentant qu'une faible aberration optique que nous utilisons à la place de l'objectif.

Une image de la mire est acquise à une distance z_c du plan image et les coordonnées (u_n, v_n) des points de référence $P_n, n=1 \dots N$ sont extraits. Les valeurs moyennes des facteurs d'échelle d_x et d_y sont alors données par les expressions suivantes, issues du modèle *sténopé* :

$$d_x = \frac{f_{fixe} \times L}{(z_c - f_{fixe}) \times U} \quad d_y = \frac{f_{fixe} \times L}{(z_c - f_{fixe}) \times V} \quad (2.6)$$

Avec :

- U et V , dimensions horizontales et verticales moyennes d'un carreau de la mire (*pixel*).
- L , dimension réelle d'un carreau de la mire (*mm*).

Nous déterminons également une erreur de modélisation absolue $\sigma d_x = \sigma d_y$ afin de prendre en compte les incertitudes sur les grandeurs expérimentales :

- $\delta L = 0,5 \text{ mm}$, erreur de mesure des côtés des carreaux de la mire,
- $\delta z_c = 0,5 \text{ mm}$, erreur de mesure de la distance à la mire,
- $\delta U = \delta V = 0,5 \text{ pixel}$, erreur commise lors de l'extraction des indices de référence de l'image numérisée,
- $\delta f_{fixe} = 0,025 \text{ mm}$, erreur de modélisation de la focale fixe.

Nous obtenons alors comme valeurs moyennes des paramètres de facteur d'échelle :

$$d_x = 16,048 \text{ } (\mu\text{m/pixel}) \text{ et } d_y = 10,967 \text{ } (\mu\text{m/pixel}) \text{ avec } \sigma d_x, \sigma d_y = 0,29 \%$$

Bien que le capteur CCD de la caméra ait des pixels carrés de 11 μm de côté, l'image est fortement modifiée par le système de vision et par les caractéristiques de la carte d'acquisition.

$$r_{\text{échelle}} = \frac{d_x}{d_y} = 1,47 \quad (2.7)$$

La Figure 2.16 illustre la déformation de l'image numérisée en bout de chaîne d'acquisition.

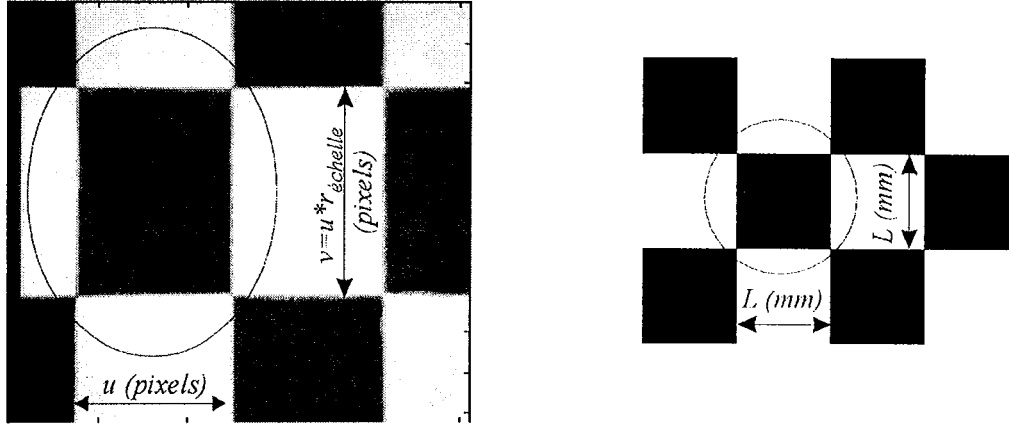


Figure 2.16 : Déformation d'échelle de l'image numérisée.

2.2.2.3.3 Longueur focale f et distance inter plan d_{ip}

La détermination de la longueur focale f et de la distance inter plan d_{ip} du modèle sténopé étendu peut être réalisée à l'aide de deux images d'un même point de référence, acquises à consigne de focale et de focus constantes, mais pour des profondeurs différentes : $z_{c,proche}$ et $z_{c,proche} + \Delta z$ [Cahn von Seelen 97].

La Figure 2.17 décrit la procédure de recherche des paramètres f et d_{ip} .

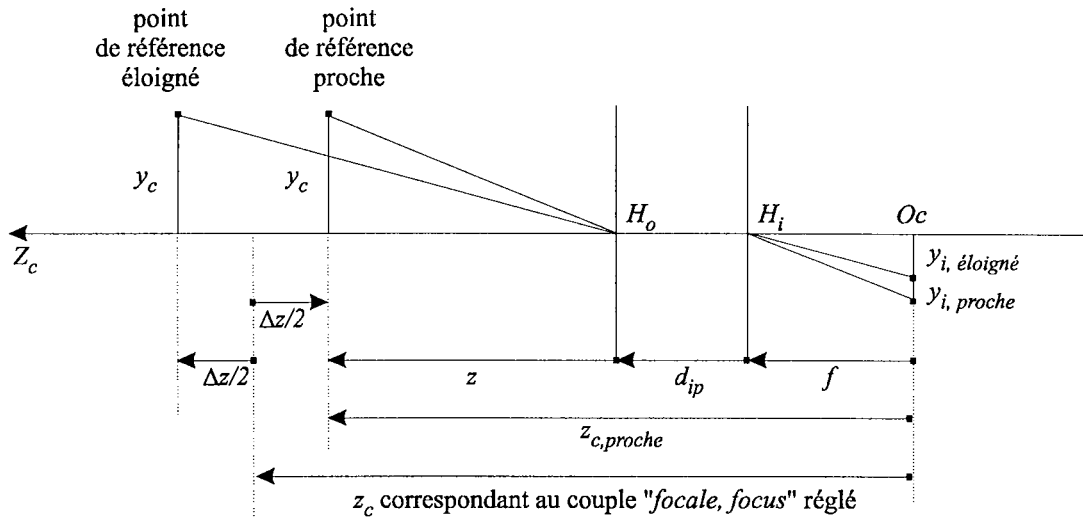


Figure 2.17 : Détermination de la longueur focale f et de la distance inter plan d_{ip} .

A partir des triangles semblables, nous obtenons deux équations pour deux inconnues : la longueur focale f et la distance apparente à l'objet z .

$$\frac{z}{y_c} = \frac{f}{y_{i,proche}} \quad \frac{(z + \Delta z)}{y_c} = \frac{f}{y_{i,éloigné}} \quad (2.8)$$

Avec : y_c , hauteur du point de référence dans le repère $R_{caméra}$ (mm)
 $y_{i,proche}$ et $y_{i,éloigné}$, projection du point de référence dans le repère R_{image} (mm)

Nous en déduisons l'expression de la longueur focale f :

$$f = \frac{y_{i,proche} \times y_{i,éloigné}}{y_c \times (y_{i,proche} - y_{i,éloigné})} \times \Delta z \quad (2.9)$$

La distance inter plans d_{ip} est donnée par :

$$d_{ip} = z_{c,proche} - f \times \left(1 + \frac{y_c}{y_{i,proche}} \right) \quad (2.10)$$

Nous réalisons une première campagne de mesure afin d'estimer avec précision la distance focale f .

Soient $C_{i, i=1 \dots 16}$ « $focus_i$, $focale_i$ » le couple de consignes réglées et z_c la profondeur correspondante.

1. La mire est placée à la distance $z_{c,proche} = z_c - \Delta z/2$ et une première image img_{proche} est acquise. La mire est alors déplacée de Δz et une seconde image $img_{éloigné}$ est acquise. La différence de distance en profondeur Δz est fixée à 600 mm afin d'obtenir deux images nettes. Ainsi les coordonnées $(u_{k,proche} , v_{k,proche})$ et $(u_{k,éloigné} , v_{k,éloigné})$ des points de référence de l'image numérisée pourront être extraites avec une précision de $\pm 0,5$ pixel puisque l'image ne présentera pas de flou.
2. De chaque image sont extraits les coordonnées des points de référence $P_n, n=1$ à N correspondants aux coins des carrés de la mire. Les distances horizontales $(U_{proche}, U_{éloigné})$

et verticales ($V_{proche}, V_{éloigné}$) moyennes séparant deux points de référence adjacents dans le plan de l'image numérisée sont alors calculées.

3. La mire étant composée de carrés (côtés identiques), les distances moyennes $y_{i,proche}$ et $y_{i,éloigné}$ correspondantes dans le plan du capteur CCD sont données par :

$$y_{i,proche} = \frac{1}{2} \times (d_x \times U_{proche} + d_y \times V_{proche}) \quad (2.11)$$

$$y_{i,éloigné} = \frac{1}{2} \times (d_x \times U_{éloigné} + d_y \times V_{éloigné})$$

La longueur focale f moyenne est alors obtenue grâce à l'équation (2.9).

Nous déterminons également une erreur de modélisation δf afin de prendre en compte les incertitudes sur les grandeurs expérimentales :

- $\sigma d_x, \sigma d_y = 0,29 \%$, erreur absolue de modélisation des facteurs d'échelle,
- $\delta \Delta z = 0,5 \text{ mm}$, erreur de mesure de la différence de profondeur,
- $\delta U = \delta V = 0,5 \text{ pixel}$, erreur commise lors de l'extraction des points de référence de l'image numérisée,
- $\delta y_c = 0,5 \text{ mm}$, erreur de mesure des côtés des carreaux de la mire.

Une fois la longueur focale f modélisée, nous réalisons une seconde campagne de mesure afin d'estimer la distance inter plan d_{ip} .

Pour chaque couple de consignes réglées $C_i, i=1...16$ « $focus_i, focale_i$ », nous conservons la distance $z_{c,proche}$ fixe et égale à 2 mètres. L'incertitude sur la mesure de z_c intervenant dans le calcul de d_{ip} , nous ne pouvons pas placer la mire à une grande distance.

La distance inter plan d_{ip} moyenne est alors obtenue grâce à l'équation (2.10).

Nous déterminons également une erreur de modélisation δd_{ip} afin de prendre en compte les incertitudes sur les grandeurs expérimentales :

- $\sigma_{d_x}, \sigma_{d_y} = 0,29 \%$, erreur de modélisation des facteurs d'échelle,
- $\delta z_{c,proche} = 0,5 \text{ mm}$, erreur de mesure de la distance à la mire,
- $\delta U = \delta V$, erreur commise lors de l'extraction des points de référence de l'image numérisée. Nous la supposons égale au pas de quantification pour les consignes de faible focus, puis variant de 1 à 3 pixels pour les consignes de focus élevées (flou de l'image),
- $\delta y_c = 0,5 \text{ mm}$, erreur de mesure des côtés des carreaux de la mire,
- δf , erreur de modélisation de la focale.

Le Tableau 2.1 présente les paramètres de longueur focale f et de distance inter plan d_{ip} du modèle sténopé étendu en fonction des couples C_i , $i=1$ à 16 de consignes «*focale_i* , *focus_i* » susceptibles d'être utilisés.

C_i couple de consignes « <i>focale_i</i> <i>focus_i</i> »	z_c distance du point de référence (mm)	f longueur focale (mm)	δf incertitude sur f (mm)	d_{ip} distance inter plan (mm)	δd_{ip} Incertitude sur d_{ip} (mm)
C_1	1600	6,60	0,05	142,90	4,77
C_2	2600	11,15	0,08	137,65	4,85
C_3	3600	15,82	0,12	105,44	4,97
C_4	4600	20,55	0,15	60,3	5,06
C_5	5600	25,20	0,18	31,66	5,16
C_6	6600	30,13	0,22	-57,48	9,29
C_7	7600	34,89	0,25	-108,98	9,48
C_8	8600	39,44	0,28	-113,76	10,05
C_9	9600	44,19	0,32	-163,45	10,87
C_{10}	10600	48,79	0,35	-187,16	18,60
C_{11}	11600	53,53	0,38	-225,42	18,78
C_{12}	12600	58,16	0,42	-248,90	19,34
C_{13}	13600	62,89	0,45	-294,78	20,27
C_{14}	14600	67,37	0,48	-284,51	20,98
C_{15}	15600	72,08	0,52	-323,88	21,61
C_{16}	16600	76,63	0,55	-330,48	22,36

Tableau 2.1 : longueur focale f et distance inter plan d_{ip} modélisées pour chaque couple de consignes C_i .

L'incertitude δd_{ip} sur la distance inter plan d_{ip} correspond à un décalage en profondeur du plan principal H_o . Les erreurs δx et δz de la projection d'un pixel sur un plan ne dépendent donc pas de la distance au plan mais uniquement de son angle par rapport au plan image. L'influence de δd_{ip} est particulièrement sensible dans le cas des petites focales.

L'incertitude δf sur la longueur focale f correspond à un décalage en profondeur des deux plans principaux H_o et H_i . Les erreurs δx et δz de la projection d'un pixel sur un plan dépendent non seulement de la distance au plan mais également de son angle par rapport au plan image. Les erreurs de projection peuvent donc devenir très importantes dans le cas de petites valeurs de focale et/ou d'objets éloignés.

La Figure 2.18 illustre la sensibilité de la projection d'un pixel de l'image dans le plan en fonction de l'incertitude sur les paramètres f et d_{ip} .

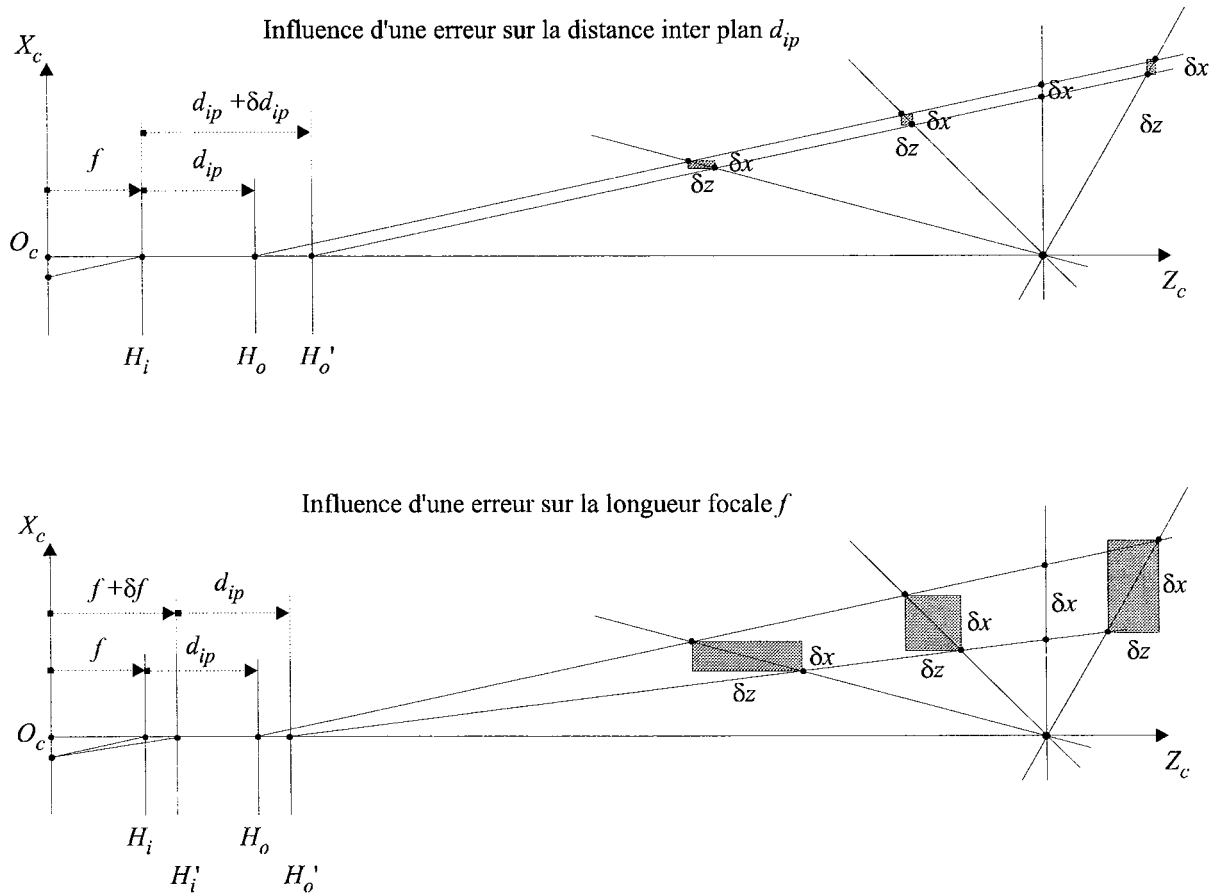


Figure 2.18 : Sensibilité de la projection d'un pixel de l'image aux paramètres f et d_{ip} .

2.2.3 Caractéristiques de la tourelle porte caméra

Les actionneurs de la tourelle sont des moteurs à courant continu équipés d'un réducteur d'un rapport de 1125. Les capteurs de recopie sont des codeurs optiques comportant 500 points par tour. La résolution angulaire en site et/ou azimuth de la tourelle est donc de $1/(500 \times 1125) = 0,00064^\circ$.

Les mesures expérimentales d'asservissement ont montré que la dérive des capteurs était négligeable et qu'il n'existait pas de jeu mécanique mesurable par nos moyens.

La précision de la mécanique de la tourelle « poignet de robot » permet de supposer que les axes de rotation en site et en azimuth sont concourants en un seul point. Les origines O_a et O_s des repères R_{azimut} et R_{site} liés aux axes de rotation de la tourelle peuvent donc être considérées comme étant confondues.

Une phase d'initialisation automatique permet d'orienter la caméra selon des angles de site et d'azimut de zéro degré [Dietel 96]. Nous utilisons pour cela les quatre butées fixes limitant l'amplitude des mouvements en site et en azimuth. Ce sont des capteurs de fin de course de type détecteurs de proximité inductifs.

La tourelle tourne tout d'abord vers le bas et la gauche jusqu'à la première paire de butées fixes ($site_{-60^\circ}, azimuth_{+159^\circ}$) puis vers le haut et la droite jusqu'à la seconde paire de butées fixes ($site_{+54^\circ}, azimuth_{-159^\circ}$). A partir des incréments comptés entre les différentes butées, nous calculons les incréments correspondants aux consignes d'azimut $\alpha = 0^\circ$ et de site $\beta = 0^\circ$.

La précision angulaire de positionnement obtenue à l'aide des butées inductives est de $\pm 0,8^\circ$ selon les deux axes.

Un décalage maximal en site $\delta\beta$ de $0,8^\circ$ est compatible avec notre application, puisque nous n'effectuons pas de mesure de hauteur précise des arêtes utilisées comme amers environnementaux. Par la suite, nous considérons que la consigne de site $\beta = 0^\circ$ permet d'obtenir un plan image sensiblement perpendiculaire au sol.

Le décalage en azimuth $\delta\alpha$ est plus gênant car il risque de fausser la précision des mesures lors de la projection des primitives laser dans le plan image (et vice versa). Il devra donc être pris en compte lors de la calibration complète du système de perception.

2.3 LE TELEMETRE LASER

Le système de télémétrie retenu est composé d'un télémètre laser Sick PLS200 réalisant une profilométrie 2D et monté sur une tourelle un axe permettant un balayage supplémentaire en site (Figure 2.19). Cette tourelle laser permet l'obtention d'une perception tridimensionnelle de l'environnement [Siadat 99] mais n'est pas utilisée pour nos travaux en raison du temps d'acquisition élevé de cette fonctionnalité.

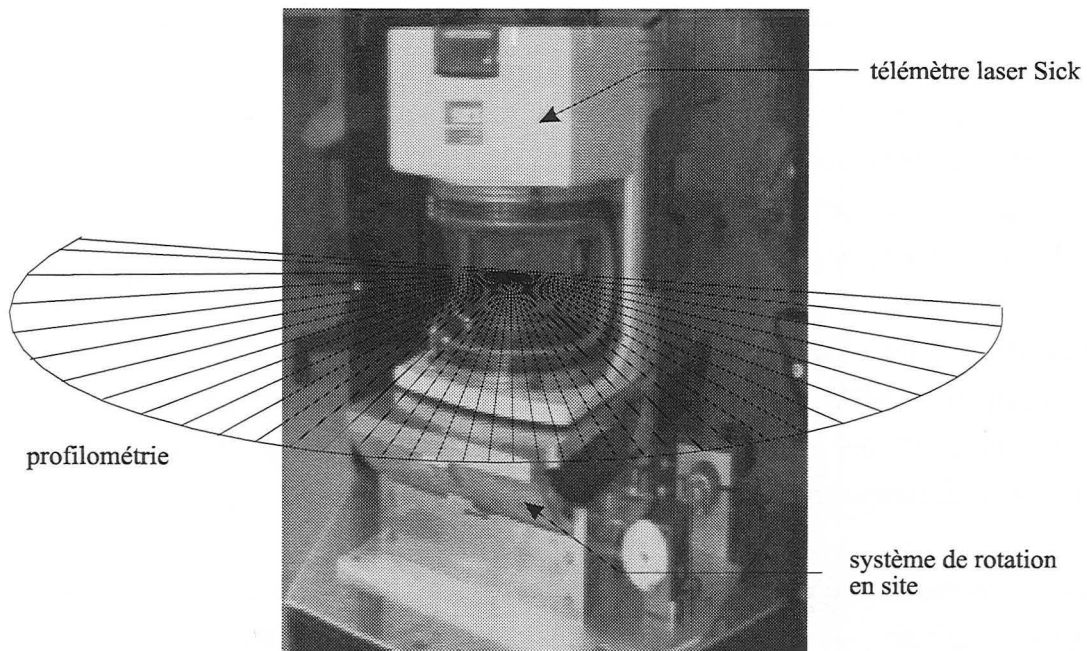


Figure 2.19 : Télémètre laser Sick PLS200.

Nous présentons dans les paragraphes suivants le principe de fonctionnement du télémètre laser, puis nous détaillons ses principales caractéristiques techniques.

2.3.1 Présentation matérielle

De même que les logiciels de traitement d'image, les algorithmes de navigation et de localisation exploitant les données laser sont implémentés sur le poste fixe.

Les données laser brutes sont donc transmises via la liaison numérique qui assure un haut débit, et la modélisation sous forme de primitives géométriques est réalisée depuis le poste fixe. La chaîne de transmission des informations de télémétrie est illustrée Figure 2.20.

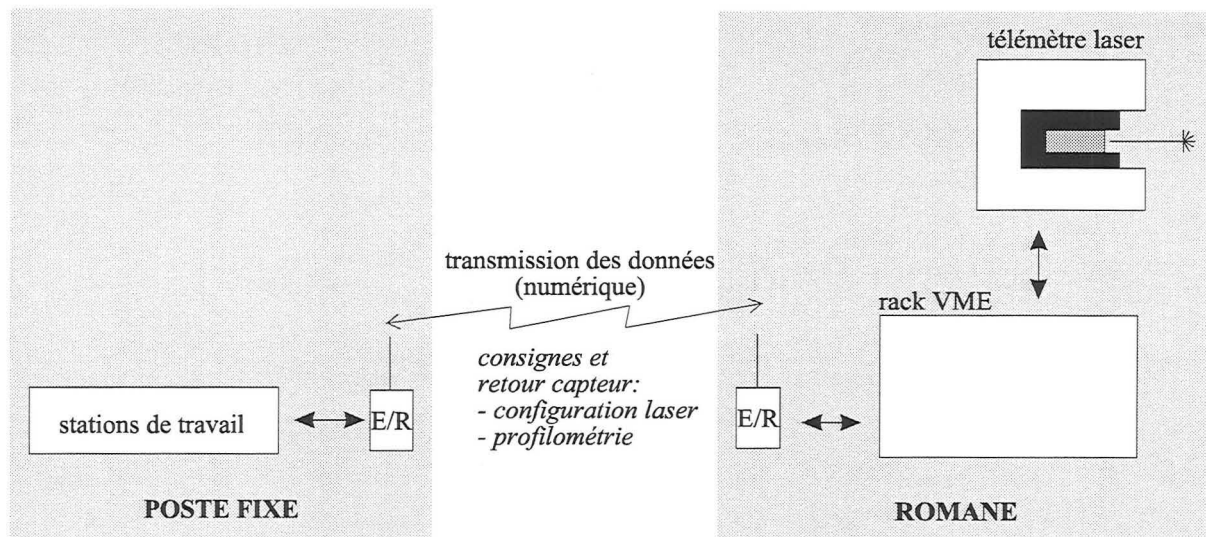


Figure 2.20 : Structure de la transmission des données laser.

2.3.1.1 Principe de fonctionnement

Le télémètre laser utilisé fonctionne selon le principe de la mesure du temps de vol de la lumière. Une impulsion lumineuse très brève est émise en même temps qu'un compteur d'impulsions est déclenché. Si la lumière rencontre un obstacle, le scrutateur reçoit le signal rétrodiffusé et le compteur est arrêté. Le temps qui s'est écoulé entre l'émission du signal et la réception du signal réfléchi permet de calculer la distance séparant le système de l'objet réfléchissant. Comme l'illustre la Figure 2.21, le faisceau lumineux est dévié par un miroir tournant, ce qui lui permet de balayer une surface en forme de demi-cercle.

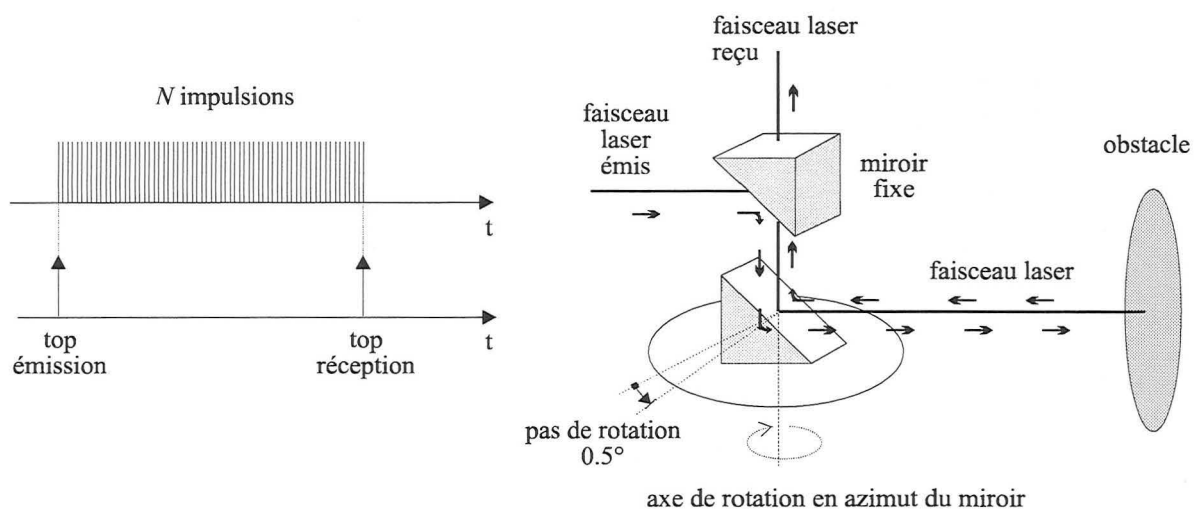


Figure 2.21 : Principe de fonctionnement du télémètre laser SICK PLS200.

L'avantage du capteur utilisé par rapport à ceux qui mesurent un déphasage est qu'il n'existe pas d'intervalle d'ambiguïté inhérent au principe de fonctionnement. Aucun traitement ultérieur n'est donc nécessaire pour accéder à la mesure de profondeur.

2.3.1.2 Acquisition d'une profilométrie

Le télémètre laser possédant un miroir tournant à une fréquence de 25 Hz, il lui faut 40 ms pour réaliser un tour. Une profilométrie de 180° est ainsi réalisée en 20 ms avec un pas de mesure angulaire $\theta = 0,5^\circ$ ce qui correspond à 361 points de mesure. Les 20 ms supplémentaires sont nécessaires au miroir pour retrouver sa position initiale. Le temps de transmission via la liaison série RS232 étant de ~250 ms, le télémètre laser peut donc fournir 4 profilométries brutes par seconde.

2.3.2 Caractéristiques du télémètre laser

Une profilométrie laser *Profil* est constituée de 361 points de mesure P_i caractérisés par leurs coordonnées polaires (d_i, θ_i) exprimées dans le repère laser R_{laser} . Comme l'illustre la Figure 2.22, l'origine O_l du repère R_{laser} est située au centre du balayage et les mesures de distance se font dans le sens trigonométrique (mesure P_0 sur l'axe des abscisses positif).

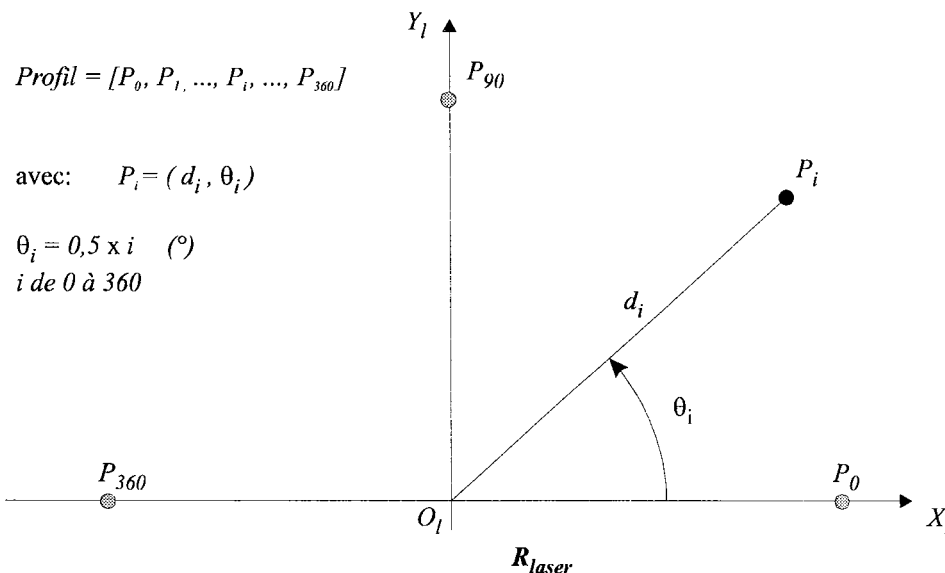


Figure 2.22 : Coordonnées d'un point de mesure p_i dans le repère laser R_{laser} .

Nous présentons ci après les caractéristiques principales du télémètre laser devant être prises en compte afin de déterminer les limites d'utilisation du capteur.

2.3.2.1 Réponse du télémètre pour des cibles de natures différentes

La longueur d'onde du faisceau de lumière émise par le laser étant située dans l'infrarouge, il est totalement insensible à la lumière ambiante, mais pas aux effets thermiques. Toutefois il est nécessaire de s'assurer que l'information de profondeur délivrée par le télémètre soit bien indépendante de la nature de l'objet visé. En effet, pour que la mesure de distance soit fiable, il faut que l'influence de la nature de la cible, ainsi que son orientation par rapport au faisceau laser soit très faible.

La courbe décrite Figure 2.23 donne pour différents types de matériaux le taux de réémission du faisceau laser nécessaire en fonction de la distance à la cible (données constructeur [Sick 95]). Il apparaît que tous les objets peuvent être détectés à moins de 4 mètres quelque soit l'angle d'incidence du faisceau (réémission minimale de 2%). Un plus fort taux de réémission est ensuite nécessaire lorsque la distance augmente. Il se pose alors des limites d'utilisation en fonction du type de matériau de la cible ainsi que l'angle d'incidence du faisceau.

Les essais réalisés dans le cadre de notre application ont démontré que le télémètre laser détecte les murs et les portes de l'environnement sur la plage de distance utilisée (0,1 à 16 mètres) quelque soit l'angle d'incidence du faisceau.

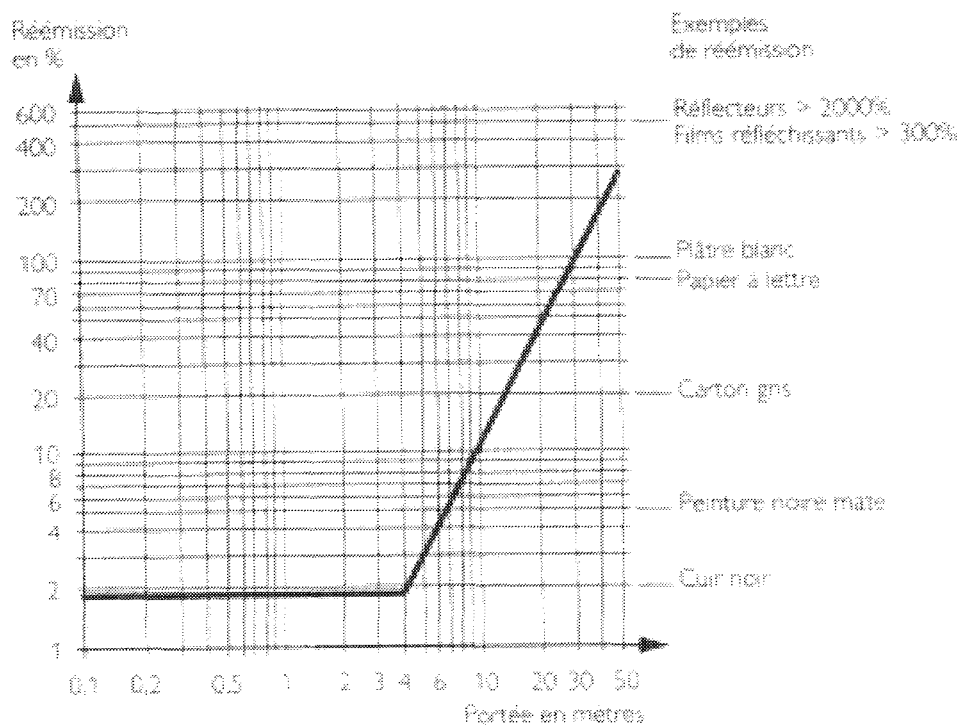


Figure 2.23 : Réponse du télémètre pour des cibles de natures différentes.

2.3.2.2 Divergence du faisceau

La surface de l'impact du faisceau sur une cible est variable en fonction de la distance, en raison de la divergence du faisceau laser. Le diamètre du spot à l'impact augmente ainsi avec la distance. Ceci induit donc une perte d'énergie entre le faisceau laser à la sortie de l'émetteur et le faisceau laser à l'entrée du récepteur (Figure 2.24).

Cette perte d'énergie fixe la portée maximale du faisceau laser telle que la mesure retournée soit correcte, à 50 mètres.

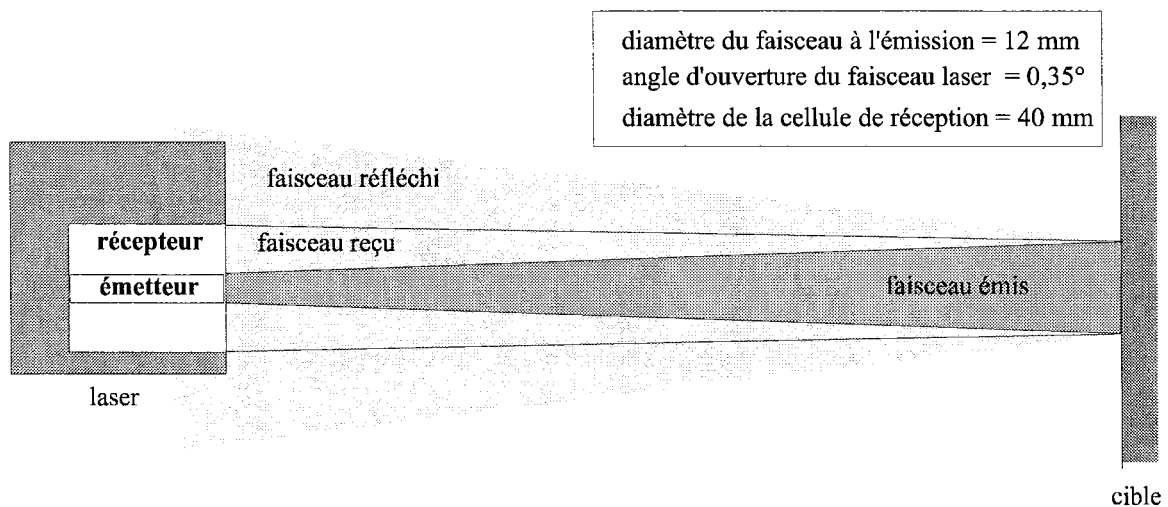


Figure 2.24 : Divergence du faisceau laser.

De plus, pour obtenir une mesure de profondeur d correcte, il faut que la cible reçoive la totalité du spot, sinon il y a phénomène de chevauchement du faisceau laser entraînant la formation de deux impacts à des profondeurs différentes.

Ce chevauchement est illustré par la Figure 2.25 et entraîne une erreur de mesure sur la profondeur d par calcul d'un point de *moyennage*.

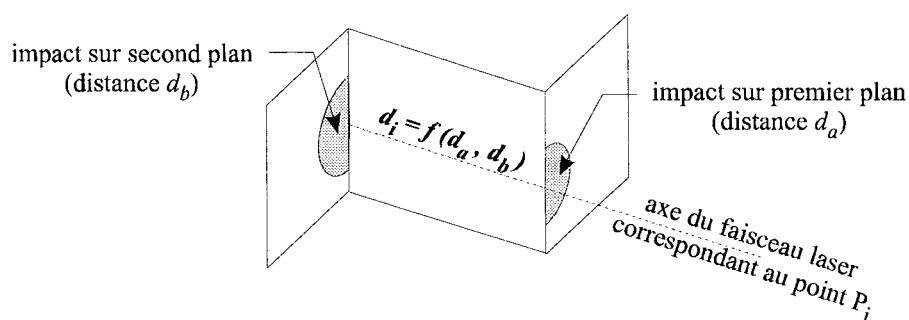


Figure 2.25 : Point de moyennage.

2.3.2.3 Résolution angulaire

La divergence du faisceau ainsi que le pas de mesure angulaire θ limitent la taille des objets pouvant être correctement détectés. La Figure 2.26 illustre le fait que l'on ne peut pas détecter avec précision des structures fines lorsque celles-ci sont trop éloignées du capteur.

Soit : D_{spot} , diamètre de l'impact du spot (mm).
 l_{spot} , distance séparant deux impacts successifs du spot (mm).
 L_{min} , largeur minimale de la cible assurant une détection correcte (mm).

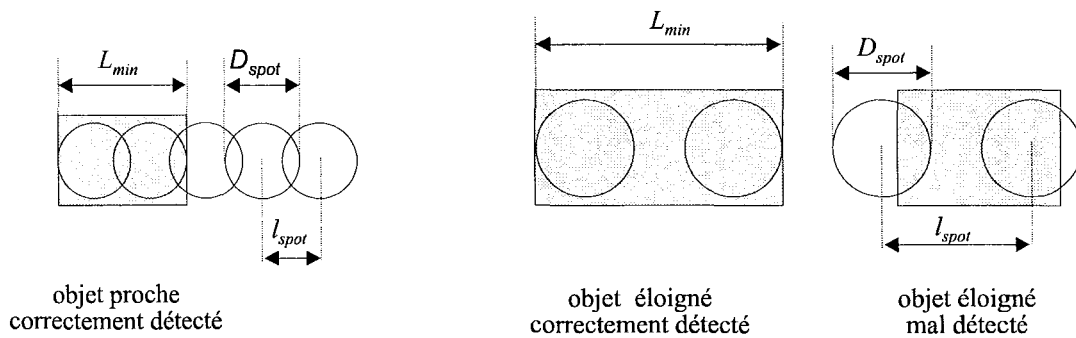


Figure 2.26 : Chevauchement des impacts laser.

Etant donné l'angle d'ouverture du faisceau laser $\lambda_{laser} = 0,35^\circ$ ainsi que le pas de mesure angulaire $\theta = 0,5^\circ$, les spots se chevauchent jusqu'à une portée de 300 cm. Au delà, la détection d'un objet n'est assurée que si sa largeur L_{objet} est supérieure ou égale à $L_{min} = D_{spot} + l_{spot}$. La courbe donnée Figure 2.27 fait apparaître que sur la plage de distance que nous utilisons (de 0,1 à 16 mètres), nous sommes assurés de percevoir des structures d'une largeur minimale de ~23 cm.

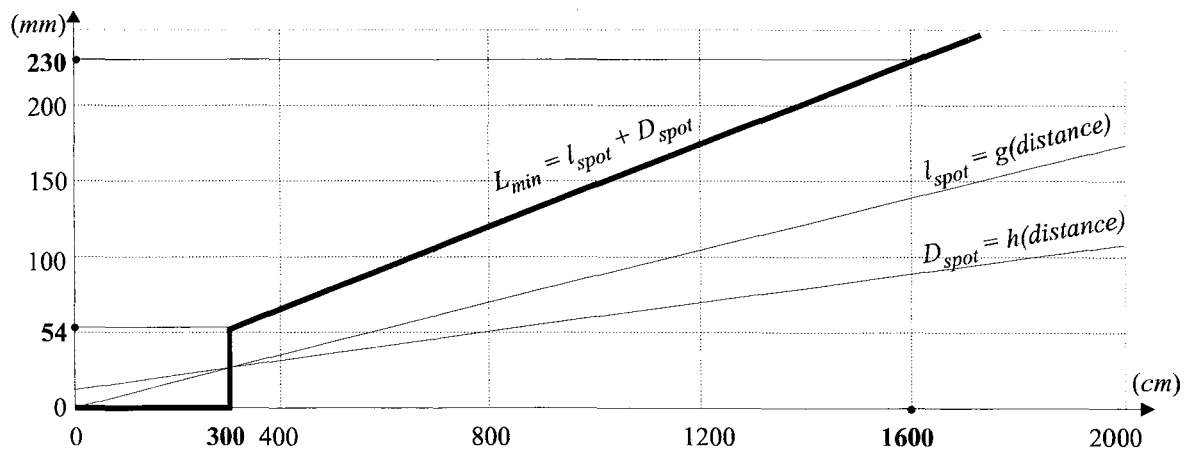


Figure 2.27 : Largeur L_{min} d'un objet pouvant être détecté en fonction de la distance.

2.3.2.4 Résolution de profondeur

La résolution de profondeur du télémètre laser Res_{laser} est donnée par :

$$Res_{laser} = R_{base} + R_{dynamique} \quad (2.12)$$

La résolution de base Res_{base} est liée à l'imprécision sur la mesure du temps de vol. L'erreur maximum de discrétisation $r_{discrétisation}$ introduite par le capteur dépend de la fréquence interne $f_{compteur}$ du compteur du télémètre :

$$Res_{base} = \pm \frac{r_{discrétisation}}{2} = \pm \frac{c}{2 \times f_{compteur}} = \pm 50 \text{ mm} \quad (2.13)$$

Avec :

- $c = \text{vitesse de la lumière } (3.10^8 \text{ m/s})$
- $f_{compteur} = 3 \text{ GHz}$

La résolution dynamique Res_{dyn} du capteur correspond à la somme de toutes les erreurs systématiques introduites par la divergence du faisceau, le diamètre du spot et le taux de réémission de surface. La courbe décrite Figure 2.28 fait apparaître un comportement linéaire de la résolution dynamique Res_{dyn} en fonction de la portée (données constructeur [Sick 95]).

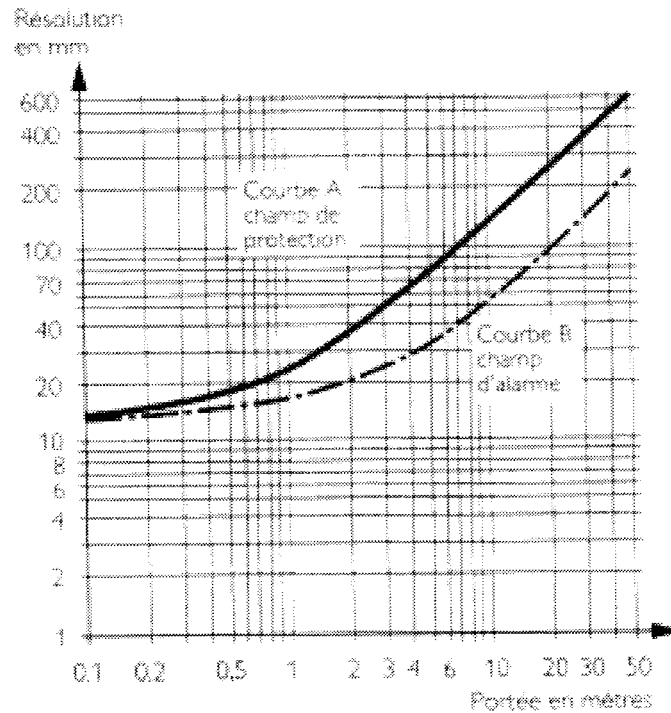


Figure 2.28 : Courbe B, résolution dynamique R_{dyn} en fonction de la distance à la cible (échelles logarithmiques).

Afin de vérifier expérimentalement les valeurs données par le constructeur, nous avons réalisé une série de 100 profilométries dans un environnement stable. Pour chaque faisceau laser d'angle θ_i , nous avons calculé la profondeur moyenne $d_{i \text{ moyen}}$, l'écart type δd_i ainsi que l'écart maximal $\Delta d_i = \max(d_i) - \min(d_i)$ correspondant.

La Figure 2.29 présente pour les différentes profondeurs moyennes $d_{i \text{ moyen}}$ classées par ordre croissant, l'écart type δd_i ainsi que l'écart maximal Δd_i correspondant.

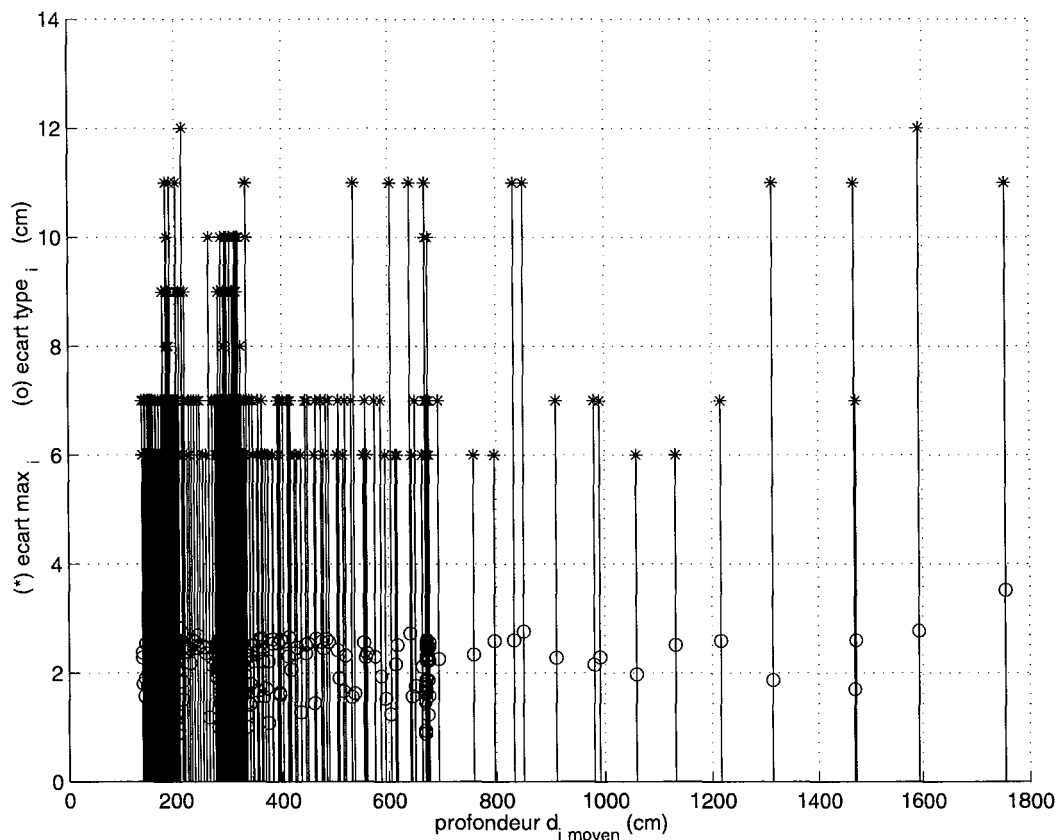


Figure 2.29 : Ecart type δd_i (o) et écart maximal Δd_i (*) classés en fonction des mesures de profondeur moyenne $d_{i \text{ moyen}}$.

Nous pouvons remarquer que l'écart type δd_i est sensiblement constant et égal à 2,5 cm quelle que soit la profondeur. Réaliser une moyenne de profilométries permet donc d'améliorer la précision des mesures, à condition toutefois que le robot soit à l'arrêt et que l'environnement soit stable.

L'écart maximal Δd_i varie de 6 à 12 cm sur la plage de profondeur utile. Bien que relativement importante pour des points de mesure isolés, l'influence de ce biais pourra être diminuée par l'utilisation de primitives de type segment : chaque segment étant défini par une étape de régression portant sur plusieurs points de mesure, la précision des mesures sera améliorée.

2.4 EXPRESSION DES PRIMITIVES SENSORIELLES DANS UN REPERE COMMUN

Les données acquises par les deux capteurs mis en œuvre sont exprimées d'une part dans le repère R_{laser} pour la télémétrie, et d'autre part dans l'image numérisée pour les informations vision. Or la stratégie de perception mise en œuvre dans le cadre de nos recherches repose sur une approche de type fusion de ces informations sensorielles. Il est donc nécessaire de pouvoir les exprimer dans un repère commun, à savoir le repère R_{robot} , qui est utilisé pour la navigation de ROMANE.

Nous présentons dans les paragraphes suivants les différents repères propres aux capteurs utilisés ainsi que la procédure de calibration mise en œuvre, puis nous détaillons les transformations utilisées dans le cadre de notre application.

2.4.1 Les repères du robot

La Figure 2.30 illustre les positions respectives des différents repères liés au robot et à ses capteurs :

R_{robot} , ayant pour origine Or (x_{or}, y_{or}, z_{or}) le milieu de la voie arrière du robot. L'axe X_r est orienté dans le sens de progression du robot, l'axe Y_r est parallèle au plan du sol.

R_{azimut} , ayant pour origine Oa (x_{oa}, y_{oa}, z_{oa}) l'intersection des axes de rotation en site et en azimut de la tourelle. Nous considérons que l'angle de site β de la tourelle porte-caméra est réglé à zéro degré (plan image perpendiculaire au sol) et que les repères R_{site} et R_{azimut} sont par conséquent confondus. L'axe X_a correspondant à une consigne d'azimut $\alpha = 0^\circ$ forme un angle égal au décalage $\delta\alpha$ avec l'axe X_r . L'axe Z_a est parallèle à l'axe Z_r .

$R_{caméra}$, ayant pour origine Oc (x_{oc}, y_{oc}, z_{oc}) le centre optique du capteur CCD. Le positionnement de la caméra étant réalisé à l'aide de picots de centrage sur la tourelle, nous pouvons considérer que les axes Z_a et Y_c sont colinéaires, et que les axes X_a et Z_c sont parallèles.

R_{laser} , ayant pour origine Ol (x_{ol}, y_{ol}, z_{ol}) le centre du balayage du faisceau laser. Les profilométries étant réalisées dans un plan horizontal, la coordonnée z_l sera toujours nulle. L'axe Y_l correspondant à la mesure laser P_{90} forme un angle égal au décalage $\delta\theta$ avec l'axe X_r .

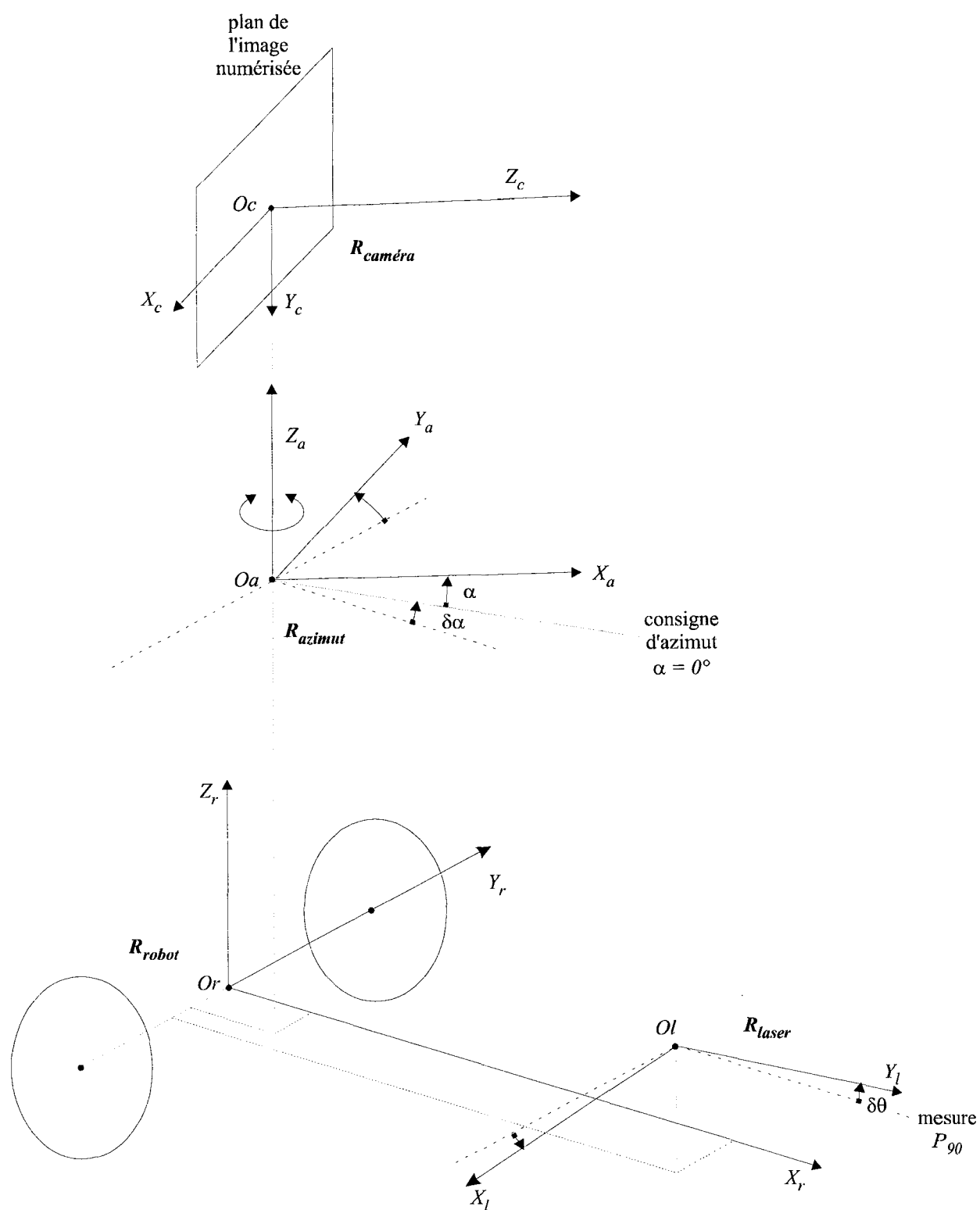


Figure 2.30 : Positions respectives des repères R_{robot} , R_{azimut} , $R_{caméra}$ et R_{laser} .

2.4.1.1 Transformation du repère R_{robot} vers $R_{caméra}$ et R_{robot} vers R_{laser}

Soient (x_r, y_r, z_r) les coordonnées d'un point P exprimées dans le repère R_{robot} .

- Les coordonnées (x_c, y_c, z_c) de sa projection dans le repère $R_{caméra}$ sont données par une suite de translations et de rotations de repères :

$$R_{robot} \text{ à } R_{azimut} : \begin{bmatrix} x_a \\ y_a \\ z_a \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\alpha + \delta\alpha) & \sin(\alpha + \delta\alpha) & 0 & -xoa_r \\ -\sin(\alpha + \delta\alpha) & \cos(\alpha + \delta\alpha) & 0 & -yoa_r \\ 0 & 0 & 1 & -zoa_r \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_r \\ y_r \\ z_r \\ 1 \end{bmatrix} \quad (2.14)$$

Avec : (xoa_r, yoa_r, zoa_r) , les coordonnées de l'origine Oa du repère R_{azimut} exprimées dans le repère R_{robot} (cm).

α , la consigne de rotation en azimuth de la tourelle (deg).

$\delta\alpha$, le décalage angulaire correspondant à une consigne α de 0° (deg).

$$R_{azimut} \text{ à } R_{caméra} : \begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & zoca \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_a \\ y_a \\ z_a \\ 1 \end{bmatrix} \quad (2.15)$$

Avec : $zoca = 10$ cm, (donnée constructeur).

- Les coordonnées (x_l, y_l, z_l) de sa projection dans le repère R_{laser} sont données par une translation et une rotation de repère :

$$R_{robot} \text{ à } R_{laser} : \begin{bmatrix} x_l \\ y_l \\ z_l \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\delta\theta - \pi/2) & \sin(\delta\theta - \pi/2) & 0 & -yol_r \\ -\sin(\delta\theta - \pi/2) & \cos(\delta\theta - \pi/2) & 0 & -xol_r \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_r \\ y_r \\ z_r \\ 1 \end{bmatrix} \quad (2.16)$$

Avec : (xol_r, yol_r) les coordonnées de l'origine Ol du repère R_{laser} exprimées dans le repère R_{robot} (cm).

$\delta\theta$, le décalage angulaire correspondant au point de mesure laser P_{90} (deg).

2.4.1.2 Procédure de calibration

Nous considérons que $xoc_r = xoa_r$ et que $yoc_r = yoa_r$. Puisque nous ne réalisons pas de mesure précise dans le plan vertical, nous pouvons nous contenter d'une mesure directe des paramètres zoc_r et zol_r . Seuls les paramètres $(xoc_r, yoc_r, \delta\alpha)$ et $(xol_r, yol_r, \delta\theta)$ doivent donc être identifiés.

Nous avons choisi pour cela de mettre à profit l'excellente résolution angulaire de la tourelle porte caméra ($0,00064^\circ$ par pas d'incrément). Les consignes de focale et de focus du module de vision seront conservées constantes pour les expérimentations. La position (u_o, v_o) du centre de l'image numérisée sera donc également fixe.

La mire verticale plane que nous utilisons est présentée Figure 2.31. Elle est constituée d'un damier dont les carreaux constituent des arêtes verticales A_k détectables par la vision, ainsi que de deux ouvertures dont les arêtes forment des discontinuités de profondeur détectables par le télémètre laser. Les points de décrochement P_k du faisceau laser se situent dans le prolongement des arêtes verticales A_k correspondants aux bords des carreaux et de la mire.

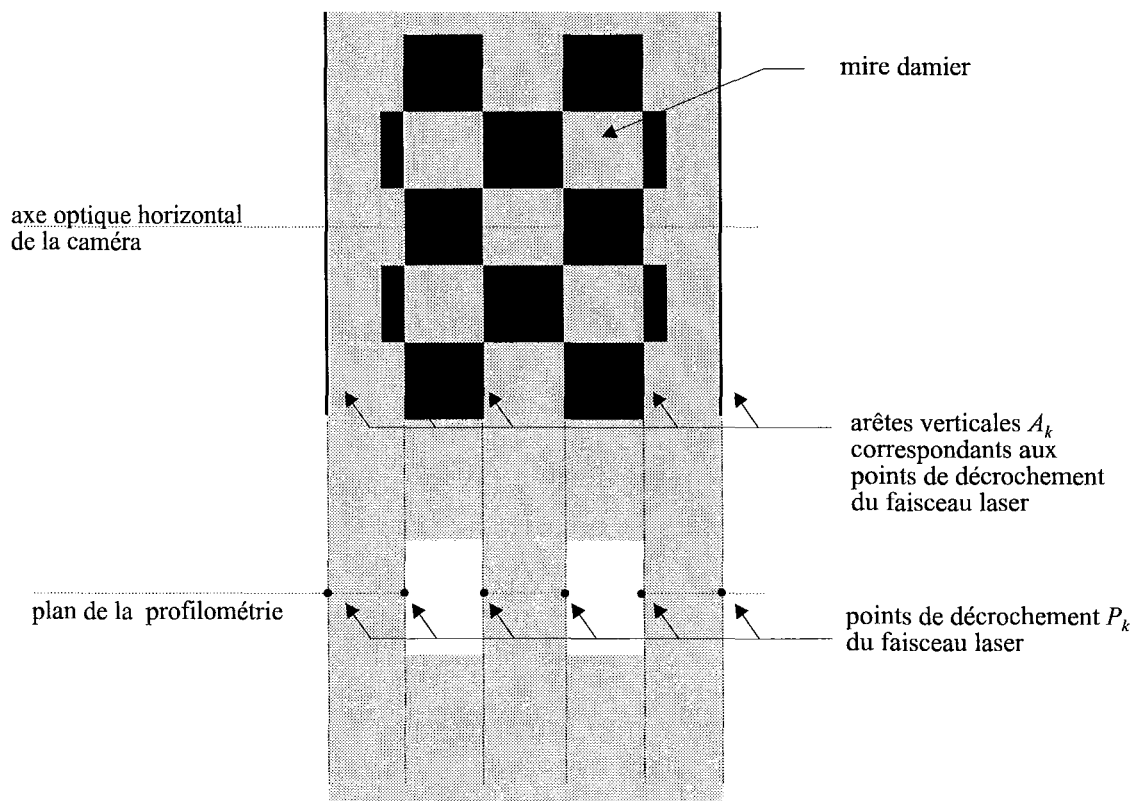


Figure 2.31 : Mire mixte vision/télémétrie laser.

Nous positionnons régulièrement cette mire sur un arc de cercle ayant pour centre l'origine Or du repère R_{robot} et de rayon R (Figure 2.32). Le couple C_i de consignes de focale et de focus de la caméra est choisi en fonction du rayon R .

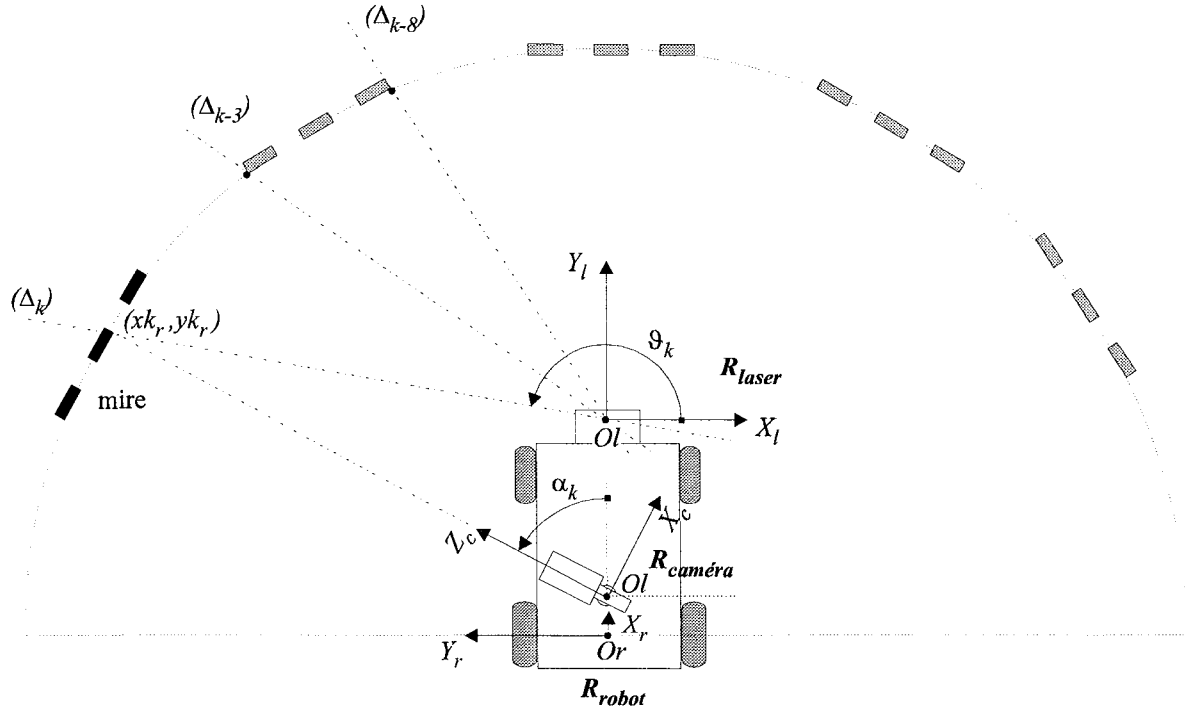


Figure 2.32 : Calibration du système de perception.

Soit (xk_r, yk_r) les coordonnées de la projection dans le plan (X_r, Y_r) de chaque arête verticale A_k , $k=1$ à N .

Pour chaque arête verticale A_k , nous réalisons l'acquisition des données sensorielles suivantes :

- **Télémétrie.** La moyenne de 100 profilométries est réalisée afin d'améliorer la précision des mesures. Puis, nous recherchons dans la profilométrie résultante la discontinuité correspondant à l'arête A_k . Soient (d_k, ϑ_k) les coordonnées polaires du point de décrochement P_k du faisceau laser exprimées dans le repère R_{laser} .
- **Vision.** Nous réglons manuellement la consigne d'azimut α_k de la tourelle porte-caméra de telle sorte que l'arête verticale A_k se trouve sur l'axe optique vertical Y_c de la caméra (colonne pixel u_o).

Toute droite (Δ_k) passant par le point de référence de coordonnées (xk_r, yk_r) et par l'origine Ol du repère R_{laser} a pour équation dans le repère R_{robot} :

$$(\Delta_k) \quad Yk_r = a_k \times Xk_r + b_k \quad (2.17)$$

$$\text{Avec :} \quad \left| \begin{array}{l} a_k = \tan(\theta_k + \delta\theta - \pi/2) \\ b_k = yk_r - xk_r \times a_k \end{array} \right.$$

Les coordonnées (xol_r, yol_r) de l'origine Ol du repère R_{laser} , exprimées dans le repère R_{robot} , ainsi que la valeur de $\delta\theta$, sont déterminées en recherchant le point de convergence des N droites (Δ_k) . Pour cela, il est nécessaire de minimiser le critère J_{laser} :

$$J_{laser} = \sum_{k=1}^N (a_k \times xol_r + b_k - yol_r)^2 \quad (2.18)$$

$$J_{laser} = \sum_{k=1}^N (\tan(\vartheta_k + \delta\vartheta - \pi/2) \times xol_r + yk_r - xk_r \times \tan(\vartheta_k + \delta\vartheta - \pi/2) - yol_r)^2$$

Nous avons obtenu comme valeurs estimées des paramètres :

$$xol_r = 58,3 \text{ cm} \quad yol_r = 0,25 \text{ cm} \quad zol_r = 43 \text{ cm} \quad \delta\theta = 0,24^\circ$$

De même, les coordonnées (xoc_r, yoc_r) de l'origine Oc du repère $R_{caméra}$, exprimées dans le repère R_{robot} , ainsi que la valeur de $\delta\alpha$, sont déterminées en minimisant le critère $J_{caméra}$:

$$J_{caméra} = \sum_{k=1}^N (\tan(\alpha_k + \delta\alpha) \times xoc_r + yk_r - xk_r \times \tan(\alpha_k + \delta\alpha) - yoc_r)^2 \quad (2.19)$$

Nous avons obtenu comme valeurs estimées des paramètres :

$$xoc_r = 2,15 \text{ cm} \quad yoc_r = 0,54 \text{ cm} \quad zoc_r = 133,8 \text{ cm}$$

Comme nous l'avons vu lors de la description technique de la tourelle porte-caméra (paragraphe 2.2.3), le décalage $\delta\alpha$ change lors de chaque initialisation de la tourelle et varie de $-0,8^\circ$ à $+0,8^\circ$. Nous avons donc mis au point une procédure simple permettant de le calculer sans faire appel à une mire sophistiquée.

Nous recherchons dans l'environnement du robot un angle de mur dont l'arête verticale constituera le point de référence P .

Nous réalisons pour cela la moyenne de 100 profilométries. Une segmentation des points de mesure suivie d'une étape de régression (détaillée au chapitre 3), permet alors de calculer précisément les coordonnées (x_l, y_l) de l'intersection de ces deux droites qui constitue le point de référence P (Figure 2.33).

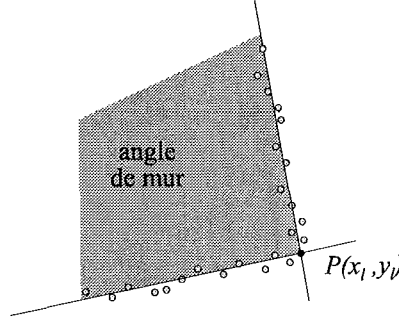


Figure 2.33 : Recherche des coordonnées d'une arête de référence P .

Soit (x_r, y_r) les coordonnées du point P exprimées dans le repère R_{robot} :

$$\begin{bmatrix} x_r \\ y_r \\ z_r \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\pi/2 - \delta\theta) & \sin(\pi/2 - \delta\theta) & 0 & xol_r \\ -\sin(\pi/2 - \delta\theta) & \cos(\pi/2 - \delta\theta) & 0 & yol_r \\ 0 & 0 & 1 & zol_r \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_l \\ y_l \\ z_l \\ 1 \end{bmatrix} \quad (2.20)$$

En considérant $\delta\alpha = 0^\circ$, l'angle d'azimut théorique $\alpha_{théorique}$ permettant de placer l'arête verticale correspondant au point de référence P sur l'axe optique vertical Y_c est donné par :

$$\alpha_{théorique} = a \tan \left(\frac{y_r - yoc_r}{x_r - xoc_r} \right) \quad (2.21)$$

Les consignes de focus et de focale sont réglées, puis la consigne α permettant de positionner réellement l'arête verticale sur l'axe optique vertical Y_c est recherchée manuellement. La valeur du décalage en azimut de la tourelle est alors donnée par :

$$\delta\alpha = \alpha - \alpha_{théorique} \quad (2.22)$$

2.4.2 Exploitation dans le cadre de notre application

La stratégie de perception mise en œuvre repose sur l'inspection par le module de vision de certaines zones de l'environnement mises en évidence dans les informations télémétriques.

Compte tenu des hypothèses liées à l'environnement, nous pouvons considérer que toute primitive *segment* laser définit un plan vertical. Chaque paire ($segment_k$, $segment_{k+1}$) forme ainsi une primitive *coin* dont le sommet S correspond à une arête verticale susceptible d'être répertoriée comme amer environnemental. Le module de vision réalise donc l'inspection de certaines primitives *coin* afin dans un premier temps, de valider l'hypothèse de la présence d'une arête potentielle correspondant au sommet S , puis le cas échéant de rechercher de nouvelles arêtes verticales en analysant les plans verticaux constituant la primitive *coin*.

Les paramètres du module de vision doivent par conséquent être réglés en fonction des coordonnées du sommet S exprimées initialement dans le repère R_{laser} . Puis la projection dans le repère R_{robot} des segments verticaux issus du traitement de l'image doit être calculée en tenant compte des erreurs de modélisation.

2.4.2.1 Réglage des paramètres du module de vision

Soit S , le point intersection de deux primitives laser $segment_k$ et $segment_{k+1}$ successives (Figure 2.34).

Ses coordonnées (x_s, y_s) exprimées dans le repère R_{robot} sont obtenues à l'aide de la matrice de transformation (2.20).

Le couple C_i des consignes de focale et de focus est alors choisi en fonction de la distance du sommet S dans le repère $R_{caméra}$: $\|O_c S\| = \sqrt{(x_{oc_r} - x_{s_r})^2 + (y_{oc_r} - y_{s_r})^2}$.

La consigne d'angle α permettant de placer le point S sur l'axe optique vertical de la caméra est donnée par :

$$\alpha = a \tan \left(\frac{y_{s_r} - y_{oc_r}}{x_{s_r} - x_{oc_r}} \right) + \delta \alpha \quad (2.23)$$

2.4.2.2 Transformation du repère de l'image numérisée vers R_{robot}

La projection d'une arête verticale a extraite de l'image numérisée sur une primitive $segment_k$ laser correspond à un polygone d'incertitude défini par 4 points a_1 a_2 a_3 et a_4 déterminés en prenant en compte :

- les erreurs liées au traitement de l'image numérisée. Toute arête verticale a extraite de l'image numérisée est ainsi caractérisée par les indices colonnes extrêmes u_a^{min} et u_a^{max} .
- les erreurs liées à la modélisation de la primitive $segment_k$ laser. Toute primitive $segment_k$ laser est ainsi caractérisée par la distance moyenne εd_k séparant les points de mesure du segment les modélisant.

Nous déterminons $\Delta seg_k^{intérieur}$ et $\Delta seg_k^{extérieur}$ les droites extrêmes support de la primitive $segment_k$ laser, exprimées dans le repère R_{robot} (Figure 2.34).

- les erreurs liées à la modélisation des paramètres intrinsèques du module de vision, à savoir la longueur focale f , la distance inter plan d_{ip} ainsi que le facteur d'échelle d_x .

Soit Δu_a la ligne de visée correspondant à l'arête verticale d'indice colonne u_a dans l'image numérisée, exprimée dans le repère R_{robot} :

$$\Delta u_a : \quad Y a_r = a \times X a_r + b \quad (2.24)$$

$$\text{Avec : } \begin{cases} a = \tan \left((u_o - u_a) \times \frac{d_x}{f} \right) + \alpha + \delta \alpha \\ b = y o c_r + (f + d_{ip}) \times \sin(\alpha + \delta \alpha) - a \times (x o c_r + (f + d_{ip}) \times \cos(\alpha + \delta \alpha)) \end{cases}$$

Nous déterminons alors les droites Δu_a^{min} et Δu_a^{max} correspondant aux lignes de visée extrêmes (Figure 2.34).

$$\Delta u_a^{min} \text{ est fonction de } \left(u_a^{min}, d_x^{min} = \frac{d_x}{1 + \sigma d_x}, f^{max} = f + \delta f, d_{ip}^{max} = d_{ip} + \delta d_{ip} \right)$$

$$\Delta u_a^{max} \text{ est fonction de } \left(u_a^{max}, d_x^{max} = \frac{d_x}{1 - \sigma d_x}, f^{min} = f - \delta f, d_{ip}^{min} = d_{ip} - \delta d_{ip} \right)$$

Les coordonnées des points définissant le polygone d'incertitude correspondent à :

- a_1 , intersection des droites Δu_a^{min} et $\Delta seg_k^{extérieur}$
- a_2 , intersection des droites Δu_a^{min} et $\Delta seg_k^{intérieur}$
- a_3 , intersection des droites Δu_a^{max} et $\Delta seg_k^{intérieur}$
- a_4 , intersection des droites Δu_a^{max} et $\Delta seg_k^{extérieur}$

Afin de discuter de la précision de la projection, nous définissons les incertitudes de projection δ_x et δ_y à partir des coordonnées moyennes (x_a, y_a) du centre de gravité du polygone d'incertitude défini par les points a_1 a_2 a_3 et a_4 (Figure 2.35).

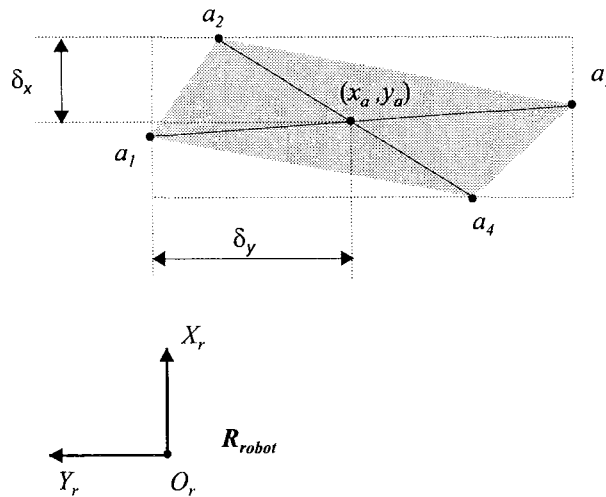


Figure 2.35 : Incertitudes de projection δ_x et δ_y .

Les incertitudes de projection δ_x et δ_y dépendent non seulement des paramètres intrinsèques du module de vision et des erreurs de mesure, mais également de l'angle φ formé par la primitive *segment* laser et le plan image (Figure 2.34).

La Figure 2.36 illustre les incertitudes δ_x et δ_y commises lors de la projection d'un pixel de l'image acquise pour le couple de consigne C_5 à la profondeur z_c de 5600 mm. Seuls les pixels d'indice compris entre $u_o = 283$ et $u_{max} = 512$ sont représentés, le comportement des pixels d'indice compris entre 1 et u_o étant obtenu par symétrie.

Les incertitudes δ_x et δ_y ont été calculées en tenant compte des erreurs de modélisation de la focale f , de la distance inter plan d_{ip} ainsi que du facteur d'échelle d_x . Nous avons également

pris en compte les erreurs sur les mesures, à savoir une erreur d'extraction du pixel de l'image de ± 1 pixel et une erreur εd de 10 mm sur la modélisation du segment laser (La Figure 2.37 illustre le résultat obtenu avec une erreur de mesure εd de 15 mm).

Les courbes en gras représentent les couples « indice de pixel, angle φ » pour lesquels les incertitudes δ_x et δ_y sont inférieures à un seuil de 25 mm.

Nous pouvons constater que les incertitudes de projection δ_x et δ_y deviennent rapidement élevées lorsque nous projetons des pixels éloignés du centre optique u_o pour des valeurs d'angle φ importantes.

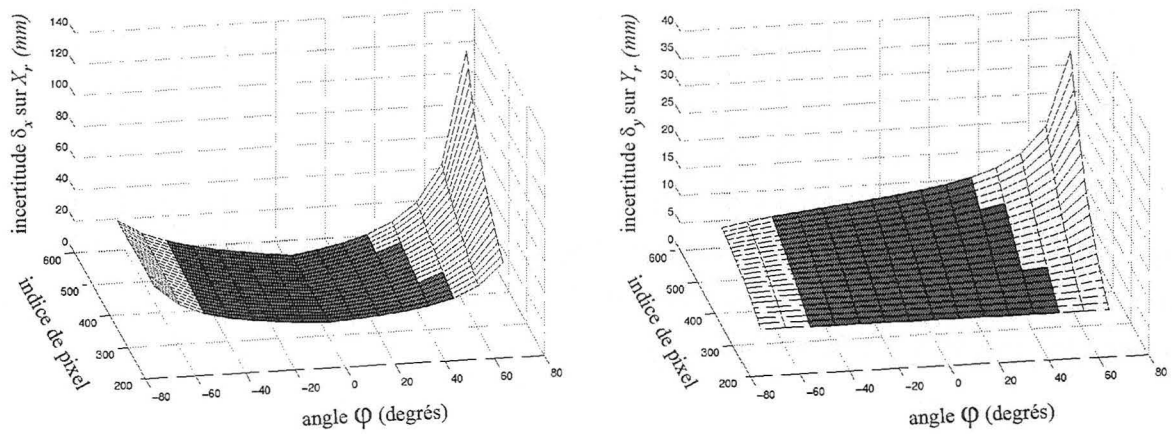


Figure 2.36 : Incertitudes δ_x et δ_y sur la projection dans le repère R_{robot} d'un pixel de l'image pour le couple de consigne C_5 ($\varepsilon d = 10$ mm).

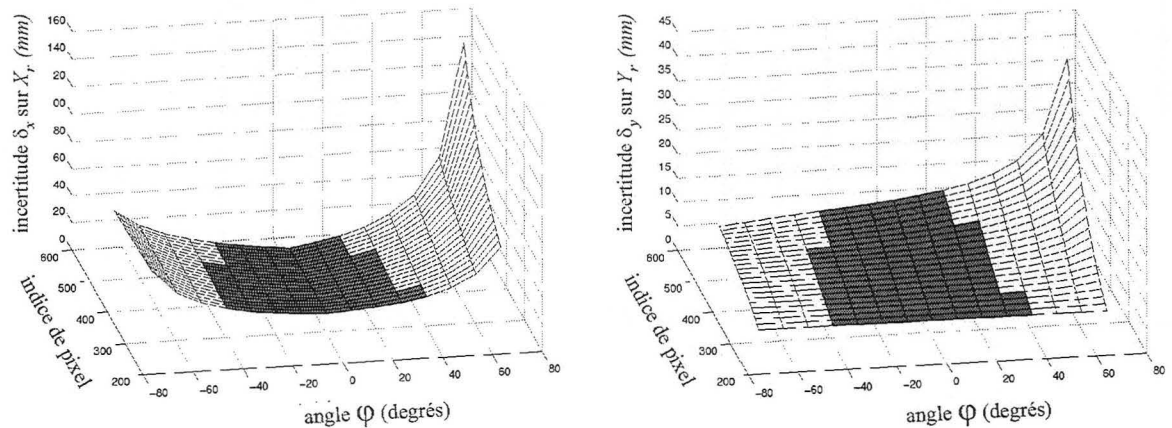


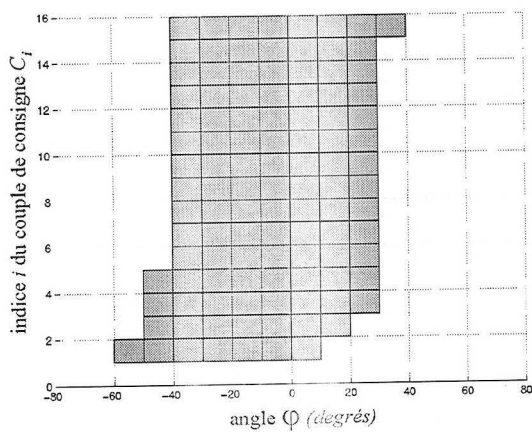
Figure 2.37 : Incertitudes δ_x et δ_y sur la projection dans le repère R_{robot} d'un pixel de l'image pour le couple de consigne C_5 ($\varepsilon d = 15$ mm).

Par conséquent nous choisissons de déterminer pour chaque couple de consigne C_i les indices des pixels pouvant être analysés par la vision tout en garantissant une précision suffisante sur la localisation des points projetés.

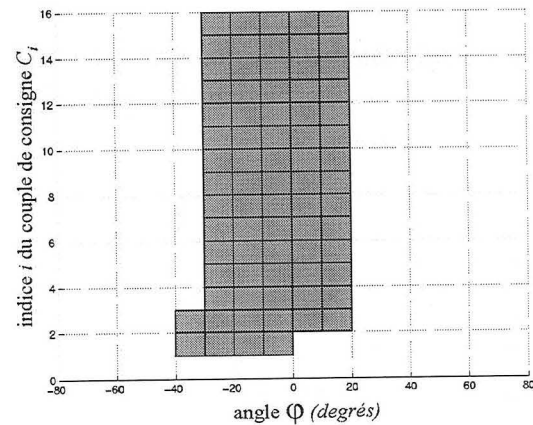
A chaque couple de consigne C_i nous faisons donc correspondre une table d'indexation $Indices_i$ permettant de retrouver pour une valeur d'angle φ et d'erreur εd données, l'indice maximal de la colonne pixel garantissant des incertitudes de projection δ_x et δ_y inférieures à un seuil de 25 mm.

La Figure 2.38 illustre le résultat de la projection du pixel d'indice $u = 512$ correspondant à l'extrémité droite de l'image pour différentes orientations de la primitive *segment* laser représentant le plan droit de la primitive *coin*. Une erreur εd de 10 mm puis de 15 mm est simulée sur le segment. Seuls les couples de consignes pour lesquels les incertitudes de projection δ_x et δ_y sont inférieures au seuil de 25 mm sont indiqués.

Nous pouvons constater que seuls les segments présentant un angle φ par rapport au plan image compris entre -30° et 40° pourront être analysés sur toute la largeur de l'image. De plus, le couple C_I qui correspond à une distance z_c de 1600 mm ne permettra pas l'analyse sur toute la largeur de l'image de segments dont l'angle φ est négatif.



erreur εd de 10 mm



erreur εd de 15 mm

Figure 2.38 : Projection du pixel d'indice 512 engendrant des incertitudes δ_x et δ_y inférieures à un seuil de 25 mm.

2.5 CONCLUSION

La calibration d'un système de vision monoculaire équipé d'un objectif à focale et focus variables est une étape délicate et complexe à réaliser. Afin d'exprimer de façon simplifiée mais réaliste les différents paramètres du processus de formation de l'image, nous avons opté pour le modèle sténopé étendu. De plus, pour limiter les campagnes de mesure, les paramètres de ce modèle n'ont été identifiés que pour les consignes de focale et de focus susceptibles d'être utilisées.

L'étude du télémètre laser a démontré que ses caractéristiques conviennent parfaitement à notre application. Ce type de capteur se révèle être un élément précieux de tout système de perception évolué.

Nous avons présenté une procédure de calibration visant à exprimer les coordonnées des repères R_{laser} et $R_{caméra}$ dans le repère commun R_{robot} en tenant compte d'un décalage aléatoire en azimuth de la tourelle se produisant lors de chaque initialisation. Nous avons pour cela retenu l'hypothèse réaliste d'un plan image sensiblement perpendiculaire au plan du sol.

La stratégie de perception mise en œuvre reposant sur la coopération de ces deux capteurs, nous avons alors détaillé les différentes transformations permettant d'exprimer les coordonnées des primitives laser dans le repère R_{robot} .

Afin de mettre en œuvre la stratégie de perception définie au chapitre 1, nous avons ensuite détaillé l'expression des coordonnées des primitives laser dans le repère R_{robot} , le calcul des consignes d'orientation en azimuth, de focale et de focus du module de vision, la définition de fenêtres d'intérêt au sein de l'image et enfin la projection dans le repère R_{robot} des arêtes verticales mises en évidence dans l'image.

Dans le chapitre suivant, les différentes étapes de la construction d'un modèle local de l'environnement du robot à l'aide des informations sensorielles de la télémétrie et de la vision seront présentées.

Chapitre 3

Construction du modèle local de l'environnement

3.1 INTRODUCTION

La construction d'une représentation cohérente et interprétable de l'environnement local d'un robot mobile est une étape délicate à réaliser car elle doit satisfaire différentes contraintes :

- Le modèle local doit contenir une information spatiale sur l'environnement. Il est en effet nécessaire de connaître la disposition des différents obstacles pouvant gêner la progression du robot.
 - Les primitives géométriques utilisées doivent mettre en évidence les différentes structures de l'environnement. Il s'agit alors de réaliser le meilleur compromis entre de nombreuses primitives décrivant en détail chaque objet mais très lourdes à exploiter, et un nombre réduit de primitives représentant plus grossièrement l'environnement.
 - Les amers environnementaux doivent pouvoir être extraits de l'ensemble des primitives segmentées. Il s'agit alors de minimiser la prise en compte de primitives observées ne correspondant à aucun amer (*faux positifs*), et la non détection d'amers présents (*vrais négatifs*).
 - Le temps nécessaire pour engendrer les primitives observées doit être compatible avec la tâche de localisation.
-

Afin de satisfaire ces contraintes, nous avons défini une stratégie de modélisation de l'environnement local du robot, décomposée en trois phases et décrite par le schéma présenté Figure 3.1.

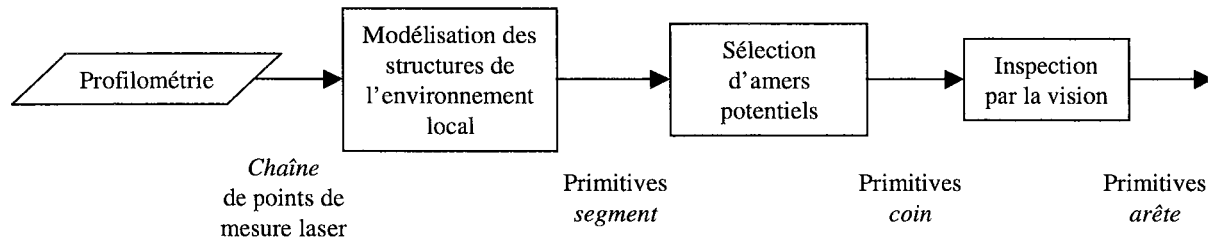


Figure 3.1 : Construction du modèle local de l'environnement.

Un modèle de l'environnement local définissant l'espace horizontal navigable par le robot est tout d'abord créé. Nous utilisons pour cela le capteur télémétrique 2D afin de déterminer des primitives *segment* correspondant aux plans verticaux de l'environnement. Le robot dispose ainsi d'une représentation de l'espace libre lui permettant d'entreprendre des déplacements tout en évitant les obstacles (paragraphe 3.2).

Des primitives laser de plus haut niveau sont ensuite définies pour la tâche de localisation. Il s'agit de primitives *coin* formées à partir de deux primitives *segment* successives. Le sommet de chaque primitive *coin* est la projection dans le plan horizontal d'une arête verticale susceptible de correspondre à un amer du modèle global de l'environnement. Nous déterminons alors pour chaque primitive *coin* un critère d'intérêt visant à sélectionner celles susceptibles d'être répertoriées en tant qu'amers (paragraphe 3.3).

Le module de vision est ensuite utilisé afin d'inspecter successivement ces arêtes potentielles. Si l'analyse valide la présence d'une arête verticale pouvant être considérée comme étant un amer, alors le module de vision réalise l'analyse des plans verticaux droit et gauche de la primitive *coin* afin d'y rechercher de nouvelles arêtes verticales non perçues par la télémétrie (paragraphe 3.4).

Nous obtenons ainsi un ensemble de primitives observées de type *arête verticale* qui seront mises en correspondance avec les amers environnementaux du modèle global afin de déterminer la localisation du robot.

3.2 MODELE LOCAL TELEMETRIQUE

Le modèle local télémétrique est destiné à servir de support à la recherche d'amers environnementaux par la vision.

Un modèle fondé sur une représentation horizontale polygonale par segments de droite convient parfaitement à notre étude puisque nous cherchons à détecter les structures formées par les plans verticaux de l'environnement.

Cette représentation de l'information sous forme de primitives géométriques simples induit un gain en espace mémoire puisqu'elle réduit le volume des données, ainsi qu'un gain en efficacité puisqu'elle fournit un nombre réduit de primitives à manipuler. Par contre, elle impose des contraintes fortes sur les opérateurs de chaînage et de segmentation telles que l'insensibilité au bruit ainsi qu'une implémentation aisée permettant un temps de calcul réduit.

Nous présentons ci-après les principales méthodes d'approximation polygonale couramment utilisées sur des données télémétriques, puis nous détaillons l'algorithme de segmentation mis en œuvre.

3.2.1 L'approximation polygonale

L'approximation polygonale consiste à définir un partitionnement d'une chaîne de points de mesure en plusieurs parties, appelées portions. Chacune de ces portions est ensuite approchée par un segment.

Cette modélisation nécessite donc d'extraire des points de la chaîne une liste de points de coupure qui définissent le partitionnement de la chaîne. En fonction de l'application, soit les extrémités des segments et les points de coupure sont identiques, soit il faut définir des segments par une approximation aux moindres carrés.

Les différentes méthodes existantes peuvent être rangées en deux classes :

Les *méthodes de fusion* [Roberts 65, Wall 83] : le principe consiste à parcourir la chaîne du premier point jusqu'au dernier. Au fur et à mesure du parcours, les points rencontrés sont fusionnés en un même segment tant que l'ensemble de ces points vérifie un certain critère. Le dernier point vérifiant le critère définit un point de coupure et correspond à l'extrémité du segment. On recommence alors le processus à partir de ce point.

D'un point de vue efficacité, ce type d'algorithme est remarquable. Il fonctionne avec un espace minimum en mémoire puisqu'il se contente de parcourir une fois la chaîne. Du fait de la faible complexité des opérations de segmentation, il est également envisageable de segmenter la chaîne en temps réel au fur et à mesure de son acquisition.

Les méthodes de fusion présentent toutefois un défaut majeur : les points de coupure obtenus sont généralement situés au delà des points de coupure réels. Le principe de l'algorithme consiste en effet à fusionner les points tant que le cumul des erreurs ne dépasse pas un certain seuil. Ce défaut est particulièrement sensible pour la détection des angles qui sont alors mal localisés.

Les *méthodes de découpage* [Duda 73, Ayache 89] : dans un premier temps, la chaîne est approchée grossièrement : il n'y a qu'un segment défini par les deux extrémités de la chaîne. L'existence d'un point caractéristique est alors recherchée sur la portion de chaîne correspondant à ce segment. Si ce point existe, le segment est remplacé par deux segments : l'un qui va d'une extrémité de la portion de la chaîne au point caractéristique et l'autre qui va du point caractéristique à l'autre extrémité de la chaîne. Ce processus est alors répété sur chacun des segments.

La présence de bruit peut provoquer la création de points de coupure erronés. C'est pourquoi il est nécessaire de faire suivre l'étape de découpage par une étape de fusion : si il existe deux segments consécutifs dont la fusion ne crée pas un nouveau point caractéristique, alors ils sont fusionnés. Ce type d'algorithme assure alors une très bonne localisation des points de coupure. Toutefois, sa nature récursive nécessite de disposer de la totalité des points de la chaîne avant de commencer la procédure de segmentation.

3.2.2 Mise en évidence des structures formées par des plans verticaux

Dans le cadre de notre application, nous souhaitons créer des primitives de type *segment* correspondant à des structures formées par des plans verticaux de l'environnement (porte, mur, mobilier, ...). Comme ces primitives sont destinées à servir de support au module de vision afin d'extraire de l'image des amers de type arêtes verticales, elles doivent satisfaire à différentes contraintes illustrées par la Figure 3.2 :

- Les points de coupure doivent être localisés avec précision sans quoi les sommets des primitives *coin* seront mal localisés. L'orientation de la caméra étant faussée, l'arête verticale correspondante au sommet risque de ne pas être incluse dans la fenêtre d'intérêt

centrale, ce qui risque de provoquer un rejet du sommet de la primitive *coin* en tant qu'amer (vrai négatif).

- Les primitives *segment* doivent approcher au mieux les structures de l'environnement. Le calcul des coordonnées des arêtes détectées par la vision se faisant par projection sur les primitives segments, la localisation de l'arête dans le repère R_{robot} serait erronée.
- Les primitives *segment* doivent correspondre exclusivement à des structures réelles de l'environnement. Elles ne doivent en aucun cas intégrer un espace libre tel qu'un carrefour de couloir. En effet, il y aurait alors un risque de détecter une arête n'appartenant pas à la primitive coin analysée, ce qui provoquerait la génération d'un amer n'existant pas (*faux positif*).

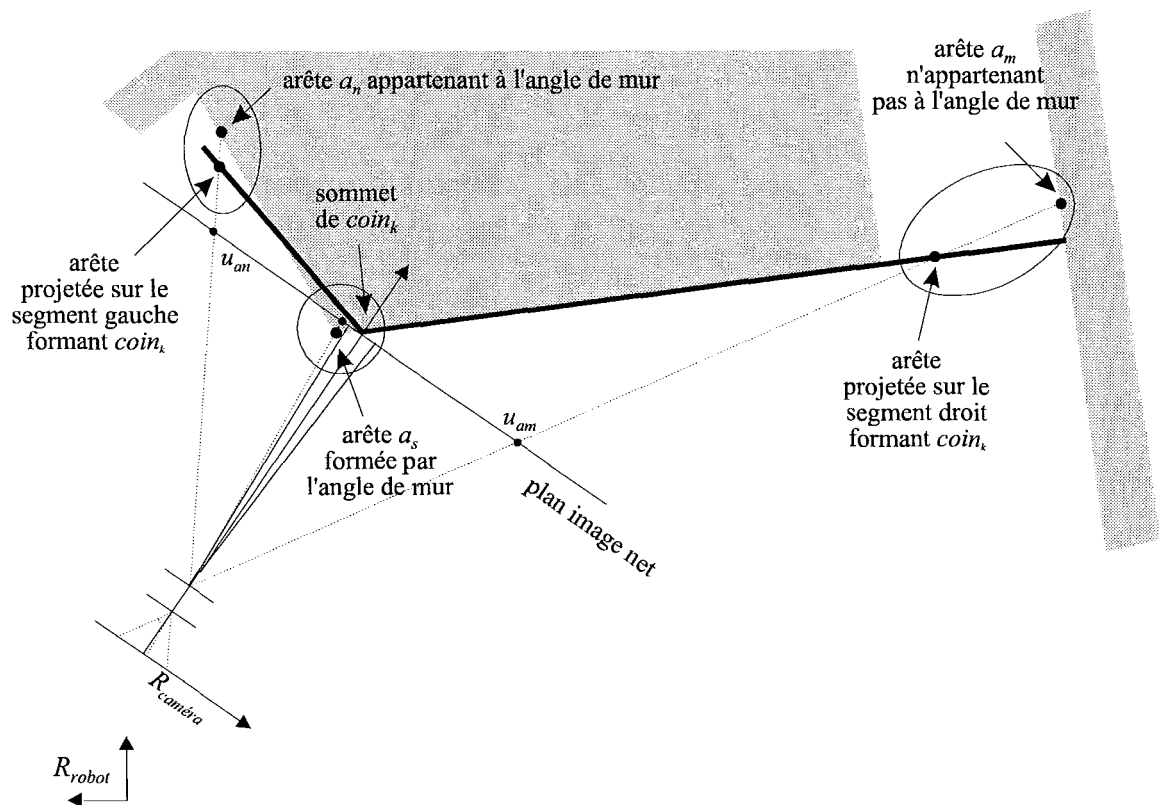


Figure 3.2 : Mauvaise modélisation par une primitive $coin_k$ d'un angle de mur entraînant une erreur de localisation des arêtes a_s , a_n et a_m dans le repère R_{robot} .

Afin de s'assurer d'un partitionnement de la chaîne de points de mesure satisfaisant à ces contraintes, nous proposons d'utiliser successivement un algorithme de fusion puis un algorithme de découpage. Les portions ainsi engendrées feront ensuite l'objet d'une approximation aux moindres carrés afin de définir des points de coupure les plus précis possibles.

3.2.2.1 Acquisition de la profilométrie laser

Le télémètre laser réalise une profilométrie horizontale à une hauteur constante couvrant un arc de cercle de 180 degrés avec un pas angulaire de 0,5° degré par mesure. Nous obtenons ainsi pour chaque profilométrie une chaîne de 361 points de mesure P .

$$chaîne = \{P_0, P_1, P_2, ..., P_k, ..., P_{360}\} \quad (3.1)$$

Chaque point de mesure P_k étant défini par ses coordonnées polaires $[d_k, \theta_k]$, nous exprimons ses coordonnées cartésiennes (x_k, y_k) dans le repère R_{laser} .

$$x_k = d_k \times \cos(\theta_k) \quad y_k = d_k \times \sin(\theta_k) \quad (3.2)$$

3.2.2.2 Filtrage de la chaîne de mesure

Un filtrage est tout d'abord réalisé afin d'éliminer de la chaîne les points de mesure susceptibles d'engendrer des primitives non représentatives de l'environnement. Il s'agit de points de mesure P_k caractérisés soit :

- par une profondeur d_k très supérieure à celle de ses deux voisins de la chaîne P_{k-1} et P_{k+1} . Ces points correspondent généralement à des discontinuités causées par de légers interstices entre deux plans verticaux (La Figure 3.3 illustre un exemple correspondant au cas d'une porte ouverte avec un espace libre entre ses gonds).

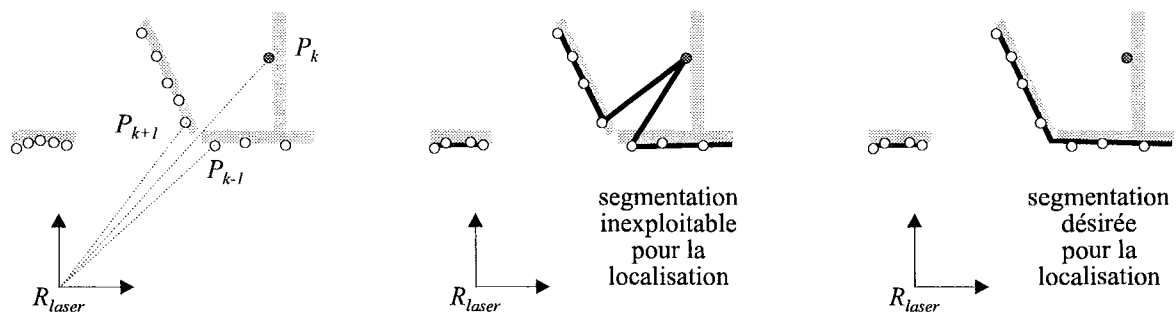


Figure 3.3 : Exemple de point de mesure P_k entraînant une erreur de segmentation au niveau d'une porte.

- par une profondeur d_k proche de la distance maximale pouvant être mesurée par le télémètre (50 m). Il s'agit alors d'un faisceau n'engendrant aucun écho (tir à l'infini).

3.2.2.3 Fractionnement de la chaîne

Nous recherchons ensuite quelles sont les fractions de la chaîne correspondant avec une quasi certitude à des plans verticaux de l'environnement. La chaîne initiale est donc fractionnée en plusieurs chaînes partielles :

$$chaîne_k = \{ (P_{début}, \dots, P_{fin}), statut_k \} \quad (3.3)$$

$$chaîne = \{ chaîne_1, chaîne_2, \dots, chaîne_k, \dots, chaîne_n \}$$

Avec : $P_{début}, \dots, P_{fin}$, points de mesure P constituant la chaîne.

$statut_k$, chaîne réelle ou virtuelle. Les chaînes *réelles* sont définies par plus de deux points de mesure P et correspondent à un ou plusieurs plans verticaux de l'environnement. Les chaînes *virtuelles* sont définies par deux points de mesure P et sont susceptibles de correspondre à un espace vide de l'environnement.

Le fractionnement de la chaîne initiale est réalisé par un algorithme de fusion dont le principe est illustré par la Figure 3.4.

L'algorithme est initialisé par le premier point de mesure, $P_{début} = P_0$. Etant donné tout nouveau point P_k , si la distance l_k séparant le point P_k de son voisin P_{k-1} est supérieure à un seuil l_{fusion} , alors on considère que l'extrémité P_{fin} de la chaîne *réelle* courante a été trouvée au point P_{k-1} . Les points P_{k-1} et P_k définissent une chaîne *virtuelle* correspondant à une zone pour laquelle on ne peut assurer qu'il existe un plan vertical. Une nouvelle chaîne partielle initialisée par $P_{début} = P_k$ est ensuite créée et l'analyse se poursuit.

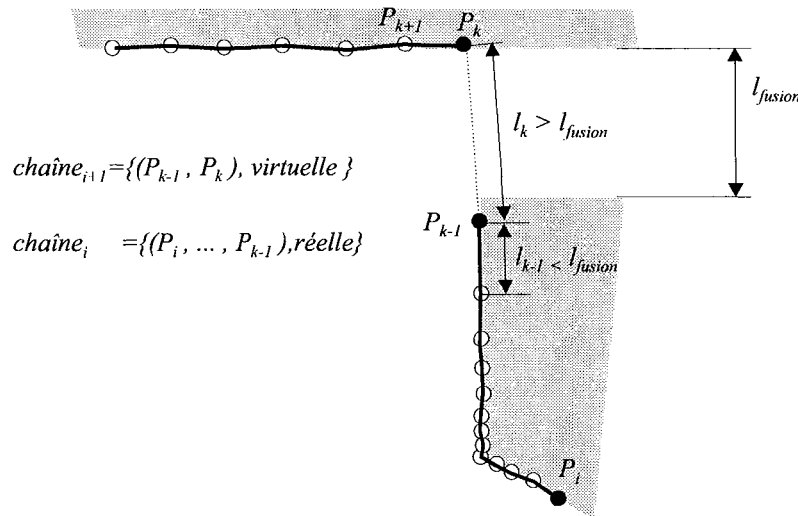


Figure 3.4 : Fractionnement de la chaîne initiale par une méthode de fusion.

Ce fractionnement de la chaîne initiale permet de mettre en évidence les structures de type plan vertical de l'environnement. Toutefois, il est important de noter que les extrémités des chaînes partielles générées ne correspondent pas obligatoirement aux extrémités des plans qu'elles modélisent.

Le choix du seuil l_{fusion} dépend entièrement de l'environnement dans lequel évolue le robot. Il doit correspondre à la plus petite structure vide que l'on veut être capable de détecter : largeur d'un couloir (155 cm), encadrement d'une porte (72,5 à 145 cm), ...

Pour notre application, nous avons choisi $l_{fusion} = 60$ cm afin d'être assuré que les chaînes partielles réelles engendrées correspondent uniquement à des plans verticaux.

La Figure 3.5 montre une profilométrie fractionnée par cette méthode. Nous pouvons constater qu'une seule chaîne partielle peut correspondre à plusieurs plans verticaux.

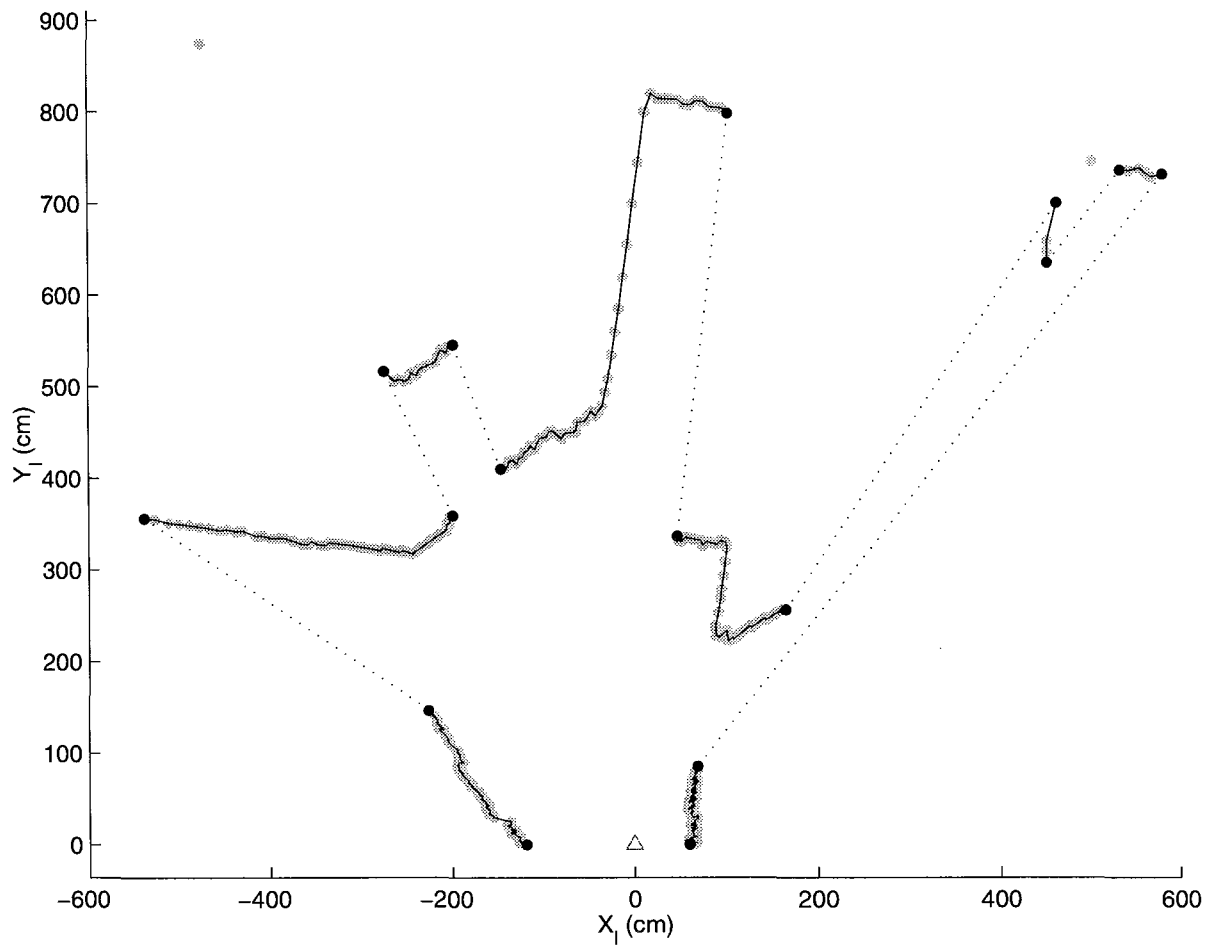


Figure 3.5 : Chaînes obtenues après filtrage et fractionnement de la profilométrie A.

3.2.2.4 Partitionnement des chaînes partielles

Les chaînes trouvées précédemment sont partitionnées afin d'isoler les différents plans verticaux les uns des autres.

$$chaîne_k = \{portion_1, portion_2, \dots, portion_i, \dots, portion_m\} \quad (3.4)$$

$$portion_i = \{(P_{droit}, \dots, P_{gauche}), statut_i\}$$

Avec :

- $P_{droit}, \dots, P_{gauche}$, points de mesure P constituant la portion.
- $statut_i$, une *portion réelle* est définie à partir d'une chaîne réelle et correspond à un plan vertical de l'environnement. Une *portion virtuelle* est définie à partir d'une chaîne virtuelle.

Afin de trouver les points de coupure de chaque chaîne, nous avons tout d'abord recours à un algorithme de découpage de type *Iterative End Point Fit* [Crowley 89] dont le principe est illustré par la Figure 3.6.

Chaque chaîne est tout d'abord approchée par une seule portion définie par les extrémités $P_{début}$ et P_{fin} de la chaîne considérée. Soit (Δ) la droite passant par les extrémités de la portion. On recherche alors le point P de la portion le plus éloigné de la droite (Δ) . Si cette distance est supérieure à un seuil $l^l_{découpage}$, alors ce point P constitue un point de coupure créant deux nouvelles portions composées des points $\{P_{début}, \dots, P\}$ et $\{P, \dots, P_{fin}\}$ auxquelles on applique le même processus.

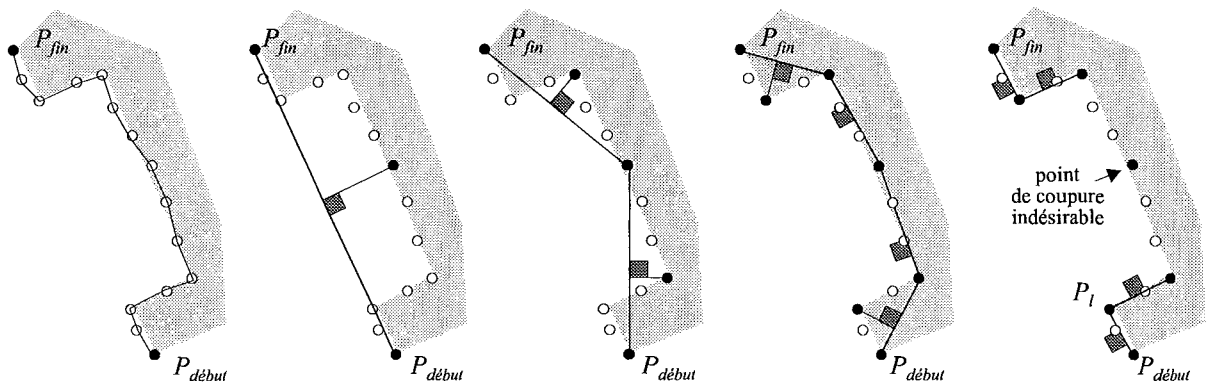


Figure 3.6 : Partitionnement d'une chaîne (1^{ère} passe).

Comme nous pouvons le constater dans l'exemple donné Figure 3.6, les points de coupure sont bien localisés, mais le bruit de mesure peut créer des points de coupure indésirables. Il est donc nécessaire de fusionner les portions jugées non représentatives pour notre application.

Soient P_a , P_b et P_c trois points de coupe successifs. Si la distance séparant le point P_b de la droite (Δ) passant par P_a et P_c est inférieure à un seuil $l^2_{découpage}$, alors le point P_b n'est plus considéré comme étant un point de coupe et l'analyse se poursuit avec les trois points de coupe suivants comme l'illustre la Figure 3.7. Le seuil $l^2_{découpage}$ peut être choisi supérieur à $l^1_{découpage}$ afin de diminuer le nombre de segments créés tout en conservant des points de coupe précis.

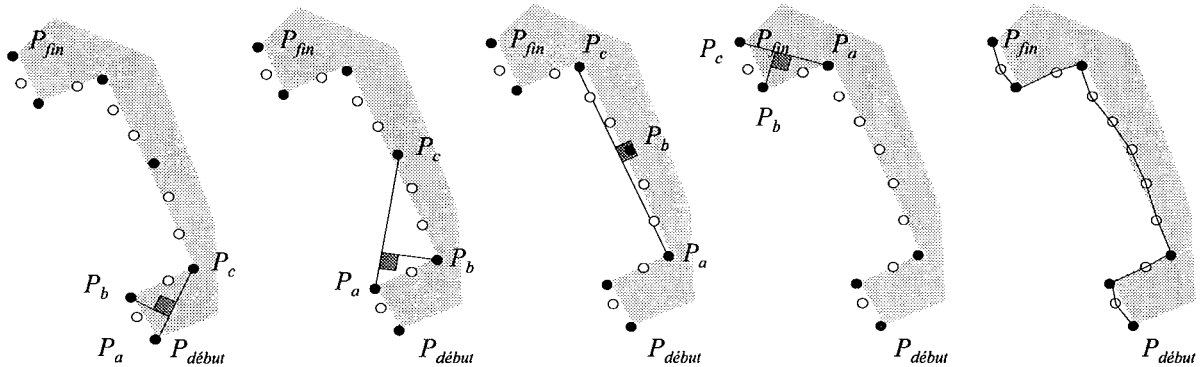


Figure 3.7: Partitionnement d'une chaîne (2^{de} passe).

La détermination des seuils $l^1_{découpage}$ et $l^2_{découpage}$ est délicate car il s'agit de réaliser le meilleur compromis entre de nombreuses petites portions représentant fidèlement l'environnement mais délicates à exploiter, et un nombre réduit de portions plus longues modélisant grossièrement celui-ci.

Afin de fixer des valeurs de seuil exploitables sur l'ensemble de l'espace de manœuvre du robot, nous avons déterminé trois zones caractéristiques de l'environnement :

- une zone *A*, caractéristique d'un couloir ou d'une pièce vide. Elle est constituée de longues parois devant engendrer un faible nombre de portions.
- une zone *B*, caractéristique d'un couloir ou d'une pièce à faible encombrement. Elle est constituée de longues parois et de quelques objets devant engendrer un nombre limité de portions.
- une zone *C*, caractéristique d'un couloir ou d'une pièce à fort encombrement. Elle est constituée de longues parois et de beaucoup d'objets devant engendrer un nombre important de portions.

Dans chaque zone, nous avons acquis 50 profilométries pour des positions différentes du robot. Nous avons alors appliqué différentes valeurs de seuil $l^1_{\text{découpage}}$ (avec $l^2_{\text{découpage}} = 2 * l^1_{\text{découpage}}$) à l'ensemble de ces profilométries.

Le Tableau 3.1 fait apparaître pour chaque zone :

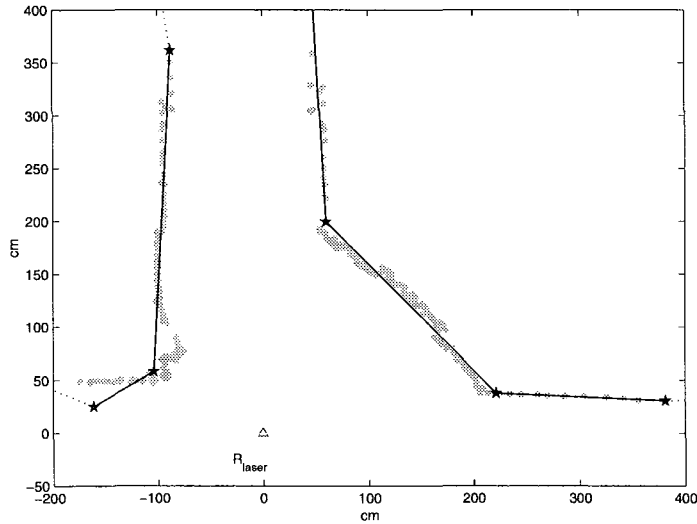
- M , le nombre optimal moyen de portions que nous souhaiterions obtenir (appréciation subjective),
- N , le nombre moyen de portions générées par l'algorithme de segmentation,
- ε_d , la distance moyenne séparant un point de mesure P de son segment (le détail de la segmentation est donnée au paragraphe 3.2.2.5). Cette mesure est représentative de l'erreur de modélisation, et donc de la précision des primitives *segment*.

Seuil $l^1_{\text{découpage}}$ (cm)		14	12	10	8	6	4
Zone A	N portions générées	4	5	6	8	11	14
	$M=7$ portions						
	Erreur moyenne ε_d (cm)	3,02	1,85	1,55	1,35	1,17	0,90
Zone B	N portions générées	13	16	17	20	24	28
	$M=19$ portions						
	Erreur moyenne ε_d (cm)	3,1	2,7	2,09	1,64	1,42	1,28
Zone C	N portions générées	25	28	29	32	38	46
	$M=33$ portions						
	Erreur moyenne ε_d (cm)	3,57	3,39	3,09	2,41	2,29	1,78

Tableau 3.1 : Partitionnement de la chaîne de mesure pour 3 environnements types.

La Figure 3.8 présente un exemple de segmentation réalisée avec différents seuils $l^1_{\text{découpage}}$. Nous pouvons constater qu'un seuil élevé ne permet pas de garantir une bonne localisation des points de coupure, alors qu'un seuil trop faible génère de nombreux petits segments inexploitable pour notre application. Un seuil $l^1_{\text{découpage}}$ de 8 cm permet d'obtenir une segmentation conforme à nos objectifs, à savoir un nombre réduit de primitives ainsi qu'une faible erreur de modélisation.

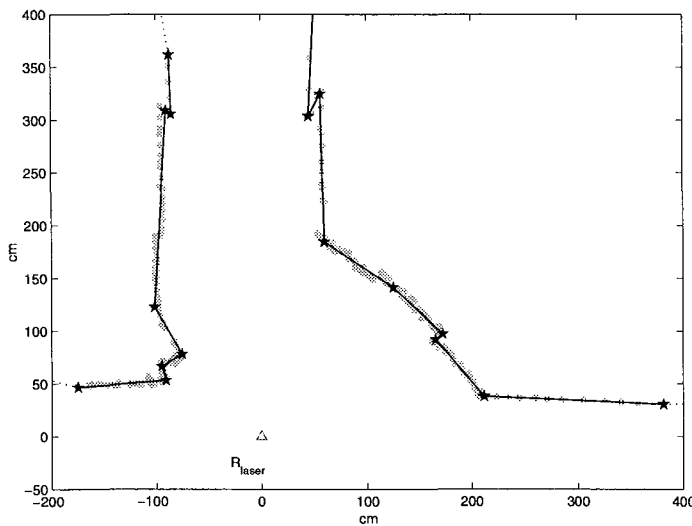
La Figure 3.9 montre un partitionnement réalisé avec les valeurs de seuil $l^1_{\text{découpage}} = 8$ cm et $l^2_{\text{découpage}} = 2 * l^1_{\text{découpage}}$.



$$l^l_{\text{découpage}} = 14 \text{ cm.}$$

Un seuil trop élevé ne permet pas de garantir une bonne localisation des points de coupure.

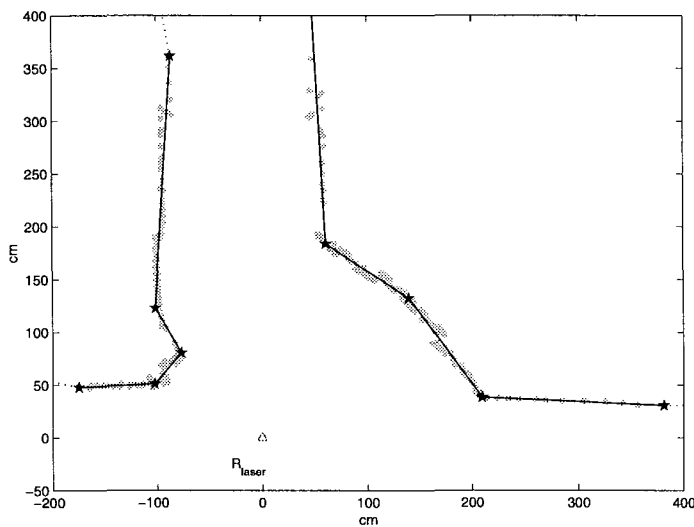
La segmentation résultante est donc trop approximative pour être exploitée dans le cadre de notre application.



$$l^l_{\text{découpage}} = 4 \text{ cm}$$

Un seuil trop faible génère un très grand nombre de points de coupure.

Les segments résultants sont alors trop petits pour être exploités dans le cadre de notre application.



$$l^l_{\text{découpage}} = 8 \text{ cm}$$

Un seuil intermédiaire permet d'obtenir des points de coupure bien localisés.

Les segments résultants sont alors suffisamment longs et représentatifs pour être exploités dans le cadre de notre application.

Figure 3.8 : Résultat du partitionnement et de la segmentation pour différentes valeurs de seuil $l^l_{\text{découpage}}$.

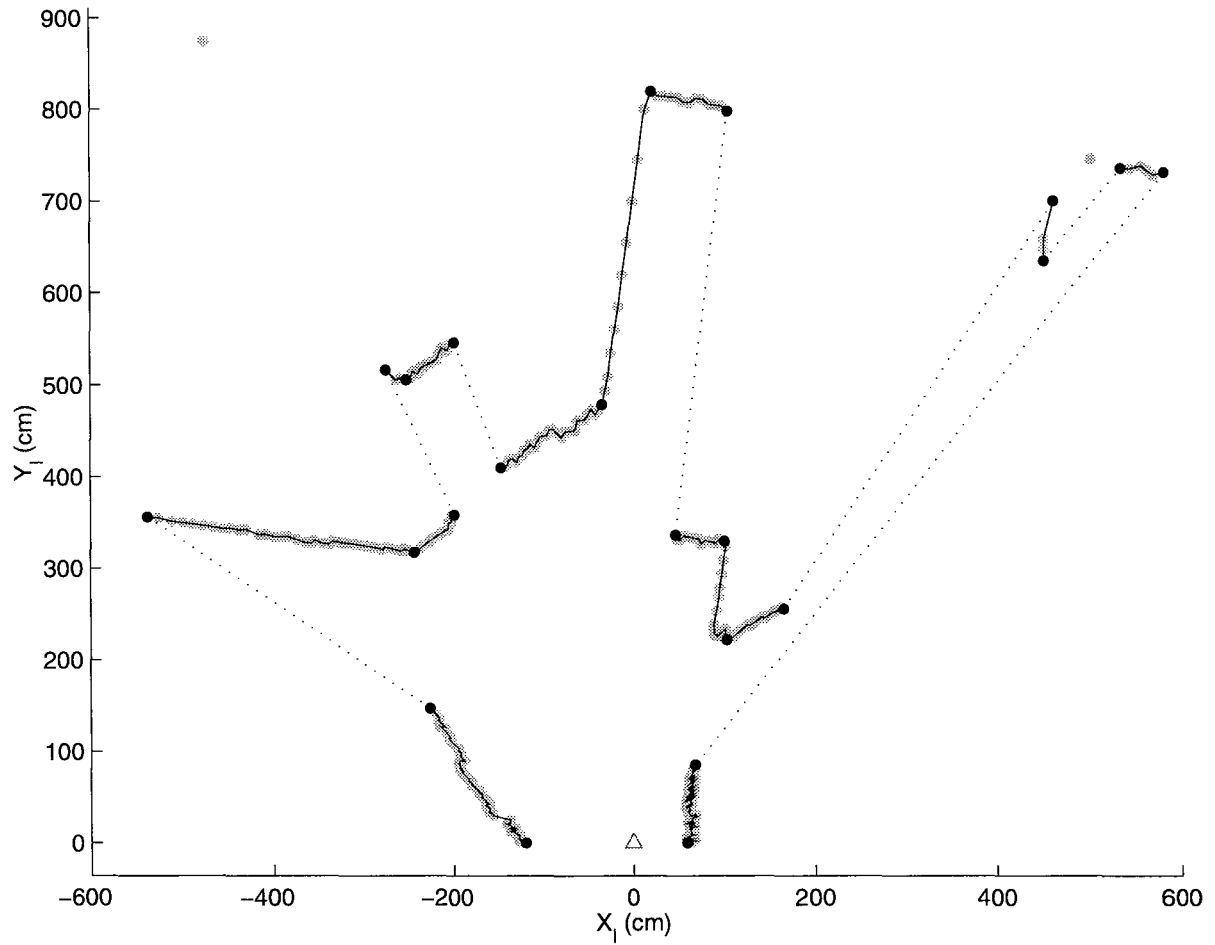


Figure 3.9 : Partitionnement de la chaîne de mesure décrivant la profilométrie A.

3.2.2.5 Segmentation des portions

Les portions trouvées précédemment sont segmentées afin d'obtenir des primitives approchant au mieux les points de mesure P .

$$portion_i = \{(P_{droit}, \dots, P_{gauche}), statut_i\} \Rightarrow segment_i = \{(p_i, p_{i+1}), ed_i, statut_i\} \quad (3.5)$$

Avec : p_i, p_{i+1} , les points extrémités du segment.

ed_i , distance moyenne séparant les points de mesure P du segment.

$statut_i$, segment réel ou virtuel. Seuls les *segments réels* sont assurés de correspondre à un plan vertical de l'environnement.

Nous déterminons tout d'abord pour chaque $portion_i$ constituée des n points de mesure $\{P_g, \dots, P_{g+n}\}$, la droite (Δ_i) approchant le mieux, au sens des moindres carrés linéaires, les points de mesure P de coordonnées (x, y) :

$$(\Delta_i) \quad Y = a_i \times X + b_i \quad (3.6)$$

Avec :

$$a_i = \frac{n \sum_g^{g+n} x_g y_g - \sum_g^{g+n} x_g \sum_g^{g+n} y_g}{n \sum_g^{g+n} x_g^2 - \left(\sum_g^{g+n} x_g \right)^2} \quad \text{et} \quad b_i = \frac{\sum_g^{g+n} x_g^2 \sum_g^{g+n} y_g - \sum_g^{g+n} x_g \sum_g^{g+n} x_g y_g}{n \sum_g^{g+n} x_g^2 - \left(\sum_g^{g+n} x_g \right)^2}$$

La distance moyenne ed_i séparant les points de mesure P du segment est alors donnée par :

$$ed_i = \frac{1}{n\sqrt{a_i^2 + 1}} \sum_g^{g+n} |y_g - a_i x_g - b_i| \quad (3.7)$$

Les points de coupure définis par les points de mesure P_g et P_{g+n} n'étant plus valides du fait de la régression (ils n'ont aucune raison d'appartenir à la droite Δ_i), nous recherchons les coordonnées des points extrêmes p_i et p_{i+1} du segment.

Soient $portion_i = \{p_i, \dots, P_k\}$ et $portion_{i+1} = \{P_k, \dots, P_{k+n}\}$ deux portions successives. La Figure 3.10 détaille la procédure de recherche des points extrêmes d'un segment. Le point de coupure P_k est éliminé et remplacé par le point extrême p_{i+1} dont les coordonnées sont données par :

- la projection du point de coupure P_k sur la droite (Δ_i) si $portion_i$ est réelle et $portion_{i+1}$ virtuelle (**cas n°1**),
- la projection du point de coupure P_k sur la droite (Δ_{i+1}) si $portion_i$ est virtuelle et $portion_{i+1}$ réelle (**cas n°2**),
- le point d'intersection des deux droites porteuses (Δ_i) et (Δ_{i+1}) si $portion_i$ et $portion_{i+1}$ sont toutes les deux réelles (**cas n°3**).

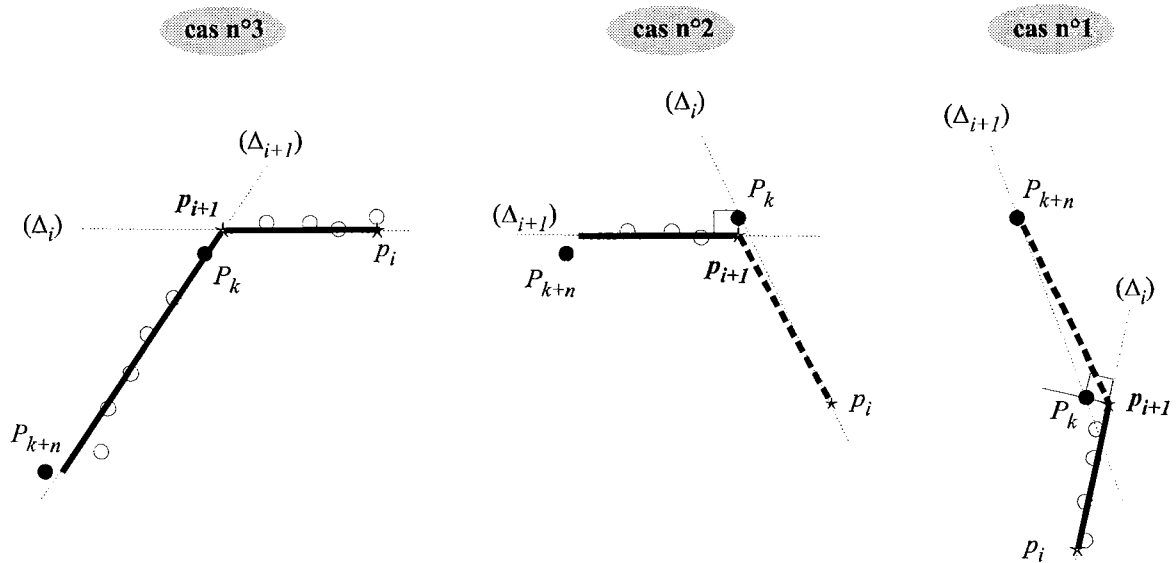


Figure 3.10 : Recherche des points extrêmes des segments.

La Figure 3.11 montre le résultat de la segmentation d'une profilométrie : les segments réels engendrés par cette procédure approchent bien les points de mesure et leurs points extrémités sont correctement localisés.

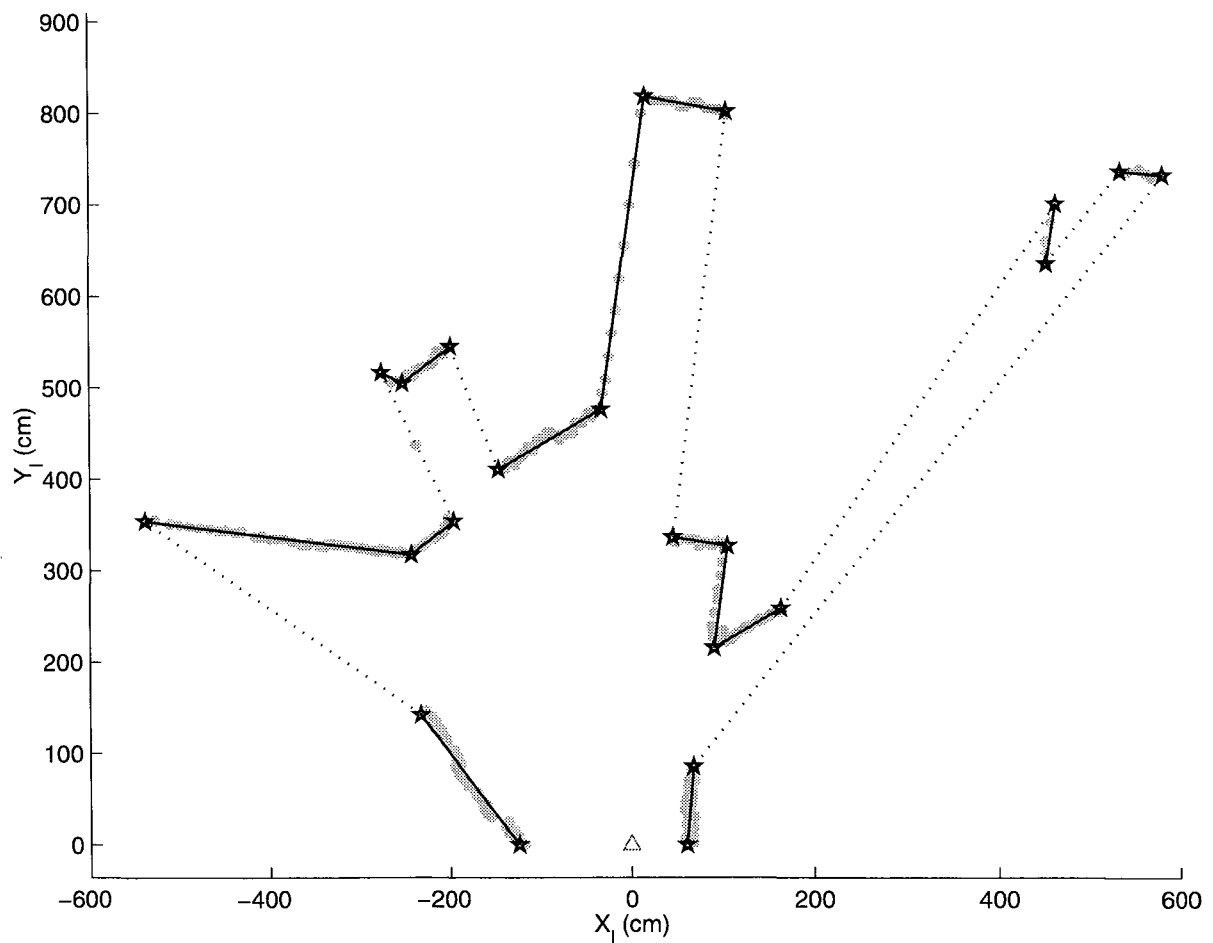


Figure 3.11 : Profilométrie A segmentée.

3.3 MISE EN ÉVIDENCE DE CARACTÉRISTIQUES POTENTIELLES

Le modèle local télémétrique est donc décrit par une *chaîne* constituée de primitives *segment* correspondant chacun à un plan vertical de l'environnement :

$$chaîne = \{segment_1, segment_2, ..., segment_i, segment_{i+1}, ..., segment_M\} \quad (3.8)$$

$$segment_i = \{(p_i, p_{i+1}), ed_i, statut_i\}$$

La stratégie de perception mise en œuvre étant basée sur la confirmation et l'enrichissement des informations télémétriques par les informations vision, nous limitons dans un premier temps le modèle télémétrique en recherchant quelle partie est accessible au module de vision.

Nous définirons ensuite des primitives *coin* formées à partir de deux primitives *segment* successives. Le sommet de chaque primitive *coin* correspond donc à une arête verticale susceptible d'être répertoriée comme amer.

L'analyse de l'ensemble des primitives *coin* par le module de vision conduisant à un temps de calcul prohibitif, nous attribuons à chacune d'elle un critère d'intérêt représentatif, dans la perspective d'une inspection par la vision, de l'intérêt de la structure modélisée.

3.3.1 Espace accessible par le système de vision

Les repères R_{laser} et $R_{caméra}$ n'étant pas confondus, certaines zones du modèle local télémétrique ne sont pas toujours accessibles au module de vision. Une partie des primitives *segment* ne pourra donc pas être inspectée par la vision.

3.3.1.1 Points extrémités p accessibles au module de vision

Soit L_p la liste des points extrémités des segments du modèle télémétrique local :

$$L_p = \{p_1, p_2, ..., p_i, p_{i+1}, ..., p_M, p_{M+1}\} \quad (3.9)$$

Nous recherchons la liste l_p des points p perceptibles par le module de vision compte tenu de la position de l'origine O_c du repère $R_{caméra}$ dans le repère R_{laser} .

Il s'agit donc de déterminer pour chaque point $p_i, i=1 \dots M+1$ si le segment $[O_c, p_i]$ coupe ou non l'un des $segment_k, k=1 \text{ à } M$ d'extrémités p_k et p_{k+1} . Ce phénomène d'occlusion est illustré par la Figure 3.12.

Soient (dp_i, θ_{p_i}) les coordonnées polaires d'un point p_i exprimées dans le repère $R_{caméra}$ (consigne d'azimut $\alpha = 0^\circ$), et (Δp_i) la droite passant par les points O_c et p_i .

- Si $\theta_{p_{k+1}} < \theta_{p_i}$ ou $\theta_{p_i} < \theta_{p_k}$, alors le $segment_k$ ne masque pas p_i et le segment suivant est testé.
- Sinon p_i appartient au secteur angulaire défini par les points p_k et p_{k+1} . Nous recherchons alors les coordonnées (x_{int}, z_{int}) du point P_{int} , intersection de la droite (Δp_i) avec le $segment_k$. Le point p_i est considéré comme non perceptible par le module de vision si :

$$dp_i > \sqrt{x_{int}^2 + z_{int}^2} \quad (3.10)$$

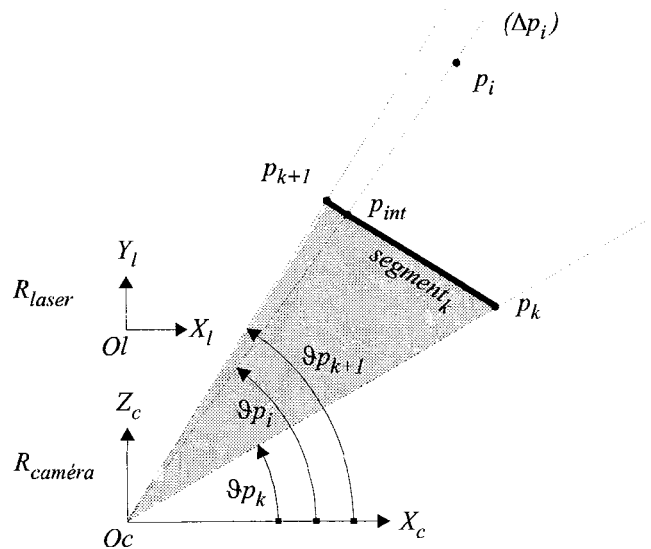


Figure 3.12 : Occlusion du point p_i du modèle local télémétrique due au $segment_k$.

Nous construisons ainsi la liste ordonnée des points p perceptibles par le module de vision :

$$l_p = \{p_1, \dots, p_i, p_j, \dots, p_{M+1}\} \quad (3.11)$$

3.3.1.2 Primitives segments accessibles au module de vision

Connaissant la liste l_p des points extrémités p accessibles, nous construisons la *chaîne_{vision}* décrivant les primitives *segment* pouvant être inspectés par le module de vision.

Soient p_i et p_j ($i < j$), deux points consécutifs de la liste l_p . Les 4 configurations permettant de construire le modèle *chaîne_{vision}* sont détaillées ci après et illustrées par la Figure 3.13.

- Si $p_j = p_{i+1}$, **alors** *segment_i* est entièrement accessible au module de vision (**cas n°0**).

$$chaîne_{vision} = \{chaîne_{vision}, segment_i\} \quad (3.12)$$

- Sinon une partie du modèle télémétrique local compris dans le secteur angulaire formé par les deux points p_i et p_j est partiellement inaccessible au module de vision. Les règles suivantes sont alors utilisées pour déterminer quelle est la fraction de l'environnement visible :

1. Si $\theta p_j > \theta p_{j-1}$, **alors** une partie de *segment_{j-1}* est tronquée (**cas n°1**). Soit p^* le point d'intersection de *segment_{j-1}* et de la droite (Δp_i) . Un *segment_a* virtuel ainsi qu'un *segment_b* représentant la portion visible du *segment_{j-1}* sont créés.

$$chaîne_{vision} = \{chaîne_{vision}, segment_a, segment_b\} \quad (3.13)$$

$$\text{Avec : } \left\{ \begin{array}{l} segment_a = \{(p_i, p^*), 0, \text{virtuel}\} \\ segment_b = \{(p^*, p_j), \epsilon d_{j-1}, \text{réel}\} \end{array} \right.$$

2. Si $\theta p_i < \theta p_{i+1}$, **alors** une partie de *segment_i* est tronquée (**cas n°2**). Soit p^* le point d'intersection de *segment_i* et de la droite (Δp_j) . Un *segment_a* représentant la portion visible de *segment_i* et un *segment_b* virtuel sont créés.

$$chaîne_{vision} = \{chaîne_{vision}, segment_a, segment_b\} \quad (3.14)$$

$$\text{Avec : } \left\{ \begin{array}{l} segment_a = \{(p_i, p^*), \epsilon d_i, \text{réel}\} \\ segment_b = \{(p^*, p_j), 0, \text{virtuel}\} \end{array} \right.$$

3.3.1.3 Modèle local accessible au module de vision

Le modèle local télémétrique perceptible par le module de vision est donc décrit par la *chaîne_{vision}* constituée de segments de droite réels ou virtuels.

$$\begin{aligned} chaîne_{vision} &= \{segment_1, segment_2, ..., segment_i, segment_{i+1}, ..., segment_M\} \\ segment_i &= \{(p_i, p_{i+1}), \epsilon d_i, statut_i\} \end{aligned} \quad (3.16)$$

Comme nous pouvons le constater sur la Figure 3.14, une partie non négligeable de l'environnement peut être inaccessible au module de vision si les repères $R_{caméra}$ et R_{laser} sont trop éloignés.

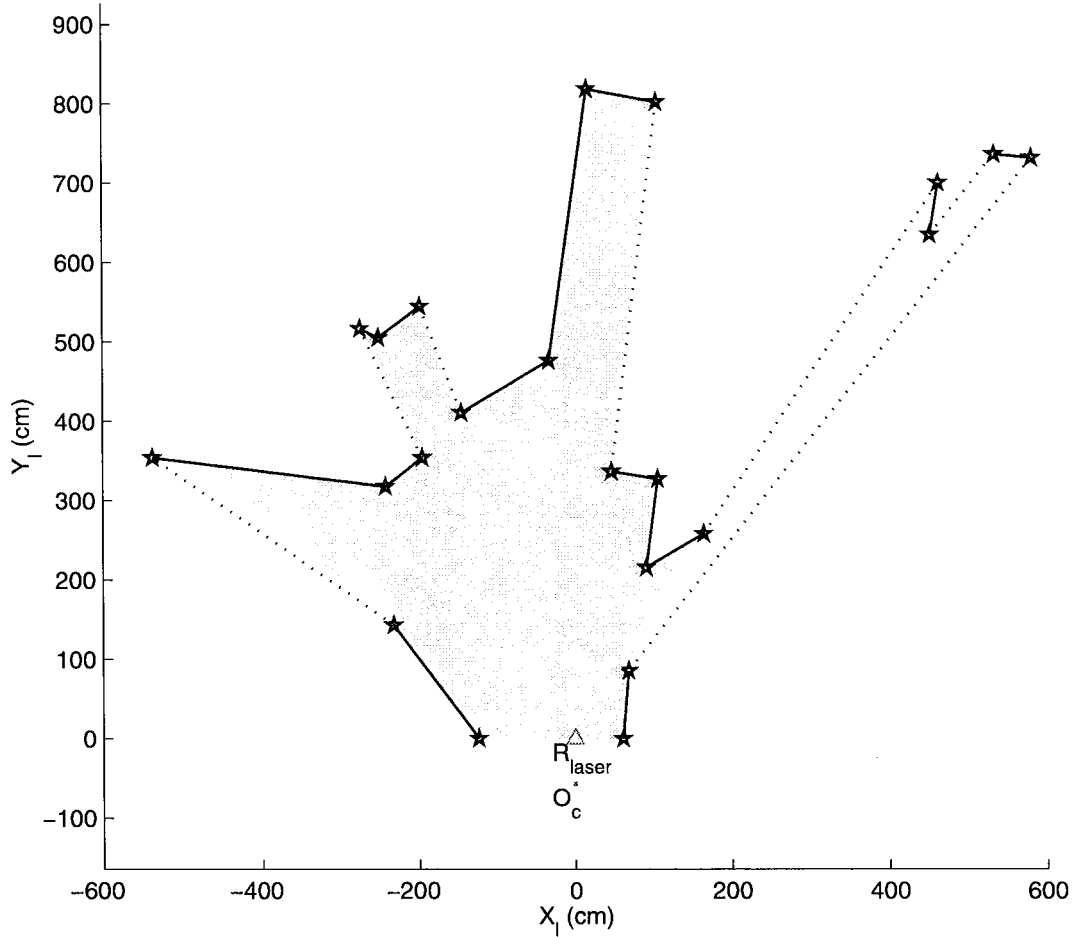


Figure 3.14 : Portion du modèle local télémétrique A accessible par le module de vision.

3.3.2 Génération de primitives coin

Soit $segment_k$ et $segment_{k+1}$ deux primitives *segment* successives du modèle local télémétrique perceptible par le module de vision :

$$segment_k = \{(p_k, p_{k+1}), \varepsilon d_k, statut_k\} \quad segment_{k+1} = \{(p_{k+1}, p_{k+2}), \varepsilon d_{k+1}, statut_{k+1}\}$$

Ces deux segments correspondent à deux plans verticaux et forment une primitive *coin_k* définie dans le plan (X_r, Y_r) par :

- $D_k = p_k$, point extrémité du plan droit.
- $S_k = p_{k+1}$, sommet de la primitive coin.
- $G_k = p_{k+2}$, point extrémité du plan gauche.
- $\varepsilon plan D_k = \varepsilon d_k$, erreur de modélisation du plan droit.
- $\varepsilon plan G_k = \varepsilon d_{k+1}$, erreur de modélisation du plan gauche.
- $plan D_k = statut_k$, statut *réel* ou *virtuel* du plan droit.
- $plan G_k = statut_{k+1}$, statut *réel* ou *virtuel* du plan gauche.
- Ω_k , angle formé par les plans gauche et droit.

Comme l'illustre la Figure 3.15, le sommet S de chaque primitive coin correspond à une arête verticale pouvant être répertoriée en tant qu'amer, alors que les plans verticaux droit et gauche sont susceptibles de contenir des amers non détectés par la télémétrie (arêtes a_i et a_j).

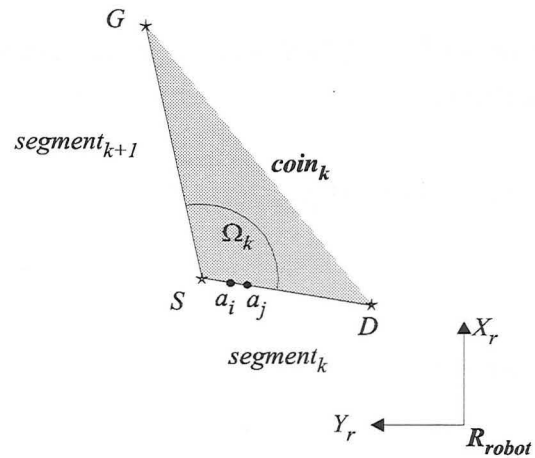
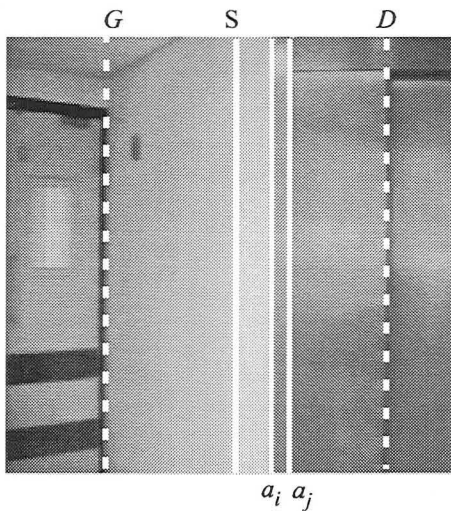


Figure 3.15 : Primitive *coin_k*.

Nous déterminons donc pour chaque primitive *coin*, les indices uD et uG des pixels délimitant la fenêtre englobant les plans droit et gauche. La projection dans le plan de l'image numérisée d'une primitive *coin* est illustrée par la Figure 3.16.

Nous utilisons les équations présentées au chapitre 2 afin de calculer l'angle d'azimut α et le couple de consigne C_i «*focale_i, focus_i*» en fonction des coordonnées du sommet S de la primitive *coin*. Puis, nous déterminons en fonction du couple C_i réglé, les valeurs extrêmes des paramètres du modèle sténopé étendu compte tenu des erreurs de modélisation :

$$\begin{cases} f^{\min} = f - \delta f \\ f^{\max} = f + \delta f \end{cases} \quad \begin{cases} d_{ip}^{\min} = d_{ip} - \delta d_{ip} \\ d_{ip}^{\max} = d_{ip} + \delta d_{ip} \end{cases} \quad \begin{cases} d_x^{\min} = d_x - \delta d_x \\ d_x^{\max} = d_x + \delta d_x \end{cases} \quad \begin{cases} u_o \\ v_o \end{cases} \quad (3.17)$$

Nous calculons alors les indices u_D^{pt} et u_G^{pt} de la projection des points extrémités D et G dans le plan de l'image numérisée.

$$u_D^{pt} = u_o + f^{\min} \times \frac{x D_c}{(z D_c - f^{\min} - d_{ip}^{\min}) \times d_x^{\max}} \quad (3.18)$$

$$u_G^{pt} = u_o + f^{\min} \times \frac{x G_c}{(z G_c - f^{\min} - d_{ip}^{\min}) \times d_x^{\max}}$$

La table d'indexation $Indices_i$ correspondant au couple de consigne C_i réglé permet de déterminer les indices maximums u_D^{lim} et u_G^{lim} des pixels pouvant être projetés avec une incertitude inférieure au seuil fixé, en fonction des angles φ_{droit} et φ_{gauche} formés par les plans droit et gauche avec le plan image et des erreurs de modélisation ε_{planD_k} et ε_{planG_k} .

Les indices uD et uG des pixels délimitant la fenêtre d'intérêt de l'image sont alors donnés par les relations :

$$uD = \text{minimum}(u_D^{pt}, u_D^{lim}) \quad uG = \text{maximum}(u_G^{pt}, u_G^{lim}) \quad (3.19)$$

Chaque primitive $coin_k$ est donc caractérisée par :

$$coin_k = \{(D_k, S_k, G_k), (\varepsilon_{planD_k}, \varepsilon_{planG_k}), (uD_k, uG_k), (planD_k, planG_k), \Omega_k\} \quad (3.20)$$

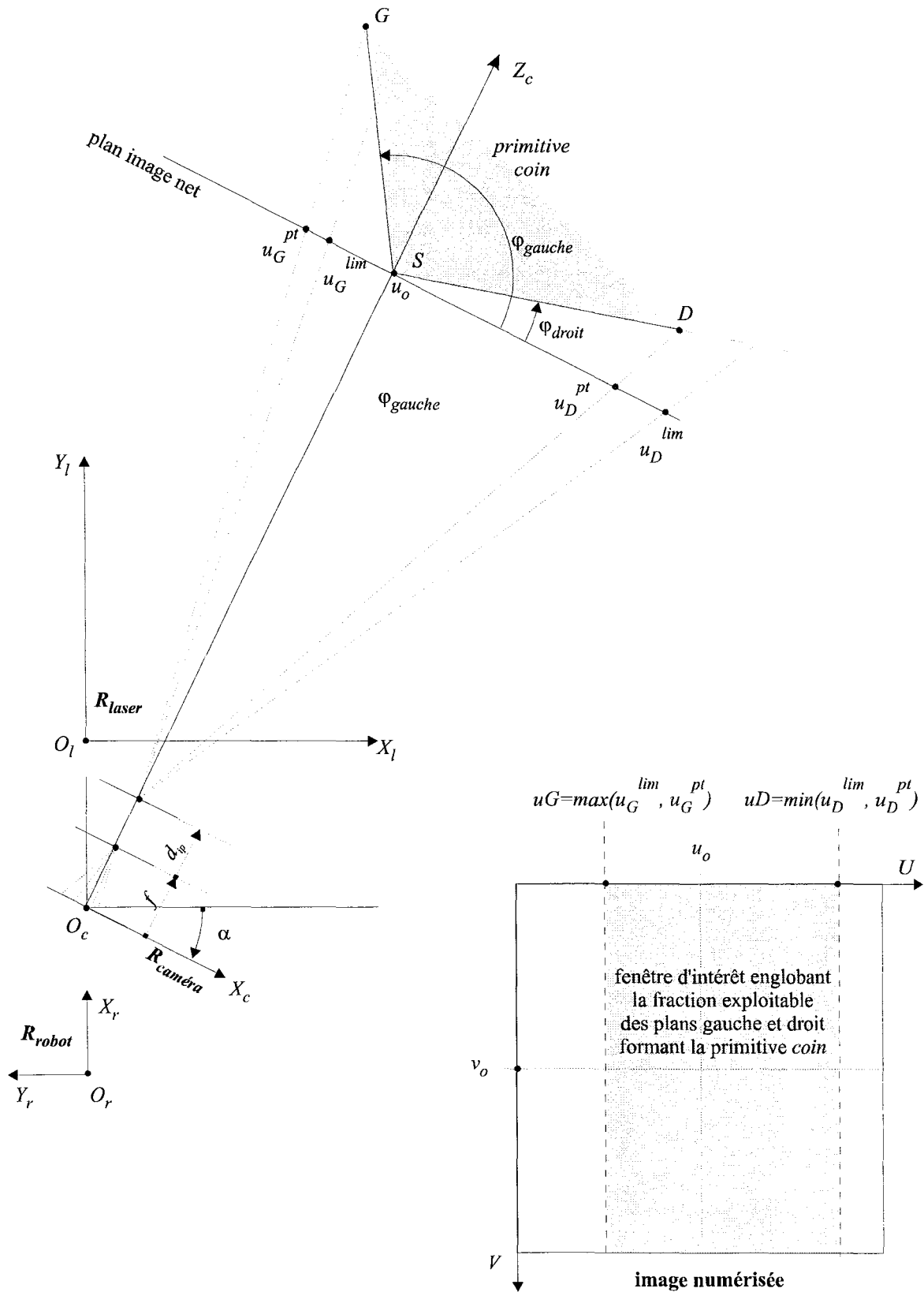


Figure 3.16 : Projection dans le plan de l'image numérisée d'une primitive coin.

3.3.3 Sélection des structures potentielles

Le module de vision est donc amené à réaliser l'inspection des primitives coins afin d'en extraire des amers environnementaux de type arête verticale.

Or, l'analyse de l'ensemble des primitives *coin* nécessiterait un temps de calcul prohibitif. De plus, des objets non référencés dans le modèle global de l'environnement peuvent être présents dans l'espace de manœuvre du robot. Certaines primitives *coin* mises en évidence dans le modèle local télémétrique ne correspondent donc pas à des amers de l'environnement.

Nous définissons par conséquent un critère d'intérêt permettant de sélectionner parmi l'ensemble des primitives *coin* celles qui modélisent des structures susceptibles de contenir des amers répertoriés dans le modèle global de l'environnement.

Ce critère d'intérêt est défini à partir de trois coefficients de pondération, ou poids de base, portant chacun sur un ou plusieurs paramètres décrivant la primitive *coin* :

- **$Poids_{type}$** , critère prenant en compte le statut réel ou virtuel des plans formant la primitive *coin*.

Trois types de primitives *coin* peuvent être définis :

Type (a) : la primitive *coin* est constituée de deux plans réels. Le sommet S correspond donc à l'intersection de deux plans verticaux. Ce type de coin est le plus intéressant car la mesure de l'angle Ω est pertinente et les deux plans verticaux sont susceptibles de contenir des amers : $Poids_{type} = 1$.

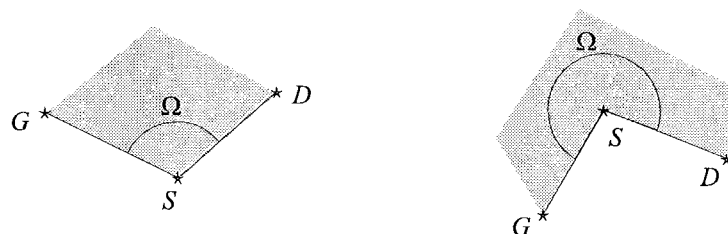


Figure 3.17 : Primitive coin de type (a).

Type (b) : la primitive *coin* est constituée d'un plan réel et d'un plan virtuel dont l'angle Ω est inférieur à 180° . Le sommet S correspond donc à une discontinuité de l'environnement. Ce type de coin est intéressant car il fait apparaître un plan vertical pouvant contenir des amers, mais la mesure de Ω n'est pas pertinente car elle dépend de

l'angle sous lequel le plan réel est regardé. Ω correspond donc à l'angle maximal que peut prendre la structure : $Poids_{type} = 0,5$.

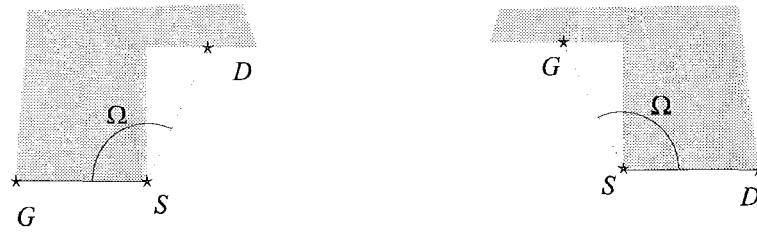


Figure 3.18 : Primitive coin de type (b).

Type (c) : la primitive *coin* est constituée d'un plan réel et d'un plan virtuel dont l'angle Ω est supérieur à 180° . Le sommet S est donc causé par la projection du segment virtuel sur le segment réel et ne correspond à aucune arête réelle de l'environnement. Ce type de coin ne présente donc que peu d'intérêt pour la vision : $Poids_{type} = 0$.



Figure 3.19 : Primitive coin de type (c).

- **$Poids_{fen\hat{e}tres}$** , critère prenant en compte la taille exploitable δu de la primitive *coin* dans le plan image.

Les amers étant répartis de façon relativement homogène au sein de l'environnement, les primitives *coin* occupant le plus de surface dans l'image doivent être favorisées puisqu'elles définissent de larges fenêtres d'analyse susceptibles de contenir des amers. Pour un réglage donné du module de vision, nous optimisons ainsi le temps de traitement des informations sensorielles.

$$\delta u = uD - uG \quad (3.21)$$

Avec : $\left| \begin{array}{l} uD \text{ et } uG, \text{ indices des pixels délimitant la fenêtre englobant les plans droit et gauche.} \end{array} \right.$

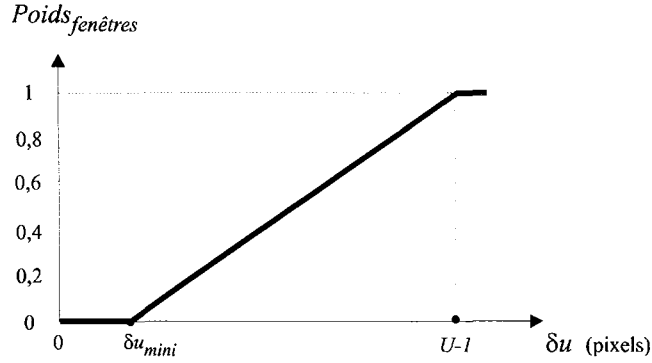


Figure 3.20 : $Poids_{fenêtres}$ lié à la projection de la primitive coin dans le plan image.

- $Poids_{\Omega}$, critère lié à l'angle Ω formé par les plans droit et gauche.

Cette fonction permet de favoriser les primitives dont l'angle Ω est compatible avec les angles Ω des amers environnementaux. Les amers modélisés sont caractérisés par des angles de 90° , 130° , 160° , 180° et 270° . Toutefois, leur mesure peut varier de 30° à 300° selon la position relative du robot. Les primitives *coin* dont l'angle n'est pas compris dans cette plage doivent donc être rejetées.

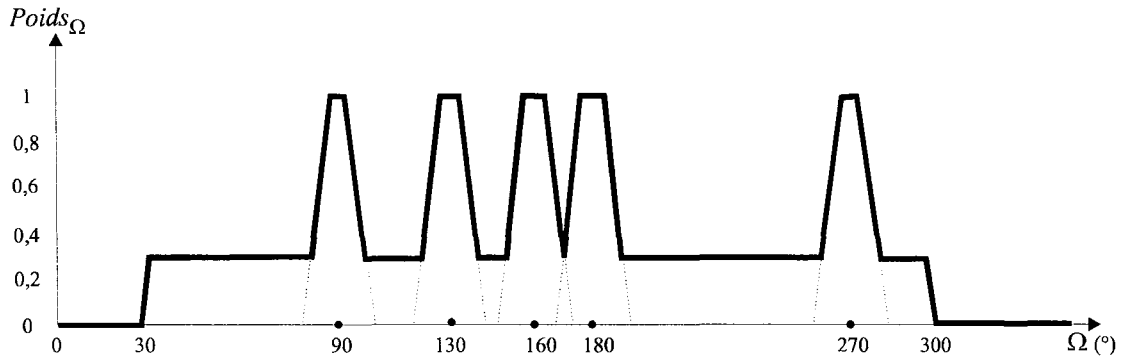


Figure 3.21 : $Poids_{\Omega}$ lié à l'angle Ω .

Pour chaque primitive *coin*, nous définissons alors un critère d'intérêt $Poids_{global}$ représentatif de l'intérêt de la structure modélisée dans la perspective d'une analyse par le module de vision. Si l'un des trois poids de base est nul, alors le $Poids_{global}$ de la primitive *coin* est nul également, sinon :

$$Poids_{global} = \frac{Poids_{type} + Poids_{fenêtre} + Poids_{\Omega}}{3} \quad (3.22)$$

Les primitives *coin* sont alors classées selon l'ordre décroissant de $Poids_{global}$.

La Figure 3.22 illustre un exemple de classement des primitives *coin* les plus utiles tirées du modèle télémétrique déjà présenté. Afin de ne pas alourdir la figure, nous n'avons représenté que les 5 premières primitives classées. Nous obtenons un résultat conforme à nos objectifs, à savoir mettre en évidence les structures de l'environnement susceptibles de contenir des amers environnementaux.

Cette sélection des primitives *coin* ne permet toutefois pas une exploitation directe des points extrémité en tant qu'amer. En effet, certaines primitives *coin* créées peuvent correspondre à des objets non référencés dans le modèle global. Ainsi, dans l'exemple donné, les primitives *coin*₈ et *coin*₁₅ correspondent respectivement à une caisse et à un pan de mur non référencés.

La mise en correspondance des points extrémités des primitives *coin* retenues, sans information de localisation a priori précise, aurait donc de fortes chances d'échouer vu le nombre important de *faux positifs*. Nous proposons par conséquent une inspection de ces primitives par le module de vision afin d'en extraire des amers pertinents.

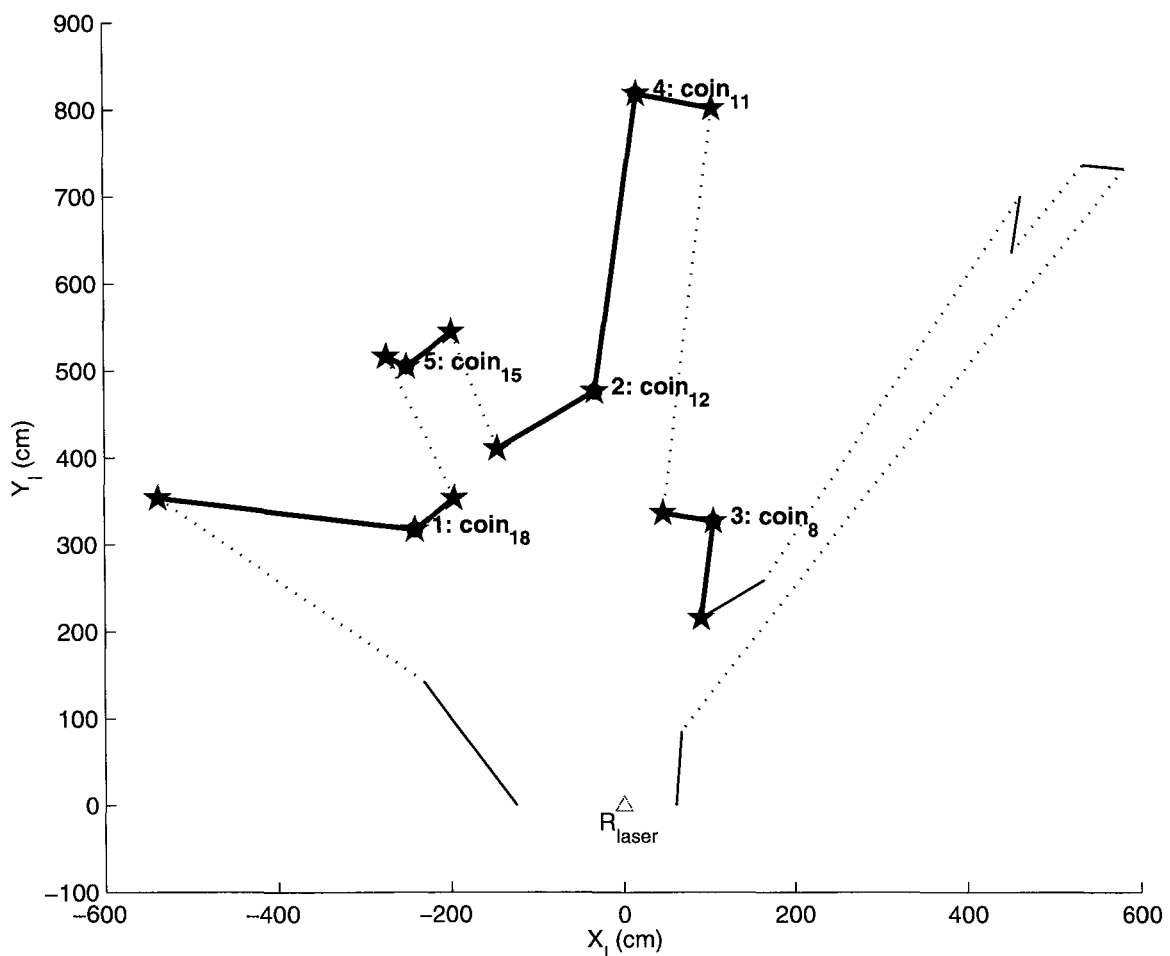


Figure 3.22 : Primitives coins classées pour l'analyse par la vision.

3.4 EXTRACTION DES AMERS

Il s'agit d'inspecter les primitives *coin* retenues comme caractéristiques potentielles à l'aide du module de vision afin de générer des primitives $a_m, m=1...M$ de type *arête verticale* suffisamment fiables pour être utilisées pour déterminer la localisation du robot mobile.

3.4.1 Stratégie de recherche d'amers

Les primitives *coin* du modèle télémétrique sont inspectées successivement par le module de vision selon l'ordre décroissant du critère d'intérêt $Poids_{global}$.

L'analyse de l'ensemble des primitives *coin* conduirait à des temps de calcul prohibitifs. Cette phase d'inspection est donc menée jusqu'à ce qu'un nombre minimum M d'arêtes verticales soient extraites (la valeur du seuil M dépend de la méthode de mise en correspondance et sera discutée au chapitre 4).

L'organigramme de l'inspection d'une primitive *coin* est illustré par la Figure 3.23. Nous en détaillons ci-après les différentes phases.

Une image centrée sur le sommet S de la primitive *coin* inspectée est acquise (les paramètres de focale et de focus étant réglés afin d'optimiser la projection des plans droit et gauche dans l'image numérisée). L'extraction des primitives vision consiste alors en une recherche de segments verticaux par analyse des gradients de luminosité.

L'analyse complète de la fenêtre englobant la primitive *coin* ne se justifie que dans la certitude d'avoir réellement une structure référencée de l'environnement. Or, nous avons fait l'hypothèse de la présence d'objets non répertoriés dans le modèle global de l'environnement.

Aussi, afin de limiter les temps de traitement de l'information vision, nous définissons d'abord une fenêtre d'intérêt de faible largeur, mais de hauteur maximale, correspondant au sommet S de la primitive *coin*.

- Si une arête verticale valide en est extraite, alors nous considérons que la primitive *coin* correspond à une structure référencée ; l'analyse des informations vision se poursuit ensuite dans les deux fenêtres d'intérêt correspondant aux plans droit et gauche.

- Dans le cas contraire, nous considérons avoir affaire à un objet non référencé ; l'analyse des plans droit et gauche ne se justifie donc pas et une nouvelle primitive *coin* est analysée.

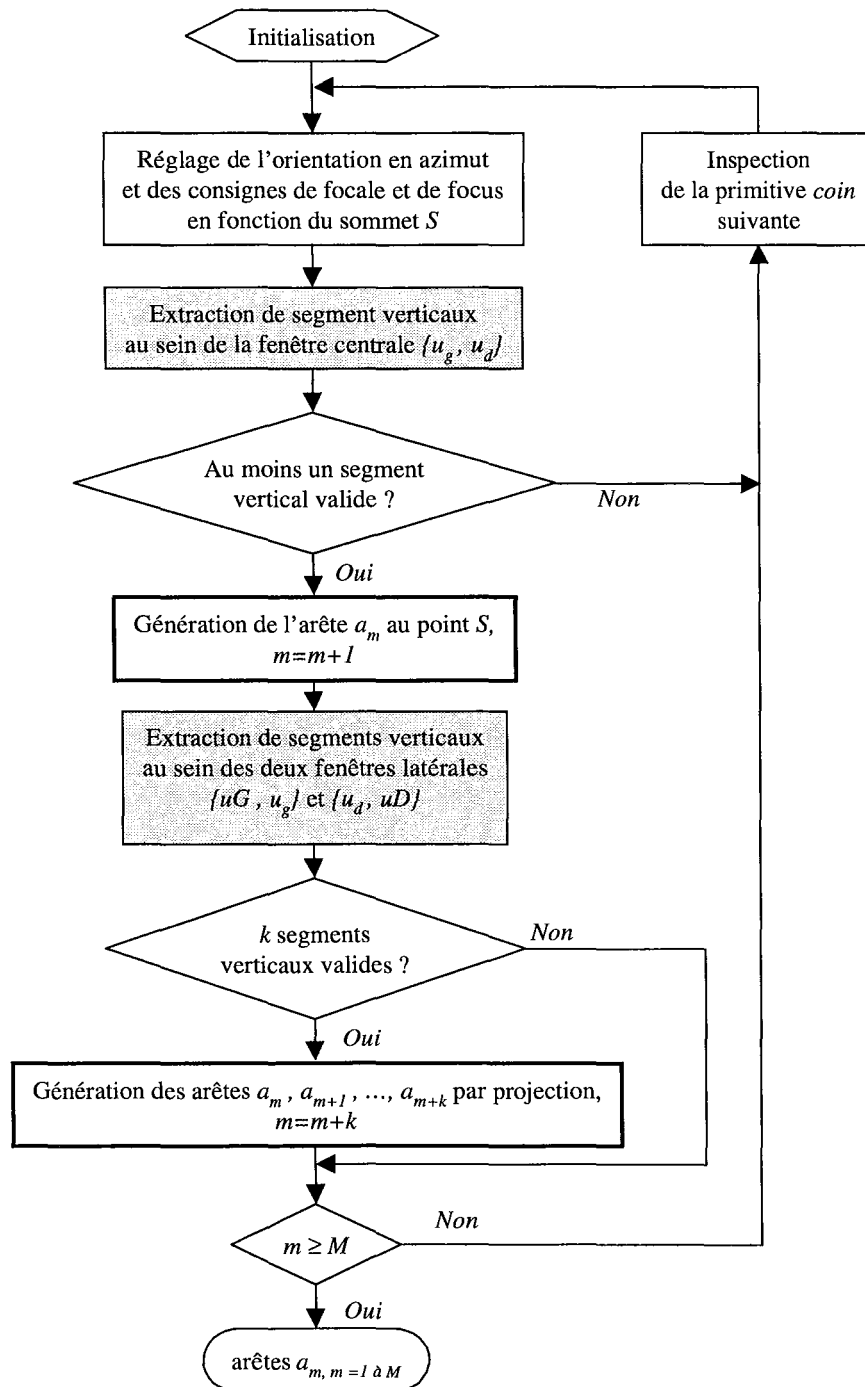


Figure 3.23 : Organigramme de l'inspection d'une primitive coin par la vision.

3.4.1.1 Réglage des paramètres du module de vision

Nous utilisons les équations présentées au chapitre 2 afin de calculer l'angle d'azimut α et le couple de consigne C_i «*focale_i, focus_i*» en fonction des coordonnées du sommet S de la primitive *coin*.

Puis, nous déterminons en fonction du couple C_i réglé, les valeurs extrêmes des paramètres du modèle sténopé étendu en tenant compte des erreurs de modélisation.

$$\begin{cases} f^{\min} = f - \delta f \\ f^{\max} = f + \delta f \end{cases} \quad \begin{cases} d_{ip}^{\min} = d_{ip} - \delta d_{ip} \\ d_{ip}^{\max} = d_{ip} + \delta d_{ip} \end{cases} \quad \begin{cases} d_x^{\min} = d_x - \delta d_x \\ d_x^{\max} = d_x + \delta d_x \end{cases} \quad \begin{cases} u_o \\ v_o \end{cases}$$

3.4.1.2 Analyse de la fenêtre centrale

Une fenêtre centrale correspondant au sommet S de la primitive *coin* est tout d'abord définie. Elle est centrée sur l'axe optique vertical et doit couvrir un secteur angulaire englobant le sommet S malgré les erreurs de segmentation des données laser.

Nous déterminons pour cela les droites $\Delta plan D^{intérieure}$ et $\Delta plan D^{extérieure}$ englobant le plan droit défini par les points D et S , et caractérisé par l'erreur de modélisation $\epsilon plan D$, respectivement $\Delta plan G^{intérieure}$ et $\Delta plan G^{extérieure}$ pour le plan gauche (Figure 3.24).

Soit s_1, s_2, s_3 et s_4 les points correspondant aux intersections des droites $\Delta plan D^{intérieure}$, $\Delta plan D^{extérieure}$, $\Delta plan G^{intérieure}$ et $\Delta plan G^{extérieure}$. Ces quatre points définissent une région d'incertitude contenant le sommet S , compte tenu des erreurs de modélisation des primitives *segment*.

Soit (x_c, z_c) les coordonnées exprimées dans le repère $R_{caméra}$ du point $s_{i,i=1...4}$ le plus éloigné de l'axe optique Z_c . Nous déterminons alors la projection maximale u_s de la cote x_c de ce point dans le plan de l'image numérisée :

$$u_s = x_c \frac{f^{\min}}{d_x^{\max} (z_c - d_{ip}^{\min} - f^{\min})} \quad (3.23)$$

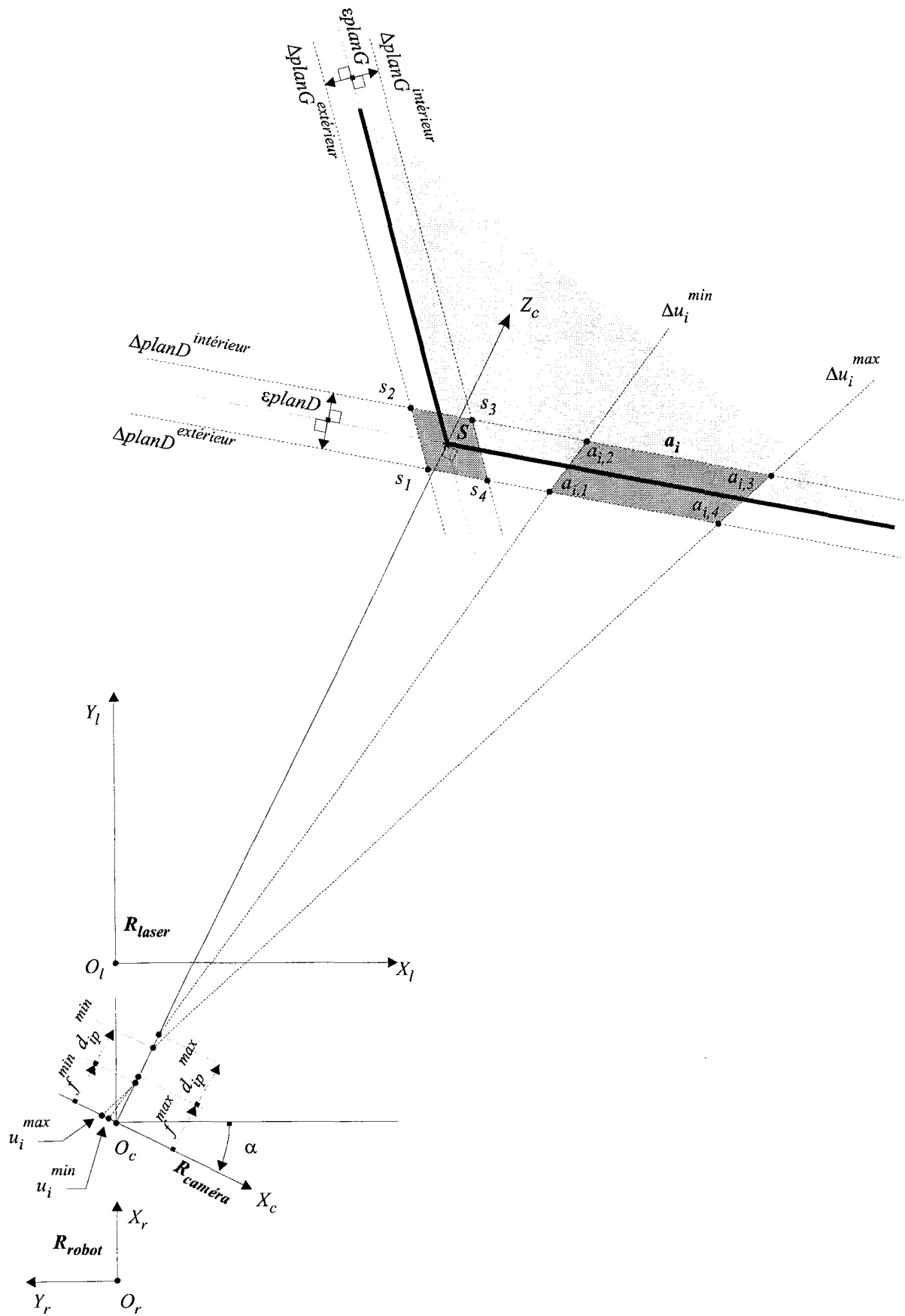


Figure 3.24 : Projection des primitives vision dans le plan (X_r, Y_r) du repère R_{robot} .

La largeur minimale de la fenêtre d'intérêt est fixée par la largeur u_{filtre} du filtre réalisant l'extraction des contours (paragraphe 3.4.2). Dans ces conditions, les limites u_g et u_d de la fenêtre centrale dans l'image numérisée sont données par :

$$\begin{cases} u_g = u_o - \max(u_s, u_{filtre}) \\ u_d = u_o + \max(u_s, u_{filtre}) \end{cases} \quad (3.24)$$

Une recherche de segments verticaux au sein de la fenêtre centrale $\{u_g, u_d\}$ permet alors :

- si au moins un segment extrait satisfait à des critères de longueur et d'orientation, de valider l'hypothèse de la présence d'une arête verticale a pouvant correspondre à un amer environnemental.

Le sommet S engendre alors une arête verticale a caractérisée par :

$$a : \{(s_1, s_2, s_3, s_4), plan\} \quad (3.25)$$

Avec :

$\left \begin{array}{l} s_1, s_2, s_3 \text{ et } s_4, \text{ les points définissant la région d'incertitude contenant le sommet } S, \\ plan, \text{ indices des plans gauche et droit de la primitive coin.} \end{array} \right.$
--

- dans le cas contraire, de valider la présence d'un objet non répertorié en tant qu'amer. Les plans gauche et droit ne doivent donc pas être inspectés et une nouvelle primitive *coin* est analysée.

La Figure 3.25 illustre ainsi l'analyse de la primitive *coin₈* du modèle local télémétrique A présenté Figure 3.22. Aucun segment vertical de longueur suffisante n'ayant pu être extrait de la fenêtre centrale représentée par deux traits pointillés blancs, la primitive *coin₈* a été validée comme étant causée par un objet non répertorié dans le modèle global de l'environnement. Elle ne génère donc aucune arête verticale et la primitive *coin* suivante est inspectée (Les zones hachurées ne sont donc pas analysées).

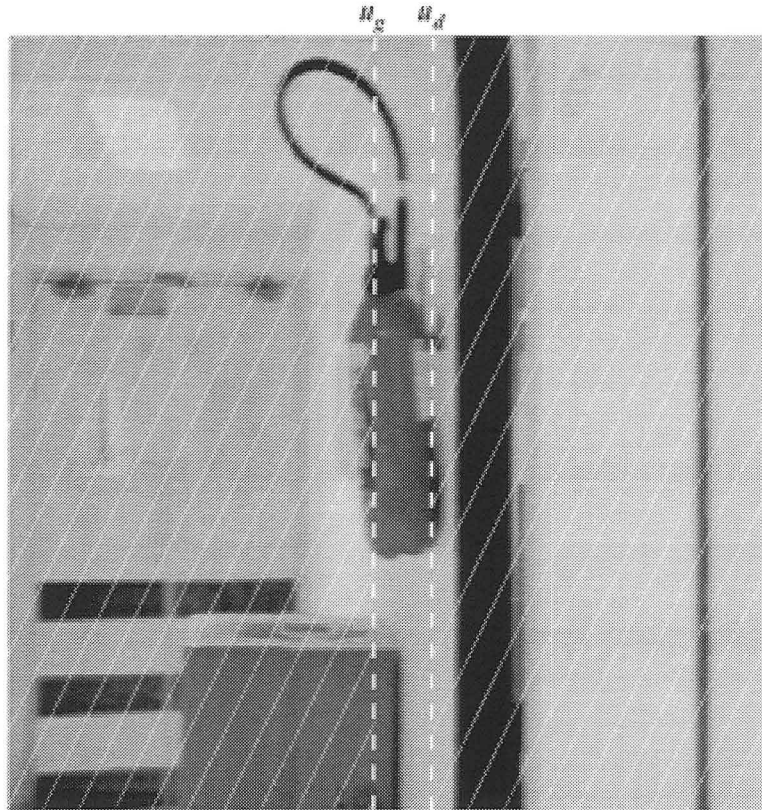


Figure 3.25 : Inspection d'une primitive coin causée par un objet non référencé (une caisse).

3.4.1.3 Analyse des fenêtres latérales

Si le sommet S de la primitive *coin* est validé comme arête verticale, alors nous recherchons de nouvelles arêtes verticales non perceptibles par télémétrie (ex : encadrements de portes, panneaux, ...) au sein des fenêtres d'intérêt latérales $\{uG, u_g\}$ et $\{u_d, uD\}$, englobant les plans gauche et droit.

Tout segment vertical extrait de l'image numérisée est caractérisé par l'indice colonne moyen u_i , ainsi que l'écart type δu_i . Nous déterminons alors les droites $\Delta u_i^{\min}$ et $\Delta u_i^{\max}$ correspondant aux lignes de visée extrêmes de la projection du pixel u_i dans le plan (X_r, Y_r) du repère R_{robot} (Figure 3.24) :

L'équation de la droite $\Delta u_i^{\min}$ est calculée à partir des paramètres $(u_i^{\min}, d_x^{\min}, f^{\max}, d_{ip}^{\max})$.
 L'équation de la droite $\Delta u_i^{\max}$ est calculée à partir des paramètres $(u_i^{\max}, d_x^{\max}, f^{\min}, d_{ip}^{\min})$.

Soit $a_{i,1}$, $a_{i,2}$, $a_{i,3}$ et $a_{i,4}$ les points intersections des droites $\Delta plan^{\text{intérieur}}$, $\Delta plan^{\text{extérieur}}$, $\Delta u_i^{\min}$ et $\Delta u_i^{\max}$. Ces quatre points définissent une région d'incertitude contenant l'arête a_i compte tenu

des erreurs liées à la modélisation des paramètres du module de vision, des primitives segments et à l'extraction du segment vertical.

Chaque arête verticale a_i est donc caractérisée par :

$$a_i : \{(a_{i,1}, a_{i,2}, a_{i,3}, a_{i,4}), plan_i\} \quad (3.26)$$

Avec : $a_{i,1}$, $a_{i,2}$, $a_{i,3}$ et $a_{i,4}$, les points définissant la région d'incertitude contenant l'arête a_i ,
 $plan_i$, indice du plan sur lequel a été projeté l'arête.

La Figure 3.26 illustre ainsi le résultat de l'analyse de la primitive $coin_{12}$ du modèle local télémétrique A présenté Figure 3.22. Un segment vertical de longueur suffisante ayant été extrait de la fenêtre centrale représentée par deux traits pointillés blancs, l'hypothèse de la présence d'une arête verticale a_i correspondant au sommet S_{12} a été validée. De nouveaux segments verticaux ont été alors recherchés dans les fenêtres d'intérêt latérales délimitées par deux traits discontinus blanc. Une nouvelle arête verticale a_j non perçue par le télémètre a été découverte (La zone hachurée n'a pas été analysée).

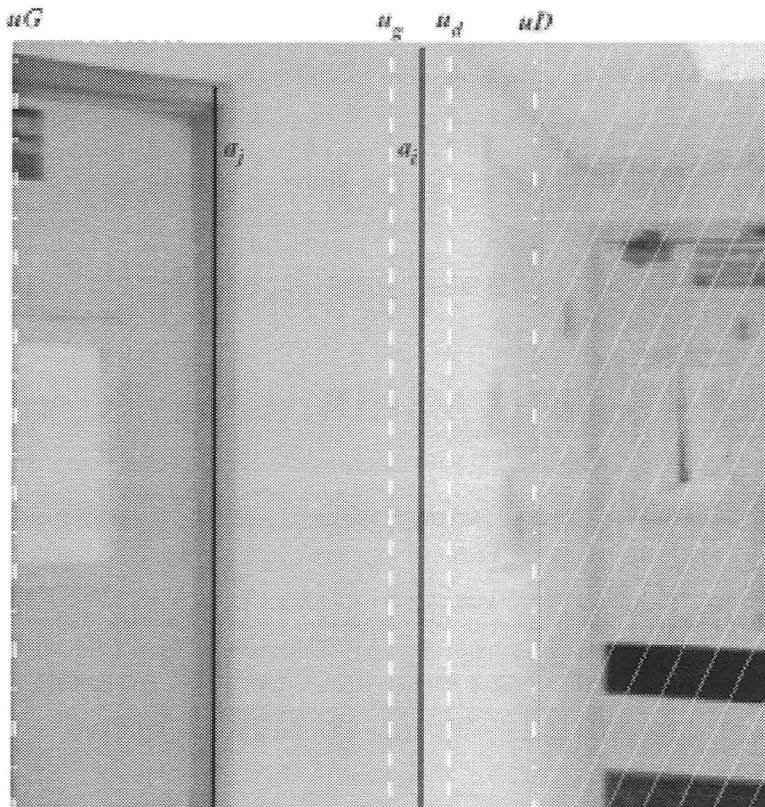


Figure 3.26 : Inspection d'une primitive coin modélisant un angle de mur référencé.

3.4.1.4 Discussion de la stratégie de perception mise en œuvre

Nous avons réalisé l'acquisition d'un modèle local de l'environnement pour 50 localisations différentes du robot. Le nombre de primitives *coin* inspectées a été limité à 6 afin de s'assurer d'un temps de traitement compatible avec la tâche de localisation.

De chaque modèle local, nous obtenons les informations suivantes :

- le nombre d'arêtes verticales extraites lors de l'inspection de chaque primitive *coin*,
- le nombre de *faux positifs* et de *vrais négatifs* correspondant à l'analyse du sommet des primitives *coin*,
- le nombre de *faux positifs* et de *vrais négatifs* correspondant à l'analyse des plans verticaux formant les primitives *coin*.

Le premier intérêt de l'inspection par la vision des primitives télémétriques est l'élimination des primitives *coin* correspondant à des objets non référencés dans le modèle global.

Les résultats expérimentaux permettent de constater que les primitives *coin* causées par des objets de faible hauteur tels que des caisses ou des pièces de mobilier de type bureaux sont rejetées à 94,4%. Il y a génération de faux amers dans le cas d'objets non référencés caractérisés par de longues arêtes telles que l'extrémité d'une porte ouverte, ou un empilement de caisses. Ceci est inévitable et devra être pris en compte lors de l'étape de mise en correspondance (Chapitre 4).

Le rejet d'amers environnementaux (*vrais négatifs*) est également possible du fait du bruit de l'image, de problèmes de reflets ou de manque de contraste. La méthode d'extraction des contours mise en œuvre et présentée au paragraphe 3.4.2 permet toutefois de rendre marginal ce phénomène (5,1% des cas).

Le second intérêt de l'inspection par la vision des primitives télémétriques est la mise en évidence d'arêtes verticales non perçues par la télémétrie.

Les résultats expérimentaux permettent de constater que les arêtes verticales correspondant à des amers environnementaux sont détectées à 81,7%. Les arêtes présentant un faible contraste dans l'image ne sont pas toujours détectées car nous avons choisi des valeurs de seuil

privilégiant la génération d'arêtes fiables. La génération de faux amers environnementaux est ainsi relativement faible (6,4% des cas).

Ces résultats sont conformes à nos besoins puisque la méthode de mise en correspondance est plus facilement perturbée par les *faux positifs* que par les *vrais négatifs*.

Le principal handicap de la méthode réside dans le temps nécessaire à la génération des arêtes observées. Les algorithmes assurant la détection des arêtes verticales de l'environnement local du robot ont tous été implémentés dans le langage interprété *Matlab* afin de simplifier leur mise au point. Leur implémentation en un langage compilé tel que le *C* assurerait un gain de temps considérable. Toutefois, certaines contraintes de temps ne pourront pas être diminuées.

Il s'agit notamment de la phase de réglage des paramètres du module de vision (orientation, focale et focus). La vitesse de rotation de la caméra étant limitée à 26,2° par seconde, l'analyse d'un grand nombre de primitives *coin* peut provoquer une perte de temps relativement importante puisque la disposition relative des primitives n'est pas un paramètre utilisé comme critère d'intérêt (il est toutefois envisageable de réaliser simultanément le traitement d'une image et le réglage des paramètres de la caméra pour l'inspection d'une nouvelle primitive *coin*).

La Figure 3.27 illustre le taux d'observation de M arêtes verticales selon le nombre N de primitives *coin* ayant été inspectées.

L'inspection de 2 primitives *coin* permet dans 93,9% des cas de générer 3 arêtes. Il s'agit là du nombre minimum d'arêtes permettant d'initialiser la procédure de mise en correspondance, dans le cas où les configurations possibles d'amers sont toutes distinctes les unes des autres. Or, il est délicat de satisfaire cette condition. De plus, comme nous l'avons vu précédemment, certaines arêtes verticales extraites par la vision peuvent ne pas correspondre à un amer. La présence d'un *faux positif* débouche alors soit sur l'impossibilité de déterminer des appariements (pas de localisation possible), soit sur des appariements erronés si la configuration relative des arêtes est proche de la configuration de certains amers (localisation faussée). Il est donc nécessaire de disposer d'un nombre supérieur d'arêtes afin de s'assurer de la fiabilité des mises en correspondances.

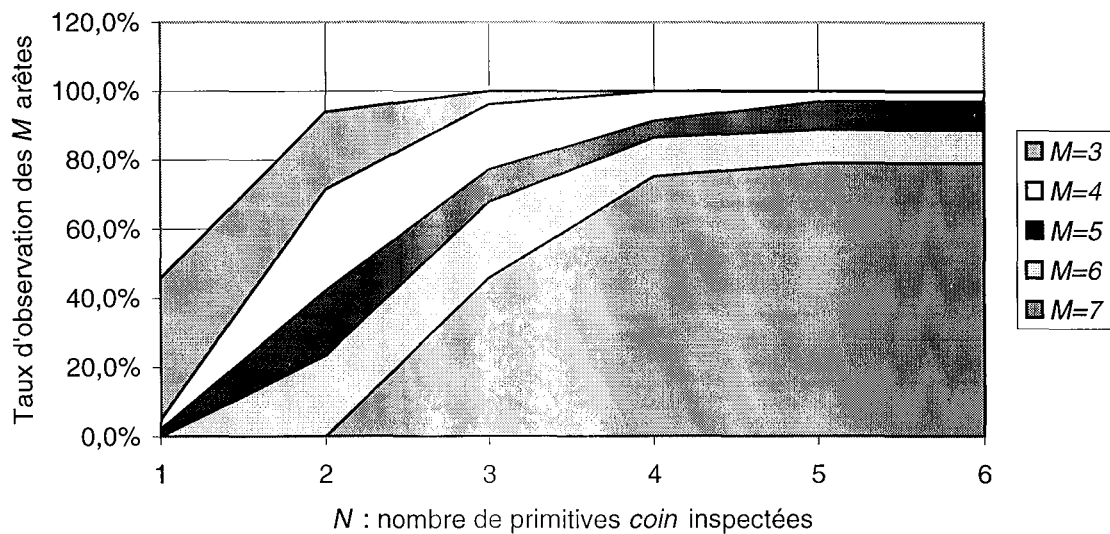


Figure 3.27 : Taux d'observation de M d'arêtes verticales en fonction des N primitives *coin* inspectées.

L'étude du graphique montre que l'inspection des 6 premières primitives *coin* classées ne garantit pas la génération d'un nombre très élevé d'arêtes verticales. Ainsi, bien que chaque primitive *coin* validée engendre en moyenne 2,79 arêtes verticales, nous n'obtenons une détection de 8 arêtes que dans 68,9% des cas. Ceci est dû à la présence de primitives *coin* causées par des objets non référencés dont l'étude n'apporte aucune information supplémentaire quand à l'existence d'arêtes verticales.

La méthode de mise en correspondance utilisée devra donc être capable de réaliser des appariements fiables à partir d'un nombre limité d'arêtes verticales observées. Nous pouvons constater que l'inspection de 3 primitives *coin* est suffisante pour obtenir 5 arêtes verticales dans 77,3% des cas (ce taux passe à 91,3% si 4 primitives *coin* sont inspectées).

La Figure 3.28 montre les arêtes verticales extraites du modèle télémétrique local A après l'analyse de quatre primitives *coin*, dont une causée par un objet non référencé. Nous pouvons qualifier l'étape de détection comme fiable puisque ces 6 arêtes verticales correspondent toutes à un amer environnemental.

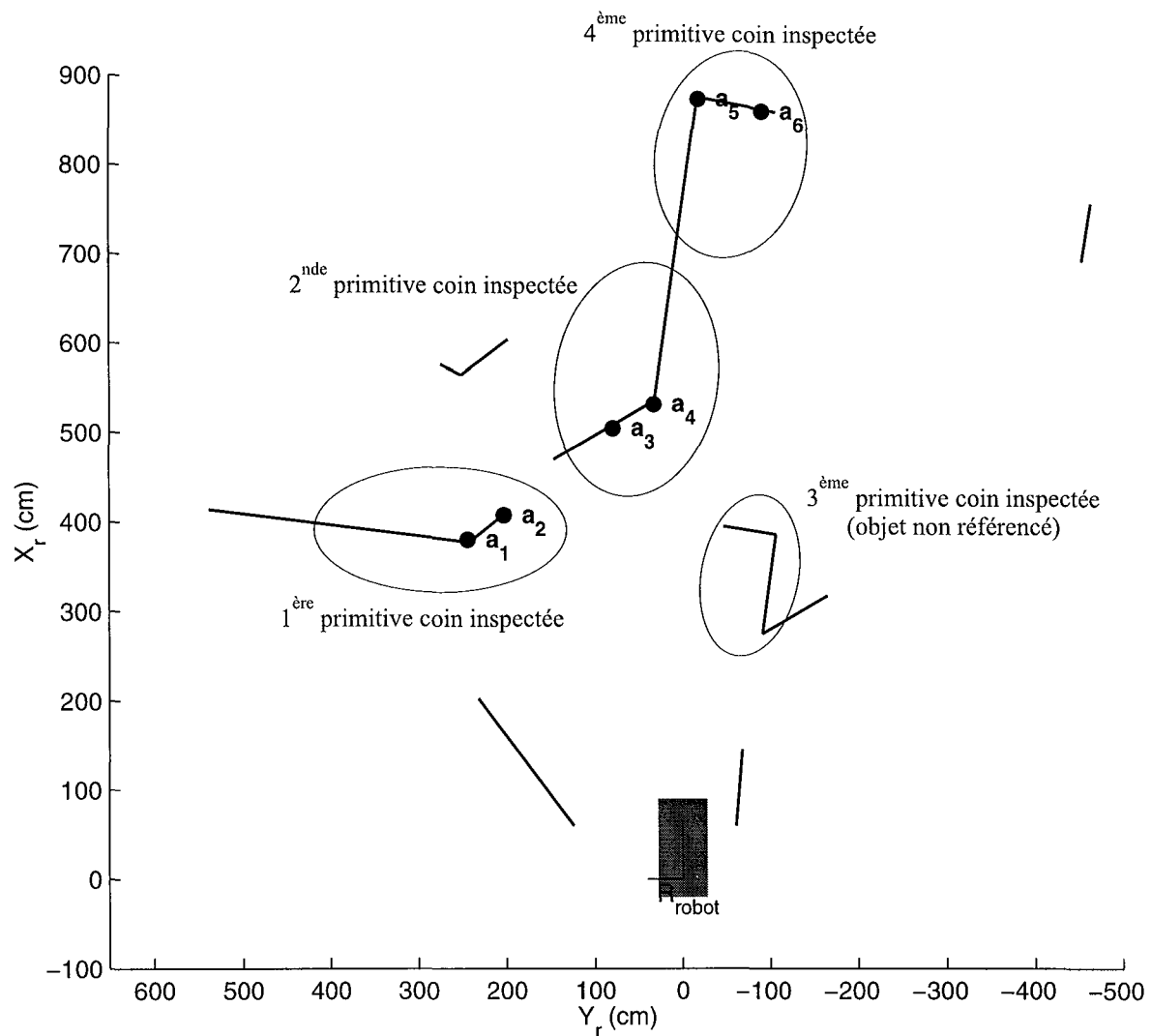


Figure 3.28 : Arêtes verticales $a_i, i=1,\dots,6$ extraites du modèle télémétrique local A.

3.4.2 Extraction des primitives vision

Nous détaillons dans les paragraphes suivants la méthode utilisée afin d'extraire d'une fenêtre d'intérêt de l'image, des primitives vision de type segments verticaux.

3.4.2.1 Choix d'une approche « dérivative »

Dans une image, les variations d'intensité représentent des changements de propriétés physiques ou géométriques de la scène. Elles constituent notamment les frontières de régions correspondant à des bords ou parties d'objets de la scène.

Les contours des amers que nous recherchons se situent ainsi entre les pixels appartenant à des régions ayant des intensités moyennes différentes. Il s'agit de contours de type « marche d'escalier ». Les approches dérivatives sont alors les plus simples pour détecter et localiser les variations d'un signal présentant une transition avec un saut d'amplitude.

La détection des contours peut ainsi être réalisée par :

Des opérateurs dérivatifs. Si l'on considère que la transition du signal est repérée par son point d'inflexion, sa localisation peut se faire par recherche soit de la valeur absolue de la dérivée première, soit du passage par zéro de la dérivée seconde.

De nombreux filtres à réponse impulsionnelle finie permettant leur implémentation par simple masque de convolution, ont été développés. On peut citer comme exemple l'opérateur gradient suivi d'une détection des maxima (opérateurs de Prewitt et de Sobel), ainsi que l'opérateur Laplacien suivi d'une étape de détection des passages par zéro (opérateurs de Marr et Hildreth).

La mise en œuvre de tels filtres est extrêmement aisée car il s'agit de masques de convolution 3 par 3. Toutefois, si ces opérateurs locaux sont rapides, leurs performances se dégradent fortement en présence de bruit.

Un filtrage optimal. La définition du filtre est basée sur un modèle de contour de type échelon perturbé par un bruit blanc Gaussien additif. Il s'agit donc, dans un premier temps, de filtrer l'image afin d'éliminer le bruit, puis de rechercher le gradient de luminosité afin de détecter les contours.

Canny a, le premier, proposé 3 critères (bonne détection, bonne localisation et non multiplicité des réponses) permettant de définir un filtre optimal à réponse impulsionnelle finie dont il a déduit un opérateur détecteur de contours [Canny 86]. Deriche en a alors extrapolé un filtre optimal à réponse impulsionnelle infinie présentant l'avantage de pouvoir être implémenté de façon récursive [Deriche 87]. Shen et Castan ont adopté la même démarche que celle exposée précédemment mais en utilisant un critère qui combine la détection et la localisation [Shen 90].

Ces filtres sont paramétrables en fonction de l'application et leurs performances sont excellentes [Cocquerez 95]. Notre choix s'est porté sur le filtre de Deriche car il est le mieux adapté à nos conditions d'utilisation et son implémentation récursive assure un temps de traitement réduit.

3.4.2.2 Détection des contours verticaux

La recherche des contours verticaux se déroule en trois étapes principales :

- calcul du gradient en chaque point de l'image et extraction des maxima locaux afin d'obtenir des contours fins, d'une épaisseur de 1 pixel,
- seuillage et chaînage des maxima locaux afin d'éliminer les transitions de faible gradient pouvant correspondre à du bruit,
- segmentation des chaînes de maxima.

3.4.2.2.1 Gradient de l'image et recherche des maxima locaux

Nous désirons extraire les structures verticales de l'image $IMG(u, v)$ correspondant à la primitive *coin* étudiée.

Nous utilisons pour cela l'opérateur bidimensionnel séparable à dérivée directionnelle suivant la direction horizontale défini par Deriche [Deriche 87].

Les structures obliques étant assez rares dans un environnement intérieur, leur présence résiduelle dans l'image filtrée ne sera pas gênante. Nous gagnerons ainsi en rapidité de calcul par rapport à un filtrage dans les directions horizontales et verticales.

L'image des contours verticaux et obliques $IMG_{filtrée}(u, v)$ est donc le résultat d'un lissage selon la direction V , suivi d'une dérivation selon la direction U :

$$IMG_{filtrée}(u, v) = (IMG(u, v) * filtre_{lisseur}(v)) * filtre_{dérivateur}(u) \quad (3.27)$$

Le $filtre_{lisseur}$ est l'intégrale du $filtre_{dérivateur}$ optimal défini par Deriche :

$$filtre_{dérivateur}(u) = c \times u \times e^{-\xi|u|} \quad \text{avec} \quad c = -\frac{(1 - e^{-\xi})^2}{e^{-\xi}}$$

$$\text{filtre}_{\text{lisseur}}(v) = b \times (\xi|v| + 1) \times e^{-\xi|v|} \quad \text{avec} \quad b = -\frac{(1 - e^{-\xi})^2}{1 + 2\xi e^{-\xi} - e^{-2\xi}}$$

Les $\text{filtre}_{\text{lisseur}}$ et $\text{filtre}_{\text{dérivateur}}$ possèdent un paramètre de réglage ξ pouvant être ajusté en vue d'obtenir la localisation ou le rapport signal sur bruit optimal souhaité. Le choix de l'échelle du filtre ξ représente donc un compromis dépendant de l'application.

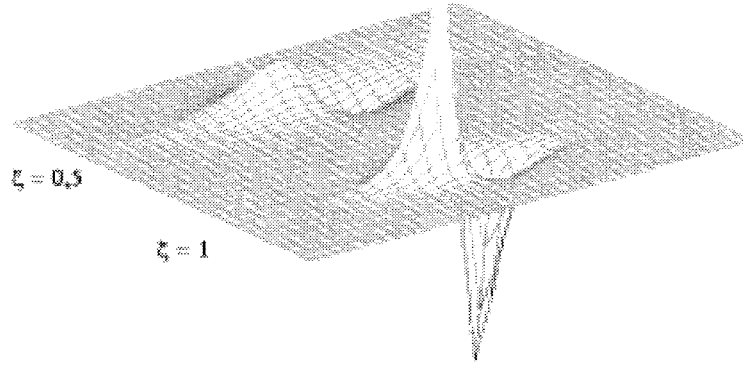


Figure 3. 29 : Impact du paramètre de réglage ξ sur la forme du filtre de Deriche.

Afin d'obtenir des contours d'une épaisseur de 1 pixel, nous définissons un contour comme étant une ligne de crête dans l'image $IMG_{\text{filtrée}}(u, v)$. Tout contour vertical est alors caractérisé par des niveaux de gris toujours positifs, de forts dénivelés le long des lignes V et de faibles dénivelés dans le sens des colonnes U .

Un contour vertical correspond donc aux maxima locaux de l'image $IMG_{\text{filtrée}}(u, v)$ dans le sens des lignes V . Tout $\text{pixel}(u, v)$ est défini comme point de contour si :

$$\begin{cases} \text{pixel}(u, v) \geq \text{pixel}(u - 1, v) \\ \text{pixel}(u, v) > \text{pixel}(u + 1, v) \end{cases} \quad \text{ou} \quad \begin{cases} \text{pixel}(u, v) \geq \text{pixel}(u + 1, v) \\ \text{pixel}(u, v) > \text{pixel}(u - 1, v) \end{cases}$$

La Figure 3.30 illustre le résultat du filtrage d'une image pour différentes valeurs du paramètre ξ . Afin d'améliorer leur visualisation, les maxima locaux ont été binarisés et sont représentés en superposition de l'image initiale. La valeur de seuillage a été choisie afin d'éliminer la majeure partie du bruit de fond de l'image (4 sur une échelle de 0 à 255).

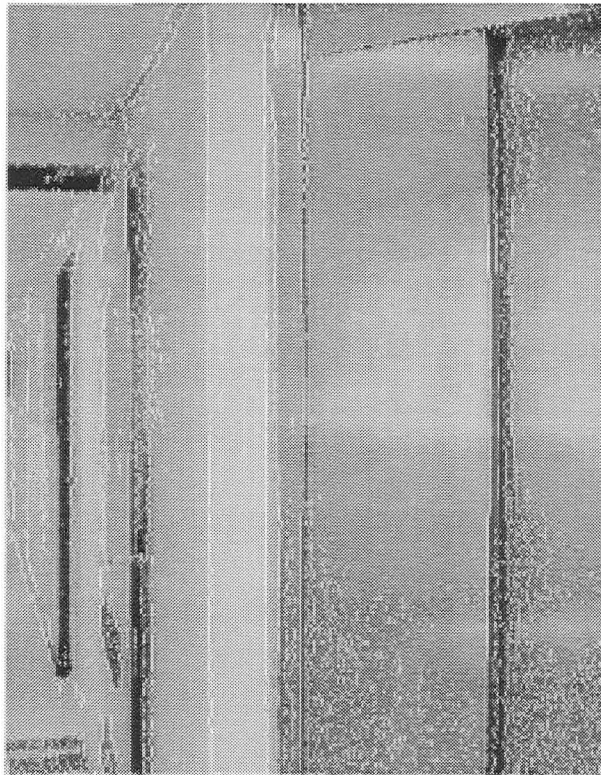
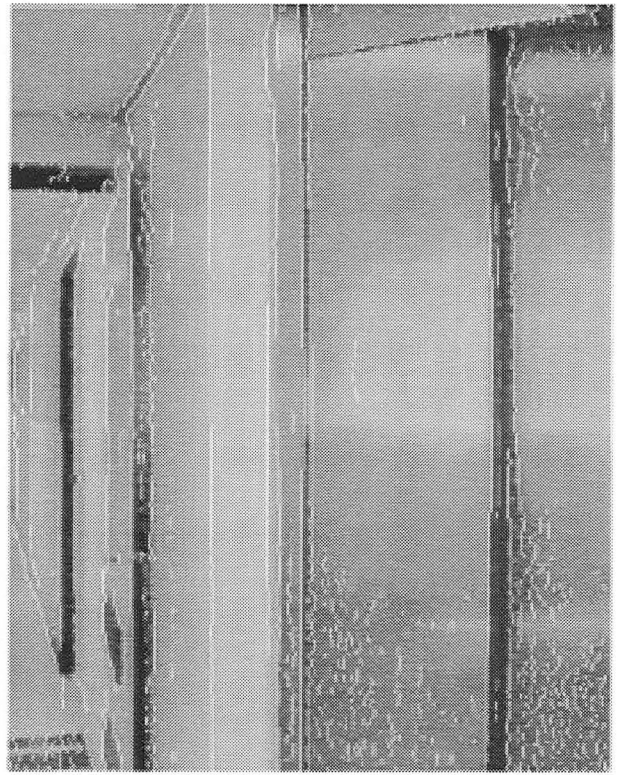
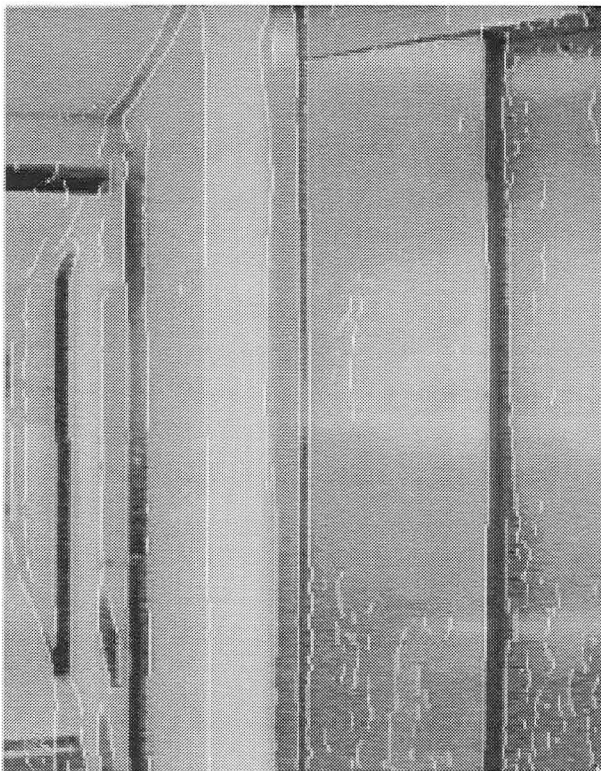
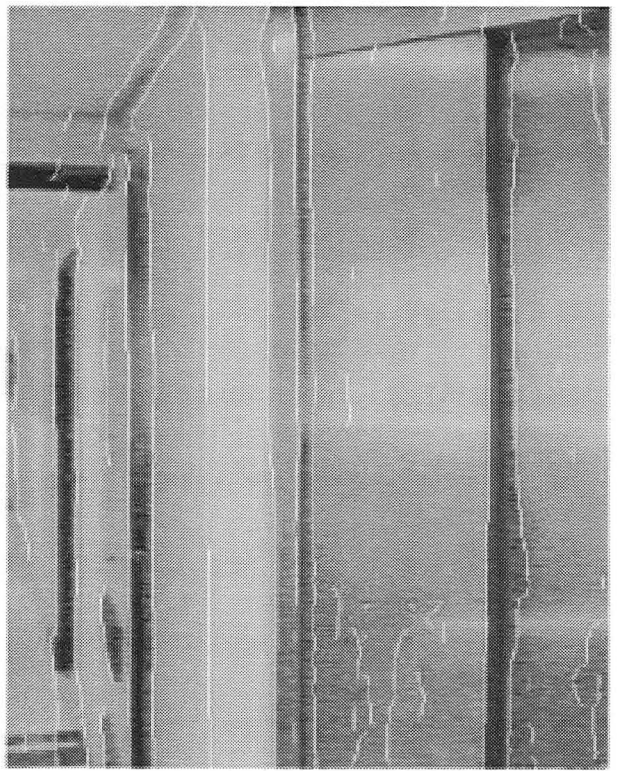
 $\xi=2$  $\xi=1,2$  $\xi=0,8$  $\xi=0,25$

Figure 3.30 : Les maxima locaux correspondant aux contours obtenus pour différentes valeurs du paramètre ξ sont représentés en superposition de l'image initiale.

- lorsque ξ **augmente**, la base du filtre diminue et porte sur un faible voisinage.

L'effet de dérivation est donc très sensible et permet de localiser avec précision la majorité des contours de l'image. Par contre, le bruit de l'image est peu filtré.

Les maxima locaux sont par conséquent très nombreux et présentent de nombreuses discontinuités. L'étape de chaînage sera donc longue car il y a beaucoup de contours à examiner.

- lorsque ξ **diminue**, la base du filtre s'agrandit et porte sur un large voisinage.

L'effet de lissage est donc fort et le rapport signal sur bruit est bon. L'influence du bruit est par conséquent diminuée. Par contre les contours détectés seront plus grossiers et délocalisés.

Les maxima locaux sont donc peu nombreux et présentent peu de discontinuités. L'étape de chaînage sera courte car il y a peu de contours à examiner.

Au vu de l'ensemble des résultats obtenus, une valeur de ξ égale à 0,8 permet d'obtenir la détection d'arêtes de faible contraste, typiques des angles de mur, tout en garantissant une bonne localisation des contours et peu de discontinuités. De plus, le faible nombre de maxima locaux obtenus permet de réaliser l'étape de chaînage en un temps réduit.

3.4.2.2.2 *Seuillage et chaînage des maxima locaux*

Les amers environnementaux apparaissent comme des chaînes verticales dans l'image des maxima locaux.

Les contours obtenus présentent des lacunes et des déformations ayant pour origine la présence de bruit, de reflets ou de différences de contraste au sein même du contour. Il est par conséquent nécessaire d'autoriser d'éventuelles discontinuités du contour dans le processus de chaînage. De plus, un seuillage des maxima locaux est nécessaire afin d'éliminer les transitions de faible amplitude correspondant à du bruit. Il s'agit donc de déterminer un seuil d'amplitude au delà duquel tout pixel sera considéré comme information pertinente et pourra donc être inclus dans la chaîne.

En raison des fluctuations provenant du bruit et apparaissant en sortie de l'opérateur de détection de contours, il y a risque de création de lacunes quand un seuil unique s_{bas} est utilisé sur l'image des maxima locaux pour obtenir les chaînes de contours.

Un seuillage par hystérésis permet de limiter ce défaut en utilisant deux valeurs de seuil s_{haut} et s_{bas} ($s_{haut} > s_{bas}$). Ainsi, les maxima locaux dont la valeur est supérieure à s_{haut} sont conservés ainsi que tous ceux dont les valeurs sont supérieures à s_{bas} et qui appartiennent à une composante connexe comprenant au moins une valeur supérieure à s_{haut} . Le risque de coupure d'un contour est alors réduit.

On se propose de tenir compte d'éventuelles discontinuités du contour en envisageant successivement 9 possibilités de chaînage. Compte tenu des amers recherchés (longues arêtes verticales), nous choisissons d'effectuer une recherche portant sur une zone de faible largeur mais de grande profondeur.

Les pixels de l'image des maxima dont l'amplitude est inférieure à s_{bas} sont tout d'abord mis à zéro. L'image des maxima est ensuite parcourue de haut en bas et de gauche à droite. Tout $maxima(u, v)$ non nul et non chaîné initialise la procédure de chaînage illustrée par la Figure 3.31 et dont l'algorithme est donné ci-après:

```

    Pour  $v_{testé} = v-1$  à  $v-3$ 
    |
    |   Pour  $u_{testé} = u-1$  à  $u+1$ 
    |   |
    |   |   Si  $maxima(u_{testé}, v_{testé}) \neq 0$  et  $maxima(u_{testé}, v_{testé})$  non chaîné
    |   |   |
    |   |   |   Alors chaîner  $maxima(u_{testé}, v_{testé})$  à  $maxima(u, v)$ 
    |   |   |   |
    |   |   |   |   Combler la chaîne si nécessaire
    |   |   |   |   |
    |   |   |   |   |    $maxima(u_{testé}, v_{testé})$  devient le nouveau pixel courant
    |   |   |   fin
    |   fin
    fin

```

La chaîne créée n'est validée que si elle est constituée de plus de N pixels et si elle contient au moins un pixel dont l'amplitude est supérieure à s_{haut} .

Une fois tous les maxima non nuls testés, la procédure de chaînage est arrêtée.

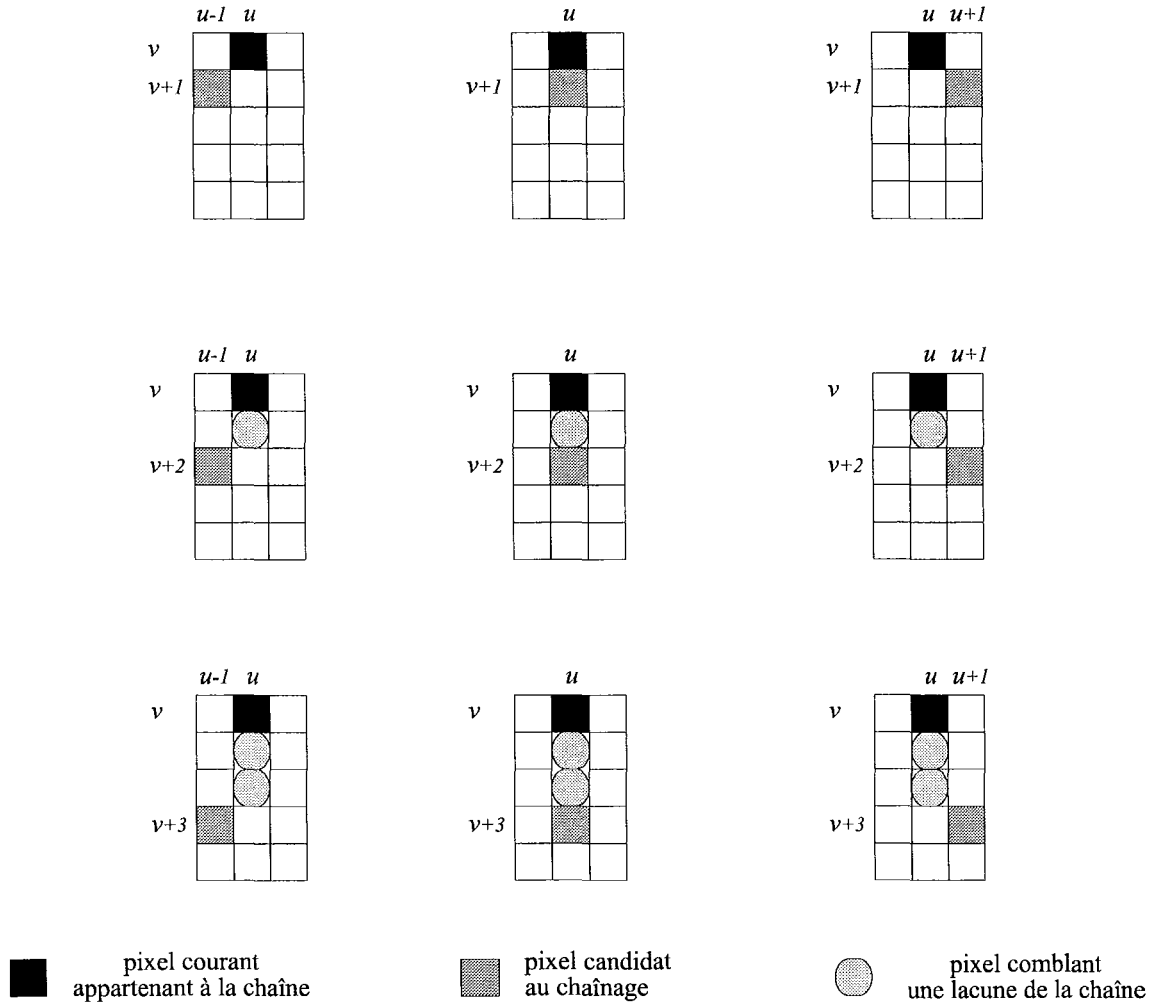


Figure 3.31 : Procédure de chaînage des maxima locaux.

Le réglage du seuil s_{bas} est relativement aisé à déterminer. Ainsi, dans le cadre de notre environnement, un seuil s_{bas} égal à 4, sur une échelle variant de 0 à 255, permet d'éliminer une grande partie du bruit de fond de l'image tout en conservant intactes les chaînes de contours correspondant à des amers.

Le réglage du seuil s_{haut} s'avère quand à lui plus délicat à déterminer et dépend du contraste des amers recherchés :

- Dans le cas de l'inspection du sommet S de la primitive *coin*, l'arête centrale correspond la plupart du temps à un angle de mur dont le contour est caractérisé par un faible contraste. Le choix des paramètres du filtre fait que théoriquement, pour un échelon d'amplitude unitaire, nous avons un maximum unitaire. Toutefois, à cause de la présence de bruit, ce maximum est égal à 0,7. Ceci signifie que pour des arêtes peu contrastées correspondant à

un échelon d'amplitude de 5 à 10 pixels, le maximum correspondant risque d'être proche de la valeur caractérisant le bruit.

Il est donc nécessaire de fixer un s_{haut} égal au s_{bas} afin d'éviter de perdre des informations. Le bruit sera donc présent sous forme de petites chaînes mais la fenêtre étant réduite, le temps de chaînage ne sera pas trop long.

- Dans le cas de la recherche d'amers supplémentaires dans les plans gauche et droit, ceux-ci sont souvent caractérisés par des contrastes moyens à forts (chambranle de porte, ...). Il est donc possible de fixer un seuil s_{haut} plus élevé permettant de prendre en compte la forte amplitude des maxima locaux.

Les résultats expérimentaux nous ont conduit à définir le seuil s_{haut} tel que 20% des pixels de l'image filtrée aient une amplitude supérieure à ce seuil. Les contours correspondants à du bruit sont alors correctement filtrés et le temps de chaînage est réduit. Par contre, les amers caractérisés par un faible contraste sont perdus.

Le nombre N de pixels validant une chaîne est fixé à 30% de la hauteur de l'image afin de tenir compte d'éventuelles occultations de l'amer.

3.4.2.2.3 Segmentation des chaînes

Les chaînes extraites sont ensuite transformées en segments de droite. On détermine pour chaque chaîne valide l'indice colonne moyen u ainsi que l'écart type δu . Afin de ne pas créer de segments imprécis ou correspondants à des contours obliques, nous ne retenons que les chaînes caractérisées par une faible déviation des indices de colonne : $\delta u \leq 2,5$ pixels.

De nombreux amers de référence utilisés pour localiser le robot sont formés par plusieurs arêtes verticales proches les unes des autres : c'est ainsi le cas des chambranles de porte. Afin de simplifier la phase de mise en correspondance, la carte globale de l'environnement ne répertorie que les principales arêtes verticales de l'environnement. Une arête de référence peut donc correspondre à plusieurs arêtes réelles.

Nous avons par conséquent choisi de fusionner les droites caractérisées par un décalage horizontal inférieur à 5 pixels et nous n'avons conservé que les droites d'une longueur supérieure à 60% de la hauteur de l'image.

La Figure 3.33 montre les chaînes obtenues à partir des maxima locaux de la Figure 3.32, pour les valeurs de seuils que nous avons présentées.

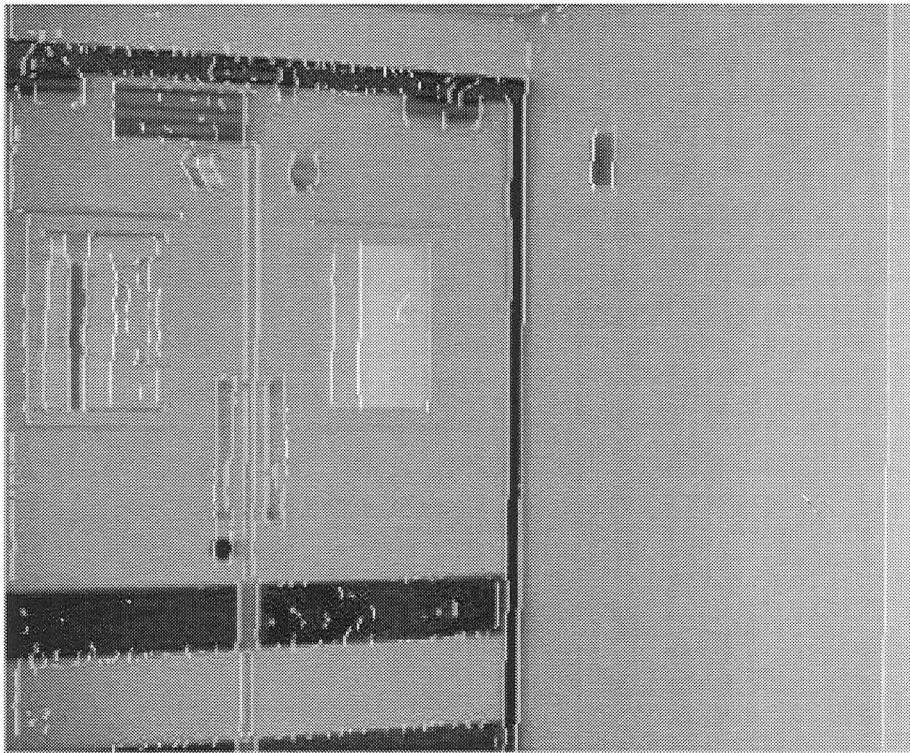


Figure 3.32 : Maxima locaux seuillés par s_{bas} .

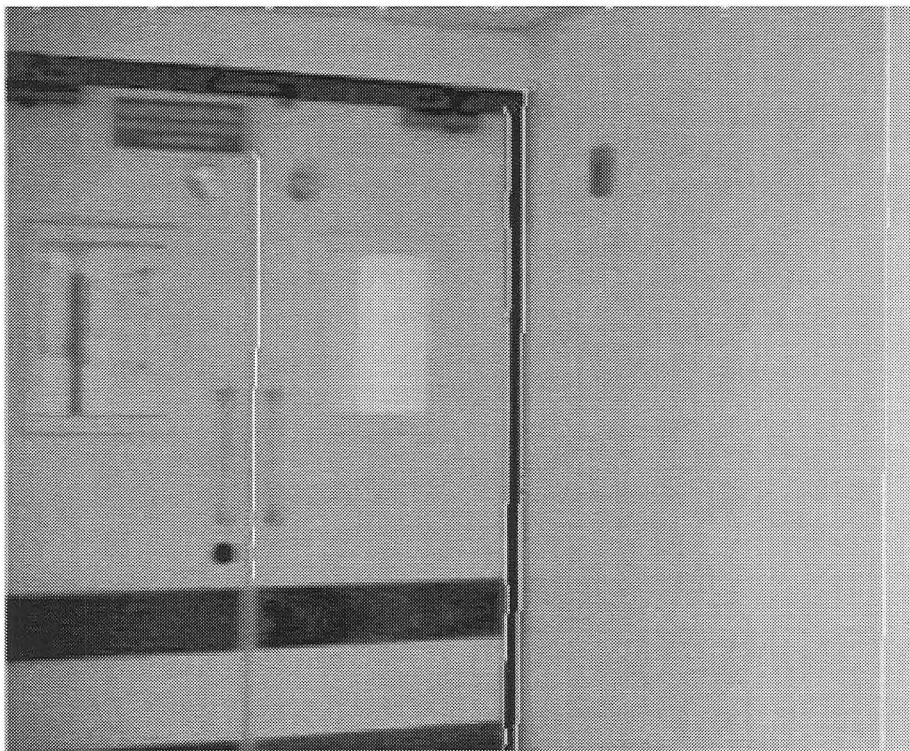


Figure 3.33 : Maxima locaux chaînés après le seuillage s_{haut} .

La Figure 3.34 illustre les segments verticaux créés à partir des chaînes de la Figure 3.33.



Figure 3.34 : Arêtes verticales extraites de l'image numérisée.

Chaque arête verticale extraite de l'image est ainsi caractérisée par l'indice colonne moyen u ainsi que l'écart type δu .

3.5 CONCLUSION

Une représentation cohérente et interprétable de l'environnement local d'un robot mobile est un élément clé de toute tâche de localisation. Afin d'obtenir une description la plus complète possible des amers environnementaux, nous avons défini une stratégie de perception basée sur la confirmation et l'enrichissement des informations télémétriques par les informations vision.

La segmentation du modèle télémétrique de l'environnement local est réalisée en ayant pour objectif de définir des primitives *coin* précises et fiables correspondant exclusivement à des plans verticaux de l'environnement. Afin d'optimiser la phase d'inspection par la vision, nous avons défini un critère d'intérêt visant à mettre en évidence les primitives *coin* susceptibles de contenir des amers environnementaux.

Le module de vision s'avère être un élément précieux puisqu'il permet, d'une part d'éliminer les primitives *coin* correspondant à des objets non référencés dans le modèle global (*faux positifs*), et d'autre part de détecter des arêtes verticales non perceptibles par la télémétrie.

La définition de fenêtres d'intérêt diminue considérablement le temps de traitement d'une image. La limite de ce système de perception est fixée principalement par le temps d'acquisition des images. Nous avons constaté que de 4 à 8 arêtes verticales pouvaient être extraites en un temps compatible avec la tâche de localisation absolue.

Puisque nous nous plaçons dans l'hypothèse où les informations de localisation a priori ne sont pas exploitables, nous avons privilégié la fiabilité des primitives observées par rapport à la quantité.

Nous présentons dans le chapitre suivant la procédure d'appariement des arêtes observées du modèle local et des amers environnementaux du modèle global.

Chapitre 4

Détermination de la localisation

4.1 INTRODUCTION

Soient $A_n, n=1...N$ les amers environnementaux dont les coordonnées sont exprimées dans le système $R_{env.}$ et $a_m, m=1...M$ les arêtes verticales observées dont les coordonnées sont exprimées dans le système R_{robot} . Calculer le vecteur Γ des paramètres (x_e, y_e, Θ_e) de la localisation du système R_{robot} par rapport au système $R_{env.}$ revient à établir la correspondance entre les arêtes observées du modèle local et les amers du modèle global. Or, les amers utilisables dans le cadre d'un environnement intérieur possèdent sensiblement les mêmes caractéristiques. Il en résulte que leur description ne suffit pas à les identifier individuellement.

Le recours à une information de localisation a priori permet de prédire la position des amers dans le modèle local du robot et ainsi de restreindre l'ensemble des appariements plausibles entre les arêtes observées et les amers environnementaux [Couroux 97]. Il s'agit alors d'un recalage de l'information de position et d'orientation du robot mobile. Ce type de solution est très efficace mais ne permet pas de vérifier de façon indépendante la viabilité des mises en correspondance au fil du temps. Afin d'assurer l'autonomie du robot mobile et notamment d'autoriser la ré-initialisation automatique de l'information de localisation, nous nous plaçons dans le cas de figure où l'information de localisation a priori est très fortement perturbée : le robot sait dans quelle zone de l'environnement il se trouve, mais il ne dispose pas d'informations suffisamment précises pour émettre des hypothèses concernant la position des amers dans le modèle local de l'environnement.

Le robot doit donc être capable de détecter et d'identifier avec confiance les amers environnementaux tout au long de son parcours. Il est par conséquent nécessaire de modéliser un grand nombre d'amers afin de s'assurer que le robot puisse toujours en percevoir suffisamment pour se localiser. Deux problèmes majeurs se posent alors. D'une part, un

modèle global précis et complet s'avère très difficile à créer. Ainsi, il est possible que certaines structures de l'environnement pouvant être utiles pour la détermination de la localisation soient perceptibles par les capteurs mais ne soient par répertoriées dans le modèle global. D'autre part, la probabilité d'obtenir un appariement erroné augmente lorsque le modèle global est lui même ambigu, c'est-à-dire qu'il contient des amers très semblables non différenciables les uns des autres. La fiabilité des mises en correspondance diminue donc lorsque le nombre d'amers environnementaux augmente.

Afin de pallier à ces inconvénients, nous proposons de recourir à une description du monde d'évolution du robot mobile portant à la fois sur les caractéristiques individuelles des amers et également sur les relations liant les amers les uns aux autres. La procédure de mise en correspondance représentée Figure 4.1 est ainsi basée sur une transformation des arêtes observées et des amers environnementaux en une représentation faisant apparaître des invariants géométriques.

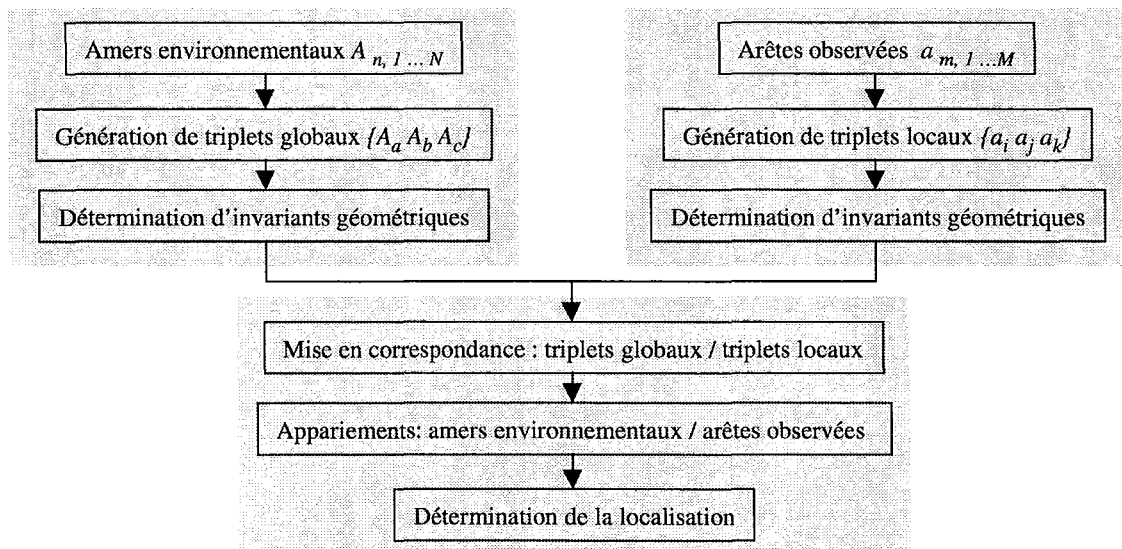


Figure 4.1 : Procédure de mise en correspondance.

Nous présentons tout d'abord les invariants géométriques utilisés, puis nous détaillons la construction du modèle global de l'environnement, ainsi que le conditionnement du modèle local (paragraphe 4.2). Nous analysons ensuite la procédure réalisant les mises en correspondance entre groupements d'amers et groupements d'arêtes observées, puis entre amers et arêtes observées (paragraphe 4.3). Nous traitons enfin la détermination de la position et de l'orientation du robot mobile à partir des appariements réalisés (paragraphe 4.4).

4.2 REPRESENTATION DES DONNEES

Le problème inhérent à tout système de reconnaissance, réside dans le grand nombre de combinaisons d'appariements plausibles entre les primitives observées et les amers du modèle, dès lors que ces amers possèdent sensiblement les mêmes caractéristiques. Le recours à des contraintes géométriques liant les amers au sein de *k-uplets* permet de simplifier la recherche des appariements.

Nous présentons ci-après les descripteurs invariants utilisés, puis nous détaillons le conditionnement des modèles global et local de l'environnement.

4.2.1 Descripteurs invariants

Soit un ensemble de configurations pouvant contenir des points, des droites et des courbes, qui sont transformées par des éléments d'un ensemble de transformations. Un invariant est une quantité numérique calculée à partir de chaque configuration qui garde la même valeur lorsque la configuration est transformée par l'un des éléments de l'ensemble des transformations.

La théorie des invariants est un thème actif de recherche en vision par ordinateur [Mundy 92]. Les premiers invariants utilisés furent les invariants topologiques, tel que la colinéarité et l'adjacence. Des invariants numériques ont aussi été utilisés pour le développement de systèmes industriels robustes de reconnaissance de forme [Bamieh 86]. Il s'agissait d'invariants globaux, des moments ou des rapports d'une aire sur le carré d'un périmètre, et les algorithmes nécessitaient de bonnes conditions d'éclairage et de contraste, et une absence d'occlusion partielle.

Récemment, de nombreux travaux ont été réalisés sur la définition d'invariants locaux, utilisables par conséquent en cas d'occlusion partielle, mais nécessitant une segmentation fiable. Ces invariants géométriques sont calculés à partir des coordonnées d'un petit nombre de primitives et sont limités à des structures contraintes telles que des ensembles coplanaires de points, de lignes et de courbes. Il peut notamment s'agir d'angles et de rapports de distances associés à des paires de segments concourants [Gross 93], des coordonnées affines associées à des quadruplets de points coplanaires [Wolfson 90], des invariants affines tels que les coordonnées barycentriques [Lee 90], ... Les méthodes utilisées dans le cadre de la vision par ordinateur s'appuient principalement sur l'utilisation d'invariants projectifs pour la modélisation et la reconnaissance d'objets tridimensionnels. Le birapport est ainsi

couramment utilisé [Mohr 91] et peut être défini à partir de nombreuses configurations planes telles que quatre droites concourantes, cinq points quelconques, quatre points sur une conique, ... Les invariants projectifs nécessitant des configurations complexes de primitives s'avèrent toutefois être sensibles au bruit et aux phénomènes d'occlusion.

Dans le cadre de notre étude, nous recherchons une représentation des arêtes observées et des amers, invariante en translation et en rotation qui sont les deux transformations euclidiennes du repère R_{robot} vers le repère $R_{env.}$.

Une première solution consiste à exprimer les coordonnées des amers dans toutes les bases possibles formées à partir de paires d'amers ordonnés. Chaque base possible du modèle global est alors indexée dans une table de hachage par les coordonnées des autres amers dans cette base. Lors de la reconnaissance, on choisit une base dans le modèle local et on exprime les coordonnées des autres arêtes dans cette base. Ces coordonnées étant invariantes, on retrouve et on vote, au moyen de la table de hachage, pour les bases modèles possédant des amers dans la même configuration [Lamdan 88]. La précision des mesures doit être excellente, sans quoi la probabilité d'obtenir de nombreux faux positifs devient très vite importante et peut conduire à des votes aberrants.

Une solution plus fiable fait intervenir les invariants de plusieurs primitives à la place des coordonnées. L'accumulation des évidences se fait toujours dans l'espace des appariements. Si nous prenons l'exemple d'un quadruplet ordonné $\{A_a A_b A_c A_d\}$, les invariants caractéristiques de la forme constituée de ces quatre amers sont utilisés pour construire la table de hachage, au lieu des coordonnées de A_a dans la base construite à partir de $A_b A_c A_d$. Au niveau de l'accumulation des évidences, il s'agit alors d'incrémenter les appariements possibles individuellement au lieu de le faire pour la base. Si $a_i a_j a_k a_l$ est un quadruplet d'arêtes compatible dans le modèle local, on votera alors pour chaque appariement $(a_i A_a)$ du quadruplet.

La configuration d'amers que nous avons retenu est celle du triplet car elle ne nécessite que 3 primitives. Ce point est très important, car comme nous l'avons souligné au chapitre précédent, le modèle local est constitué en moyenne de 5 à 7 arêtes observées. Une constellation formée d'un petit nombre de primitives est donc nécessaire afin d'obtenir un nombre important de constellations candidates. Il est ainsi également possible de décrire dans le modèle global des zones de l'environnement caractérisées par une faible présence d'amers.

Les invariants utilisés sont les 3 longueurs L des côtés et les 3 angles α du triangle formé par tout triplet d'amers $\{A_a A_b A_c\}$ [Atiya 93]. En théorie, deux longueurs et un angle suffisent à

définir un triplet unique. Toutefois, la présence d'incertitude sur les mesures des coordonnées des arêtes observées conduit à utiliser des paramètres redondants.

Si l'on modélise une vaste portion de l'environnement, il est probable que plusieurs triplets d'amers présenteront des valeurs de paramètres semblables. Ceci est dû au fait que dans un milieu structuré, les amers environnementaux sont souvent répartis de façon sensiblement uniforme. Nous proposons par conséquent d'enrichir la description du triplet grâce à deux paramètres propres aux amers utilisés ; à savoir l'angle Ω formé par les plans verticaux support de l'arête et le statut *réel* ou *virtuel* des segments liant les arêtes (Figure 4.2).

Chaque triplet d'amers $\{A_a A_b A_c\}$ est ainsi caractérisé par le vecteur $P_{A\{abc\}}$ constitué de 12 paramètres invariants en rotation et translation :

$$P_{A\{abc\}} = [L_{ab} \ L_{bc} \ L_{ca} \ \alpha_a \ \alpha_b \ \alpha_c \ \Omega_a \ \Omega_b \ \Omega_c \ \text{statut}_{ab} \ \text{statut}_{bc} \ \text{statut}_{ca}] \quad (4.1)$$

Avec :

- L_{ab} , L_{bc} et L_{ca} , longueurs des côtés du triangle.
- α_a , α_b et α_c , angles internes du triangle.
- Ω_a , Ω_b et Ω_c , angles des arêtes verticales.
- statut_{ab} , statut_{bc} et statut_{ca} , statut *réel* ou *virtuel* des segments liant les arêtes.

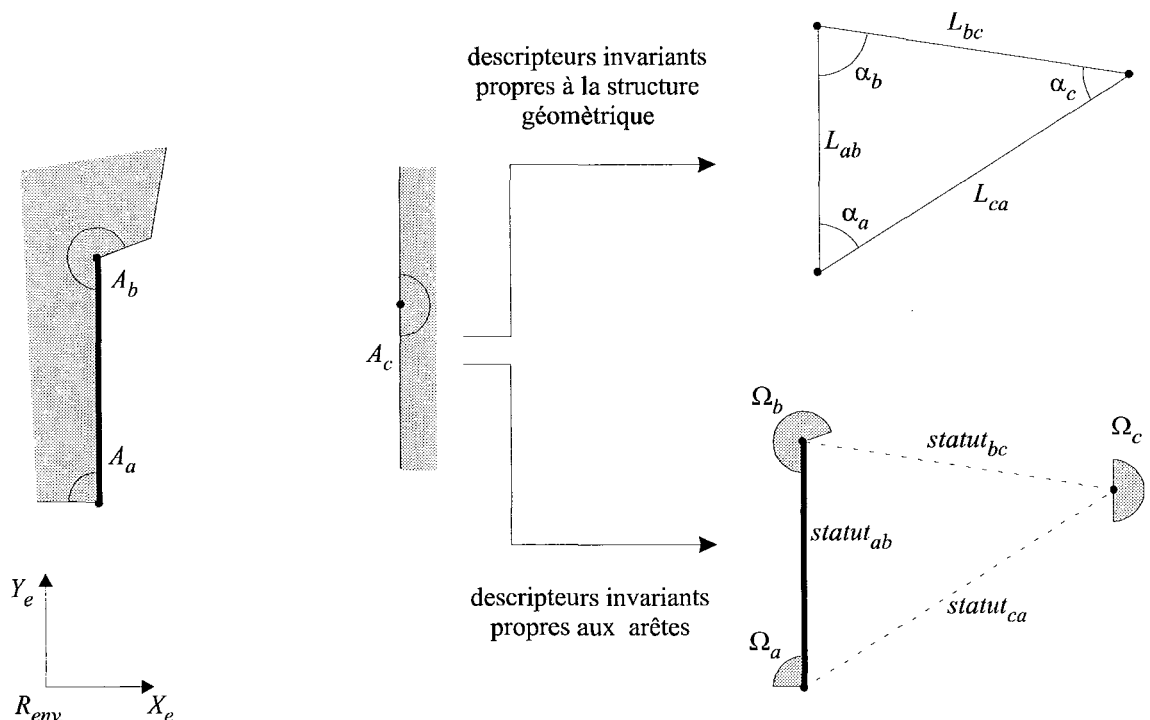


Figure 4.2 : Descripteurs invariants associés à tout triplet d'amers $\{A_a A_b A_c\}$.

4.2.2 Le modèle global de l'environnement

Les amers sont recensés dans une carte théorique de l'environnement décrivant l'espace au sein duquel va évoluer le robot.

Nous présentons tout d'abord les paramètres utilisés pour modéliser les amers, puis nous détaillons la procédure de génération automatique de triplets d'amers.

4.2.2.1 Les amers environnementaux

Les amers environnementaux sont de longues arêtes verticales telles que celles provoquées par un angle de mur, un encadrement de porte ou encore une différence de teinte entre 2 murs. Dans le cadre de nos expérimentations, 64 amers sont modélisés afin de décrire un couloir de 30 m sur 10 m communiquant avec différentes pièces.

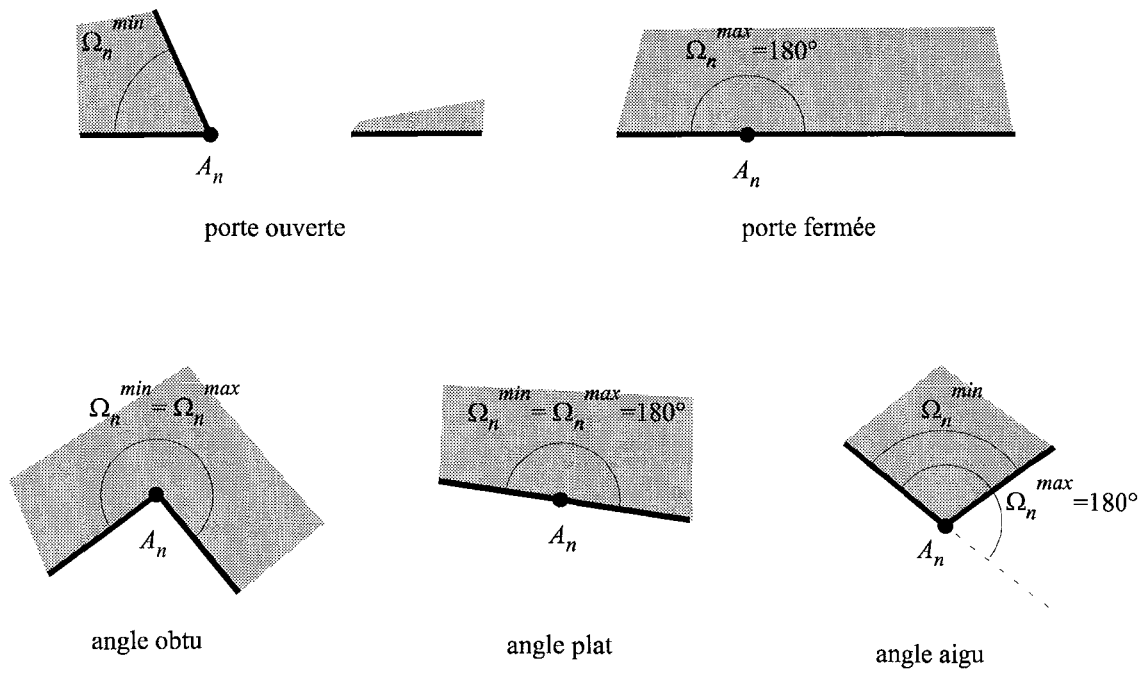
Chaque amer A_n est caractérisé par les paramètres :

- (x_n, y_n) , coordonnées de la projection de l'arête A_n dans le plan (X_e, Y_e) du repère de référence R_{env} .

Elles ont été obtenues à partir d'une campagne de mesure combinant l'exploitation de données télémétriques (le robot suivant une trajectoire rectiligne connue), de plans d'architecte du site et de mesures directes à l'aide d'une chaîne d'arpenteur. Les coordonnées des arêtes présentent donc une incertitude liée aux principes de mesure que nous estimons à environ un centimètre.

- Ω_n , angle formé par le(s) plan(s) support(s) de l'arête A_n .

La valeur de l'angle Ω_n est comprise entre deux bornes Ω_n^{min} et Ω_n^{max} . En effet, l'angle Ω d'un amer correspondant à un chambranle de porte change selon que cette dernière est ouverte ou fermée. De même, un angle de mur aigu peut être perçu, selon la position relative du robot, comme un angle de Ω_n^{min} à 180° . Seuls les angles de mur obtus ou plats n'admettent pas de plage de variation. La Figure 4.3 présente les différents cas possibles.

Figure 4.3 : Plage de variation de l'angle Ω d'un amer.

- V_n , liste des amers appartenant au(x) plan(s) support(s) de A_n .

Cette information permet de déterminer le statut *réel* ou *virtuel* du segment liant deux amers. Toutefois, si deux amers appartiennent à un même plan vertical contenant une porte, alors, le statut du segment dépend de l'ouverture ou de la fermeture de la porte. Une marque A_* est donc utilisée en supplément pour les amers appartenant à un plan vertical contenant une ou plusieurs portes, et une liste $\mathcal{P}$ des amers toujours reliés est créée (Figure 4.4).

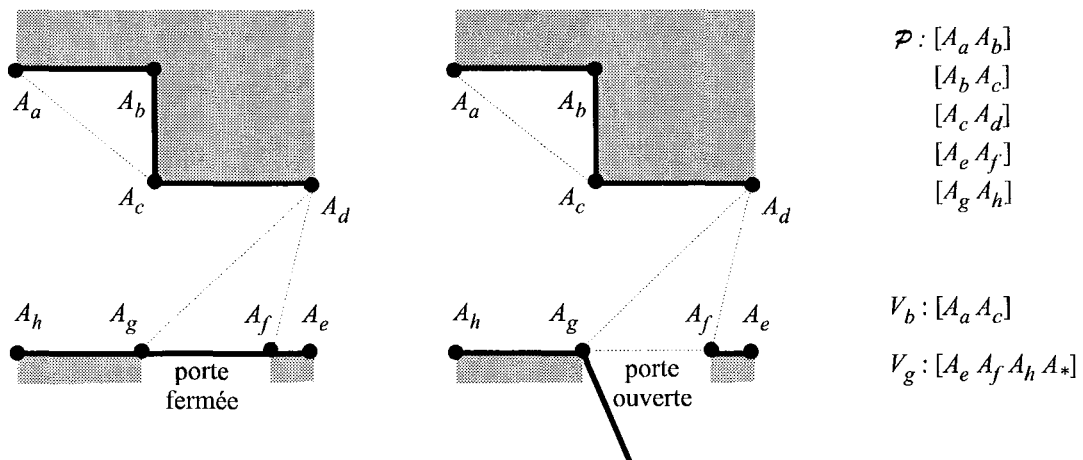


Figure 4.4: Amers appartenant à un même plan vertical.

4.2.2.2 Formation de triplets consistants

Toute configuration de 3 amers $A_a A_b$ et A_c correspond à 6 triplets distincts : $\{A_a A_b A_c\}$, $\{A_b A_c A_a\}$, $\{A_c A_a A_b\}$, $\{A_a A_c A_b\}$, $\{A_c A_b A_a\}$ et $\{A_b A_a A_c\}$.

Si nous définissons un triplet d'arêtes observées $\{a_i a_j a_k\}$ à partir de la séquence angulaire $\vartheta_i > \vartheta_j > \vartheta_k$, il est possible de limiter la modélisation aux seuls triplets orientés susceptibles d'être perçus par le robot ; à savoir les triplets $\{A_a A_b A_c\}$, $\{A_b A_c A_a\}$, $\{A_c A_a A_b\}$ (Figure 4.5).

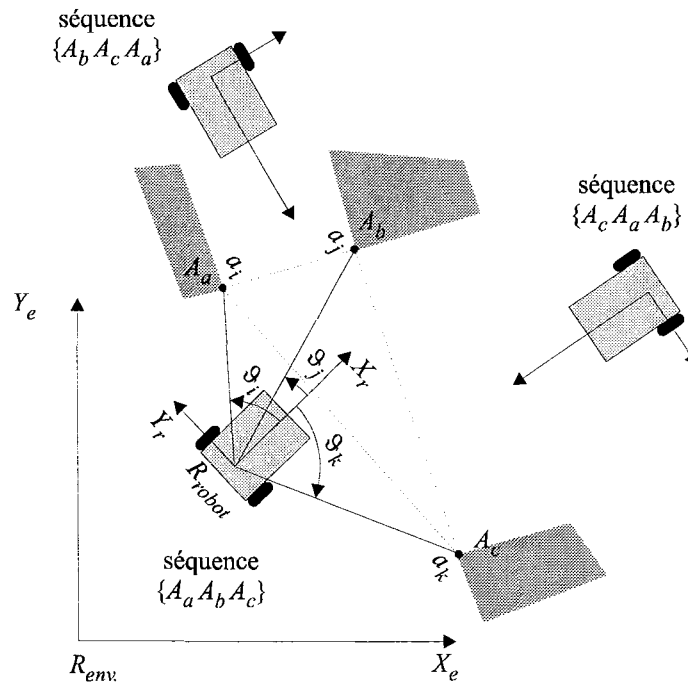


Figure 4.5 : Triplet d'arêtes observées.

Le nombre de combinaisons possibles de 3 amers pris parmi N est donné par :

$$N_{\text{triplets}} = \frac{3N!}{3!(N-3)!} \quad (4.2)$$

Les 64 amers modélisant le couloir utilisé pour nos expérimentations génèrent ainsi 124992 triplets orientés pour lesquels nous déterminons les invariants géométriques L et α à partir des coordonnées (x,y) des amers.

Afin de construire un modèle global exploitable, il est nécessaire de réduire cette quantité colossale de combinaisons théoriques. Nous choisissons par conséquent d'éliminer les triplets impossibles à percevoir, soit du fait d'une trop grande distance séparant les amers, soit du fait de phénomènes d'occultation résultant de la disposition relative des amers (Figure 4.6).

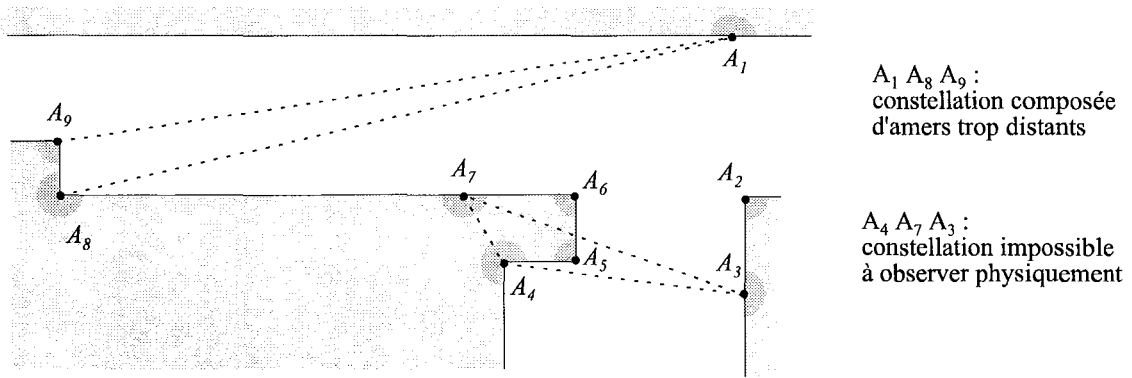


Figure 4.6 : Constellations de triplets d'amers délicates à percevoir.

- Les triplets d'amers dont une longueur L est supérieure à un seuil de 10 m sont éliminés. Bien que le système de perception autorise la détection d'amers distants l'un de l'autre de $2 \times 16 = 32$ m, la configuration du site expérimental fait que dans 99 % des cas, les triplets d'arêtes observées sont caractérisés par des distances L inférieures à ce seuil de 10 m.
- Seuls les triplets pouvant être perçus physiquement par le robot sont validés. Les phénomènes d'occultation d'amers provoqués par la présence de plans verticaux opaques (les murs) doivent ainsi être pris en compte. Nous considérons pour cela que le robot se trouve à l'intérieur du couloir, ou dans une pièce communiquant avec le couloir. Soit A_a , A_b et A_c trois arêtes successives, la Figure 4.7 illustre les 4 cas pouvant se présenter :

Cas n°1 : les arêtes A_a et A_c appartiennent au(x) même(s) plan(s) que l'arête A_b ($A_a \in V_b$ et $A_c \in V_b$). Seul le triplet $\{A_a A_b A_c\}$ est perceptible.

Cas n°2 : l'arête A_c n'appartient pas au(x) même(s) plan(s) que les arêtes A_a et A_b ($A_a \in V_b$ et $A_c \notin V_b$). Le triplet $\{A_a A_b A_c\}$ est perceptible, ainsi que le triplet $\{A_c A_a A_b\}$ si l'angle α est inférieur ou égal à 180° .

Cas n°3 : l'arête A_a n'appartient pas au même(s) plan(s) que les arêtes A_b et A_c ($A_a \notin V_b$ et $A_c \in V_b$). Le triplet $\{A_a A_b A_c\}$ est perceptible, ainsi que le triplet $\{A_b A_c A_a\}$ si l'angle α est inférieur ou égal à 180° .

Cas n°4 : les trois arêtes appartiennent à des plans différents ($A_a \notin V_b$ et $A_c \notin V_b$). Les triplets $\{A_a A_b A_c\}$, $\{A_c A_a A_b\}$ et $\{A_b A_c A_a\}$ sont perceptibles.

L'élimination de ces triplets inconsistants conduit à une réduction de 86% du volume des données puisque seuls 17548 triplets d'amers sont validés.

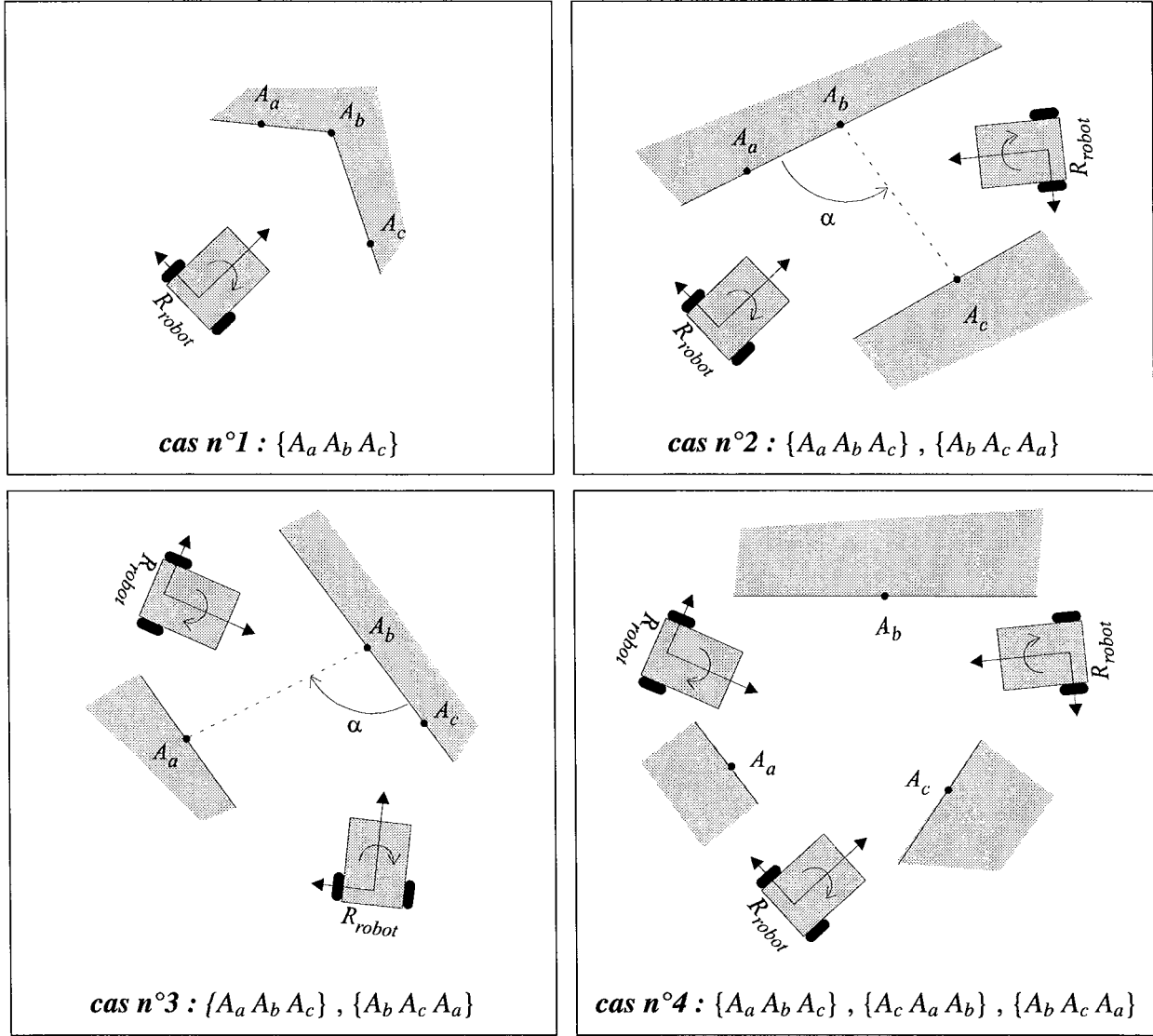


Figure 4.7 : Triplets d'amers perceptibles selon la configuration du site.

A chaque triplet d'amer validé $\{A_a A_b A_c\}$, nous affectons alors les paramètres Ω_a , Ω_b et Ω_c propres aux amers.

Les invariants $statut_{ab}$, $statut_{bc}$ et $statut_{ca}$ correspondant au statut *réel* ou *virtuel* des segments formant tout triplet $\{A_a A_b A_c\}$ sont ensuite déterminés. Puisque les portes peuvent être ouvertes ou fermées, un segment reliant un amer A_a représentant un encadrement de porte à un autre amer A_b est susceptible de prendre les deux états. Il peut donc s'avérer nécessaire de générer de deux à quatre triplets pour une même configuration d'amers. L'algorithme présenté Tableau 4.1 et Tableau 4.2 exploite les listes des amers appartenant aux mêmes plans verticaux afin de déterminer l'ensemble des triplets possibles.

Nous obtenons finalement un modèle global de l'environnement modélisé par 23880 triplets d'amers caractérisés par 12 invariants géométriques L , α , Ω et $statut$.

Appartenance au mêmes plans verticaux ?		Liens particuliers entre amers ?		$statut_{abc}$
si	$A_a \in V_b \quad A_b \in V_c \quad A_c \in V_a$	alors	si $A_* \in (V_a V_b V_c), ([A_a A_b], [A_b A_c]) \notin \mathcal{P}$	alors h, e, c et a
			sinon si $A_* \in (V_a V_b), [A_a A_b] \in \mathcal{P}$	alors h et e
			sinon si $A_* \in (V_b V_c), [A_b A_c] \in \mathcal{P}$	alors h et c
		sinon		alors h
sinon si	$A_a \in V_b \quad A_b \in V_c \quad A_c \notin V_a$	alors	si $A_* \in (V_a V_b V_c), ([A_a A_b], [A_b A_c]) \in \mathcal{P}$	alors g
			sinon si $A_* \in (V_a V_b), [A_a A_b] \in \mathcal{P}$	alors g et e
			sinon si $A_* \in (V_b V_c), [A_b A_c] \in \mathcal{P}$	alors g et c
		sinon		alors g, e, c et a
sinon si	$A_a \notin V_b \quad A_b \in V_c \quad A_c \in V_a$	alors	si $A_* \in (V_a V_b V_c), ([A_b A_c], [A_c A_a]) \in \mathcal{P}$	alors d
			sinon si $A_* \in (V_b V_c), [A_b A_c] \in \mathcal{P}$	alors d et c
			sinon si $A_* \in (V_c V_a), [A_c A_a] \in \mathcal{P}$	alors d et b
		sinon		alors d, c, b et a
sinon si	$A_a \in V_b \quad A_b \notin V_c \quad A_c \in V_a$	alors	si $A_* \in (V_a V_b V_c), ([A_a A_b], [A_c A_a]) \in \mathcal{P}$	alors f
			sinon si $A_* \in (V_a V_b), [A_a A_b] \in \mathcal{P}$	alors f et b
			sinon si $A_* \in (V_c V_a), [A_c A_a] \in \mathcal{P}$	alors f et e
		sinon		alors f, e, b et a
sinon si	$A_a \in V_b \quad A_b \notin V_c \quad A_c \notin V_a$	alors	si $A_* \in (V_a V_b), [A_a A_b] \notin \mathcal{P}$	alors e et a
		sinon		alors e
sinon si	$A_a \notin V_b \quad A_b \in V_c \quad A_c \notin V_a$	alors	si $A_* \in (V_b V_c), [A_b A_c] \notin \mathcal{P}$	alors c et a
		sinon		alors c
sinon si	$A_a \notin V_b \quad A_b \notin V_c \quad A_c \in V_a$	alors	si $A_* \in (V_c V_a) \text{ et } [A_c A_a] \notin \mathcal{P}$	alors b et a
		sinon		alors b
sinon				alors a

Tableau 4.1 : Statut possible des segments liés par une configuration d'amers $\{A_a A_b A_c\}$.

	$statut_{abc}$							
	a	b	c	d	e	f	g	h
$statut_{ab}$	virtuel	virtuel	virtuel	virtuel	réel	réel	réel	réel
$statut_{bc}$	virtuel	virtuel	réel	réel	virtuel	virtuel	réel	réel
$statut_{ca}$	virtuel	réel	virtuel	réel	virtuel	réel	virtuel	réel

Tableau 4.2 : Statut des segments composant le triplet $\{A_a A_b A_c\}$.

4.2.2.3 Le modèle global

Pour chaque triplet d'amer $\{A_a A_b A_c\}$, on ajoute la paire $\langle \{A_a A_b A_c\}, P_{A\{abc\}} \rangle$ à la liste $\mathcal{L}$ (Figure 4.8). La longueur q de cette liste est donc égale au nombre de triplets d'amers.

$$\begin{array}{l}
 \langle \text{triplet} , \quad \text{vecteur des paramètres invariants} \quad \rangle \\
 \\
 \mathcal{L}_1 : \quad \langle \{A_a A_b A_c\} , [L_{ab} \ L_{bc} \ L_{ca} \ \alpha_a \ \alpha_b \ \alpha_c \ \Omega_a \ \Omega_b \ \Omega_c \ \text{seg}_{ab} \ \text{seg}_{bc} \ \text{seg}_{ca}] \rangle \\
 \mathcal{L}_2 : \quad \langle \{A_a A_b A_d\} , [L_{ab} \ L_{bd} \ L_{da} \ \alpha_a \ \alpha_b \ \alpha_d \ \Omega_a \ \Omega_b \ \Omega_d \ \text{seg}_{ab} \ \text{seg}_{bd} \ \text{seg}_{da}] \rangle \\
 \mathcal{L}_3 : \quad \langle \{A_c A_d A_e\} , [L_{cd} \ L_{ce} \ L_{ec} \ \alpha_c \ \alpha_d \ \alpha_e \ \Omega_c \ \Omega_d \ \Omega_e \ \text{seg}_{cd} \ \text{seg}_{de} \ \text{seg}_{ec}] \rangle \\
 \dots \quad \quad \quad \dots \quad \quad \quad \dots \\
 \mathcal{L}_q : \quad \langle \{A_f A_h A_q\} , [L_{fh} \ L_{hq} \ L_{qf} \ \alpha_f \ \alpha_h \ \alpha_q \ \Omega_f \ \Omega_h \ \Omega_q \ \text{seg}_{fh} \ \text{seg}_{hq} \ \text{seg}_{qf}] \rangle
 \end{array}$$

Figure 4.8 : Liste $\mathcal{L}$ décrivant le modèle global.

Les courbes présentées Figure 4.9 et Figure 4.10 donnent le nombre moyen de triplets d'amers caractérisés par des valeurs d'invariants proches. L'incertitude sur la valeur des paramètres est donnée en centimètres pour $L_{ab} \ L_{bc} \ L_{ca}$ et en degrés pour $\alpha_a \ \alpha_b \ \alpha_c$.

Les invariants L et α sont très discriminants mais ne garantissent pas l'unicité des triplets. A chaque triplet d'amers $\{A_a A_b A_c\}$ décrivant le couloir peuvent ainsi correspondre de 1 à 20 triplets présentant des valeurs de paramètres proches. Il s'agit principalement de configurations d'amers causées par les chambranles de porte en vis-à-vis.

Pour cette raison, les paramètres Ω et *statut* qui ne fournissent que peu d'informations sélectives si on les considère séparément, sont très importants lorsqu'ils sont intégrés en fin de traitement puisqu'ils permettent de diviser par 2 à 3 le nombre de triplets semblables.

Il reste toutefois évident qu'un seul triplet d'arêtes observées ne peut garantir une localisation unique du robot.

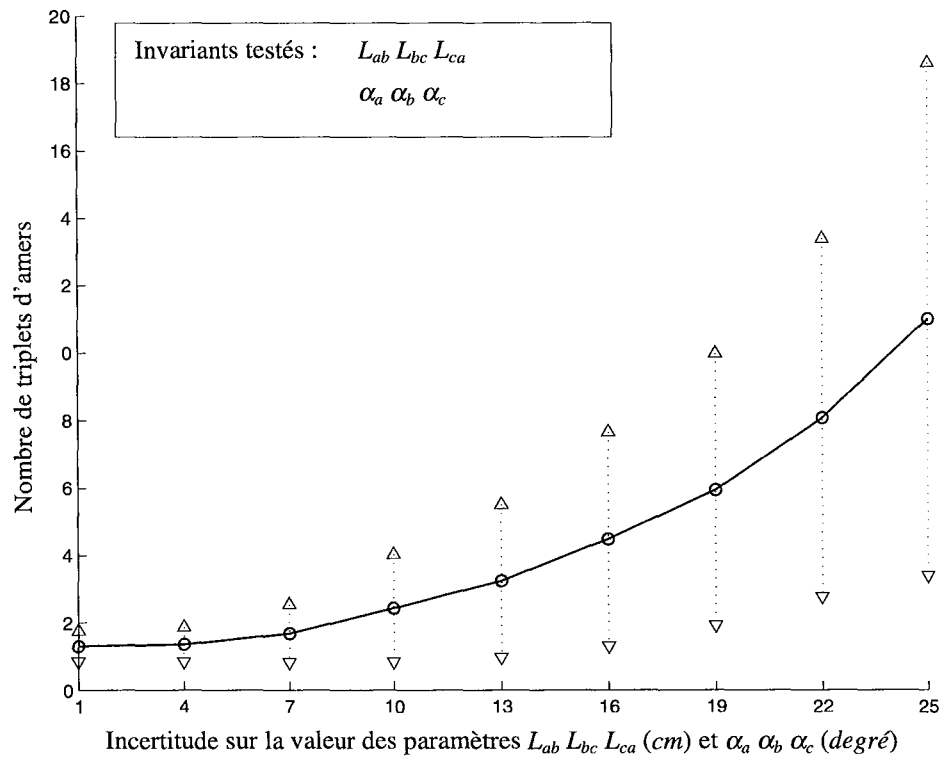


Figure 4.9 : Nombre moyen de triplets d'amers caractérisés par des valeurs d'invariants L et α proches.

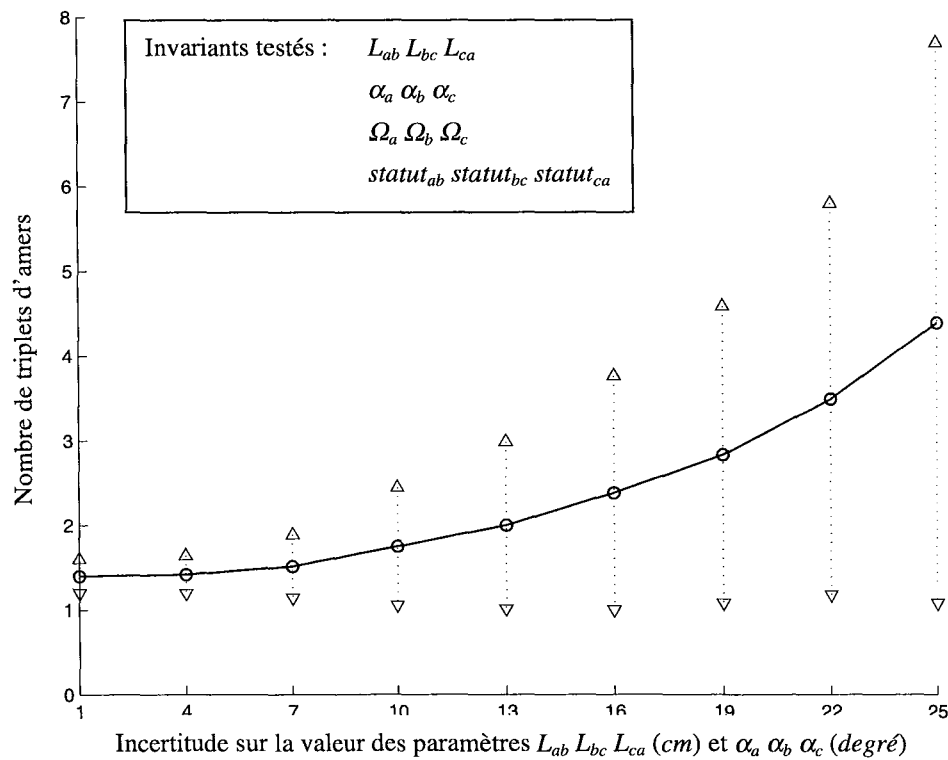


Figure 4.10 : Nombre moyen de triplets d'amers caractérisés par des valeurs d'invariants L , α , Ω et statut proches.

4.2.3 Modèle local de l'environnement

Le modèle local est constitué de M arêtes verticales observées a_m caractérisées par :

$$a_m : \{a_{m,1}, a_{m,2}, a_{m,3}, a_{m,4}, plan_m\} \quad (4.3)$$

Avec : $a_{m,1}$, $a_{m,2}$, $a_{m,3}$ et $a_{m,4}$, les points définissant la région d'incertitude contenant a_m ,
 $plan_m$, indices des plans droits et gauche de la primitive coin.

4.2.3.1 Constitution des triplets d'arêtes

La génération de tous les triplets d'arêtes observées $\{a_i, a_j, a_k\}$ constitués à partir de la séquence angulaire $\vartheta_i > \vartheta_j > \vartheta_k$ conduit à un total de M_{triplets} possibles.

$$M_{\text{triplets}} = \frac{M!}{3!(M-3)!} \quad (4.4)$$

La Figure 4.11 présente un exemple de triplets formés à partir de 6 arêtes observées du modèle local.

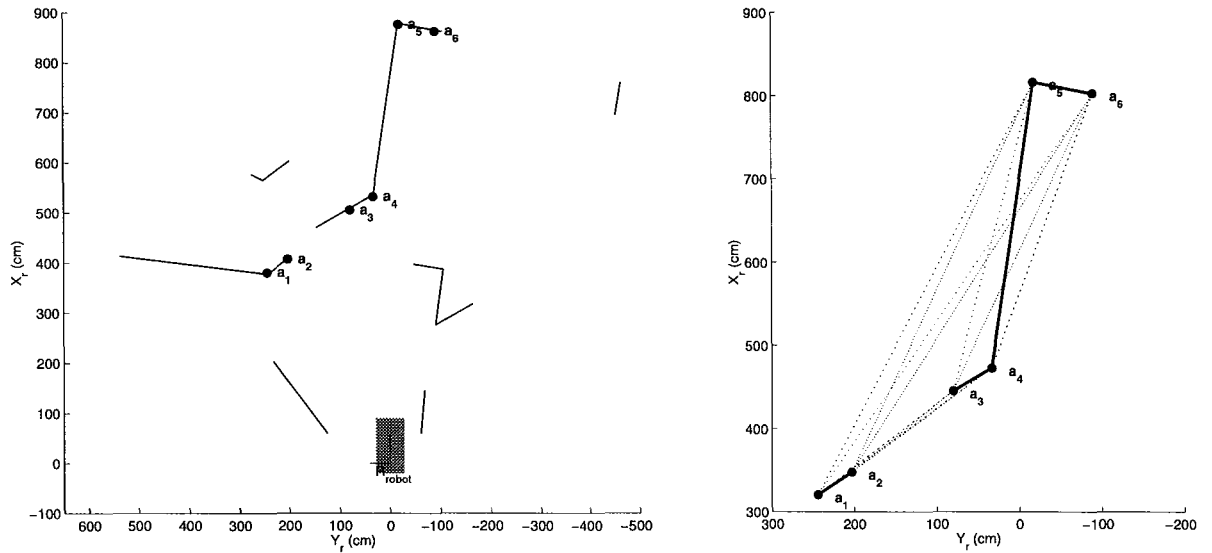


Figure 4.11 : Les triplets formés à partir des arêtes observées $a_i, i=1 \dots 6$.

Les erreurs liées à la modélisation des arêtes observées a_i, a_j, a_k doivent être prises en compte lors du calcul des paramètres invariants caractérisant chaque triplet.

A la différence des méthodes statistiques définissant un domaine d'équiprobabilité suivant une loi de répartition, les méthodes dites à « erreur bornée » définissent un domaine quelconque de valeurs possibles à partir des erreurs limites sur les mesures. Cette formulation ne nécessite donc pas de fortes suppositions quand à la distribution des erreurs de mesure.

L'approche retenue consiste à propager la borne de l'erreur de mesure dans la méthode de calcul des invariants afin d'obtenir une plage de recherche assurant de ne pas manquer l'appariement avec le triplet d'amer correspondant. Pour chaque triplet d'arêtes observées $\{a_i, a_j, a_k\}$ nous déterminons donc les vecteurs $P_{a\{ijk\}}^{\min}$ et $P_{a\{ijk\}}^{\max}$ des paramètres invariants.

$$P_{a\{ijk\}}^{\min} = \begin{bmatrix} L_{ij}^{\min} & L_{jk}^{\min} & L_{ki}^{\min} \\ \alpha_i^{\min} & \alpha_j^{\min} & \alpha_k^{\min} \\ \Omega_i^{\min} & \Omega_j^{\min} & \Omega_k^{\min} \\ statut_{ij} & statut_{jk} & statut_{ki} \end{bmatrix} \quad P_{a\{ijk\}}^{\max} = \begin{bmatrix} L_{ij}^{\max} & L_{jk}^{\max} & L_{ki}^{\max} \\ \alpha_i^{\max} & \alpha_j^{\max} & \alpha_k^{\max} \\ \Omega_i^{\max} & \Omega_j^{\max} & \Omega_k^{\max} \\ statut_{ij} & statut_{jk} & statut_{ki} \end{bmatrix}$$

Les erreurs minimales et maximales commises sur les paramètres L , α et Ω sont calculées à partir des erreurs de modélisation des arêtes a_i , a_j et a_k .

- La distance maximale $L_{ij}^{\max}$ entre a_i et a_j apparaît toujours pour les points extrêmes des régions d'incertitude. La distance minimale $L_{ij}^{\min}$ entre a_i et a_j apparaît également pour les points extrêmes, excepté lorsqu'une perpendiculaire à un segment d'une région passe par un point extrême de l'autre région.

Soit $a_{i,l}$ un point extrême de la région contenant a_i et $a_{j,m}$, $a_{j,n}$ deux points extrêmes adjacents de la région contenant a_j . La distance minimale entre le point $a_{i,l}$ et le segment $[a_{j,m}, a_{j,n}]$ est obtenue à partir de :

$$\lambda = \frac{(a_{i,l} - a_{j,m}) \cdot (a_{j,n} - a_{j,m})}{\|a_{j,n} - a_{j,m}\|^2}$$

Si $\lambda \in [0,1]$, alors la distance est donnée par $\|a_{i,l} - (a_{j,m} + \lambda(a_{j,n} - a_{j,m}))\|$.

Si λ est négatif, alors la distance est donnée par $\|a_{i,l} - a_{j,m}\|$.

Si λ est positif, alors la distance est donnée par $\|a_{i,l} - a_{j,n}\|$.

Ce calcul est réalisé pour toutes les combinaisons de points extrêmes de a_i et de segments de a_j , et vice versa. Le minimum de toutes ces distances correspond à L_{ij}^{min} .

- L'angle minimum α_j^{min} formé par les arêtes a_i , a_j et a_k apparaît toujours pour les points extrêmes des régions d'incertitude. L'angle maximum α_j^{max} apparaît parfois pour les points extrêmes mais également quelquefois en choisissant deux points extrêmes de a_i et a_k , et un point appartenant à un des segments délimitant a_j . Afin de limiter les calculs, nous proposons une solution approchée consistant à rechercher le minimum et le maximum des angles définis par les points extrêmes de a_i et a_k et les points extrêmes et milieu des segments de a_j .
- Les angles Ω_j^{min} et Ω_j^{max} sont calculés suivant le même principe que les angles α_j^{min} et α_j^{max} , mais en substituant à a_i et a_k les régions d'incertitude des points extrémités gauche et droit des plans supports de l'arête a_j .

La Figure 4.12 illustre la détermination des invariants L^{min} , L^{max} et α_j^{min} , α_j^{max} .

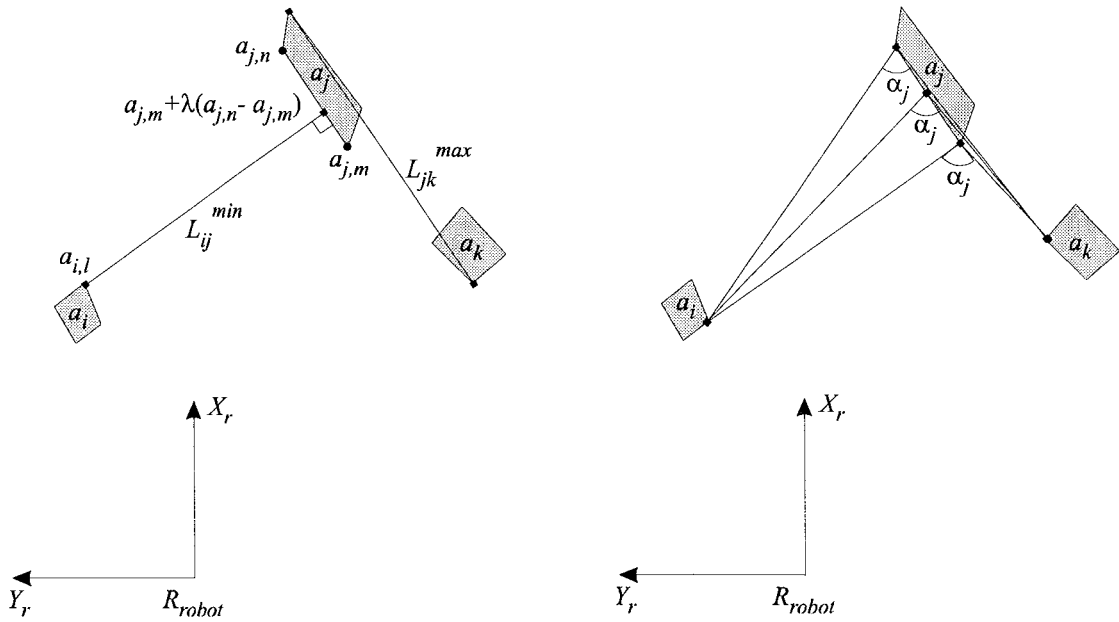


Figure 4.12 : Incertitude sur les invariants L et α .

Les paramètres invariants $statut_{ij}$, $statut_{jk}$ et $statut_{ki}$ sont quand à eux déduits des paramètres $plan_i$, $plan_j$ et $plan_k$ des arêtes : si deux arêtes appartiennent à un même plan vertical, alors le statut du segment les reliant est défini comme *réel*, sinon il est défini comme *virtuel*.

4.2.3.2 Fiabilité du modèle local

La procédure de localisation repose sur la mise en correspondance de triplets d'arêtes observées avec les triplets d'amers sur la base de paramètres invariants compatibles.

Deux constatations s'imposent alors :

- Plus les intervalles d'incertitude sur les paramètres L , α et Ω caractérisant les triplets d'arêtes seront grands et plus la fiabilité des appariements sera faible. En effet, de larges plages de variation imposent moins de contraintes sur les conditions d'appariement et augmentent donc la probabilité d'appariements erronés.

Cette tolérance peut toutefois être relativement grande à condition qu'il existe un nombre suffisant d'arêtes observées permettant de réaliser des recoupements d'hypothèses.

- La présence d'un ou de plusieurs *faux positifs* parmi les arêtes observées diminue considérablement le nombre de triplets consistants, mais génère également des triplets parasites susceptibles d'être appariés avec des triplets d'amers.

Le Tableau 4.3 donne le nombre de triplets d'arêtes observées correspondant réellement à des triplets d'amers en fonction du nombre d'arêtes observées de type faux positifs. La présence d'un *faux positif* est particulièrement sensible lorsque le nombre des arêtes observées est inférieur à 5.

		Nombre d'arêtes observées de type <i>faux positif</i>			
		0	1	2	3
<i>M</i> arêtes observées	3	1	0	0	0
	4	4	1	0	0
	5	10	4	1	0
	6	20	10	4	1
	7	35	20	10	4
	8	56	35	20	10

Tableau 4.3 : Nombre de triplets d'arêtes $\{a_i a_j a_k\}$ consistants selon le nombre d'arêtes observées faux positifs .

4.3 MISE EN CORRESPONDANCE

La détermination des mises en correspondance se fait dans un premier temps entre triplets d'amers et triplets d'arêtes observées sur la base des invariants calculés. Les appariements entre amers et arêtes observées sont ensuite déduits des hypothèses formées.

4.3.1 Mise en correspondance de triplets

Le modèle global consiste en une liste $\mathcal{L}$ de longueur q formée d'une suite de paires $\langle \text{triplets}, \text{vecteur} \rangle$. Le premier terme de chaque paire correspond aux indices des amers formant le triplet ordonné et le second terme correspond au vecteur $P_{A\{abc\}}$ des 12 descripteurs invariants du triplet considéré.

Nous désirons trouver au sein de cette base de données les triplets d'amers qui approchent au mieux l'ensemble des triplets d'arêtes observées. Les descripteurs invariants sont pour cela directement comparés afin de déterminer si deux triplets se trouvent dans la même configuration géométrique l'un par rapport à l'autre. L'analyse séquentielle pour chaque triplet d'arêtes observées des vecteurs $P_{A\{abc\}}$ de l'ensemble des triplets d'amers conduirait à des temps de calcul extrêmement importants. Il est donc nécessaire de définir une méthode de stockage et d'accès rapide aux invariants associés aux amers.

Nous proposons une procédure de mise en correspondance consistant à déterminer, dans un premier temps, l'ensemble des triplets d'amers compatibles avec les descripteurs invariants L et α du triplet d'arêtes considéré. Ces paramètres sont en effet très sélectifs et sont donc bien adaptés à une recherche par indexation. Les descripteurs invariants Ω et *statut* de chaque triplet d'amers candidat sont ensuite comparés avec ceux du triplet d'arêtes considéré et seuls les appariements compatibles sont conservés dans un ensemble $\mathcal{S}$ constitué de paires $\langle \{A_a A_b A_c\}, \{a_i a_j a_k\} \rangle$.

Des listes $\mathcal{L}^p_{,p=1\dots 6}$ sont ainsi créées pour les descripteurs invariants $L_{ab} L_{bc} L_{ca}$ et $\alpha_a \alpha_b \alpha_c$. Soit $\mathcal{L}^p$ la liste relative au descripteur invariant p , et $\mathcal{L}_{m,p}$ la valeur du $p^{\text{ème}}$ paramètre invariant du triplet pointé par l'indice m dans la liste $\mathcal{L}$. Pour chaque triplet d'amers de $\mathcal{L}$, on ajoute la paire $\langle m, \mathcal{L}_{m,p} \rangle$ à la liste $\mathcal{L}^p$.

Chaque liste $\mathcal{L}^p$ est donc constituée par une suite de paires $\langle \text{indice}, \text{valeur} \rangle$. Le premier terme de la paire correspond à l'indice pointant le triplet dans $\mathcal{L}$, et le second terme correspond à la valeur numérique du descripteur considéré (Figure 4.13).

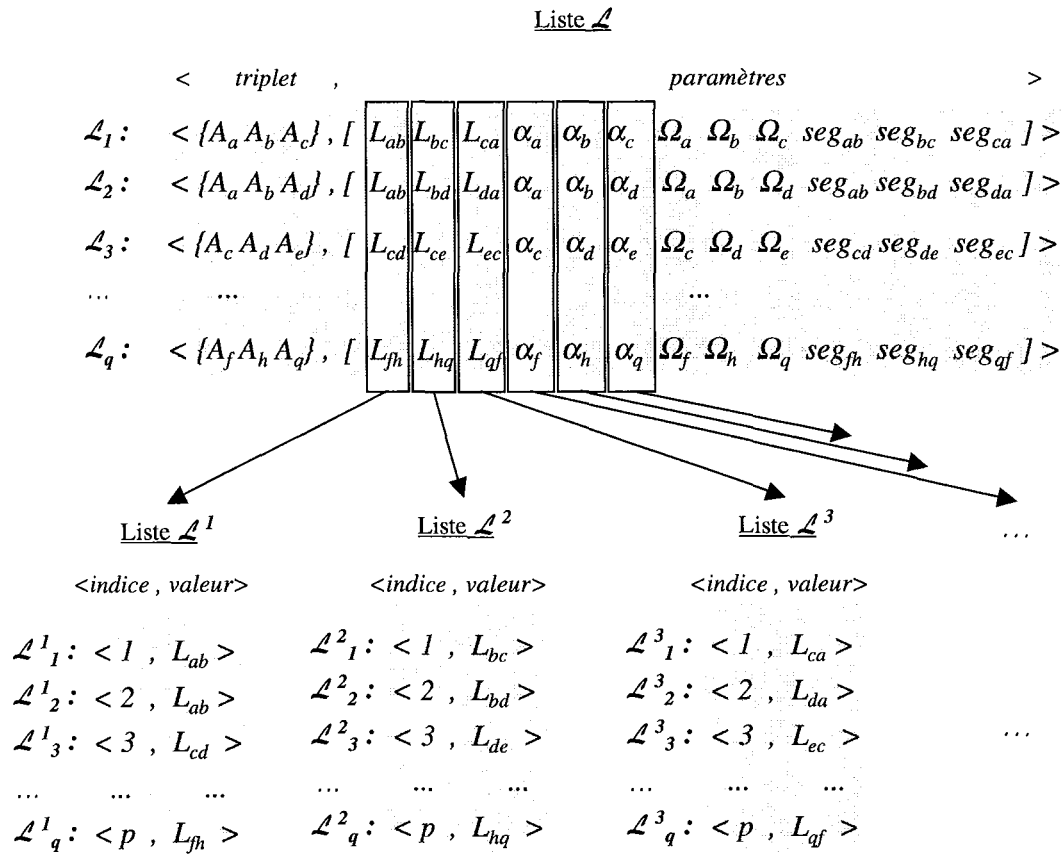


Figure 4.13 : Définition des listes $\mathcal{L}^p$ relatives aux descripteurs invariants L et α .

Nous proposons deux implémentations assurant une analyse rapide des listes $\mathcal{L}^p$. La première repose sur l'analyse de tables de hachage créées entièrement *hors ligne*, et la seconde sur l'indexation de listes ordonnées générées partiellement *en ligne*.

4.3.1.1 Tables de hachage

Une table de hachage discrétise l'espace des invariants et indexe les triplets d'amers à partir de leurs invariants. Pour chaque valeur d'invariant caractérisant un triplet d'arêtes observées, la table de hachage donne ainsi l'accès direct aux triplets d'amers possédant une valeur de paramètre compatible. Une table de hachage $\mathcal{T}^p$ est donc créée pour chaque liste $\mathcal{L}^p$.

Soit $\text{valeur}^{\min}$ et $\text{valeur}^{\max}$ les valeurs extrêmes du descripteur invariant relatif à la liste $\mathcal{L}^p$ considérée. Ces valeurs correspondent au premier et au dernier indice de la table de hachage $\mathcal{T}^p$. Un pas de hachage constant δ est alors défini en fonction de l'erreur ε sur la mesure du paramètre invariant.

- Si le pas de hachage δ est grand devant l'erreur ε , alors le nombre de lignes à indexer par triplet d'arêtes observées est plafonné à deux. La recherche est donc extrêmement rapide. La contrepartie est un étalement non minimal des valeurs du paramètre et donc un nombre sans doute élevé de faux positifs.
- Si le pas de hachage est très faible δ devant l'erreur ε , alors l'effet de la discrétisation est amorti au niveau de l'étalement des valeurs du paramètre, mais le nombre d'indexations par invariant peut être très élevé et surtout la taille de la table augmente.

Il y a donc un compromis à réaliser entre la place mémoire de la table et la perte de la sélectivité. Nous avons choisi un pas de hachage de 1 cm pour les paramètres de longueur L et un pas de 1 degré pour les paramètres d'angle α .

Les Figure 4.14 et Figure 4.15 illustrent la distribution du nombre de triplets d'amers indexés par chaque entrée dans le cas des tables de hachage $\mathcal{T}^1$ et $\mathcal{T}^4$ relatives aux paramètres L_{ab} et α_a .

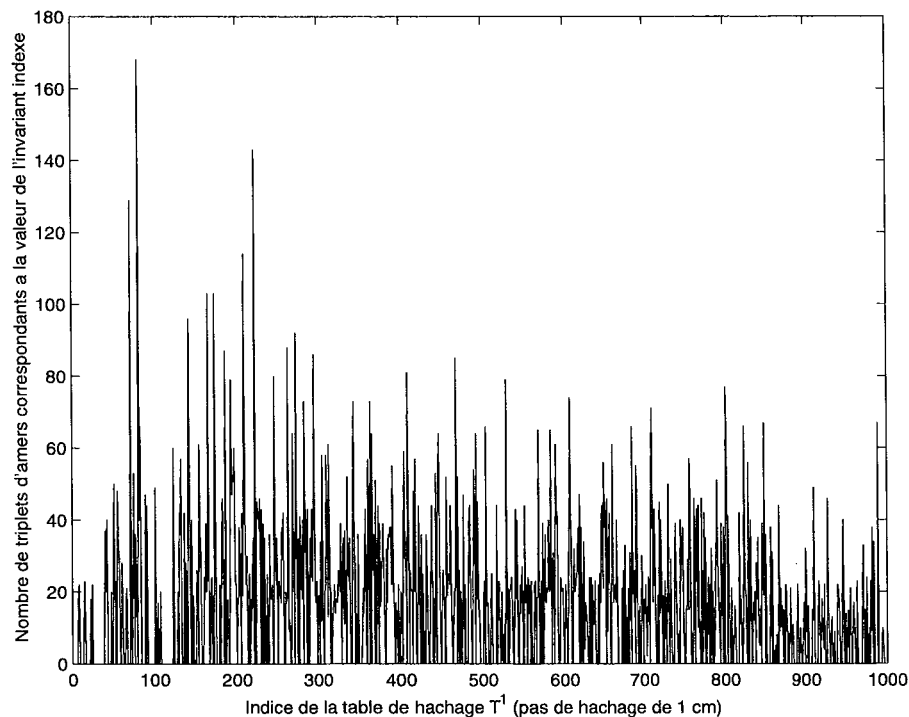


Figure 4.14 : Distribution de la table de hachage $\mathcal{T}^1$.

Le pas de hachage de 1 cm est très sélectif car il assure une répartition relativement uniforme des triplets par rapport aux indices. Le paramètre de longueur L se prête donc parfaitement à ce type de représentation.

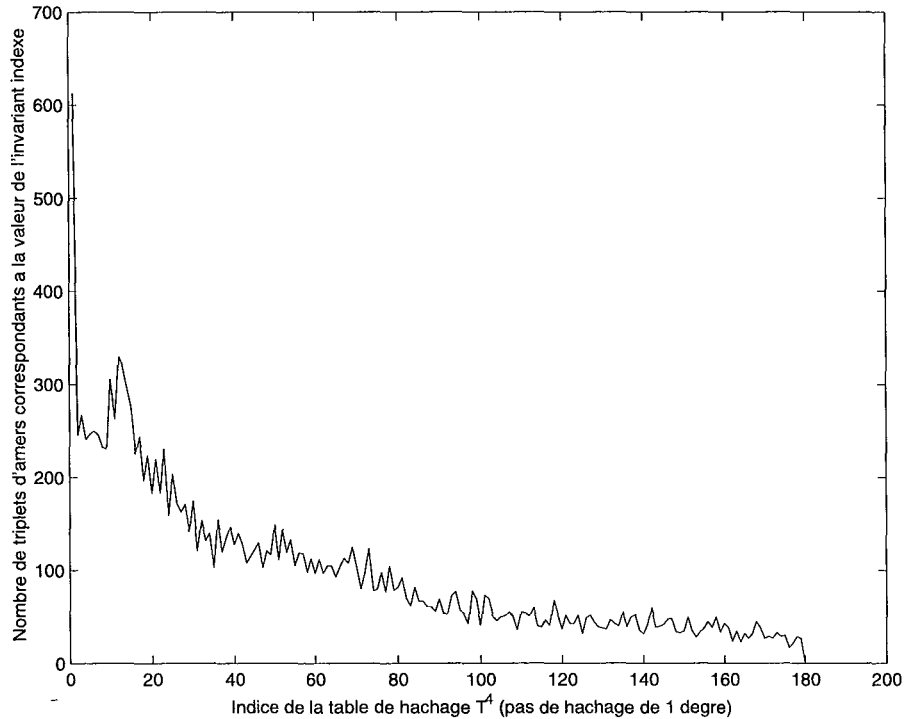


Figure 4.15 : Distribution de la table de hachage 7^4 .

Le pas de hachage de 1 degré implique un grand nombre de triplets compatibles pour chaque indice. Ceci est dû à la plage de variation limitée du descripteur invariant et implique beaucoup de votes consistants pour une même valeur de paramètre.

La mise en correspondance par table de hachage permet de voter directement pour les triplets d'amers compatibles avec la plage de valeur ($P_{a\{ijk\}}^{\min}[p] ; P_{a\{ijk\}}^{\max}[p]$) du paramètre invariant p caractérisant un triplet d'arêtes observées (Figure 4.16). Le nombre d'opérations nécessaires est lié à l'incertitude sur la mesure divisée par le pas de hachage, ainsi qu'à l'accès aux indices de la table.

La procédure de recherche complète consiste donc pour chaque triplet d'arêtes observées $\{a_i a_j a_k\}$ à analyser successivement les 6 tables de hachage. Seuls les triplets d'amers ayant reçu 6 votes favorables sont considérés comme possédant une configuration géométrique compatible avec celle du triplet d'arêtes considéré. Les tables de hachage étant toutes calculées *hors ligne*, seule la recherche des appariements est réalisée *en ligne*. La mise en correspondance est donc extrêmement rapide dès lors que les descripteurs invariants sont suffisamment sélectifs. La recherche doit toutefois se faire pour les 6 tables avant de savoir s'il existe ou non au moins un triplet global compatible. Les paramètres Ω et *statut* des triplets d'amers candidats sont ensuite comparés avec ceux du triplet d'arêtes considéré et seuls les appariements compatibles sont conservés dans un ensemble $\mathcal{S}$ constitué de paires $\langle \{A_a A_b A_c\}, \{a_i a_j a_k\} \rangle$. L'algorithme de mise en correspondance est présenté Figure 4.17.

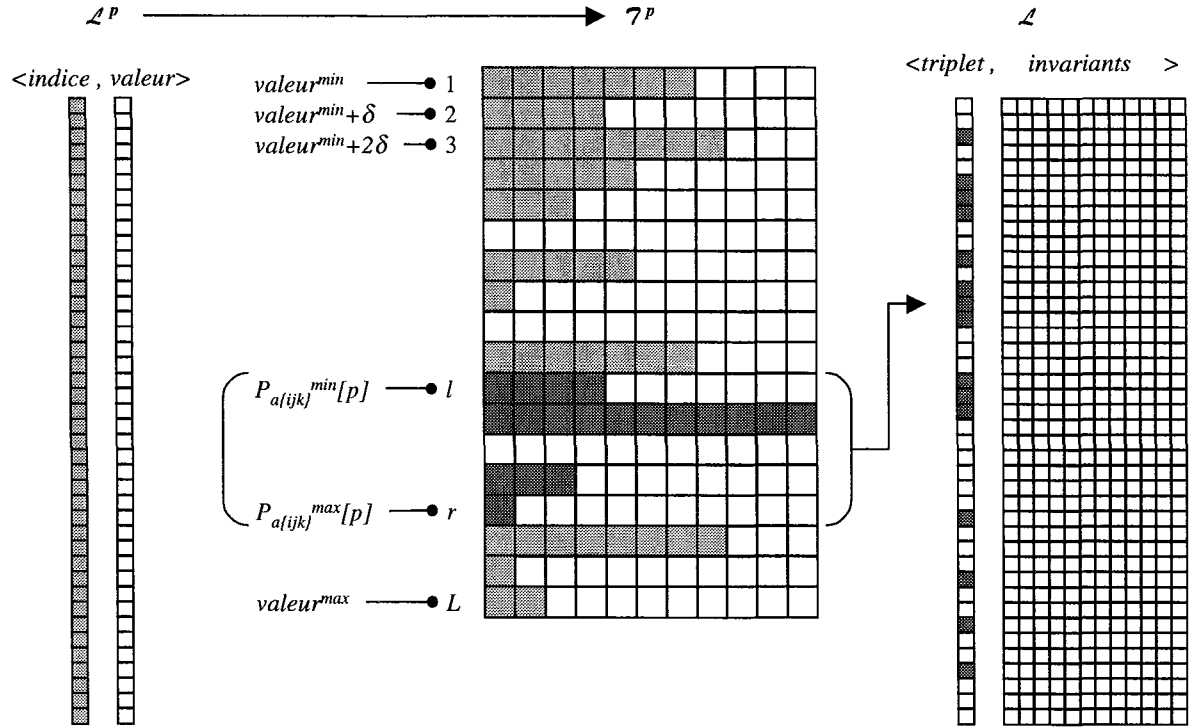


Figure 4.16 : Table de hachage relative à un descripteur invariant.

Pour chaque triplet d'arêtes observées $\{a_i, a_j, a_k\}$

Pour $p = 1, \dots, 6$

Calculer le pointeur inf de la table 7^p correspondant à $P_{a(ijk)}^{min}[p]$

Calculer le pointeur sup de la table 7^p correspondant à $P_{a(ijk)}^{max}[p]$

Pour chaque indice de 7^p pointeur, pointeur $\in [inf, sup]$

Marquer les triplets $\{A_a, A_b, A_c\}$ pointés par $\mathcal{L}_{indice}$

Fin

Fin

Pour chaque triplet d'amers $\{A_a, A_b, A_c\}$ ayant été marqué 6 fois

Ajouter la paire $\langle \{A_a, A_b, A_c\}, \{a_i, a_j, a_k\} \rangle$ à l'ensemble $\mathcal{S}$ si :

$$\Omega_i^{max} \geq \Omega_a^{min}$$

$$\Omega_i^{min} \leq \Omega_a^{max}$$

$$statut_{ij} \equiv statut_{ab}$$

$$\Omega_j^{max} \geq \Omega_b^{min}$$

$$\Omega_j^{min} \leq \Omega_b^{max}$$

$$statut_{jk} \equiv statut_{bc}$$

$$\Omega_k^{max} \geq \Omega_c^{min}$$

$$\Omega_k^{min} \leq \Omega_c^{max}$$

$$statut_{ki} \equiv statut_{ca}$$

Fin

Fin

Figure 4.17 : Algorithme de mise en correspondance de triplets par tables de hachage.

4.3.1.2 Listes ordonnées

La représentation des descripteurs invariants sous forme de listes ordonnées permet de réduire le volume des triplets d'amers à traiter au fur et à mesure de la recherche. Chaque liste $\mathcal{L}^p$ est donc triée par ordre croissant du second élément de chaque paire.

Soit $(P_{a\{ijk\}}^{\min}[p], P_{a\{ijk\}}^{\max}[p])$ la plage de variation de l'invariant p caractérisant un triplet d'arêtes observées. Nous recherchons les pointeurs *inf* et *sup* de $\mathcal{L}^p$ tels que les champs *indice* de $\mathcal{L}^p_{inf,...,sup}$ correspondent aux pointeurs dans $\mathcal{L}$ des triplets d'amers compatibles.

Le recours à un algorithme de recherche binaire assure un temps de mise en correspondance réduit. Il consiste à déterminer la position de l'indice compatible en divisant successivement par deux la longueur de l'intervalle dans lequel l'indice peut se trouver. L'algorithme de recherche binaire décrit ci-après est illustré par la Figure 4.18.

1 Recherche du pointeur *inf* :

- 1.1 *début* = 1 et *fin* = q .
- 1.2 *pointeurmin* = (*début* + *fin*) / 2.
- 1.3 Si *début* > *fin*, alors il n'y a pas de triplets d'amers compatibles.
- 1.4 Si le champ *valeur* pointé par $\mathcal{L}^p_{pointeurmin}$ est supérieur ou égal à $P_{a\{ijk\}}^{\min}[p]$, et si le champ *valeur* pointé par $\mathcal{L}^p_{pointeurmin-1}$ est inférieur à $P_{a\{ijk\}}^{\min}[p]$, alors *inf* est égal à *pointeurmin* (saut au pas 2).
- 1.5 Si le champ *valeur* pointé par $\mathcal{L}^p_{pointeurmin}$ est inférieur à $P_{a\{ijk\}}^{\min}[p]$, alors *début* est égal à *pointeurmin* + 1 et retour au pas 1.2.
- 1.6 Si le champ *valeur* pointé par $\mathcal{L}^p_{pointeurmin}$ est supérieur ou égal à $P_{a\{ijk\}}^{\min}[p]$, alors *fin* est égal à *pointeurmin* - 1 et retour au pas 1.2.

2 Recherche du pointeur *sup* :

- 2.1 *début* = *pointeurmin* et *fin* = q .
- 2.2 *pointeurmax* = (*début* + *fin*) / 2.
- 2.3 Si le champ *valeur* pointé par $\mathcal{L}^p_{pointeurmax}$ est inférieur ou égal à $P_{a\{ijk\}}^{\max}[p]$, et si le champ *valeur* pointé par $\mathcal{L}^p_{pointeurmax+1}$ est supérieur à $P_{a\{ijk\}}^{\max}[p]$, alors *sup* est égal à *pointeurmax* (arrêt de la recherche)
- 2.4 Si le champ *valeur* pointé par $\mathcal{L}^p_{pointeurmax}$ est inférieur ou égal à $P_{a\{ijk\}}^{\max}[p]$, alors *début* est égal à *pointeurmax* + 1 et retour au pas 2.2.
- 2.5 Si le champ *valeur* pointé par $\mathcal{L}^p_{pointeurmax}$ est supérieur à $P_{a\{ijk\}}^{\max}[p]$, alors *fin* est égal à *pointeurmax* - 1 et retour au pas 2.2.

Le nombre de comparaisons maximum permettant de trouver le pointeur *inf* se produit lorsque la valeur de paramètre recherchée est trouvée lors de la dernière comparaison (intervalle de longueur 1), ce qui correspond à un total de $O(\log_2(q))$ opérations. Ce nombre maximum d'opérations est indépendant du nombre de triplets compatibles ainsi que de l'incertitude sur la valeur du paramètre. Ainsi, avec une longueur de liste q égale à 23880, les indices *inf* et *sup* sont trouvés en un maximum de $2 * \log_2(q) = 28$ comparaisons.

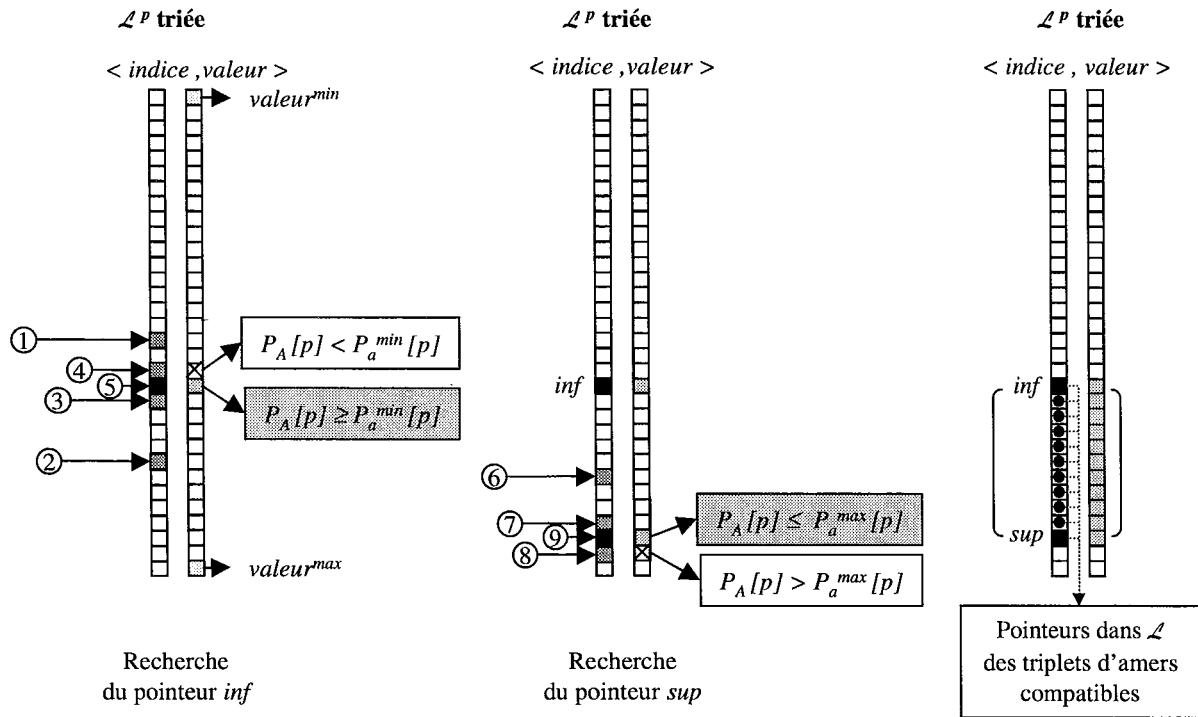


Figure 4.18 : Recherche binaire dans une liste ordonnée.

La procédure de mise en correspondance consiste pour chaque paramètre $P_{a\{ijk\}}[p], p=1 \dots 6$ du triplet local considéré, à rechercher successivement dans la liste $\mathcal{L}^p$ correspondante les indices des triplets d'amers dont la valeur de paramètre est comprise entre $P_{a\{ijk\}}^{\min}[p]$ et $P_{a\{ijk\}}^{\max}[p]$.

Afin d'accélérer la recherche des indices, l'analyse de la liste courante n'est réalisée que pour les triplets d'indices retenus dans la liste précédente. Chaque descripteur testé permettant de diminuer la taille de la liste résultante, le temps perdu à trier les listes est compensé par la vitesse d'analyse des listes réduites. Seule la liste $\mathcal{L}^1$ est donc créée *hors ligne*, les 5 autres listes étant calculées *en ligne*.

La Figure 4.19 illustre la procédure de recherche des listes ordonnées.

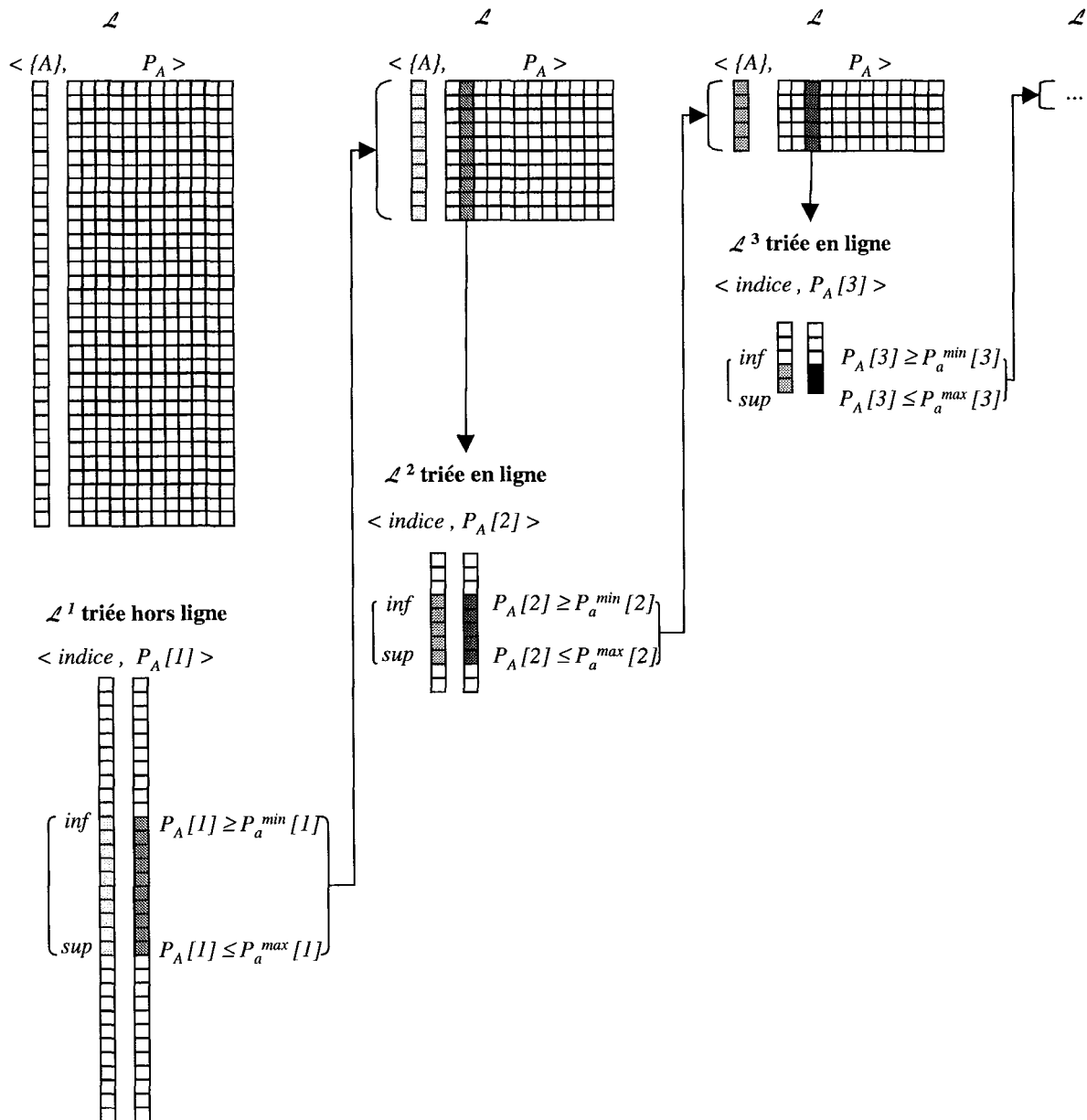


Figure 4.19 : Mise en correspondance de triplets par listes ordonnées.

Afin d'optimiser cette implémentation, il est nécessaire de déterminer le paramètre invariant le plus discriminant du point de vue des appariements afin de réduire au maximum la longueur de la liste au premier passage. La Figure 4.20 donne pour les descripteurs invariants L et α , la longueur moyenne de la liste $\mathcal{L}^2$ résultante. La longueur L des côtés du triangle correspond au paramètre le plus discriminant puisqu'il permet une réduction d'un facteur de 21,6 de la longueur de la liste des triplets d'amers candidats à l'appariement.

La Figure 4.21 donne la longueur moyenne de la liste $\mathcal{L}$ courante pour les 6 premiers paramètres invariants analysés dans l'ordre : L_{ab} L_{ba} L_{ca} α_a α_b α_c .

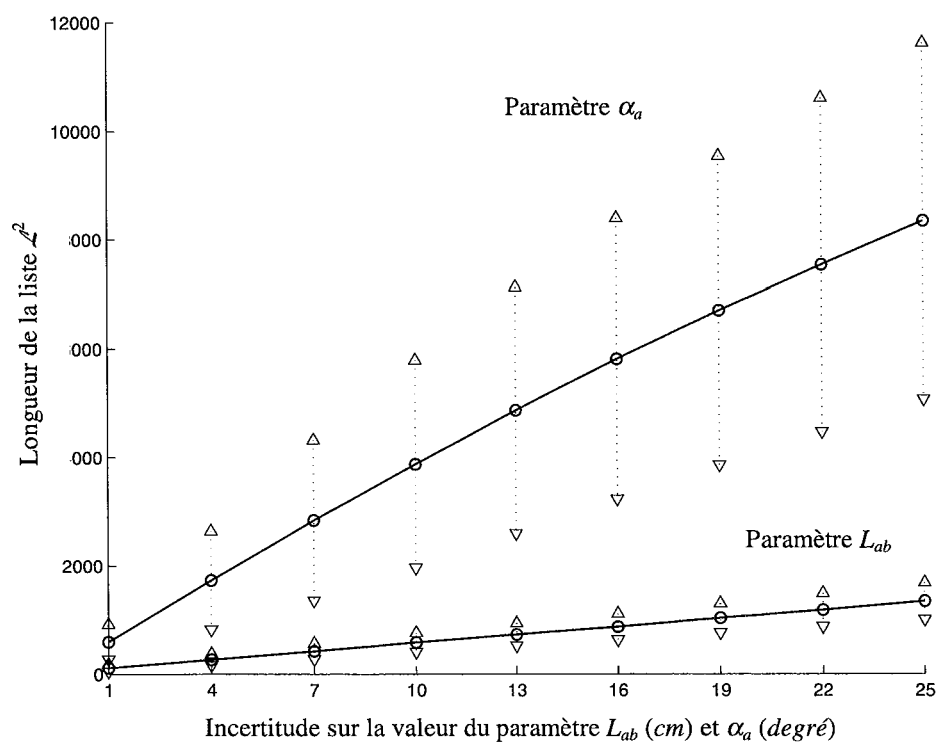


Figure 4.20 : Pouvoir discriminant du premier critère testé.

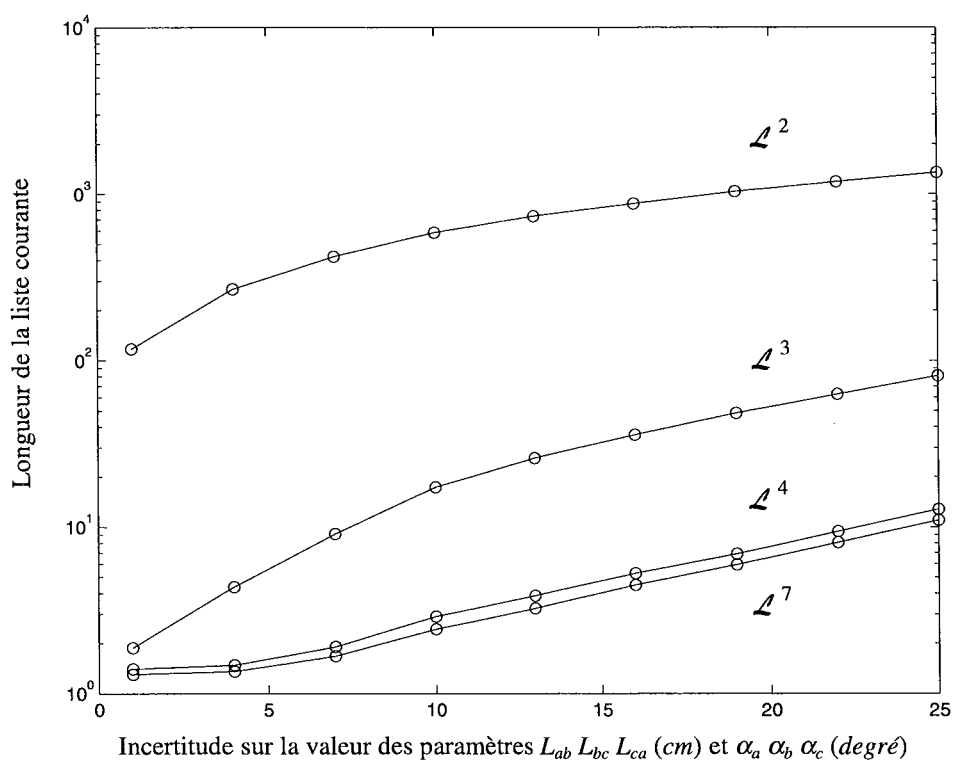


Figure 4.21 : Restriction de la liste L courante au fur et à mesure de l'analyse.

Un triplet d'amers est apparié au triplet d'arêtes considéré si et seulement si son indice a été retenu dans la dernière liste. Les paramètres Ω et *statut* des triplets d'amers candidats sont ensuite comparés avec ceux du triplet d'arêtes considéré et seuls les appariements compatibles sont conservés dans un ensemble $\mathcal{S}$ constitué de paires $\langle \{A_a A_b A_c\}, \{a_i a_j a_k\} \rangle$. L'algorithme est donné Figure 4.22.

```

Pour chaque triplet d'arêtes observées  $\{a_i a_j a_k\}$ 
   $\mathcal{L}$  : liste définissant le modèle global
  Pour  $p = 1, \dots, 6$ 
    Calculer et trier  $\mathcal{L}^p$  de  $\mathcal{L}$  ( $\mathcal{L}^1$  a été calculée hors ligne)
    Trouver le plus petit pointeur inf tel que : champ valeur de  $\mathcal{L}_{inf}^p \geq P_{a\{ijk\}}^{min}[p]$ 
    Trouver le plus grand pointeur sup tel que : champ valeur de  $\mathcal{L}_{sup}^p \leq P_{a\{ijk\}}^{max}[p]$ 
    Pour chaque paire  $\langle indice, valeur \rangle$  de  $\mathcal{L}_{pointeur}^p$ ,  $pointeur \in \{inf, sup\}$ 
      Ne conserver dans  $\mathcal{L}$  que les paires  $\langle \{A_a A_b A_c\}, P_{A\{abc\}} \rangle$  pointées par  $\mathcal{L}_{indice}$ 
    Fin
  Fin

  Pour chaque triplet d'amers  $\{A_a A_b A_c\}$  de  $\mathcal{L}$ 
    Ajouter la paire  $\langle \{A_a A_b A_c\}, \{a_i a_j a_k\} \rangle$  à l'ensemble  $\mathcal{S}$  si :
       $\Omega_i^{max} \geq \Omega_a^{min}$        $\Omega_i^{min} \leq \Omega_a^{max}$        $statut_{ij} \equiv statut_{ab}$ 
       $\Omega_j^{max} \geq \Omega_b^{min}$        $\Omega_j^{min} \leq \Omega_b^{max}$        $statut_{jk} \equiv statut_{bc}$ 
       $\Omega_k^{max} \geq \Omega_c^{min}$        $\Omega_k^{min} \leq \Omega_c^{max}$        $statut_{ki} \equiv statut_{ca}$ 
    Fin
  Fin

```

Figure 4.22 : Algorithme de mise en correspondance de triplets par listes ordonnées.

A la différence de l'implémentation sous forme de table de hachage qui nécessite impérativement l'analyse des six tables $\mathcal{T}^p$ ainsi qu'un vote final, cette implémentation permet de réduire la liste des triplets compatibles au fur et à mesure de l'analyse.

Ainsi, lorsque la liste $\mathcal{L}$ courante est vide, la recherche est arrêtée. Les triplets d'arêtes constitués à partir d'un ou plusieurs faux positifs sont donc rapidement éliminés.

De plus, la liste $\mathcal{L}$ résultante donne directement les indices des triplets d'amers compatibles.

4.3.1.3 Performances de la mise en correspondance

Le modèle global est composé de 64 amers ayant engendré 23880 triplets d'amers. Nous avons réalisé une série d'essais portant sur 50 modèles locaux, comportant chacun de 3 à 8 arêtes verticales. Nous avons ainsi une base de données de 1185 triplets locaux. Parmi ces triplets, 384 ne correspondent pas à des triplets d'amers car ils ont été générés à partir d'un ou plusieurs faux positifs.

La Figure 4.23 illustre le temps de recherche des triplets d'amers compatibles avec le descripteur invariant L_{ij} en fonction de l'incertitude sur la valeur du paramètre.

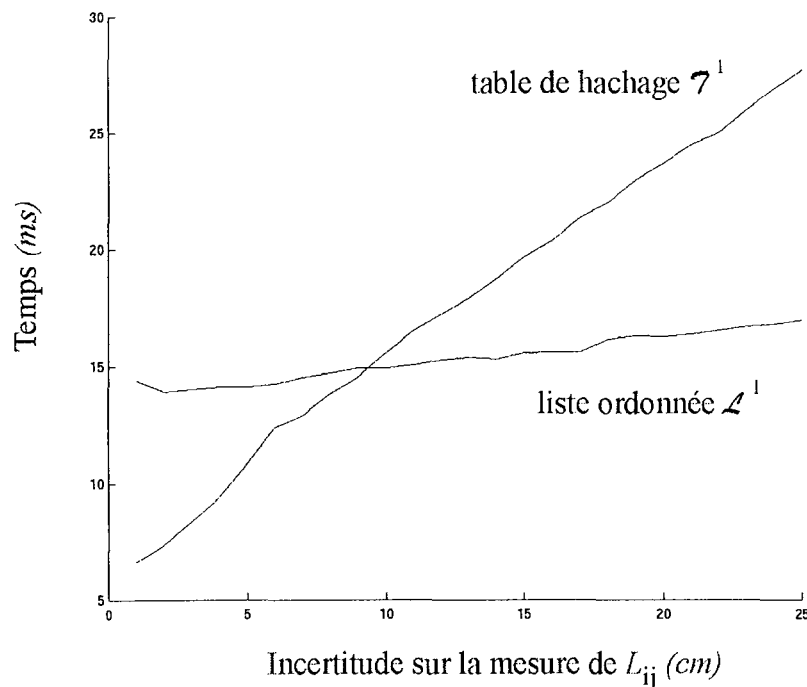


Figure 4.23 : Temps de recherche moyen lié au descripteur invariant L_{ij} .

L'implémentation par table de hachage est particulièrement rapide dès lors que l'incertitude est relativement faible. Le temps de recherche croît linéairement avec l'incertitude puisque l'on accède à un nombre de lignes fonction du pas de hachage de 1 cm.

L'implémentation par listes ordonnées assure un temps de calcul sensiblement constant car la recherche est fonction de la longueur de la liste (23880 triplets) et non pas de l'incertitude. Nous pouvons constater qu'à partir d'une incertitude de 10 cm, l'implémentation sous forme de listes ordonnées devient plus intéressante.

La Figure 4.24 illustre le temps de recherche des triplets d'amers compatibles avec les 6 premiers paramètres invariant L_{ij} L_{jk} L_{ki} α_i α_j α_k .

L'implémentation par tables de hachage est très performante pour les paramètres de longueur L (indices 1 à 3) mais devient moins rapide pour les paramètres d'angle α (indices 4 à 6). Ceci s'explique par le fait qu'une discrétisation par un pas de hachage de 1° n'est pas suffisamment sélective au niveau des entrées : il y a beaucoup de triplets compatibles avec une même valeur d'invariant. Le temps correspondant à l'indice 6 comprend la recherche de l'invariant ainsi que la mise en commun des votes permettant de déterminer les triplets compatibles.

L'implémentation par liste assure un temps de calcul extrêmement réduit dès le 3^{ème} paramètre testé. La longueur de la liste courante est alors suffisamment faible pour obtenir un temps supplémentaire sensiblement constant. Le deuxième avantage est que la liste finale contient les triplets d'amers compatibles et ne nécessite par conséquent aucun vote supplémentaire.

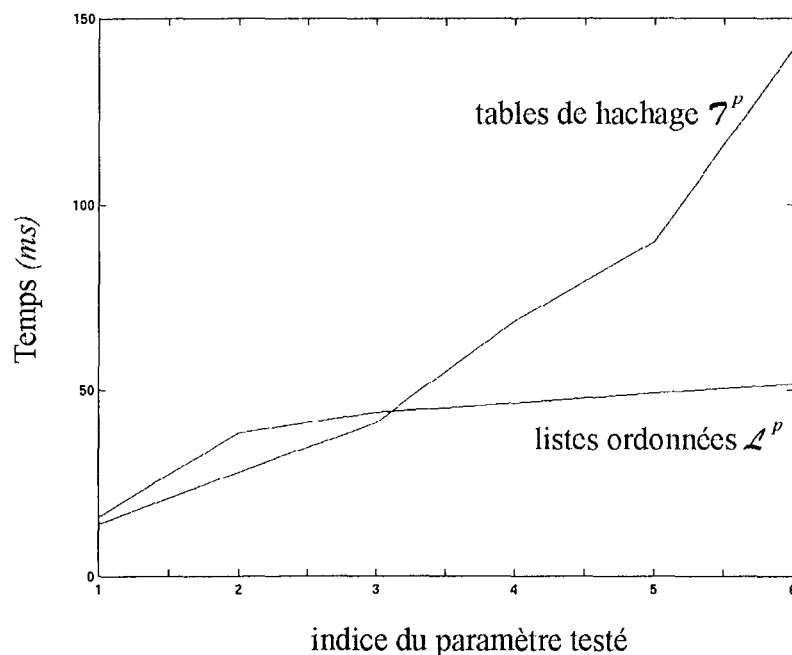


Figure 4.24 : Temps de recherche des triplets d'amers pour les 6 premiers invariants.

La qualité du système d'indexation peut être jugée à la réduction des prédictions qu'il réalise : la méthode de reconnaissance apparie en moyenne 2,8 triplets d'amers à chaque triplet d'arêtes observées (écart type 2,06).

4.3.2 Appariements de primitives

Nous construisons à partir de l'ensemble $\mathcal{S}$ des mises en correspondance de triplets d'arêtes observées $\{a_i a_j a_k\}$ et d'amers $\{A_a A_b A_c\}$, le tableau T_{triplets} faisant correspondre pour chaque triplet d'amers $\{A_a A_b A_c\}$ les arêtes observées a appariées.

L'amer le plus souvent mis en correspondance avec une même arête observée est considéré comme étant le meilleur candidat à l'appariement. Nous fixons pour cela une butée app_{mini} représentant le nombre minimum de mises en correspondance devant être réalisées entre une arête observée a et un amer A afin de valider cet appariement.

Soit m , le nombre d'arêtes observées. Si chaque arête observée correspond à un amer du modèle global et si tous les triplets possibles d'amers ont bien été mis en correspondance, alors chaque arête observée a devrait apparaître dans $app_{\text{mini}} = (m-1)(m-2)/2$ triplets d'amers.

Il est toutefois possible que le nombre de triplets d'amers réellement appariés soit inférieur à app_{mini} . En effet, une ou plusieurs arêtes peuvent être des faux positifs et certains triplets d'amers peuvent ne pas avoir été appariés du fait d'erreurs de modélisation des arêtes observées ou des amers.

La procédure d'appariement donnée ci-après est illustrée par la Figure 4.25.

1. m = nombre d'arêtes observées.
2. $T = T_{\text{triplets}}$, $app_{\text{mini}} = (m-1)(m-2)/2$
3. Pour chaque arête observée a , éliminer du tableau T les amers A n'ayant pas été appariés au moins app_{mini} fois.
4. Éliminer du tableau T toutes les lignes contenant moins de 3 amers. On ne conserve ainsi que les triplets viables.
5. Le tableau T donne pour chaque arête a le ou les amers A susceptibles d'être appariés. Déterminer alors pour chaque arête observée a , le nombre d'occurrence de l'amer A apparié.

Si le tableau T résultant est vide, alors retour au pas 2 avec $m = m-1$.

Du fait de l'existence de configurations de triplets d'amers semblables, il est possible que plusieurs amers A soient appariés à une même arête observée a . Dans ce cas, puisqu'il est impossible de déterminer un appariement unique, il est nécessaire d'éliminer du tableau T les lignes faisant référence à ces appariements.

4.4 DETERMINATION DE LA POSITION ET DE L'ORIENTATION DU ROBOT

A chaque arête observée a_i , dont les coordonnées sont exprimées dans le repère R_{robot} , est apparié un amer A_a dont les coordonnées (xA_a, yA_a) sont exprimées dans le repère R_{env} . Déterminer la localisation du robot consiste à trouver le vecteur Γ des paramètres (x_e, y_e, Θ_e) permettant de superposer les arêtes observées aux amers environnementaux.

Nous proposons deux méthodes de calcul, l'une reposant sur la minimisation d'un critère de distance au sens des moindres carrés, et l'autre sur l'exploitation des régions d'incertitude définissant chaque arête.

4.4.1 Estimation par les moindres carrés

Nous définissons les coordonnées moyennes (xa_i, ya_i) de l'arête observée a_i comme étant le centre de gravité de la région d'incertitude définie par les points $a_{i,1}$, $a_{i,2}$, $a_{i,3}$ et $a_{i,4}$.

L'estimation du vecteur Γ des paramètres (x_e, y_e, Θ_e) est alors réalisée en minimisant la distance séparant chaque arête de l'amer qui lui est apparié.

Nous déterminons tout d'abord par trilatération une première localisation estimée à partir des deux appariements $\langle a_i, A_a \rangle$ et $\langle a_j, A_b \rangle$ présentant la plus grande occurrence (Figure 4.26).

$$\begin{cases} \tilde{x}_e = xA_b + da_j \cos(\alpha + \beta) \\ \tilde{y}_e = yA_b + da_j \sin(\alpha + \beta) \\ \tilde{\Theta}_e = \phi a_i - \vartheta a_j \end{cases} \quad (4.5)$$

Avec :

- α , angle formé par l'axe des abscisses X_e et la droite passant par les amers A_a et A_b .
- ϕa_i , angle formé par l'axe des abscisses X_e et la droite passant par l'amer A_a et l'origine du repère R_{robot} de coordonnées $(\tilde{x}_e, \tilde{y}_e)$.
- $[da_i, \vartheta a_i]$ et $[da_j, \vartheta a_j]$, coordonnées polaires des arêtes a_i et a_j dans le repère R_{robot} .
- $\beta = \arccos\left(\frac{da_{ij}^2 + da_j^2 - da_i^2}{2 \times da_{ij} \times da_j}\right)$ et $da_{ij} = \sqrt{(xa_i - xa_j)^2 + (ya_i - ya_j)^2}$

Cette méthode de calcul est très rapide et donne généralement une bonne localisation estimée du robot. Toutefois, elle ne permet pas de déterminer si les appariements réalisés sont bien consistants les uns avec les autres. En effet, un minimum est toujours trouvé et seule la définition de seuils permet de valider ou d'invalidier la localisation calculée. De plus, la formulation par les moindres carrés ne fournit pas d'indication quand à la précision de la localisation.

4.4.2 Recours à l'incertitude des mesures

La modélisation des erreurs de mesure sous la forme de régions d'incertitude permet la propagation de cette incertitude sur le vecteur Γ des paramètres (x_e, y_e, Θ_e) . Il est ainsi possible de définir des bornes quand à l'erreur de localisation en définissant l'ensemble $(x_e^*, y_e^*, \Theta_e^*)$ des paramètres compatibles avec les incertitudes de mesure des arêtes observées.

Soit deux arêtes observées a_i et a_j appariées aux amers A_a et A_b . Nous supposons dans un premier temps qu'il n'y a aucune erreur de mesure sur les arêtes observées. Déterminer la localisation du robot consiste alors à rechercher la rotation Θ_e de R_{robot} qui oriente le segment $[a_i a_j]$ parallèlement au segment $[A_a A_b]$, puis à déterminer les translations x_e et y_e qui superposent les points extrémités des segments. En étendant ce raisonnement aux régions d'incertitudes des arêtes observées, il est possible de calculer les positions extrêmes du robot compatibles avec les mesures.

Les angles extrêmes de rotation Θ_e^{min} et Θ_e^{max} compatibles avec les régions d'incertitudes liées aux deux arêtes a_i et a_j se produisent lorsqu'un segment de longueur $d = \|A_a - A_b\|$ est placé de telle sorte que l'une de ses extrémités soit confondue avec l'un des points $a_{i,n}$ définissant la région liée à a_i , et l'autre extrémité appartienne à l'un des segments $[a_{j,m} a_{j,l}]$ définissant la région liée à a_j , et vice versa (Figure 4.27).

Ce point d'intersection est déterminé en recherchant le paramètre $\lambda \in [0,1]$ satisfaisant à l'équation $\|a_{j,l} + \lambda(a_{j,m} - a_{j,l}) - a_{i,n}\| = d$.

$$\lambda = \frac{-B \pm \sqrt{\Delta}}{2A} \quad \text{avec} \quad \begin{cases} A = (x_{a_{j,m}} - x_{a_{j,l}})^2 + (y_{a_{j,m}} - y_{a_{j,l}})^2 \\ B = 2[(x_{a_{j,l}} - x_{a_{i,n}})(x_{a_{j,m}} - x_{a_{j,l}}) + (y_{a_{j,l}} - y_{a_{i,n}})(y_{a_{j,m}} - y_{a_{j,l}})] \\ C = (x_{a_{j,l}} - x_{a_{i,n}})^2 + (y_{a_{j,l}} - y_{a_{i,n}})^2 - d^2 \\ \Delta = B^2 - 4AC \quad (\lambda \text{ est calculé si } \Delta \geq 0) \end{cases}$$

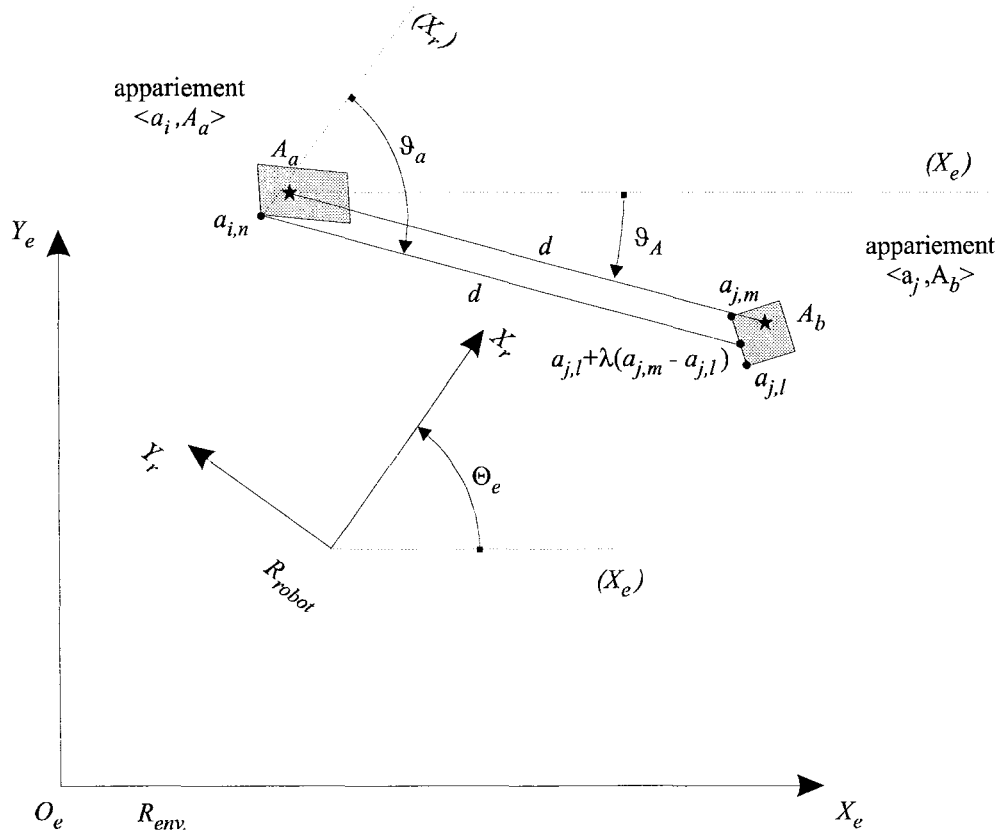


Figure 4.27 : Détermination d'une orientation Θ_e possible du robot.

L'angle Θ_e permettant d'orienter le segment $[a_{i,n} a_{j,l} + \lambda(a_{j,m} - a_{j,l})]$ parallèlement au segment $[A_a A_b]$ est alors donné par :

$$\Theta_e = \vartheta_A - \vartheta_a \quad \text{avec} \quad \begin{cases} \vartheta_a = \text{atan} \left(\frac{ya_{j,l} - ya_{i,n} + \lambda(ya_{j,m} - ya_{j,l})}{xa_{j,l} - xa_{i,n} + \lambda(xa_{j,m} - xa_{j,l})} \right) \\ \vartheta_A = \text{atan} \left(\frac{yA_b - yA_a}{xA_b - xA_a} \right) \end{cases} \quad (4.8)$$

A chaque paire d'arêtes a_i et a_j correspond un maximum de $4 \times 4 \times 2 = 32$ valeurs de Θ_e pour lesquelles nous définissons l'intervalle minimum $\{\Theta_e^{\min}, \Theta_e^{\max}\}$.

L'intersection des intervalles propres à chaque paire d'arêtes observées appariées donne alors l'ensemble Θ_e^* des orientations possibles du robot.

Nous déterminons alors pour chaque orientation Θ_e appartenant à l'ensemble Θ_e^* l'ensemble des positions (x_e^*, y_e^*) du robot (Le pas de discrétisation est de $0,2^\circ$).

Pour chacun des quatre points $a_{i,n}$ définissant la région d'incertitude liée à l'arête observée a_i , la position (x_e, y_e) du robot dans R_{env} est donnée par :

$$\begin{cases} x_e = xA_a + da_{i,n} \cos(\theta a_{i,n} + \Theta_e) \\ y_e = yA_a + da_{i,n} \sin(\theta a_{i,n} + \Theta_e) \end{cases} \quad (4.9)$$

Ces quatre points délimitent un polygone comprenant l'ensemble des positions du robot compatibles avec l'appariement $\langle a_i, A_a \rangle$. L'intersection des polygones correspondant à chaque appariement donne alors l'ensemble des positions (x_e^*, y_e^*) pour l'orientation Θ_e considérée (Figure 4.28).

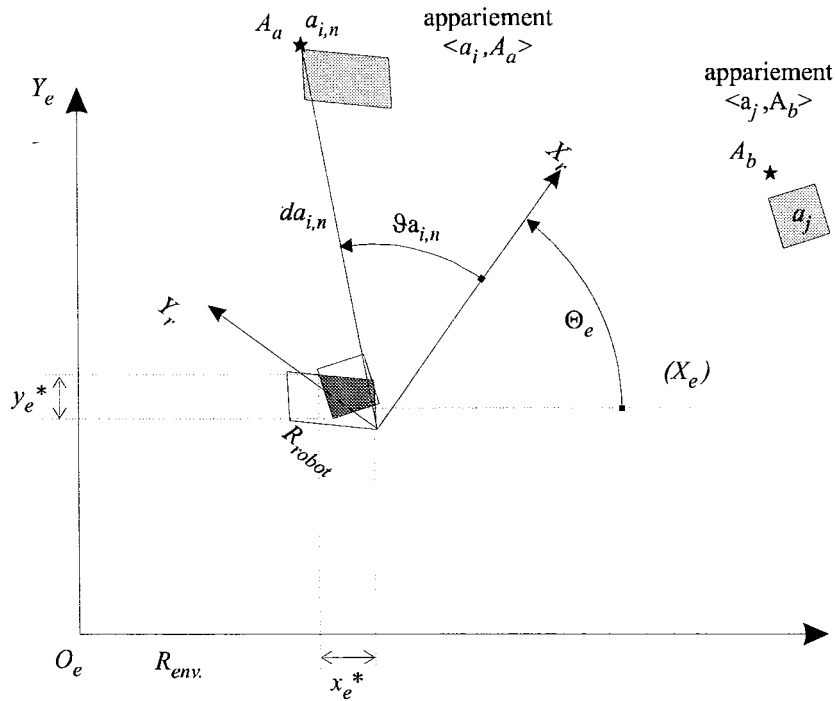


Figure 4.28 : Détermination d'un ensemble de positions (x_e^*, y_e^*) possibles du robot.

L'intersection de deux polygones convexes donne un polygone convexe. Il s'avère que le nombre de points décrivant l'intersection de deux polygones est souvent plus élevé que le nombre de points des deux opérandes. Ainsi, des opérations répétées conduisent à des calculs de plus en plus complexes (Figure 4.29).

Une solution consiste à approcher les polygones d'incertitude par des rectangles. Cette représentation basée sur les intervalles nécessite ainsi un nombre constant d'opérations. Toutefois, les résultats conduisent à une estimation pessimiste de l'ensemble des localisations possibles du robot.

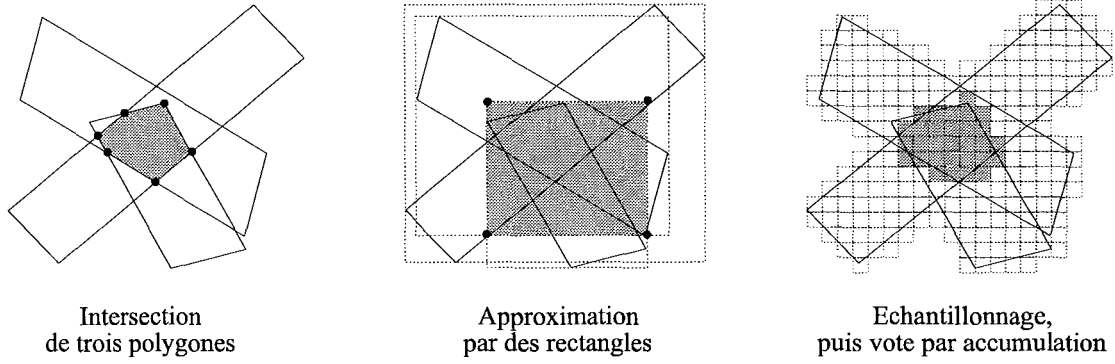


Figure 4.29 : Intersection de plusieurs polygones.

Les régions d'incertitude étant relativement petites, nous proposons de réaliser un échantillonnage des polygones avec un pas de 1 cm, puis de déterminer par un système de vote les cellules communes aux différents polygones.

L'union des cellules propres à chaque orientation Θ_e définit ainsi un ensemble de positions possibles du robot que nous approchons par un rectangle afin de déterminer les vecteurs Γ_{moyen} $[x_e \ y_e \ \Theta_e]$ et $\Gamma_{incertitude}$ $[\delta x_e \ \delta y_e \ \delta \Theta_e]$ délimitant la plage de variation des paramètres de localisation (Figure 4.30).

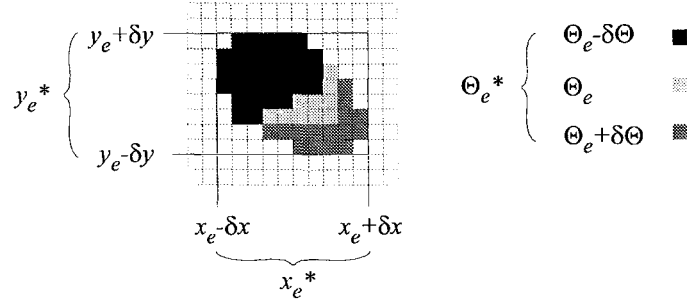


Figure 4.30 : Approximation de l'ensemble des localisations du robot.

Le temps de calcul nécessaire pour la détermination de la localisation varie de 1 à 8 secondes en fonction de l'incertitude sur les mesures ainsi que de la répartition relative des arêtes.

Cette méthode est donc plus lente que l'estimation par les moindres carrés mais permet d'une part, de définir l'ensemble des positions du robot compatibles avec les erreurs de mesures, et d'autre part ne retourne aucune valeur si les appariements réalisés ne sont pas consistants. Ainsi le problème lié à la définition de seuils heuristiques est évité.

4.4.3 Performance de la localisation

Le modèle global de l'environnement représente un couloir de 30 m sur 10 m communiquant avec différentes pièces et possédant 8 portes à simple battant et 11 portes à double battants.

La Figure 4.31 présente les 64 primitives de type arêtes verticales utilisées comme amers. Les segments fins correspondent aux murs et aux portes.

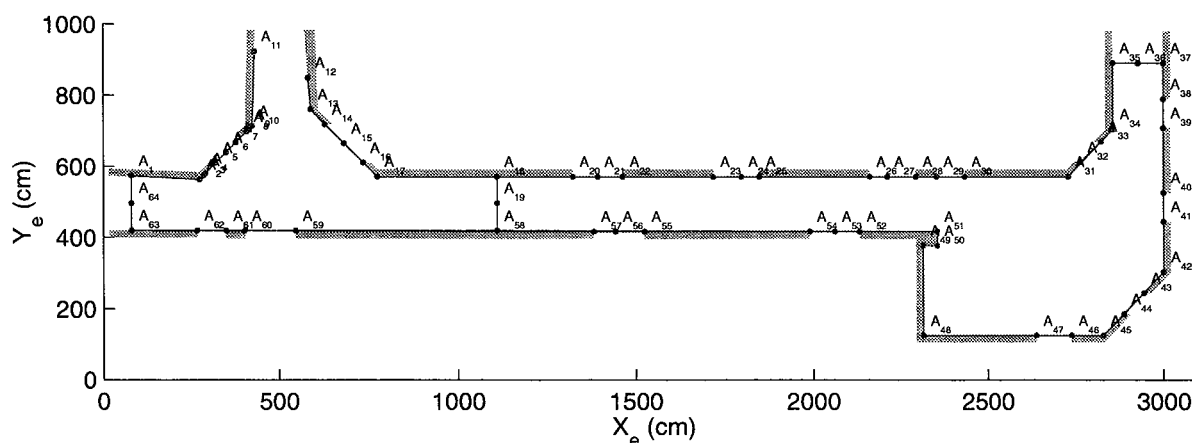


Figure 4.31 : Carte globale de l'environnement.

Nous avons réalisé l'acquisition de 50 modèles locaux pour différentes positions du robot dans son environnement.

Nous étudions dans un premier temps les limites liées à l'exploitation immédiate du modèle télémétrique, puis nous analysons l'apport de la vision. Nous concluons enfin sur les mises en échec du système de localisation présenté.

4.4.3.1 Exploitation du modèle télémétrique

Les sommets des primitives *coin* du modèle local télémétrique sont exploités directement par la méthode de mise en correspondance.

Le principal intérêt de cette démarche réside dans le gain de temps réalisé au niveau de la perception. En effet, le traitement des données télémétriques est réalisé relativement rapidement par rapport au temps nécessaire pour l'acquisition et le traitement des images.

Le Tableau 4.4 donne le temps de calcul $T_{segmentation}$ correspondant à la génération des primitives *segment* et *coin*, à la détermination de l'accessibilité du module de vision ainsi qu'à la sélection des amers potentiels.

Nombre de primitives <i>segment</i>	5	8	11	14	17	20	23	26	29	32	39
$T_{segmentation}$ moyen (s)	3,27	4,52	5,02	5,51	6,39	6,53	6,64	6,89	7,22	7,54	9,48
Ecart type de $T_{segmentation}$ (s)	0,06	0,22	0,19	0,28	0,37	0,38	0,40	0,40	0,57	0,58	0,78

Tableau 4.4 : Temps de calcul $T_{segmentation}$ (SUN SPARCstation 20).

Compte tenu de l'environnement dans lequel évolue le robot, seul un nombre restreint d'amers correspondant à des angles de mur sont susceptibles d'être perçus simultanément par le robot en un point donné de l'espace. Nous n'avons recours par conséquent qu'aux 6 premières primitives *coin* dont le critère d'intérêt $Poids_{global}$ est supérieur à 33% afin de déterminer la localisation du robot.

Le Tableau 4.5 donne le pourcentage d'occurrence des primitives *coin* dont le critère d'intérêt est supérieur à 33%.

Nombre de primitives <i>coin</i>	3	4	5	6	7	8	9	10
Occurrence (%)	97,2	87,9	69,7	48,5	39,4	27,3	6,1	3,1

Tableau 4.5 : Occurrence des primitives *coin* dans le modèle télémétrique .

L'utilisation immédiate des sommets des primitives *coin* en tant qu'arêtes observées permet dans 68,9 % des cas de déterminer la localisation correcte du robot.

Les mises en échec sont principalement dues au fait que les faux positifs ne peuvent pas être discernés par la télémétrie. Or, les triplets parasites qu'ils provoquent peuvent être appariés à des triplets d'amers du fait de l'incertitude sur les mesures. La méthode de mise en correspondance n'est alors efficace que si les primitives *coin* correspondant réellement à des amers sont plus nombreuses que les faux positif et/ou si les faux positifs ne créent pas de constellations référencées dans le modèle global.

Il est donc important de s'assurer de la viabilité des primitives sensorielles. Ceci justifie pleinement l'emploi du module de vision.

Nous pouvons envisager une stratégie de localisation reposant sur le traitement en parallèle des informations sensorielles (Figure 4.32). La localisation du robot est déterminée à partir des sommets des primitives *coin* et simultanément le module de vision réalise la recherche d'arêtes verticales. Les phases de mise en correspondance et de localisation étant nettement plus rapides que la détection des arêtes, le module de vision peut être désactivé si une localisation valide est calculée. Sinon, on poursuit l'inspection par la vision et on traite la ou les images acquises.

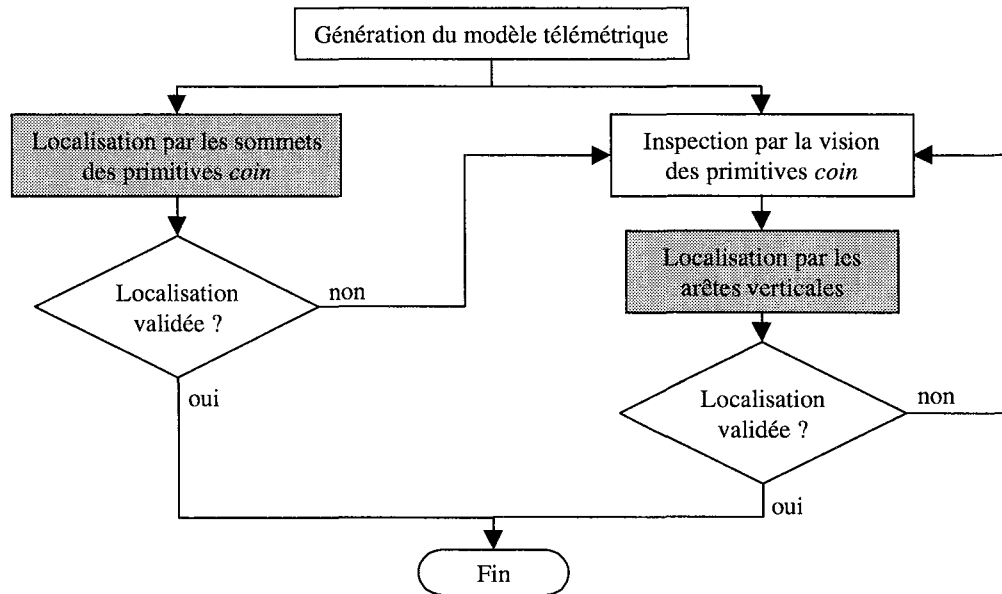


Figure 4.32 : Traitement parallèle des informations.

Le modèle local présenté Figure 4.33 illustre le résultat d'une localisation calculée uniquement à l'aide des sommets des primitives *coin* du modèle télémétrique. Bien que le sommet a_2 soit un faux positif, les appariements ont été correctement réalisés.

Le modèle local présenté Figure 4.34 illustre les limites de fiabilité des primitives *coin*. Les arêtes a_3 , a_4 et a_6 ne correspondant pas à des amers, seul le triplet $\{a_1, a_2, a_5\}$ est valide. Or les triplets générés par les faux positifs ont été mis en correspondance avec des triplets d'amers du fait des incertitudes liées aux mesures. Les appariements résultants étant erronés et inconsistants les uns avec les autres, l'ensemble des localisations $(x_e^*, y_e^*, \Theta_e^*)$ possibles du robot est un ensemble vide. La figure de droite présente la localisation déterminée par la méthode des moindres carrés. Les écarts de distance δD et d'angle $\delta \Phi$ résiduels entre amers et arêtes appariées étant supérieurs aux seuils fixés, la localisation calculée est là aussi invalidée.

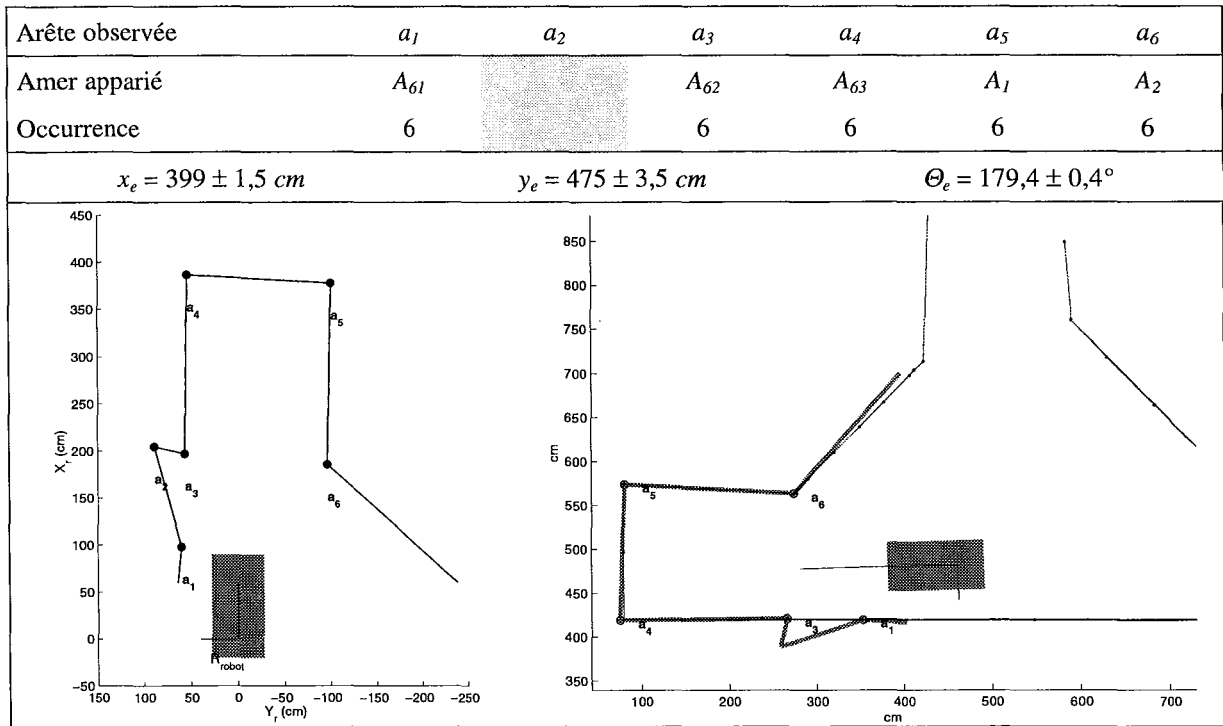


Figure 4.33 : Modèle local A et localisation correspondante.

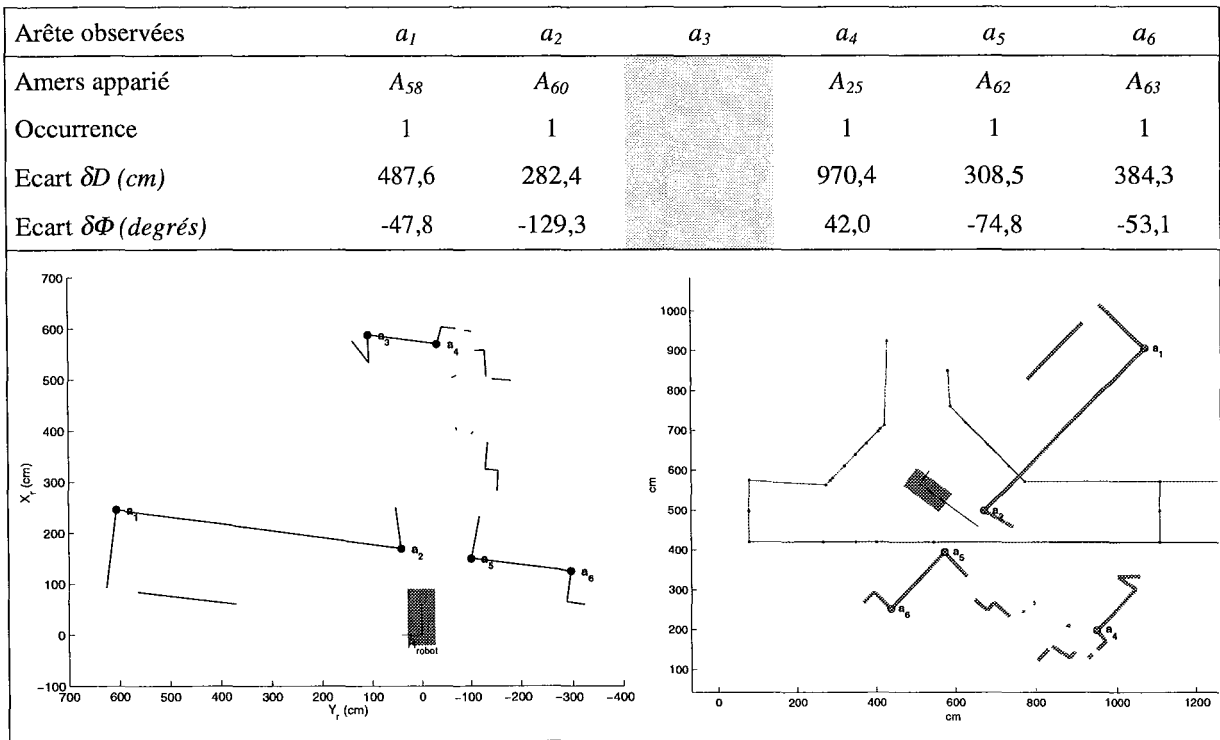


Figure 4.34 : Modèle local B et localisation correspondante.

4.4.3.2 Apport de la vision

L'emploi du module de vision s'avère nécessaire lorsqu'il y a peu de primitives *coin*, et/ou lorsque beaucoup de primitives *coin* sont issues d'objets non référencés (c'est-à-dire que la localisation initiale par télémétrie a échoué).

L'inspection des primitives *coin* par le module de vision permet en effet de détecter des arêtes que la télémétrie n'a pas été capable de discerner et d'éliminer la plupart des faux positifs du modèle télémétrique.

Le modèle local présenté Figure 4.35 illustre ainsi le cas où les arêtes verticales mises en évidence conduisent à une localisation correcte du robot, là où l'utilisation directe des primitives *coin* a échoué (cf. modèle télémétrique précédent). Seule l'arête a_3 est un faux positif causé par l'extrémité non référencée d'une porte. Les occurrences des appariements sont relativement faibles car ce faux positif a engendré des triplets compatibles avec des triplets d'amers. Toutefois, les arêtes observées référencées sont suffisamment nombreuses pour éliminer ces appariements fantômes.

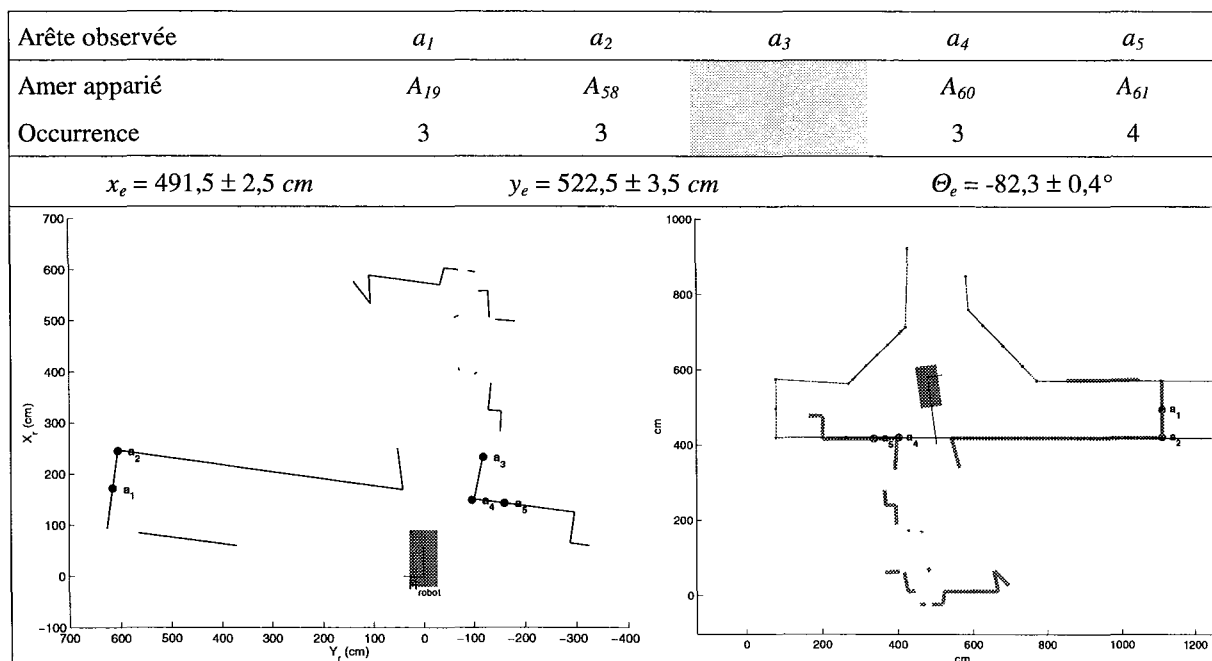


Figure 4.35 : Modèle local Bbis et localisation correspondante.

Le modèle local présenté Figure 4.36 illustre l'intérêt du recours à la vision lorsqu'un faible nombre de primitives *coin* sont extraites du modèle télémétrique. L'inspection par le module de vision a conduit à la génération de 5 arêtes observées et a rendu possible le calcul de la localisation du robot.

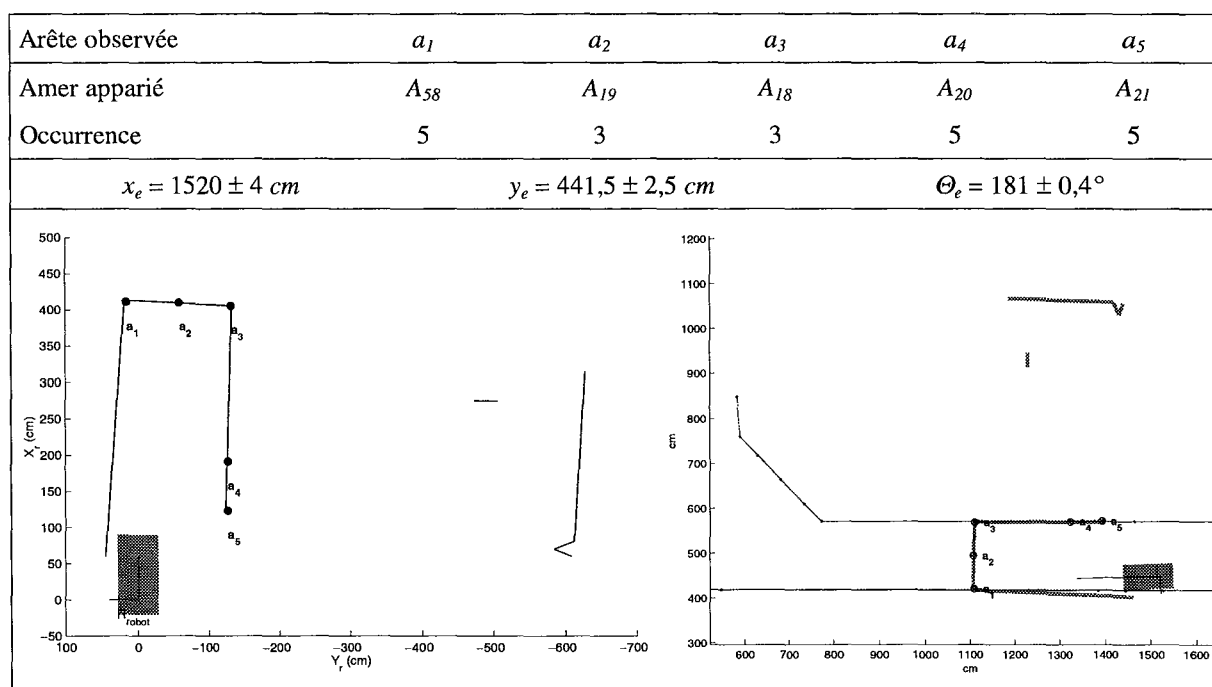


Figure 4.36 : Modèle local C et localisation correspondante.

Le modèle local présenté Figure 4.37 illustre la détermination de la localisation alors que le robot se trouve dans une pièce débouchant sur le couloir modélisé. L'utilisation directe des sommets des primitives *coin* ayant abouti à un ensemble vide de localisation, le recours à la vision s'est avéré nécessaire.

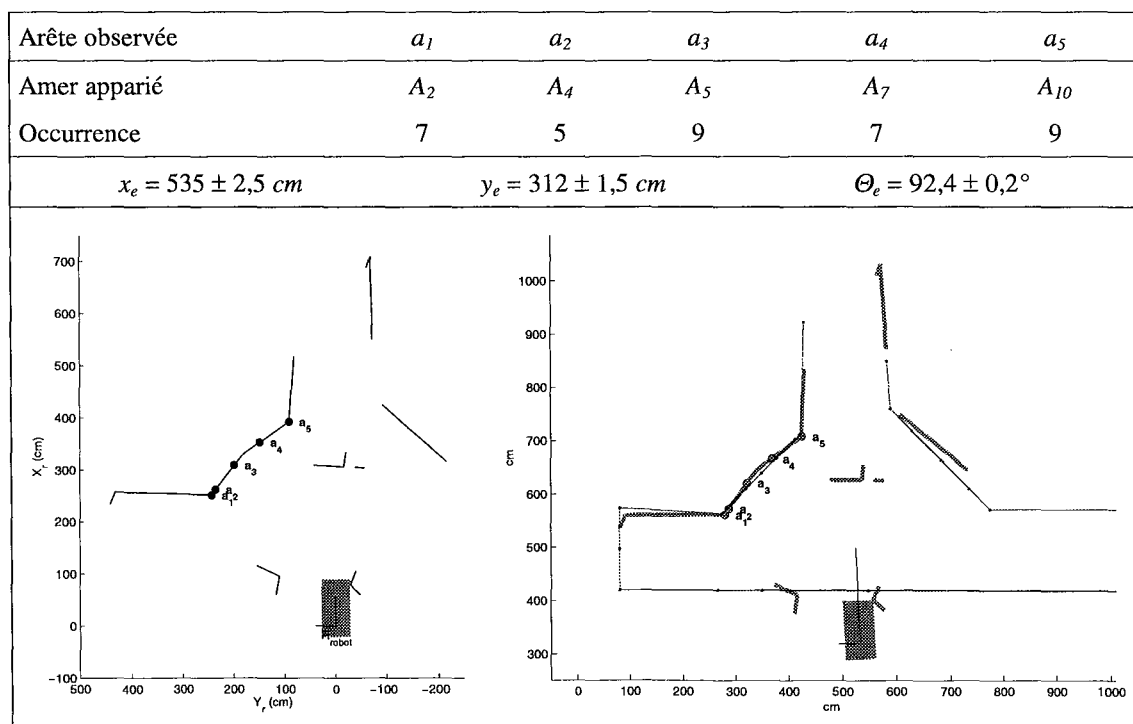


Figure 4.37 : Modèle local D et localisation correspondante.

4.4.3.3 Incertitude du modèle global

Les erreurs liées à la modélisation des amers du modèle global peuvent diminuer la précision de la localisation.

- Certains amers du modèle global représentent en fait deux arêtes. Leurs coordonnées correspondent donc en fait à une valeur moyenne.

Nous avons choisi de ne pas représenter tous les amers car la sensibilité du télémètre laser ne permet pas de détecter toutes les structures (Le réglage d'un seuil de segmentation inférieur à 8 cm produit de nombreuses petites primitives *segment* inexploitable).

- Les coordonnées de l'ensemble des amers sont exprimées dans un même repère $R_{env.}$. Les erreurs de mesure s'accumulent donc lors de la génération du modèle global et limitent par conséquent la précision de la représentation pour les amers éloignés de l'origine du repère.

Le modèle local présenté Figure 4.38 illustre le problème dû aux erreurs de modélisation des amers de la carte globale. Si l'arête a_3 est effectivement un faux positif, l'arête a_6 correspond quand à elle à l'amer A_{37} . Les erreurs sur les coordonnées de l'amer font que l'appariement n'a pas été réalisé.

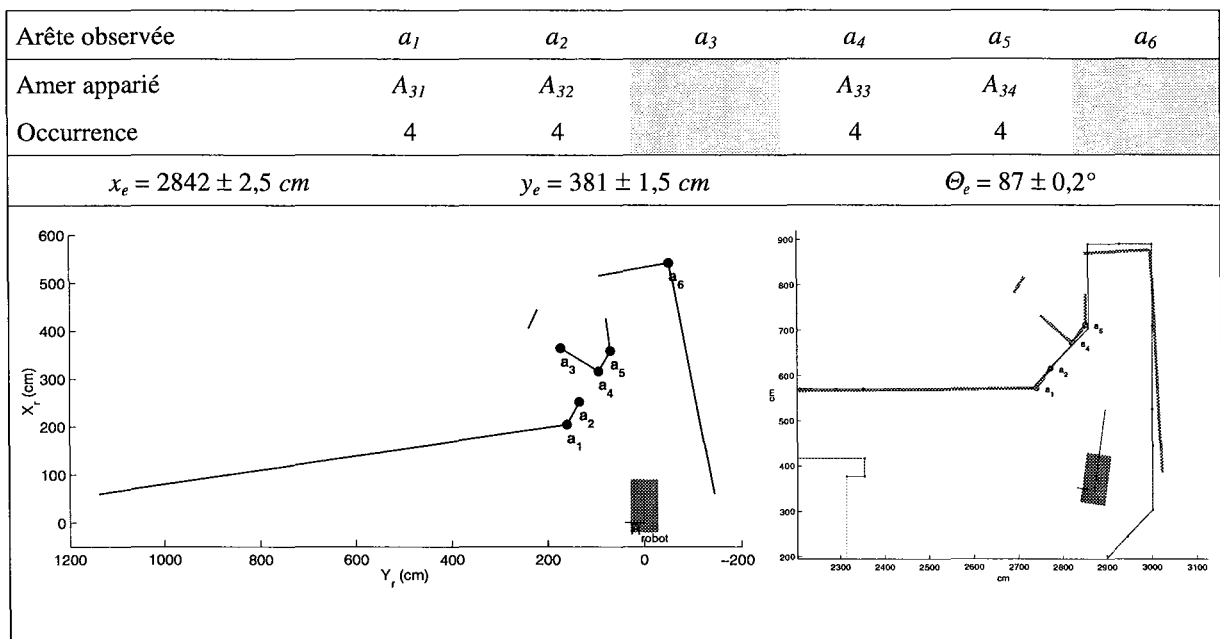


Figure 4.38 : Modèle local E et localisation correspondante.

4.4.3.4 Mises en échec

La procédure de localisation est mise en échec principalement dans trois situations :

- Lorsque le robot se trouve dans une zone caractérisée par la présence d'un faible nombre d'angles de mur. Le manque de primitives *coin* ne permet pas alors d'initier la procédure de mise en correspondance.

Une première solution consiste en une analyse aléatoire des parois verticales. Les mesures seraient toutefois fortement entachées d'erreurs du fait de l'incertitude sur les paramètres du modèle de la caméra et de l'angle ϕ entre le plan image et la surface du mur. La Figure 4.39 illustre ainsi le cas où le robot est parallèle aux murs du couloir. De plus, le temps d'acquisition et de traitement des images risque d'être élevé puisque nous n'avons pas d'indices pour initialiser la recherche des arêtes verticales.

Une seconde solution consiste à déplacer le robot vers une zone de l'environnement plus riche en primitives *coin*. Bien qu'il ne dispose pas d'information de localisation absolue, le robot peut en effet entreprendre un déplacement puisque qu'il dispose du modèle télémétrique pour détecter les zones navigables.

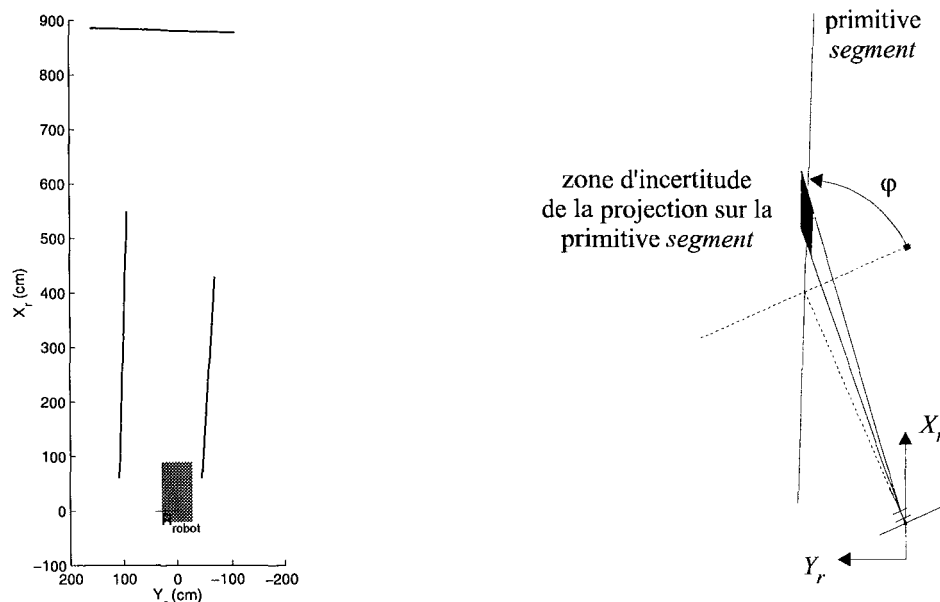


Figure 4.39 : Mise en échec causée par un manque de primitive *coin*.

- Seule une faible fraction du modèle local est effectivement modélisée dans le modèle global. Certaines structures de l'environnement telles que des armoires par exemple, peuvent alors provoquer la détection par le module de vision d'arêtes verticales non

référéncées dans le modèle global. Des appariements erronés sont alors susceptibles d'apparaître si les constellations sont proches de celles du modèle global. Une localisation faussée risque donc d'être calculée.

La Figure 4.40 illustre ainsi le cas où le robot se trouve à l'intérieur d'une pièce communiquant avec le couloir. Seule la portion de modèle local correspondant aux arêtes a_4 et a_5 est effectivement référencée dans le modèle global.

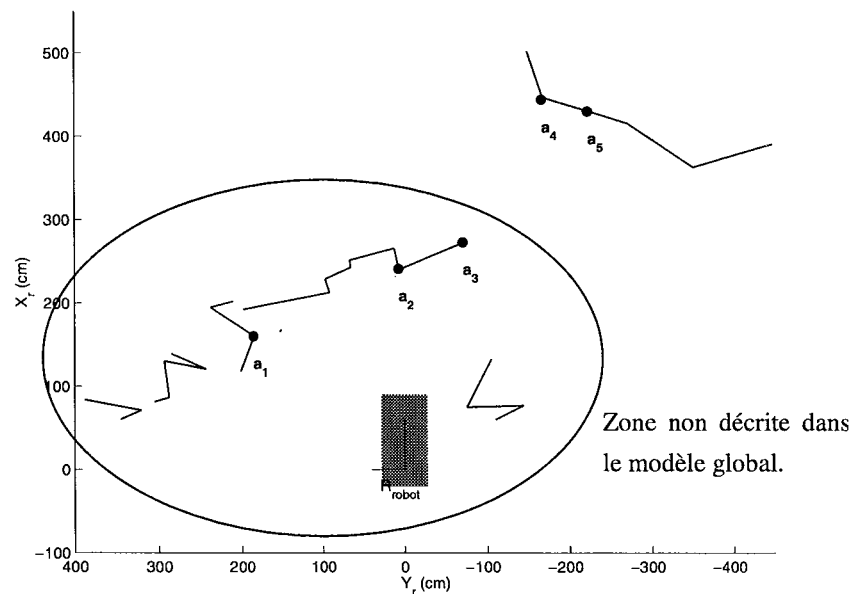


Figure 4.40 : Mise en échec causée par une faible représentativité du modèle local.

- Les arêtes observées se trouvent dans une configuration référencée dans plusieurs zones de l'environnement. Les appariements ne sont donc pas uniques et débouchent sur une impossibilité de localiser le robot. Le cas est plus fréquent dès lors que moins de 4 arêtes sont extraites du modèle local.

D'un point de vue pratique, on pourra être amené à utiliser une information de localisation a priori afin d'éliminer certains triplets : connaissant la position approximative du robot, les triplets d'amers appariés constitués à partir d'au moins un amer dont la distance au robot est supérieure à un certain seuil sont éliminés.

4.5 CONCLUSION

La détermination de la position et de l'orientation d'un robot mobile à l'aide d'amers environnementaux simples tels que des arêtes verticales est délicate à réaliser dès lors que le robot ne dispose pas d'information de localisation a priori. Nous avons par conséquent proposé une transformation des arêtes observées et des amers en une représentation faisant apparaître des invariants géométriques.

Une configuration de type triplet de points s'avère être particulièrement bien adaptée à un environnement intérieur semi-structuré car elle est très discriminante et permet de modéliser des zones caractérisées par une faible présence d'amers. Ainsi, à partir d'un modèle composé de 64 amers, nous avons généré automatiquement 23880 triplets d'amers caractérisés chacun par 12 descripteurs invariants. Le nombre moyen de triplets d'amers présentant des valeurs d'invariants proches varie alors de 1 à 5.

La procédure de mise en correspondance consiste à réaliser les appariements entre triplets d'amers et triplets d'arêtes observées, puis entre amers et arêtes observées à partir des hypothèses formées. L'incertitude sur les coordonnées des arêtes observées est propagée dans la méthode de calcul des invariants afin d'obtenir une plage de recherche assurant de ne pas manquer l'appariement avec le triplet d'amer correspondant. Une implémentation des descripteurs invariants sous la forme de listes ordonnées s'avère alors être extrêmement efficace du point de vue temps de recherche (moins de 80 *ms* par triplet d'arêtes). Nous avons pu observer que les appariements sont correctement réalisés même en présence d'arêtes observées de type *faux positifs*.

Nous avons alors proposé une méthode de localisation consistant à définir des bornes quant à l'erreur de localisation en définissant l'ensemble $(x_e^*, y_e^*, \theta_e^*)$ des paramètres compatibles avec les incertitudes de mesure des arêtes observées. Bien que plus lente qu'une estimation directe par les moindres carrés, cette méthode présente l'avantage de ne pas nécessiter la définition de seuils afin de valider la localisation calculée. Chaque arête observée contribuant à affiner la localisation du robot, une précision de l'ordre de 3 *cm* et de 0,4° est généralement obtenue.

Conclusion générale

Dans le cadre de la robotique mobile, une tâche de navigation ne peut être entreprise qu'une fois résolus les problèmes cruciaux de la perception de l'environnement et de la détermination de la position du robot au sein de cet environnement. C'est en effet à partir de la connaissance de son environnement qu'un robot mobile prend des décisions d'action et élabore des lois de commande.

Au travers des travaux présentés dans ce mémoire, nous nous sommes attachés à définir un système de perception ainsi qu'une méthode de localisation qui soient fiables et utilisables dans le cadre d'un environnement de type intérieur dynamique.

Nous avons notamment choisi de développer une méthode de localisation absolue exploitant des amers environnementaux de type arêtes verticales afin de s'affranchir des problèmes d'aménagement et de maintenance du milieu d'évolution du robot. Les solutions proposées pourront ainsi être mises en œuvre dans le cadre d'environnements différents, mais de même type, sans modifications lourdes. La seule contrainte réside dans la nécessité de déterminer les coordonnées des amers de référence du nouvel environnement considéré.

Afin d'assurer une autonomie relative au robot, nous nous sommes également fixé comme objectif de réaliser les mises en correspondances entre les amers et les arêtes observées en utilisant un minimum d'informations concernant la position du robot. Il sera ainsi possible de vérifier de façon indépendante la viabilité des mises en correspondances au fil du temps, et le cas échéant, de ré-initialiser automatiquement l'information de localisation.

Au vu des limites des systèmes de perception mono-capteur, nous avons porté notre attention sur l'utilisation d'un système de perception mixte télémétrie/vision. Compte tenu de l'environnement considéré dans notre application, notre choix c'est fixé sur l'emploi d'un télémètre laser 2D, ainsi que d'une caméra CCD monoculaire équipée d'un objectif à focale et mise au point réglables montée sur une tourelle deux axes. La complémentarité de cet

équipement sensoriel rend possible l'obtention d'informations de différentes natures concernant un même type d'objet. L'information sensorielle résultante est ainsi plus riche et plus fiable puisqu'elle reflète non seulement les données fournies par chaque capteur mais également l'information qui n'aurait pas pu être inférée par aucune des sources prises séparément.

Afin de pallier aux limites individuelles de chaque capteur nous avons mis au point une stratégie de perception active basée sur la confirmation et l'enrichissement des informations laser par les informations vision. Il s'agit d'une approche de type fusion de primitives avec focalisation d'attention au niveau d'une décision locale : la vision se met ainsi au service de la télémétrie pour valider une hypothèse et enrichir une information ponctuelle.

La segmentation du modèle télémétrique a été réalisée en ayant pour objectif de définir des primitives *segment* précises et fiables correspondant exclusivement à des structures formées par des plans verticaux de l'environnement. Nous avons pour cela défini une procédure de partitionnement de la chaîne de points de mesure faisant appel successivement à un algorithme de fusion, puis à un algorithme de découpage.

Afin d'optimiser la phase d'inspection par la vision, nous avons alors défini un critère d'intérêt visant à mettre en évidence des primitives *coin* susceptibles de correspondre à des arêtes verticales référencées comme amer. La définition de fenêtres d'intérêt diminuant considérablement le temps de traitement d'une image, la limite d'utilisation du module de vision est fixée principalement par le temps d'acquisition des images. Ce module de vision active s'avère être un élément précieux puisqu'il permet, d'une part d'éliminer les primitives *coin* correspondant à des objets non référencés dans le modèle global (*faux positifs*), et d'autre part de détecter des arêtes verticales non perceptibles par la télémétrie.

L'inspection par le module de vision permet d'extraire du modèle local de 5 à 7 arêtes observées dont les coordonnées dans le repère R_{robot} sont définies par un polygone d'incertitude prenant en compte les erreurs liées à la modélisation des paramètres du module de vision, des primitives *coin* ainsi qu'à l'extraction des primitives vision.

Le problème inhérent à tout système de reconnaissance réside dans le grand nombre de combinaisons d'appariements plausibles entre les primitives observées et les amers du modèle, dès lors que ces amers possèdent sensiblement les mêmes caractéristiques. Afin de simplifier la recherche des appariements, nous avons proposé une transformation des arêtes observées et des amers environnementaux en une représentation invariante en translation et en rotation, qui sont les deux transformations euclidiennes du repère R_{robot} vers le repère R_{env} .

Nous avons retenu une configuration de type triplet de points caractérisée par 6 descripteurs invariants propres à la structure géométrique du triangle formé et par 6 autres descripteurs invariants propres aux caractéristiques mêmes des amers. Cette représentation s'avère être très discriminante puisque seul un nombre réduit de triplets d'amers présentent des valeurs de paramètres proches.

La procédure de mise en correspondance consiste alors à réaliser les appariements entre triplets d'amers et triplets d'arêtes observées sur la base des 12 descripteurs invariants, puis entre amers et arêtes observées à partir des hypothèses formées.

La propagation de l'incertitude relative aux coordonnées des arêtes observées dans la méthode de calcul des invariants permet de définir une plage de recherche qui garantit le bon appariement avec le triplet d'amer correspondant. Une implémentation des descripteurs invariants sous la forme de listes ordonnées s'avère alors être extrêmement efficace du point de vue du temps de recherche. Nous avons pu observer que les appariements sont correctement réalisés même en présence d'arêtes observées de type *faux positifs* et/ou *vrais négatifs*.

Nous avons alors proposé une méthode de localisation consistant à définir des bornes quand à l'erreur de localisation en définissant l'ensemble $(x_e^*, y_e^*, \theta_e^*)$ des paramètres compatibles avec les incertitudes de mesure des arêtes observées. Chaque arête observée contribuant à affiner la localisation du robot, une précision de l'ordre de 3 cm et de 0,4° est généralement obtenue.

Malgré ces résultats encourageants, il reste encore quelques points à développer ou à améliorer. Parmi les perspectives de développements ultérieurs, nous pouvons envisager :

- L'amélioration du temps de traitement des informations sensorielles par la parallélisation des tâches. Il s'agit notamment de réaliser simultanément le traitement de l'image courante tout en réglant l'orientation ainsi que la focale et la mise au point de la caméra en vue de l'inspection d'une nouvelle primitive *coin*.
- L'intégration de l'incertitude de mesure dans les coordonnées des amers utilisés dans le modèle global. Ce point autoriserait la génération semi-automatique de la carte globale de l'environnement. Le robot se déplaçant dans un environnement partiellement modélisé, toute arête verticale non appariée observée de plusieurs endroits pourrait faire l'objet d'une nouvelle intégration dans le modèle global.

Références bibliographiques

-
- [Aldon 94] M.J.Aldon, L.Le Bris, « *Mobile robot localization using a light-stripe sensor* », Intelligent Vehicles'94, pages 255-259, Paris, Octobre 94.
- [Alefeld 83] G.Alefeld, J.Herzberger, « *Introduction to interval computations* », New York, Academic, 1983.
- [Arnould 93] P.Arnould, « *Etude de la localisation d'un robot mobile par fusion de données* », Thèse, Doctorat de l'Institut National Polytechnique de Lorraine, Spécialité: Automatique et Traitement Numérique du Signal, Nancy, France, 93.
- [Asensio 98] J.R.Asensio, J.M.M.Montiel, L.Montano, « *Navigation among obstacles by the cooperation of trinocular stereo vision system and laser rangefinder* », 3rd IFAC Symposium on Intelligent Autonomous Vehicles, pages 456-461, Madrid, Espagne, 25-27 mars 98.
- [Atiya 93] S.Atiya, G.D.Hager, « *Real-time vision-based robot localization* », IEEE trans. On robotics and automation, vol.9, n°6, pages 785-800, décembre 93.
- [Ayache 89] N.Ayache, « *Vision stéréoscopique et perception multisensorielle* », InterEditions, Sciences Informatique, pages 65-67, 89.
- [Ayache 91] N.Ayache, « *Artificial vision for mobile robots: stereo vision and multisensory perception* », MIT Press, Cambridge, 91.
- [Bamieh 86] B.Bamieh, R.J.P.deFigueiredo, « *A general moment-invariants /attributed-graph method for three-dimensional object-recognition from a single image* », IEEE journal of robotics and automation, vol.2, n°1, pages 31-41, 86.
- [Barshan 95] B.Barshan, H.F.Durrant-Whyte, « *Inertial navigation systems mobile robots* », IEEE trans. On robotics and automation, vol.11, n°3, pages 328-342, juin 95.
- [Blanes 98] F.Blanes, G.Benet, M.Martinez, J.Simo, « *Grid map building from reduced sonar data* », 3rd IFAC Symp. on Intelligent Autonomous Vehicles, pages 225-230, Madrid, Espagne, 25-27 mars 98.
- [Bonnifait 96] P.Bonnifait, G.Garcia, « *Continuous localization of outdoor mobile robot using odometry and relative azimuth angles of known landmarks* », Computational Engineering in Systems Applications,
-

- pages 30-35, Lille, France, juillet 96.
- [Borenstein 96] J.Borenstein, H.R.Everett, L.Feng, « *Where am I ?* », *Sensors and methods for mobile robot positioning* », Report, University of Michigan, edited and compiled by J.Borenstein, 96.
- [Born 93] M.Born E.Wolf, « *Principles of optics* », Pergamon Press, Oxford, 93.
- [Botton 96] S.Botton, F.Duquenne, Y.Egels, M.Even, P.Willis, « *G.P.S. , Localisation et navigation* », Conseil National de l'Information Géographique. Groupe Positionnement Statique et Dynamique, Edition Hermes, 96.
- [Byrne 92] R.H.Byrne, P.R.Klarer, J.B.Pletta, « *Techniques for autonomous navigation* », Sandia report SAND92-0457, Sandia National Laboratories, Albuquerque, Etats Unis, 92.
- [Cahn von Seelen 97] U. Cahn von Seelen, « *Performance evaluation of an active vision system* », Dissertation in Computer and Information Science. Faculty of the university of Pennsylvania., 97.
- [Canny 86] J.Canny, « *A computational approach to edge detection* », IEEE trans. On pattern analysis and machine intelligence, vol.8, n°6, pages 679-698, novembre 86.
- [Castellanos 98] J.A.Castellanos, J.M.Martinez, J.Neira, J.D.Tardos, « *Experiments in multisensor mobile robot localization and map building* », 3rd IFAC Symposium on Intelligent Autonomous Vehicles, pages 173-178, Madrid, Espagne, 25-27 mars 98.
- [Chen 91] S.Y.Chen, W.H.Tsai, « *Determination of robot locations by common objects shapes* », IEEE trans. On robotics and automation, vol.7, n°1, pages 149-156, février 91.
- [Cocquerez 95] J.P.Cocquerez, S.Philipp, « *Analyse d'images: filtrage et segmentation* », Editions MASSON, pages 107-152, 95.
- [Cohen 92] C.Cohen, F.V.Koss, « *A comprehensive study of three object triangulation* », SPIE Mobile robots VII, vol.1831, pages 95-106, 92.
- [Cord 98] T.Cord, T.Rupp, D.E.Lazie, « *A navigation system for mobile service robots* », 3rd IFAC Symposium on Intelligent Autonomous Vehicles, pages 781-786, Madrid, Espagne, 25-27 mars 98.
- [Couroux 97] W.Couroux, « *Méthode de localisation absolue d'un robot mobile évoluant en milieu extérieur fondée sur la recherche d'indices visuels* », Thèse, Doctorat de l'Institut National Polytechnique de Lorraine, Spécialité: Automatique et Traitement Numérique du Signal, 97.
- [Courtney 94] J.D.Courtney, A.K.Jain, « *Mobile robot localization via classification of multisensor maps* », IEEE int. Conf. On Robotics and Automation, pages 1672-1678, San Diego, Etats-Unis, 8-13 mai 94.
- [Cox 91] I.J.Cox, « *Blanche – an experiment in guidance and navigation of an autonomous mobile robot* », IEEE transactions Robotics and Automation, vol.7, n°3, pages 193-204, 91.

-
- [Crisman 93] J.D.Crisman, C.E.Thorpe, « *SCRAF: a color vision system that tracks roads and intersections* », IEEE transactions Robotics and Automation, vol.9, n°1, pages 49-58, février 93.
- [Crowley 89] J.Crowley, « *World modelling and position estimation for a mobile robot using ultrasonic ranging* », IEEE int. Conf. On Robotics and Automation, pages 674-680, Scottsdale, AZ, 14-19 mai 89.
- [Curan 95] A.Curan, K.J.Kyriakopoulos, « *Sensor based self localization for wheeled mobile robots* », Journal of Robotic Systems, vol.12, n°3, pages 163-176, 95.
- [Delahoche 96] L.Delahoche, C.Pégard, M.Mouaddib, P.Vasseur, « *Localization based on target classification with a multisensor mobile robot* », 4th int. Conf. On control, automation, robotics and vision, ICARCV'96, pages 1111-1115, Singapour, 3-6 décembre 96.
- [Deriche 87] R.Deriche, « *Using Canny's criteria to derive a recursively implemented optimal edge detector* », International journal of computer vision, pages 167-187, 87.
- [Dietel 96] R.Dietel, « *Programmierung und Anwendung des Motion Micro-Controllers LM629 zur der Videokamera eines mobilen Roboters* », Diplomarbeit, Universität Karlsruhe, Décembre 96.
- [Duda 73] R.O.Duda, P.E.Heart, « *Pattern classification and scene analysis* », Wiley, New York, 73.
- [Eberst 98] C.Eberst, « *Fusing a panoramic laser range finder with a binocular stereo system using hypothetical features* », 3rd IFAC Symposium on Intelligent Autonomous Vehicles, pages 161-166, Madrid, Espagne, 25-27 mars 98.
- [Elarbi 92] M.Elarbi Boudihir, « *Contribution au guidage de robot mobile par vision* », Thèse, Doctorat de l'Institut National Polytechnique de Lorraine, Spécialité: Automatique et Traitement Numérique du Signal, Nancy, France, 92.
- [Ellepola 96] R.Ellepola, P.Kovesi, « *Mobile robot navigation in a semi-structured environment* », 4th int. Conf. On control, automation robotics and vision, ICARCV'96, pages 914-918, Singapour, 3-6 décembre 96.
- [Everett 94] H.R.Everett, D.W.Gage, G.A.Gilbret, R.T.Laird, R.P.Smurlo, « *Real-world issues in warehouse navigation* », Proceedings SPIE Mobile Robots IX, vol.2352, Boston, Etats-Unis, 2-4 novembre 94.
- [Fennema 90] C.Fennema, A.Hanson, E.Riseman, J.R.Beveridge, R.Kumar, « *Model-directed mobile robot navigation* », IEEE trans. On systems, man and cybernetics, vol.20, n°6, pages 1352-1369, 90.
- [Fleury 92] S.Fleury, T.Baron, « *Absolute external mobile robot localization using a single image* », SPIE, vol.1831, pages 131-143, 92.
- [Forsberg 95] J.Forsberg, U.Larsson, A.Wernersson, « *mobile robot navigation using the range-weighted hough transform* », IEEE Robotics and Automation magazine, pages 18-26, mars 95.
-

- [Garcia 98] E.Garcia, H.Lamela, « *Conception of a 3D vision system for autonomous robot applications using an AMCW laser rangefinder* », 3rd IFAC symposium on Intelligent Autonomous Vehicles, IAV'98, pages 553-557, Madrid, Espagne, 25-27 mars 98.
- [Gonzalez 95] J.Gonzalez, A.Stentz, A.Ollero, « *A mobile robot iconic position estimator using a radial laser scanner* », Journal of intelligent and robotics systems, vol.13, pages 161-179, 95.
- [Gross 93] P.Gross, « *Outils géométriques pour la modélisation et la reconnaissance d'objets polyédriques* », Thèse, Doctorat de l'Institut National Polytechnique de Grenoble, Spécialité: Informatique, Grenoble, France, 93.
- [Guerrero 98] J.J.Guerrero, C.Sagués, « *Navigation from uncalibrated monocular vision* », 3rd IFAC symposium on Intelligent Autonomous Vehicles, IAV'98, pages 210-215, Madrid, Espagne, 25-27 mars 98.
- [Hardt 97] H.-J Von der Hardt, « *Contribution au pilotage et à la localisation d'un robot mobile* », Thèse, Doctorat de l'Institut National Polytechnique de Lorraine, Spécialité: Automatique et Traitement Numérique du Signal, Nancy, France, 97.
- [Hoppenot 97] P.Hoppenot, E.Colle, « *Real-time mobile robot localisation with poor ultrasonic data* », 3rd IFAC Symposium on Intelligent Components and Instruments for Control Applications, SICICA'97, pages 135-140, Annecy, France, 9-11 juin 97.
- [Huttenlocher 94] D.P.Huttenlocher, M.E.Leventon, W.J.Rucklidge, « *Visually-guided navigation by comparing two-dimensional edge images* », Technical Report TR 94-1407. Department of Computer Science, Cornell university, janvier 94.
- [Ishikawa 92] M.Ishikawa, « *The sensor fusion system : mechanisms for integration of sensory information* », Journal of Advanced robotics, vol.6, n°3, pages 335-344, 92.
- [Jasiobedzki 93] P.Jasiobedzki, M.Jenkin, E.Milios, B.Down, J.Tsotsos, T.Campbell, « *Laser eye - a new 3D sensor for active vision* », SPIE sensor fusion VII, vol.2059, pages 316-321, 93.
- [Kaske 97] A.Kaske, « *Contribution à la détection de bords de route imprécis. Implantation sur le robot ROMANE* », Thèse, Doctorat de l'Institut National Polytechnique de Lorraine, Spécialité: Automatique et Traitement Numérique du Signal, Nancy, France, 97.
- [Kortenkamp 94] D.Kortenkamp, T.Weymouth, « *Topological mapping for mobile robots using a combinaison of sonar and vision sensing* », AAAI'94, 94.
- [Kosaka 92] A.Kosaka, A.C.Kak, « *Fast vision-guided mobile robot navigation using model-based reasoning and prediction of uncertainties* », CVGIP image understanding, vol.56, n°3, pages 271-329, nov. 92.
- [Kurata 98] J.Kurata, K.T.V.Grattan, H.Uchiyama, « *Path planning for a mobile robot by integrating mapped information* », 3rd IFAC symposium on
-

-
- Intelligent Autonomous Vehicles, IAV'98, pages 319-323, Madrid, Espagne, 25-27 mars 98.
- [Lallement 98a] A.Lallement, M.Dufaut, R.Husson, « *Sensor cooperation and world perception of the autonomous mobile robot ROMANE* », 3rd IFAC symposium on Intelligent Autonomous Vehicles, IAV'98, pages 179-185, Madrid, Espagne, 25-27 mars 98.
- [Lallement 98b] A.Lallement, A.Siadat, M.Dufaut, R.Husson, « *Laser-Vision cooperation for map building and landmarks recognition* », IEEE joint conference on the Science and Technology of Intelligent Systems, ISIC/CIRA/ISAS'98, pages 387-392, Gaithersburg, Etats-Unis, 14-17 septembre 98.
- [Lallement 98c] A.Lallement, M.Dufaut, R.Husson, « *Mobile robot localisation based on reliable features generation* », 12th Deutsch-Franzosisches Institut fur Automation und Robotik annual conference, IAR'98, pages 131-136, Mulhouse, France, 19-20 novembre 98.
- [Lallement 99a] A.Lallement, M.Dufaut, R.Husson, « *Mobile robot localization based on invariant matching* », Int. Conf. On Computational Intelligence for Modelling, Control and Automation, CIMCA'99, pages 275-280, Vienne, Autriche, 17-19 février 99.
- [Lallement 99b] A.Lallement, M.Dufaut, R.Husson, « *Environmental landmarks matching for mobile robot localisation* », 3^{ème} conférence internationale sur l'Automatisation Industrielle, AIAI'99, Montréal, Canada, 7-9 juin 99.
- [Lallement 99c] A.Lallement, M.Dufaut, R.Husson, « *A reliable method for vertical edges recognition* », 5th European Control Conference, ECC'99, Karlsruhe, Allemagne, 31 aout – 3 septembre 99.
- [Lamdan 88] Y.Lamdan, H.J.Wolfson, « *Géométrie hashing: a general and efficient model-based recognition scheme* », 2nd International Conf. on Computer Vision, pages 238-249, Tampa, Etats Unis, 88.
- [Lamela 98] H.Lamela, J.R.Lopez, E.Garcia, S.Gomez, P.Acedo, G.Carpintero, « *A 2D and 3D vision system using hardware triangulation video signal processing for mobile robots* », 3rd IFAC symposium on Intelligent Autonomous Vehicles, IAV'98, pages 558-562, Madrid, Espagne, 25-27 mars 98.
- [Larsson 98] U.Larsson, F.Chenevier, A.Wernersson, « *On rate Gyros for improved navigation by reducing dead-reckoning errors* », 3rd IFAC symposium on Intelligent Autonomous Vehicles, IAV'98, pages 693-698, Madrid, Espagne, 25-27 mars 98.
- [Lavest 93] J.M.Lavest, G.Rives, M.Dhome, « *Three dimensional reconstruction by zooming* », IEEE trans. On robotics and automation, vol.9, n°2, pages 196-207, avril 93.
- [Lee 90] C.H.Lee, T.Huang, , « *Finding poit correspondences and determining motion of a rigid object from two weak perspective views* », Computer vision, graphics and image processing, vol.52, pages 309-327, 90.
-

- [Madsen 97] C.B.Madsen, C.S.Andersen, J.S.Sorensen, « *A robustness analysis of triangulation-based robot self-positioning* », 5th int. Symposium on Intelligent Robotics Systems, SIRS'97, pages 195-204, Stockholm, Suède, 8-11 juillet 97.
- [Mayora 98] K.Mayora, I.Moreno, G.Obieta, « *Perception system for navigation in a 3D outdoor environment* », 3rd IFAC symposium on Intelligent Autonomous Vehicles, IAV'98, pages 707-711, Madrid, Espagne, 25-27 mars 98.
- [Mohr 91] R.Mohr, L.Morin, C.Inglebert, L.Quan, « *Géométrique solutions to some 3D vision problems* », R.Storer éditeur, Integration and control in real time active vision, ESPRIT BRA Series, Springer-Verlag, 91.
- [Moore 66] R.E.Moore, « *Interval analysis* », Englewoods cliffs, Prentice Hall, 66.
- [Mundy 92] J.L.Mundy, « *Géométric invariance in computer vision* », MIT Press, Cambridge, Massacusetts, Etats Unis, 92.
- [Nashashibi 93] F.Nashashibi, « *Perception et modélisation d'un environnement tridimensionnel pour la navigation autonome d'un robot mobile* », Thèse de l'université Paul Sabatier de Toulouse, Toulouse, France, 19 janvier 93.
- [Roberts 65] L.Roberts, « *Machine perception of three-dimensional solids* », Optical and electro-optical info. Process. MIT Press, Cambridge, pages 159-197, 65.
- [Seki 98] H.Seki, Y.Tanaka, M.Takano, K.Sasaki, « *Positioning system for indoor mobile robot using active ultrasonic beacons* », 3rd IFAC Symposium on Intelligent Autonomous Vehicles, pages 681-686, Madrid, Espagne, 25-27 mars 98.
- [Siadat 97] A.Siadat, A.Lallement, M.Dufaut, R.Husson, « *Perception et modélisation de l'environnement du robot mobile ROMANE* », Colloque de recherche doctorale, automatique génie informatique image iignal, AGIS'97, Angers, France, décembre 97.
- [Siadat 99] A.Siadat, « *Perception et modélisation de l'environnement par télémétrie laser. Application à la navigation autonome d'un fauteuil roulant* », Thèse, Doctorat de l'Institut National Polytechnique de Lorraine, Spécialité: Automatique et Traitement Numérique du Signal, Nancy, France, 99.
- [Sick 95] Document constructeur « *Sick PLS Laser Scanner Operating Instructions* », Allemagne, 95.
- [Skrzypczynski 98] P.Skrzypczynski, « *Localisation of a mobile robot based on natural landmarks* », 3rd IFAC Symposium on Intelligent Autonomous Vehicles, pages 615-620, Madrid, Espagne, 25-27 mars 98.
- [Subbarao 88] M.Subbarao, « *Parallel depth recovery by changing camera parameters* », Int. Conf. on Computer Vision, pages 149-155, Tampa, Floride, 88.
- [Sugihara 88] K.Sugihara, « *Some location problems for robot navigation using a*
-

-
- single camera* », Computer vision, graphics and image processing, vol.42, pages 112-129, 88.
- [Tsumura 86] T.Tsumura, « *Survey of automated guided vehicle in Japanese factory* », IEEE int. conf. on Robotics and Automation, pages 1329-1334, San Francisco, Etats Unis, 7-10 avril 86.
- [Vaganay 93] J.Vaganay, M.J.Aldon, A.Fournier, « *Mobile robot attitude estimation by fusion of inertial data* », IEEE int. Conf. On Robotics and Automation, ICRA93, pages 277-282, Atlanta, Etats-Unis, mai 93.
- [Vandorpe 96] J.Vandorpe, H.Xu, H.Van Brussel, E.Aertbeliën, « *Positioning of the mobile robot LiAS using natural landmarks and a 2D range finder* », IEEE/SICE/RSJ int. conf. on multisensor fusion and integration for intelligent systems, pages 257-264, 96.
- [Vergnet 94] R.Vergnet, « *Reconnaissance d'objets géométriques et génériques en environnement extérieur : stratégies de vision active* », Thèse, Université de Paris-Sud, Centred'Orsay, France, février 94.
- [Wall 83] K.Wall, P.E.Danielsson, « *A new method for polygonal approximation of digitized curves* », Proc. Third Scandinavian Conf. On Image Analysis, pages 60-66, Lund, Suède, 83.
- [Wallner 97] F.Wallner, B.Schiele, J.L.Crowley, « *Position estimation for a mobile robot from principal components of laser range data* », 5th int. Symposium on Intelligent Robotics Systems, SIRS'97, pages 215-224, Stockholm, Suède, 8-11 juillet 97.
- [Watt 89] A.Watt, « *Fundamentals of three-dimensional computer graphics* », Addison Wesley, 89.
- [Weckesser 96] P.Weckesser, R.Dillmann, « *Sensor-fusion of intensity and laserrange images* », IEEE/SICE/RSJ int. conf. on multisensor fusion and integration for intelligent systems, pages 501-508, 96.
- [Weiss 93] I.Weiss, « *Geometric invariants and object recognition* », Int. journal of Computer Vision, vol.10, n°3, pages 207-231, 93.
- [Weiss 94] G.Weiss, C.Wetzler, E.Von.Puttkamer, « *Keeping track of position and orientation of moving indoor systems by correlation of range-finder scans* », IEEE int. conf. on intelligent robots and systems, IROS'94, pages 595-601, München, Allemagne, septembre 94.
- [Willson 94] R.G.Willson, « *Modeling and calibration of automated zoom lenses* », Rapport interne, the robotics institute, Carnegie Mellon university, CMU-RI-TR-94-03, 94.
- [Wolfson 90] H.J.Wolfson, « *Model-based object recognition by geometric hashing* », O.Faugeras éditeur, 1st european conference on computer vision, pages 526-536, Antribes, France, avril 90.
- [Yang 96] Y.B.Yang, H.T.Tsui, « *Mobile robot localization by multi-sensor fusion and scene matching* », SPIE, vol.2904, pages 298-307, 96.
-

Résumé

Les travaux présentés dans ce mémoire portent sur l'étude et la mise au point d'une méthode de localisation absolue pour robot mobile autonome évoluant dans un environnement de type intérieur dynamique. Cette méthode est basée sur l'exploitation de la complémentarité des informations délivrées par un système de perception mixte formé d'un télémètre laser 2D et d'un capteur de vision monoculaire.

Notre contribution se situe essentiellement sur le plan méthodologique avec notamment : la mise au point d'une stratégie de perception active, la proposition d'une technique robuste de segmentation, la recherche de techniques décisionnelles utilisant une approche de type fusion de primitives, l'étude d'une technique d'identification d'amers environnementaux (de type arête verticale) et enfin la mise au point d'une procédure de localisation.

La phase d'identification des amers, nécessaire à la localisation du robot, nécessite le recours à une procédure d'appariements entre arêtes segmentées et amers de référence. Une transformation invariante en translation et en rotation, appliquée sur l'ensemble des primitives, conduit à une simplification notable de cette procédure. L'identification consiste alors à réaliser une mise en correspondance entre triplets d'amers et triplets d'arêtes sur la base de 12 descripteurs invariants, puis entre amers et arêtes observées à partir des hypothèses formées. La localisation absolue du mobile est obtenue en définissant l'ensemble des positions compatibles avec les amers observés et délimitée par un polygone d'incertitude.

Les techniques présentées ont été implantées et expérimentées sur le démonstrateur ROMANE.

Abstract

The work presented in this thesis is devoted to the study and the elaboration of an absolute localisation method for the purpose of guiding an autonomous mobile robot in a dynamic indoor environment. This method is based on the exploitation of the complementarity of the information delivered by a perception system composed of a 2D laser range finder and of a monocular vision sensor.

Our contribution is of a methodological nature, with notably: the definition of an active perception strategy, the proposal of a robust laser data segmentation algorithm, the research of decisional techniques for the fusion of primitives, the study of a method of identification of environmental landmarks (i.e. vertical edges) and the definition of a localisation process.

The landmark identification stage, which is necessary for robot localisation, implies a matching process between observed edges and known landmarks. An invariant transformation of translation and rotation applied to these primitives allows for the quick and reliable establishment of the correspondences. The identification is then carried out by directly comparing observed edge triplets and known landmark triplets based on twelve invariant parameters. From this set of corresponding triplets, it is further possible to determine all consistent pairings of observed features and known landmarks. The mobile robot localisation is then computed by determining the set of locations compatible with the matched landmarks and defined by a polygon of uncertainty taking into account measurement errors.

The techniques presented here have been implanted and validated on the ROMANE prototype.

**AUTORISATION DE SOUTENANCE DE THESE
DU DOCTORAT DE L'INSTITUT NATIONAL
POLYTECHNIQUE DE LORRAINE**

o0o

VU LES RAPPORTS ETABLIS PAR :

Monsieur GRESSER Julien, Professeur, Université de Mulhouse,

Monsieur WENDEL Pierre-Louis, Professeur, ENSPS, Illkirch,

Le Président de l'Institut National Polytechnique de Lorraine, autorise :

Monsieur LALLEMENT Alex

à soutenir devant un jury de l'INSTITUT NATIONAL POLYTECHNIQUE DE
LORRAINE, une thèse intitulée :

**"Localisation d'un robot mobile par coopération entre vision monoculaire et
télémétrie laser".**

en vue de l'obtention du titre de :

DOCTEUR DE L'INSTITUT NATIONAL POLYTECHNIQUE DE LORRAINE

Spécialité : **"AUTOMATIQUE & TRAITEMENT NUMERIQUE DU SIGNAL"**

NANCY BRABOIS
2, AVENUE DE LA
FORET-DE-HAYE
BOITE POSTALE 3
F - 5 4 5 0 1
VANDŒUVRE CEDÈX

Fait à Vandoeuvre le, **23 juin 1999**

Le Président de l'I.N.P.L.,

J. HARDY



Résumé

Les travaux présentés dans ce mémoire portent sur l'étude et la mise au point d'une méthode de localisation absolue pour robot mobile autonome évoluant dans un environnement de type intérieur dynamique. Cette méthode est basée sur l'exploitation de la complémentarité des informations délivrées par un système de perception mixte formé d'un télémètre laser 2D et d'un capteur de vision monoculaire.

Notre contribution se situe essentiellement sur le plan méthodologique avec notamment : la mise au point d'une stratégie de perception active, la proposition d'une technique robuste de segmentation, la recherche de techniques décisionnelles utilisant une approche de type fusion de primitives, l'étude d'une technique d'identification d'amers environnementaux (de type arête verticale) et enfin la mise au point d'une procédure de localisation.

La phase d'identification des amers, nécessaire à la localisation du robot, nécessite le recours à une procédure d'appariements entre arêtes segmentées et amers de référence. Une transformation invariante en translation et en rotation, appliquée sur l'ensemble des primitives, conduit à une simplification notable de cette procédure. L'identification consiste alors à réaliser une mise en correspondance entre triplets d'amers et triplets d'arêtes sur la base de 12 descripteurs invariants, puis entre amers et arêtes observées à partir des hypothèses formées. La localisation absolue du mobile est obtenue en définissant l'ensemble des positions compatibles avec les amers observés et délimitée par un polygone d'incertitude.

Les techniques présentées ont été implantées et expérimentées sur le démonstrateur ROMANE.

Abstract

The work presented in this thesis is devoted to the study and the elaboration of an absolute localisation method for the purpose of guiding an autonomous mobile robot in a dynamic indoor environment. This method is based on the exploitation of the complementarity of the information delivered by a perception system composed of a 2D laser range finder and of a monocular vision sensor.

Our contribution is of a methodological nature, with notably: the definition of an active perception strategy, the proposal of a robust laser data segmentation algorithm, the research of decisional techniques for the fusion of primitives, the study of a method of identification of environmental landmarks (i.e. vertical edges) and the definition of a localisation process.

The landmark identification stage, which is necessary for robot localisation, implies a matching process between observed edges and known landmarks. An invariant transformation of translation and rotation applied to these primitives allows for the quick and reliable establishment of the correspondences. The identification is then carried out by directly comparing observed edge triplets and known landmark triplets based on twelve invariant parameters. From this set of corresponding triplets, it is further possible to determine all consistent pairings of observed features and known landmarks. The mobile robot localisation is then computed by determining the set of locations compatible with the matched landmarks and defined by a polygon of uncertainty taking into account measurement errors.

The techniques presented here have been implanted and validated on the ROMANE prototype.
