Institut National
Polytechnique de Lorraine

CRIN — INRIA Lorraine

THÈSE

pour l'obtention du

Doctorat de l'Institut National Polytechnique de Lorraine

(Spécialité Informatique)

par

Long QUAN

CONTRIBUTION DE LA VISION MONOCULAIRE À LA PERCEPTION TRIDIMENSIONNELLE

Présentée et soutenue publiquement le 4 octobre 1989, devant la commisssion composée de :

M.	**Jean-Paul Haton**	**Président**
MM.	**Serge Castan**	**Rapporteurs**
	Pierre Marchand	
	ld Masini	**Examinateurs**
	r Mohr	

Institut National
Polytechnique de Lorraine

CRIN — INRIA Lorraine

THÈSE

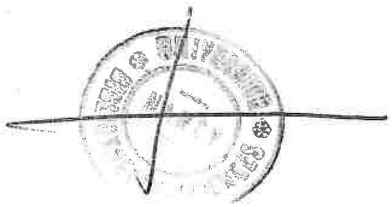
pour l'obtention du

Doctorat de l'Institut National Polytechnique de Lorraine

(Spécialité Informatique)

par

Long QUAN



CONTRIBUTION DE LA VISION MONOCULAIRE À LA PERCEPTION TRIDIMENSIONNELLE

Présentée et soutenue publiquement le 4 octobre 1989, devant la commisssion composée de :

M.	**Jean-Paul Haton**	**Président**
MM.	**Serge Castan**	**Rapporteurs**
	Pierre Marchand	
MM.	**Gérald Masini**	**Examinateurs**
	Roger Mohr	

À mes parents

Résumé

Nous présentons dans cette thèse un système de vision monoculaire pour la perception tridimensionnelle. Plus précisément les problèmes abordés sont l'appariement d'une image avec le modèle tridimensionnel de la scène et l'appariement d'images entre elles. Ce travail s'inscrit dans le projet de vision ORASIS dont l'objectif est de permettre à un robot mobile de se déplacer dans une scène d'intérieur.

Nous présentons d'abord un certain nombre de propriétés de la géométrie projective que nous exploitons dans notre système. Puis nous développons un algorithme d'extraction des informations perspectives telles que les points de fuite et la ligne d'horizon. Ensuite nous abordons le problème du regroupement des indices visuels, et développons les algorithmes de regroupement sur les indices de type segment de droite issus de l'approximation polygonale de contours. Les groupes utilisés sont : le groupe directionnel, le groupe raie, le groupe colinéaire, le groupe convexe et le groupe coplanaire. Nous développons également un algorithme linéaire permettant la détermination précise du point de vue dans le modèle. Nous décrivons, ensuite, les algorithmes pour l'appariement image/modèle et image/image. L'appariement des indices visuels avec des modèles d'objets est effectué au niveau bidimensionnels en termes de groupes invariants bidimensionnel par changement de point de vue et non pas en termes de caractéristiques tridimensionnelles. Le principe de l'algorithme repose sur la technique de prédiction/vérification. Une discussion confrontant les différentes techniques de l'appariement nous aide à justifier notre choix. Finalement nous présentons les résultats expérimentaux de l'appariement.

Mots-Clés :

- vision par ordinateur - vision monoculaire - vision à base de modèles - mouvement - inférence 2D/3D - appariement - shape-from-perspective - interprétation d'images - organisation perceptuelle - transformé de Hough - détermination spatiale - invariants perspectifs - point de fuite - ligne d'horizon - birapport - sphère gaussienne - coordonnées projectives -

Abstract

A monocular vision system is presented in this thesis. The objective is to match a monocular image against a *3D* model or against other images. This work is a part of the vision project ORASIS whose aim is to make a mobile robot move in an indoor scene.

We introduce first some properties of projective geometry that we have used in our system, then develop an algorithm capable of extracting perspective information of the scene such as vanishing points and horizon lines. Next, perceptual grouping is performed on line segments from polygonal approximation of contours for pruning the search space. The perceptual groups used are directional groups, rays, collinear groups, junctions, convex groups and coplanar groups. After that we also develope a linear algorithm for determining precisely the viewpoint in the model. The matching algorithm is implemented both for image/model matching and image/image matching based on the prediction/verification principle. The different matching techniques are discussed and we justify our choice. Finally the experimental resultats are presented.

Key-Words:

- computer vision - monocular vision - model-based vision - motion - 2D/3D inference - matching - shape-from-perspective - image interpretation - organisation perceptuelle - Hough transform - spatial determination - perspective invariant - vanishing point - horizon line - double ratio - Gaussian sphere - projective coordinates -

Remerciements

Cette thèse est pour moi l'occasion d'exprimer des remerciements à tous ceux qui m'ont conseillé et aidé à réaliser ce projet dans de bonnes conditions, et je suis heureux aujourd'hui de remercier plus particulièrement :

Monsieur Roger Mohr, Professeur à l'Institut National Polytechnique de Lorraine, qui m'a proposé un sujet de recherche passionnant, et m'a encadré et orienté tout au long de ces années de travail. Je lui suis particulièrement reconnaissant pour ses compétences, ses conseils, ses encouragements, sa confiance et également pour son enthousiasme.

Monsieur Jean-Paul Haton, Professeur à l'Université de Nancy 1 et Directeur de Recherches INRIA au CRIN, qui m'a accueilli dans l'équipe Reconnaissance des Formes et Intelligence Artificielle du CRIN, et qui me fait l'honneur de présider ce jury. Je le remercie de sa confiance et pour le grand intérêt qu'il a toujours porté à mes travaux.

Monsieur Serge Castan, Professeur à l'Univercité Paul-Sabatier à Toulouse et Directeur du laboratoire CERFIA et Monsieur Pierre Marchant, Professeur à l'Université de Nancy 1, qui ont accepté d'être rapporteurs de ce mémoire et de siéger à ce jury. Qu'ils trouvent ici l'expression de ma gratitude pour l'intérêt dont ils ont fait preuve à l'égard de ce travail.

Monsieur Gérald Masini, Chargé de Recherche au C.N.R.S, qui a accepté de faire partie du jury de soutenance. Il m'a fait un grand plaisir en ayant accepté d'examiner mon travail et de le juger.

J'exprime des remerciement chaleureux à toute l'équipe Vision du CRIN-INRIA Lorraine et plus spécialement à mon camarade de travail Eric Thirion pour de nombreuses discussions fructueuses dans le cadre du projet Orasis. Merci également à tous mes collègues et amis du CRIN, pour l'atmosphère de travail sympathique dont j'ai bénéficié. Tout particulièrement à mes camarades thésard Pierre Nugues, Yves Laprie, Marie Odile Berger, Yifan Gong et Bernard Waldmann pour tous les moments qu'ils ont consacrés à discuter ou à me relire ce mémoire.

Enfin, je tiens à remercier vivement toute ma famille pour ses encouragements et le soutient moral qu'ils m'ont prodigués, sans lesquels je n'aurait pas pu terminer ce travail.

1
Introduction

1.1 Que faire ?

Un système de *vision artificielle* (ou *vision par ordinateur*) pour un robot mobile a pour objectifs

- l'*apprentissage* d'environnement inconnu,
- la *reconnaissance* des objets connus de l'environnement,
- et la *localisation* du robot dans un environnement connu.

Apprentissage signifie construction automatique du modèle de son environnement. Ce modèle est la représentation de l'ensemble des connaissances que le robot possède sur son environnement. La reconnaissance signifie qu'une correspondance est établie entre les données visuelles et une représentation de l'objet dans le modèle. La localisation consiste à exprimer le repère lié au robot dans le repère lié au modèle.

Commençons par les capteurs visuels, dont le rôle est primordial, puisqu'ils permettent de percevoir l'environnement et de saisir des données visuelles. Selon le type de données qu'ils saisissent, on les classe en deux types :

- *capteur actif* : utilisation des faisceaux laser ou ultrasons.

  Ils sont capables de mesurer la profondeur de l'objet qu'ils percoivent. En particulier, les données visuelles sont directement tridimensionnelles.

- *capteur passif* : utilisation des caméras CCD.

  Les données sont simplement une matrice de niveaux de gris, voire de couleurs, que l'on appelle une *image*.

On appelle respectivement système de *vision active* les systèmes utilisant des capteurs actifs et système de *vision passive* les systèmes utilisant des capteurs passifs.

Cependant, le terme vision active désigne aussi la vision avec contrôle du capteur, par exemple le déplacement de la caméra. Nous ne nous intéresserons pas du tout ici aux problèmes posés par la vision active, car c'est la vision passive qui rentre à proprement parler dans le domaine de vision artificielle.

La *vision monoculaire* est un domaine de la vision passive restreint à l'utilisation d'une seule caméra. Lorsque plus de 2 caméras sont utilisées, le système est appelé *stéréovision*. Nous nous intéressons dans cette thèse principalement à la vision monoculaire.

Une caméra nous fournit donc une image bidimensionnelle constituée par les signaux lumineux émis ou réfléchis par l'environnement. Chaque élément de l'image, appellé *pixel*, est un codage complexe de lumière, reflexion, orientation et distance des objets. Une telle image présente une ambiguïté massive (voir la figure 1.1) due à la projection durant laquelle les informations de profondeur sont perdues, ce qui soulève la difficulté majeure de la vision monoculaire.

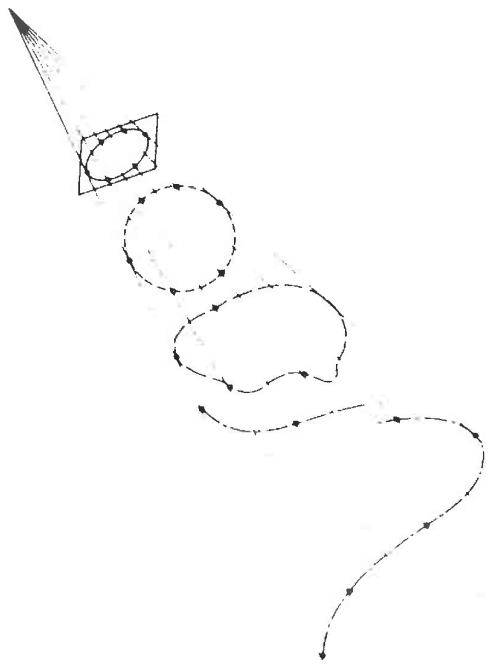


Figure 1.1. Une image monoculaire est massivement ambiguë [Barrow 81].

Si l'on regarde des objets de la scène, ils peuvent être statiques ou dynamiques et rigides ou non-rigides. Nous supposerons ici que les objets sont statiques et rigides.

Dans cette thèse, nous n'avons pas l'intention d'attaquer le problème d'apprentissage, qu'il soit géométrique ou symbolique. Ce dernier reste un problème difficile dans l'état de l'art actuel. Les difficultés du sujet font qu'il est encore prématuré de les conjuguer à celles de la vision monoculaire. Le problème de la reconnaisance et celui de la localisation reviennent en fait au même problème, celui de l'*appariement*. En effet, l'appariement des données visuelles avec la représentation de l'objet dans le modèle permet d'effectuer la reconnaissance des objets d'une part et la localisation du robot d'autre part. C'est donc à ce problème d'appariement que nous nous intéressons dans le contexte d'une vision monoculaire pour des objets statiques et rigides.

1.2 Comment faire ?

1.2.1 Méthode basée sur la reconstruction tridimensionnelle

Dans le schéma classique et dominant (illustré par la figure 1.2) d'aborder le problème, l'objectif de la première phase de la vision avant l'appariement est de parvenir à une représentation de l'image dite de *2,5-dimensional sketch*[1]. Cette représentation est essentiellement la reconstruction tridimensionnelle des indices visuels, plus précisément l'orientation et la profondeur relative au point de vue d'une surface visible. Le repère de cette description est lié à l'observateur. L'appariement ou la reconnaissance s'effectuera ensuite dans le domaine tridimensionnel sur des caractéristiques tridimensionnelles reconstruites.

L'essentiel concernant cette approche est donc de trouver les méthodes algorithmiques permettant de reconstruire ces caractéristiques *3D* qui ne sont pas explicitement présentes dans l'image.

La plupart des techniques de reconstruction commencent par l'extraction des changements d'intensité dans une image et les présenter de façon convenable. Le terme *primal sketch*[2] est utilisé par Marr [Marr 82] pour désigner une telle représentation *2D*. Dans [Marr 82], on extrait d'abord *raw primal sketch*, c'est-à-dire des changements de niveaux de gris qui sont considérés comme les indices visuels les plus importants d'une image à tout point de vue. Ensuite, en sélectionnant et regroupant ces indices visuels, on obtient une description dite de *full primal sketch* de l'image.

Par la suite, de nombreuse méthodes sont proposées pour la reconstruction des

1. *Croquis 2,5D* traduction utilisée dans [Lux 85].
2. *Croquis élémentaire* traduction utilisée dans [Lux 85].

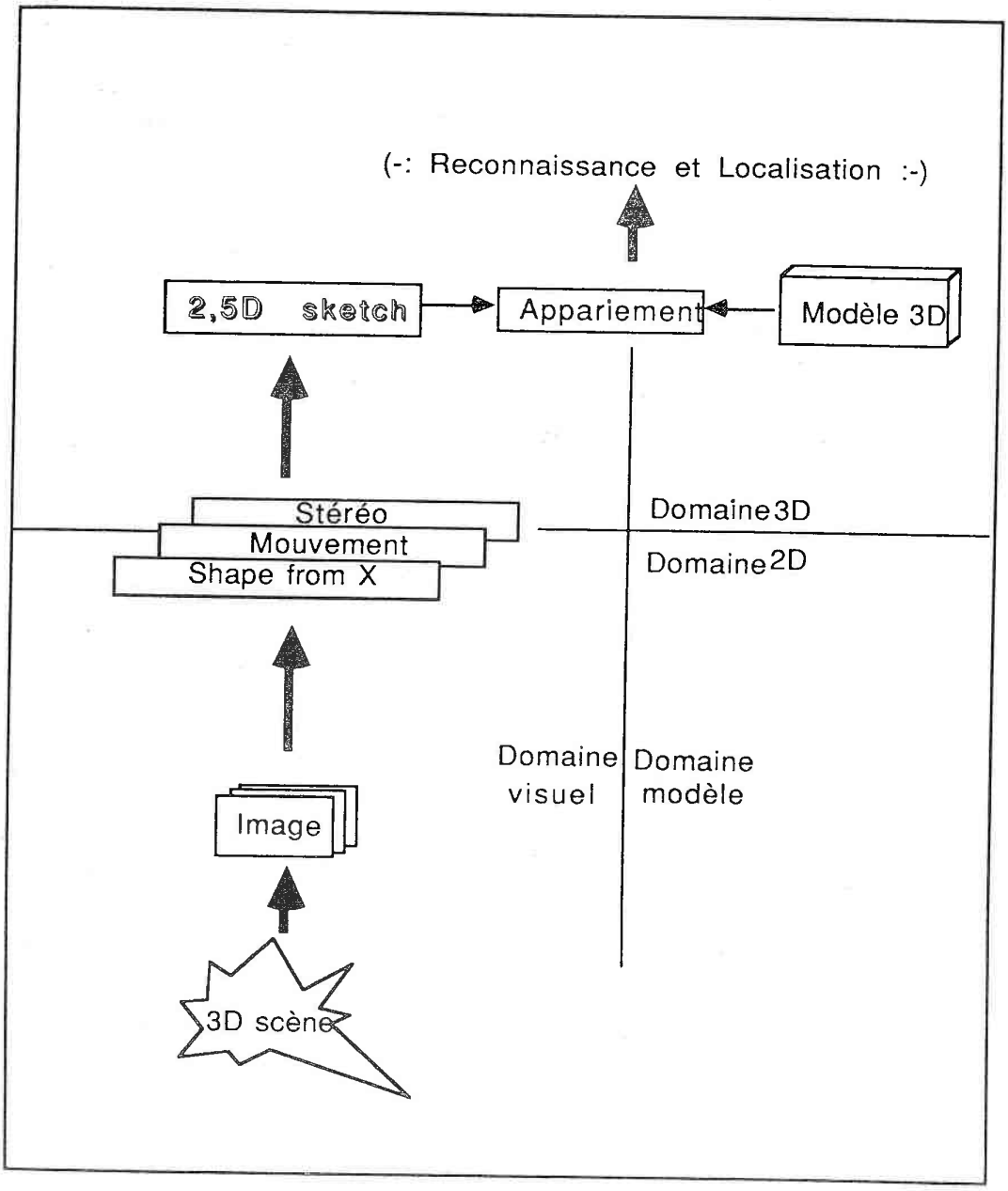


Figure 1.2. Le schéma général d'un système de vision basé sur la reconstruction tridimensionnelle.

caractéristiques *3D* de type *2,5D sketch.*[3]

Méthodes monoculaires : shape-from-X

1. *Shape-from-contour* (voir [Barrow 81] et [Brady 84]) essaye de déduire une représentation des surfaces visibles dans une image, en particulier la détermination de l'orientation des surfaces définies par leur vecteurs normaux locaux à partir des contours extraits de l'image. Pour des contours de type extrême (cf. section 1.3 pour la classification des contours), on obtient immédiatement une solution. Malheusement le problème est sous-contraint pour d'autres types de contours et la discrimination de différents types de contours est un problème ouvert. Pour avoir une solution à la fois stable et unique, il faut donc optimiser certains critères, par exemple symétrie et compacité. La solution ainsi obtenue est qualitative et heuristique.

2. *Shape-from-texture* [Ballard 82] déduit le *2,5D sketch* à partir des textures dans l'image. Une texture est l'organisation d'une surface de manière répétitive, uniforme et régulière. Il est évident que la distorsion de la texture par perspective peut fournir une telle représentation. Le grand problème reste qu'on n'a pas de bon moyen de discriminer des textures.

3. *Shape-from-shading* (voir [Ikeuchi 81] et [Horn 86]) consiste à déduire une représentation de type *2,5D sketch* (ou *needle map*) directement à partir de l'image de niveau de gris. Ceci est très compliqué car le niveau de gris ne dépend pas que de l'orientation de la surface. Il est une fonction complexe de la luminance, de la caractéristique de réflexion *etc.* Le problème est que même avec des conditions très simplifiées sur la luminance et la surface, il est difficile d'avoir une reconstruction satisfaisante. Actuellement, cette reconstruction ne peut être que très qualitative et grossière. Le récent travail de Pentland [Pentland 88] semble avoir abouti à de meilleurs résultats en utilisant une fonction de réflectance linéaire, mais en travaillant dans le domaine fréquentiel.

Stéréovision

Le principe de base de la stéréovision (cf. [Ballard 82,Ayache 88,WrobelDaucourt 88]) est d'utiliser deux images (ou plus) d'une même scène prise sous deux angles légèrement

3. Les différents termes pour décrire la représentation de type *2,5D sketch* sont proposés, notament *intrinsic image* dans [Barrow 81] et *needle map* dans [Horn 81].

différents. La première étape consiste à apparier des indices visuels homologues extraits des deux images. La connaissance de la disparité entre les deux images stéréoscopiques permet, par triangulation, de calculer la profondeur, donc de fournir une représentation complète de type *2,5D sketch.*

Mouvement

L'analyse du mouvement est également un moyen important de reconstruire des caractéristiques *3D* de type *2,5D sketch.* Il existe traditionnellement deux approches principales pour l'analyse du mouvement :

1. *mouvement discret* : cette approche consiste en la mise en correspondance des indices visuels entre les images et en la détermination du mouvement *3D* et de la structure à partir de ces correspondances. On peut se reporter aux [Faugeras 88,?]. La différence avec la stéréovision est qu'il faut en même temps estimer les paramètres de mouvement.

2. *flot optique* : cette approche consiste en le calcul du champ de l'image en déplacement, en supposant qu'il est la projection de l'actuel champ *3D* en déplacement, et en la détermination du champ *3D* à partir de celui-ci (cf. [Horn 81,?]). Les problèmes inhérents à cette approche sont que le flot ne peut être retrouvé à chaque pixel à cause de l'effet de fenêtre et des points sans gradient [Hildreth 84] et que le flot optique n'est pas en général la projection du champ *3D* [Verri 87]. Par conséquent, la reconstruction *3D* ne peut être que qualitative.

Conclusion

Le problème inhérent aux méthodes monoculaires shape-from-X est que la reconstruction est à la fois qualitative et non stable.

Le problème principal, en ce qui concerne la stéréovision et le mouvement est que trouver les correspondances peut ne pas être simple. Les résultats sont très sensibles aux bruits et aux erreurs lorsque les objets sont distants. Néanmoins, à l'heure actuelle, ces méthodes sont les plus efficaces et plus théoriquement rigoureuses que les méthodes monoculaires. De récents travaux entrepris dans l'équipe de vision de l'INRIA-Rocquencourt (cf. [Faugeras 88,?,?,?]) semblent indiquer que la stéréovision et le mouvement sont des méthodes robustes en utilisant des indices visuels de type segment de droite pour des objets rigides.

Etant donnés les nombreux problèmes sérieux posés par cette étape de reconstruction *3D*, surtout par les méthodes monoculaires shape-from-X, l'étape suivante sur l'appariement en terme des caractéristiques *3D* a été rarement réalisée ou approfondie. Sa faisabilité reste contestée, et réside probablement dans l'intégration d'informations multiples.

1.2.2 Méthode monoculaire basée sur le regroupement perceptuel

Un autre point de vue pour aborder le problème peut être schématisé par la figure 1.3. Il est cependant moins explicitement étudié que les approches précédentes. Il s'agit d'abord d'effectuer le regroupement des indices visuels dans l'image afin de constituer des indices à la fois stables et discriminants qui permettront d'accélérer la recherche. Ensuite il faut prédire le point de vue de l'image dans le modèle pour que l'appariement des indices visuels (que l'approche précédente propose d'effectuer en termes des caractéristiques 3D) avec des modèles d'objets soit effectué au niveau bidimensionnel, c'est-à-dire dans l'image. Dans ce schéma, la détermination de point de vue joue un rôle important, car il permet de projeter le modèle dans l'image en vue d'une vérification quantitative dans l'image.

C'est donc une approche complètement opposée à la précédente, qui se déroule principalement dans une image *2D* sans aucune reconstruction préliminaire des profondeurs et orientations locales des surfaces des données visuelles de l'image.

1.2.3 Pourquoi le choix ?

Avec ou sans la reconstruction *3D* ?

- Le système de vision basé sur la reconstruction *3D* a été proposé par David Marr (cf. [Marr 82]). Il argumente sur l'importance de *2,5D sketch* en démontrant le rôle important de la reconstruction tridimensionnelle dans la vision humaine. Pourtant la vision humaine possède également une très bonne performance de reconnaissance des images *2D*, tels que des dessins de trait. Un exemple typique est celui des dessins animés, dans lesquels il y a très peu de potentialité de la déduction tridimensionnelle ascendante [Charniak 85]. Les expériences psychologiques de [Hochberg 62] montrent aussi que la capacité de reconnaître des dessins est un phénomène inné chez l'homme. Le processus de regroupement perceptuel joue également un rôle fondamental dans la vision humaine selon les études de l'école de Gestalt [Wertheimer 58]. Il est donc discutable, que la reconstruction *3D* soit primordiale ou au moins soit la seule voie dans

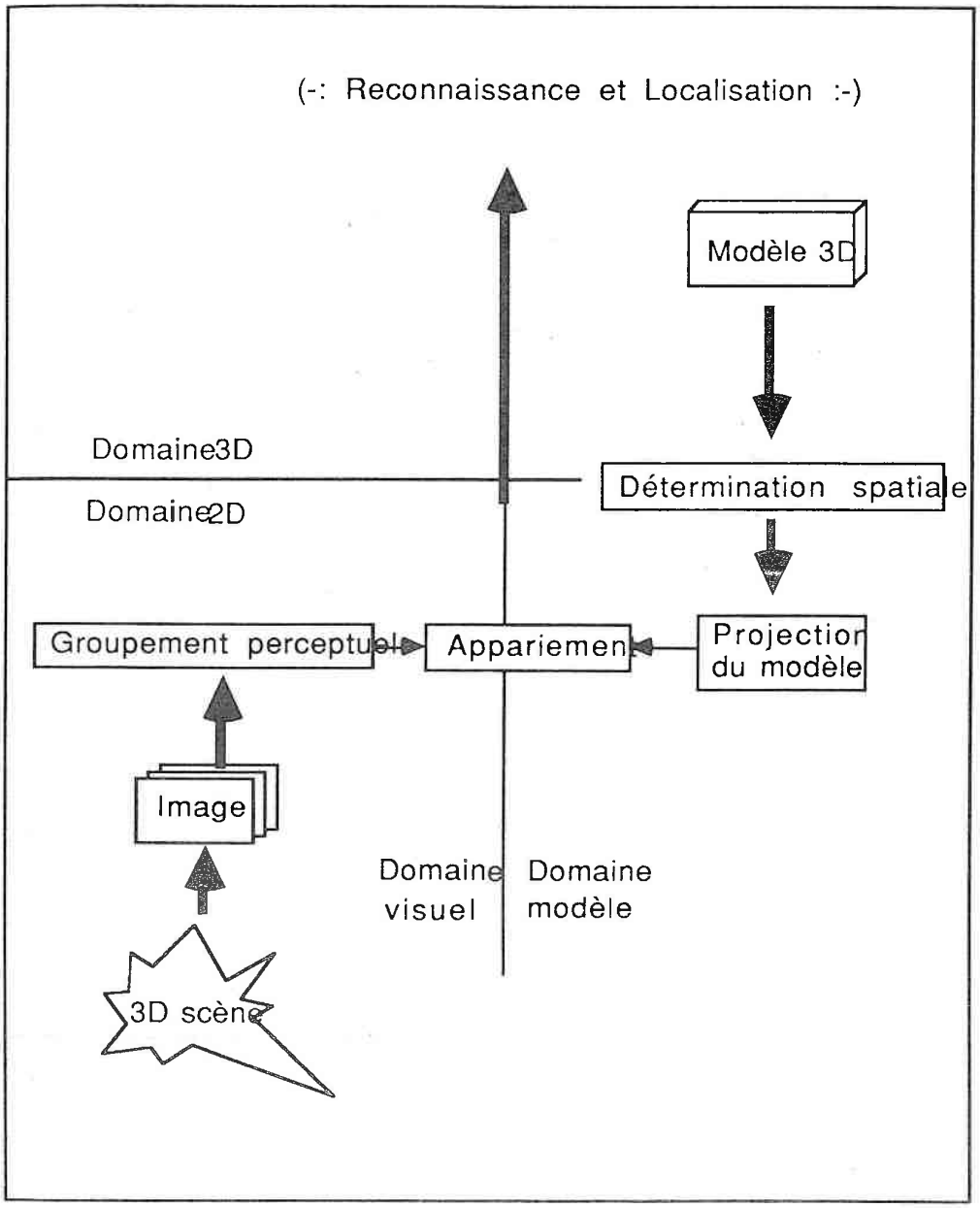


Figure 1.3. Le schéma du système de vision basé sur le regroupement perceptuel des indices visuels

la reconnaissance par la vision humaine.

- Il est discutable que la reconstruction tridimensionnelle facilite beaucoup la recherche de correspondance. Il est démontré [Grimson 84] que la reconnaissance en utilisant directement les informations *3D* provenant d'un capteur actif n'est pas une tâche facile. Le module d'appariement dans la plupart des systèmes de vision réalisés à l'heure actuelle fonctionne dans le domaine *2D*. La démarche de l'appariement de Marr effectué en termes des caractéristiques *3D* n'a pas été implantée et sa faisabilité reste contestée.

- Les informations *2D* sont plus fiables et plus précises que celle de *3D*. Les méthodes monoculaires *Shape-from-X* ne sont valables que dans les cas très particuliers et les résultats n'ont aucune garantie quantitative. Dans les cas de la stéréovision et le mouvement, la précision de la reconstruction tridimensionnelle des indices visuels est nettement inférieure à celle des localisations des indices visuels. En tout cas, la précision des informations *3D*, dérivées de leurs homologues *2D*, ne pourra jamais dépasser celle de *2D*. Un tel argument peut aussi être trouvé dans [Thirion 89] qui exploite des données stéréoscopiques.

On en conclut que l'appariement basé sur le regroupement perceptuel dans l'image présente beaucoup d'avantages par rapport à la méthode basée sur la reconstruction tridimensionnelle. Il est d'autant plus facile, à l'heure actuelle, de faire de l'appariement *2D*. Cette approche a été retenue uniquement en raison de l'appariement sans se préoccuper de la construction automatique du modèle. Il est néanmoins clair que, pour certaines tâches de vision, par exemple, l'apprentissage d'un robot dans un environnement inconnu, la reconstruction tridimensionnelle est quasi indispensable.

1.3 Contexte ORASIS

Ce travail s'intègre dans le cadre du projet PRC Communication Homme-Machine ORASIS[4] en collaboration avec CERFIA (Toulouse), CRIN-INRIA-Lorraine (Nancy), INRIA de Rocquencourt et Sophia-Antipolis, IRISA (Rennes), Laboratoire d'Electronique de l'Université de Blaise Pascal (Clermont-Ferrand) et LIFIA (Grenoble).

L'objectif principal est de développer la méthodologie de la vision de haut niveau pour que le robot (voir la figure 1.4) soit capable d'apprendre, reconnaître l'environnement

4. Objectif Robot Autonome et Système d'Intelligence Sensorielle, ORASIS signifie aussi vision en grec.

d'intérieur dans lequel il évolue et s'y localise.

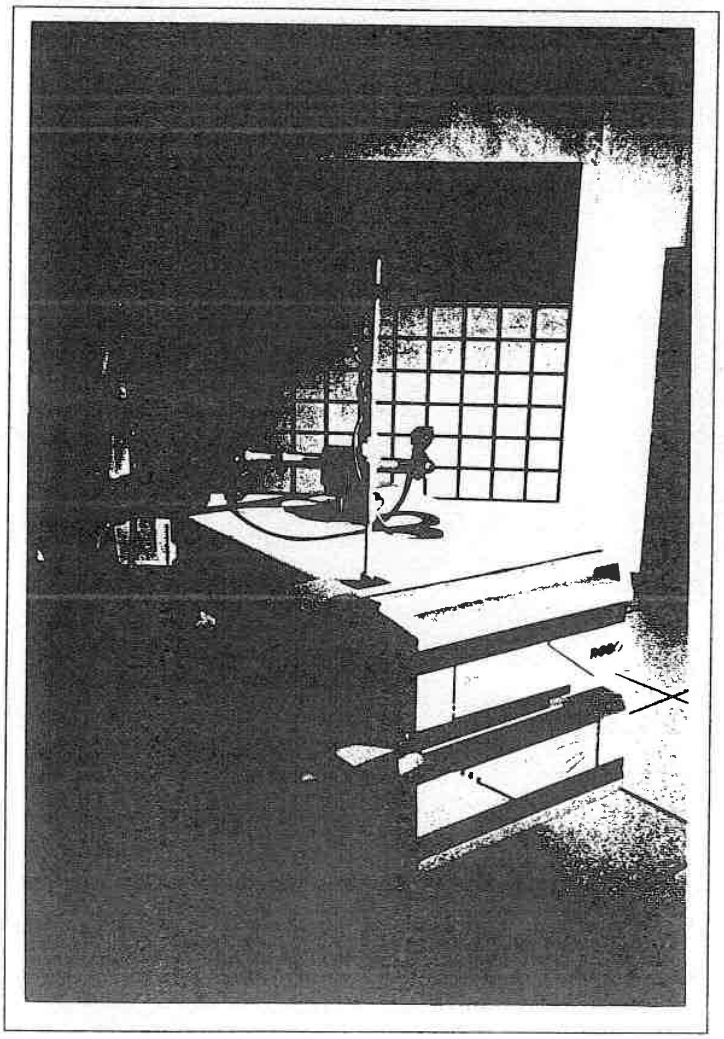


Figure 1.4. Le robot mobile construit à l'INRIA-Rocquencourt [Faugeras 87].

Cette thèse bénéficie de nombreux travaux developpés dans le projet ORASIS, notamment ceux de la vision de bas niveau.

- Le robot de l'INRIA fournit une base d'images. La figure 1.5 montre une image prise dans une scène d'intérieur.
- Segmentation d'images en contours :

  1. détection de points de contour : des points de contour sont extraits de

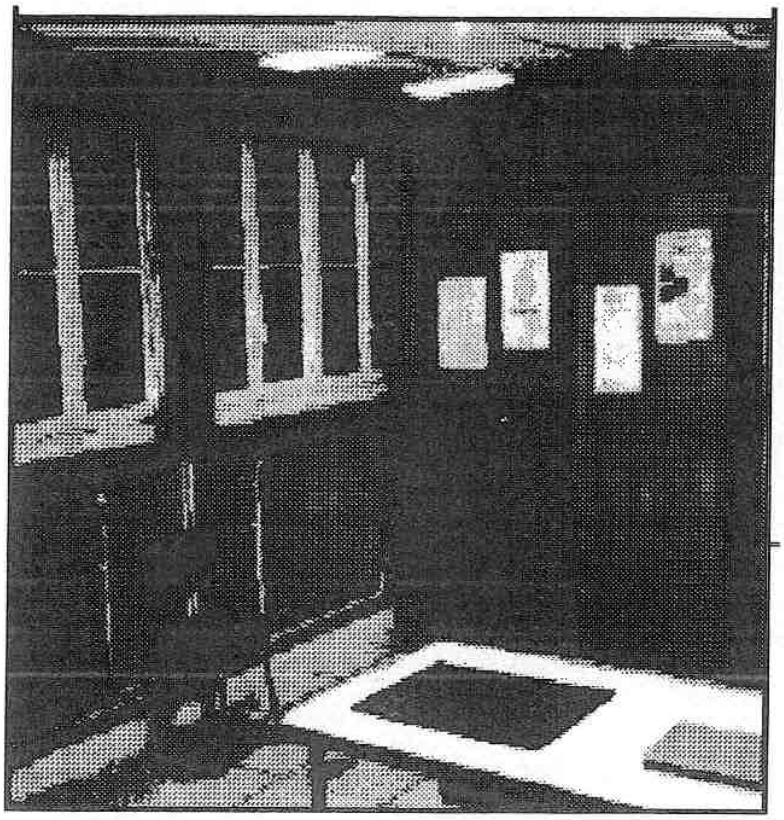


Figure 1.5. Une image d'une scène d'intérieur.

l'image aux endroits où il y a des changements de niveau de gris, passage par zéro du laplacien ou maxima locaux du gradient. Ceux-ci correspondent aux différents phénomènes physiques dans une scène réelle (cf. figure 1.6) :

- deux surfaces se rencontrent comme *plis* et *arêtes*,
- des bords tangentiels de surface comme *extrême*,
- discontinuité d'illuminance comme *ombres*,
- et discontinuité de réflectance comme *marques*.

Le détecteur de contours est développé par Deriche [Deriche 87] dérivé du travail de Canny [Canny 86].

2. chaînage de points de contour : les points de contour sont ensuite structurés en chaînes de contours. Le chaînage est effectué par la procédure de Gérard Giraudon [Giraudon 87]. Ce passage d'une image de niveau

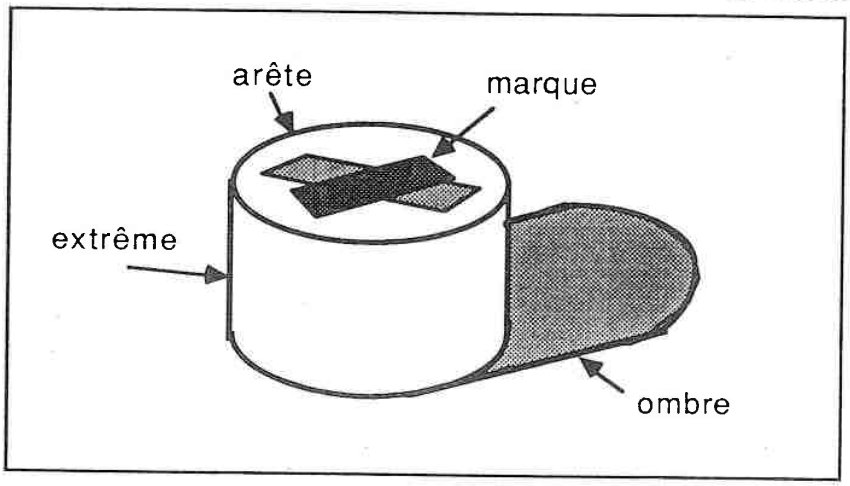


Figure 1.6. Différents types de contours.

de gris à ses chaînes de contour permet une première compression de l'information. De plus on a réalisé une structuration de l'image qui fait ressortir l'information pertinente.

3. approximation polygonale de contours : les chaînes de contours sont approximées par des chaînes de segments de droite, ce qui accroît encore la compression de l'information. L'algorithme, inspiré en partie des traveaux de Sklansky et Gonzalez [Sklansky 80] est mis au point par Marc Berthod.

- calibration de la caméra : le modèle de calibration détermine, pour chaque caméra, la transformation permettant de passer d'un point *3D* à sa projection dans l'image. A partir d'un point de l'image, il est possible de reconstituer la droite des points *3D* vérifiant l'opération inverse. Le modèle est la projection perspective sous forme d'une matrice de 3×4. Ce processus est réalisé par Giorgio Toscani [Toscani 87].

- reconstruction de segment *3D* : une représentation *3D* partielle de la scène observée est fournie sous forme de segments *3D* par le système de stéréovision trinoculaire, réalisé par Nicholas Ayache [Ayache 88].

1.4 Organisation de la thèse

Cette thèse est organisée de la manière suivante :

- Chapitre 2 : ce chapitre est consacré aux quelques rappels géométriques sur un certain nombre de propriétés de la géométrie projective et sur des opérations géométriques que nous utiliserons fréquemment.
- Chapitre 3 : c'est un chapitre technique consacré à l'extraction des informations perspectives telles que points de fuite et lignes d'horizon (cf. [Quan 87, Quan 89a]).
- Chapitre 4 : nous donnons un algorithme linéaire de détermination spatiale (cf. [Quan 89b]).
- Chapitre 5 : ce chapitre est consacré à l'organisation perceptuelle des indices visuels (cf. [Quan 88b,Quan 88c]).
- Chapitre 6 : nous discutons dans ce chapitre d'abord les approches existantes d'appariement, et ensuite nous décrivons la méthode que nous avons utilisée (cf. [Quan 89b,Mohr 88c,Quan 89c]).
- Chapitre 7 : ce chapitre est consacré à l'appariement des indices visuels dans deux images, qui constitue une application de l'approche d'appariement que nous avons employée (cf. [Quan 88a]).
- Chapitre 8 : cette thèse se termine par un chapitre de conclusion.

Les liens entre les chapitres et les données utilisées dans notre système peuvent être schématisés par la figure 1.7.

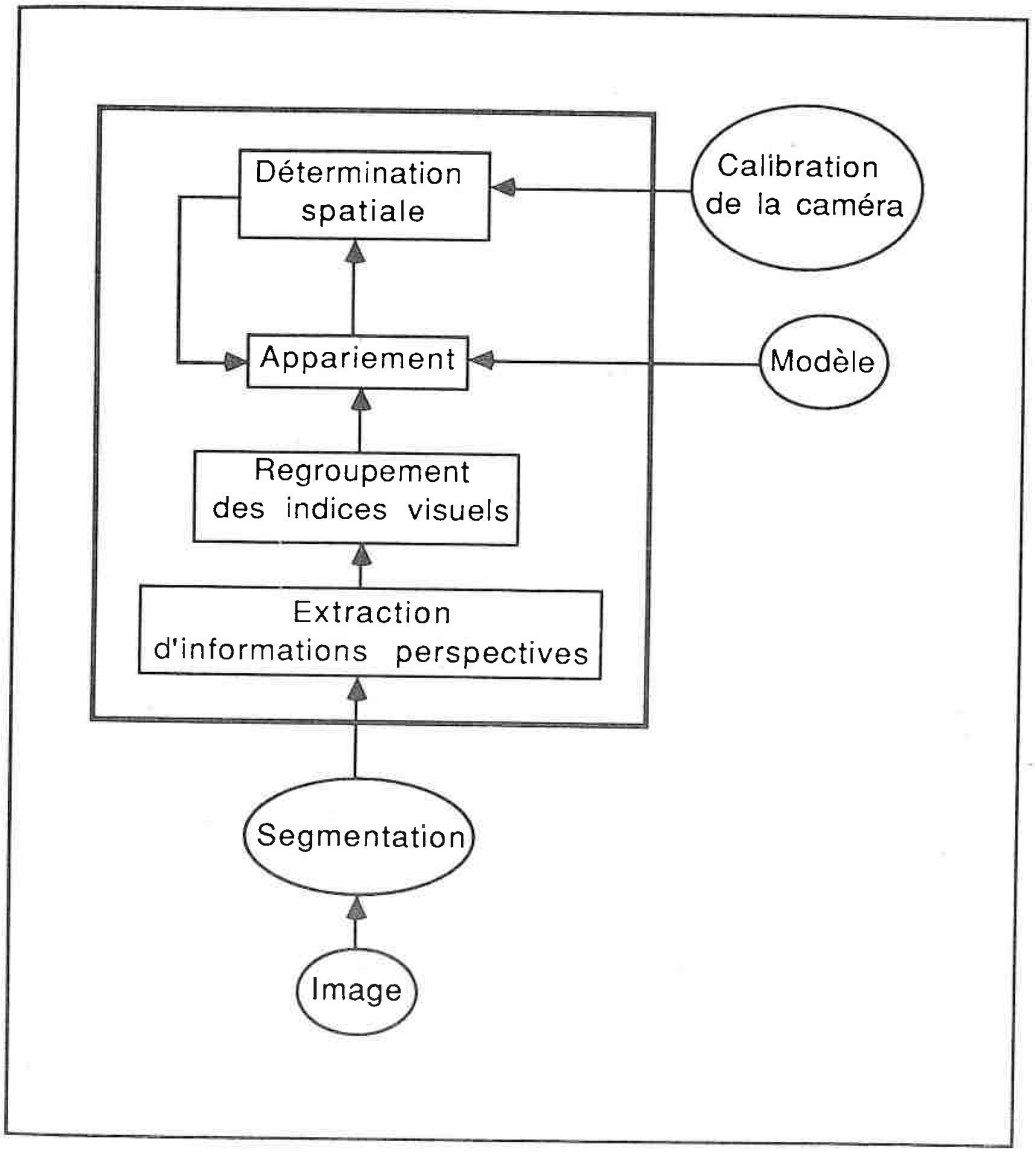


Figure 1.7. Système global.

2
Rappels géométriques

Ce chapitre comporte d'abord quelques rappels concernant un certain nombre de propriétés de la géométrie projective utilisées en vision monoculaire. Il est ensuite consacré aux opérations géométriques dans $\mathbb{R}^2$. qui serviront principalement au regroupement perceptuel dans le chapitre 4 et à l'appariement dans les chapitres 6 et 7.

Pour quelques références sur ces sujets, voir [Berger 77], [Tisseron 83], [Coxeter 80] et [Foley 82].

2.1 Etudes de quelques propriétés en géometrie projective

2.1.1 Espace projectif et coordonnées homogènes

On considère $\mathbb{R}^{n+1} - \{(0, ..., 0)\}$ que l'on munit de la relation d'équivalence

$$(x_1, ..., x_n, t) \sim (x'_1, ..., x'_n, t') \; \exists \lambda \neq 0 \text{ tq. } (x'_1, ..., x'_n, t') = \lambda(x_1, ..., x_n, t).$$

Par cette relation, on identifie $(x_1, ..., x_n, t)$ à $(x_1/t, ..., x_n/t)$ dans $\mathbb{R}^n$ pour $t \neq 0$. Si $t = 0$, on parle de points à l'infini dans la direction $(x_1, ..., x_n)$. On remarque, en faisant tendre t vers 0, qu'il correspond effectivement dans $\mathbb{R}^n$ à un point tendant vers l'infini. L'espace quotient $\mathbb{P}^n$ obtenu par cette relation d'équivalence est appelé *espace projectif* de dimension n. $(x_1, ..., x_n, t)$ sont appelés *coordonnées homogènes* de points de $\mathbb{P}^n$. Les points à l'infini sont ceux de coordonnées homogènes $(x_1, ..., x_n, 0)$.

Une *projection* est donc une application $p : \mathbb{R}^{n+1} \rightarrow \mathbb{P}^n$.

Dans le cas de $\mathbb{R}^3$ (resp. $\mathbb{R}^2$), on obtient l'espace projectif de dimension deux (resp. un), on parle alors de *plan projectif* (resp. de *droite projective*).

2.1.2 Projection perspective de $\mathbb{R}^3$

Soient un plan P et un point O de $\mathbb{R}^3$ n'appartenant pas à P. On appelle *projection perspective* (ou *projection centrale*) sur P de centre O, l'application qui envoie un point $M \neq 0$ de $\mathbb{R}^3$ sur l'intersection m de la droite OM avec P (cf. figure 2.1).

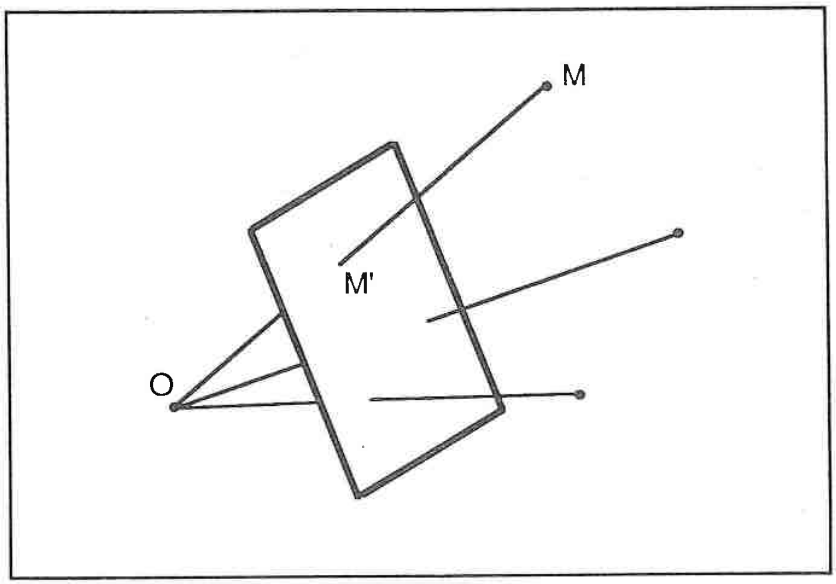


Figure 2.1. Projection perspective de $\mathbb{R}^3$.

Les *transformations homographiques* sont les transformations linéaires en coordonnées homogènes (envoyant des droites sur des droites) ; la projection perspective en est une. Une telle transformation ne conserve pas les proportions, en particulier le milieu de deux points n'est pas conservé par une telle projection (cf. figure 2.2).

2.1.3 Envoi de points à l'infini

Tout espace affine possède un complété projectif. Ce complété projectif est le point à l'infini pour une droite affine, et la droite à l'infini pour un plan affine.

Ainsi le *point de fuite* est la projection perspective du point à l'infini auquel des droites parallèles de $\mathbb{R}^3$ s'intersectent (cf. figure 2.3). Autrement dit, lorsque deux droites affines D et D' de $\mathbb{R}^3$ sont parallèles, leur direction commune *"représente* leur point d'intersection à l'infini. Chaque direction de droite apparaît ainsi comme *"représentant"* un point à l'infini, commun à toutes les droites ayant cette direction.

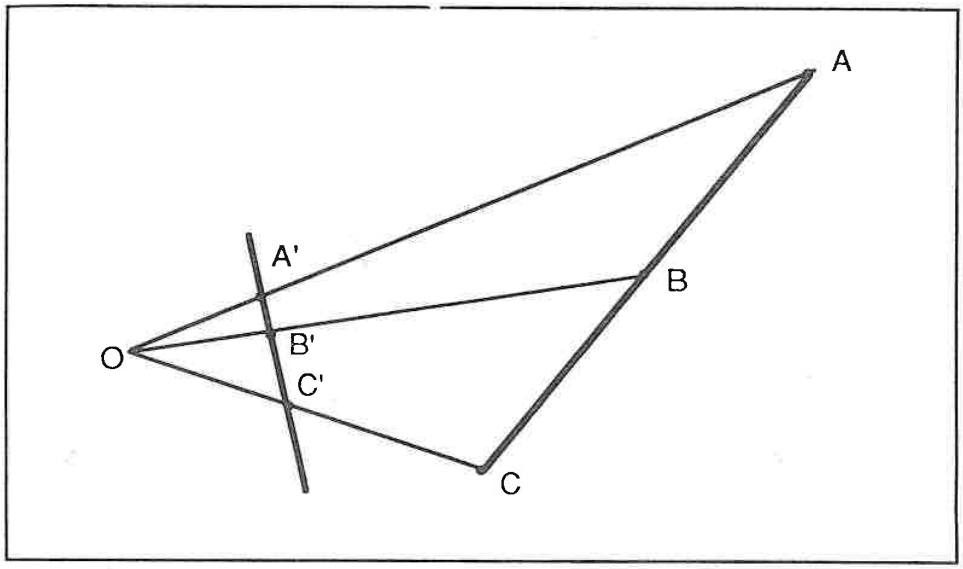


Figure 2.2. Le milieu de deux points n'est pas conservé par projection perspective.

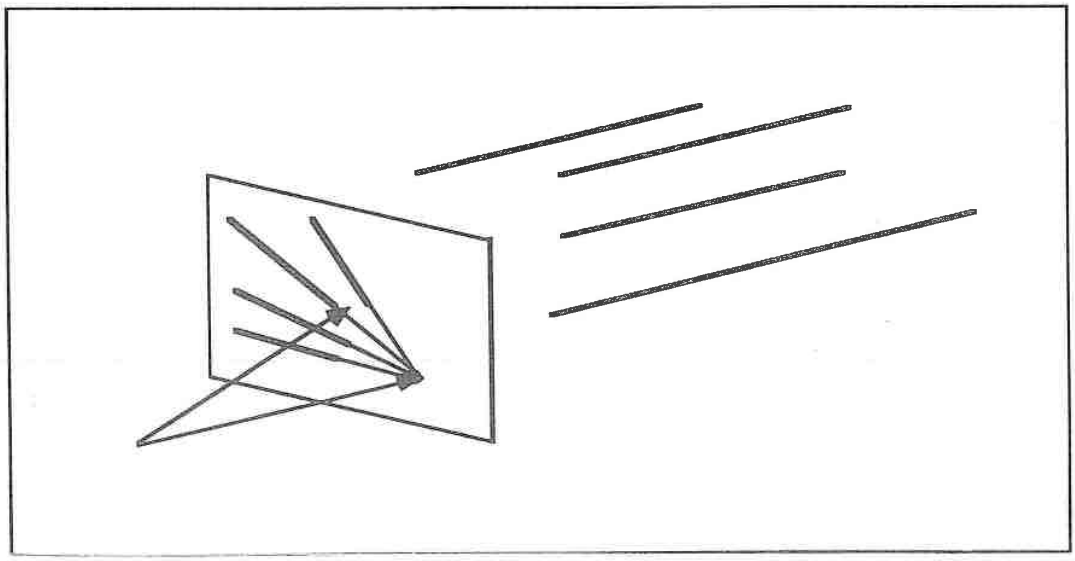


Figure 2.3. Point de fuite.

L'intérêt des points de fuite réside dans l'expression de directions de droites parallèles.

De même, la *ligne d'horizon* apparaît comme la projection perspective de la droite à l'infini, commune à tous les plans parallèles de $\mathbb{R}^3$.

2.1.4 Invariant projectif

Dans l'espace euclidien, la distance entre deux points est un invariant par le groupe des isométries. Dans l'espace projectif, les points alignés de $\mathbb{P}^n$ se conservent par homographies. De plus il existe un invariant des quadruplets (M, N, A, B) de points alignés distincts, c'est le *birapport*. Soient M, N, A, B quatre points distincts d'une droite projective, par définition le birapport, noté $[M, N, A, B]$, de ces points est :

$$[M, N, A, B] = \frac{\overline{MA}}{\overline{MB}} / \frac{\overline{NA}}{\overline{NB}}.$$

(cf. figure 2.4).

On a immédiatement les relations suivantes :

$$[M, N, A, B] = [M, N, B, A]^{-1} = [A, B, M, N]$$

$$[M, N, A, B] + [M, A, N, B] = 1.$$

Quand l'un des points est à l'infini, par exemple $B = \infty$, $[M, N, A, \infty] = \frac{\overline{MA}}{\overline{NA}}$. La connaissance du point à l'infini nous permet donc de travailler sur trois points seulement.

On dit que les points M, N, A, B sont en division harmonique si leur birapport vaut -1 ; M est alors appelé conjugué de B. Enfin, on peut noter que M, N, le milieu de M et N et le point à l'infini sont en division harmonique, c'est-à-dire que le milieu du segment est conjugué du point à l'infini.

2.1.5 Repère projectif et coordonnées projectives

Un espace vectoriel $\mathbb{R}^n$ se repère avec une base de n points, un espace affine de dimension n avec un repère affine de $n+1$ points ; un espace projectif $\mathbb{P}^n$ se repère avec un repère projectif de $n+2$ points.

On appelle *repère projectif* de $\mathbb{P}^n$ la donnée de $n+2$ points de $\mathbb{P}^n$ dont $n+1$ quelconques sont toujours projectivement indépendants.

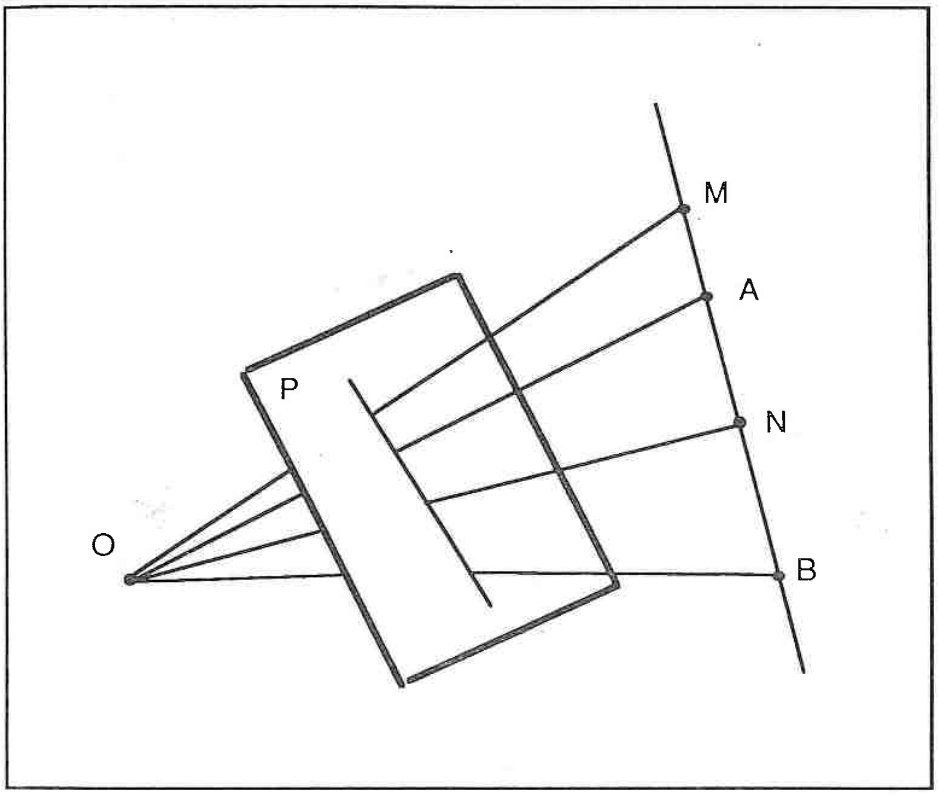


Figure 2.4. L'invariant projectif : birapport.

Soient $e_0, ..., e_n$ une base fixée dans $\mathbb{R}^{n+1}$, alors $p(e_0), ..., p(e_n), p(e_0 + ... + e_n)$ est un repère projectif de $\mathbb{P}^n$.

La donnée d'un repère projectif de $\mathbb{P}^n$ permet donc de lui associer des systèmes de coordonnées homogènes, puisque le choix de la base convenable est défini à une homothétie près. Un système de coordonnées homogènes $(x_1, ..., x_n, t)$ d'un point $m \in \mathbb{P}^n$ sera dit *coordonnées projectives* de m par rapport au repère projectif considéré.

Il faut donc trois points pour repérer une droite projective $\tilde{D}$.

Soit N, A, B trois points distincts de $\tilde{D}$, le birapport $[M, N, A, B]$ est la coordonnée projective de M dans le repère projectif N, A, B.

Ces outils permettent de formaliser l'appariement de segments de deux droites projectives.

2.1.6 Topologie d'un espace projectif

Soit S la sphère unité pour une norme sur $\mathbb{R}^{n+1}$. On définit une topologie de l'espace projectif $\mathbb{P}^n$ ($p(\mathbb{R}^{n+1})$) en posant la distance

$$\delta(M, M') = \arccos \mid x \cdot x' \mid$$

où $M, M' \in \mathbb{P}^n$, $x, x' \in S$.

Alors $\mathbb{P}^n$ s'identifie topologiquement au quotient de S par la relation d'équivalence $x = ty$, avec $x, y \in S$, $t \in \mathbb{R}$ et $\mid t \mid = 1$.

La topologie ainsi définie sur $\mathbb{P}^n$ est usuelle et $\mathbb{P}^n$ est compact.

Dans le cas de $\mathbb{R}^3$, la distance $\delta(M, M')$ est la longueur du plus petit arc de grand cercle joignant M et M' de la sphère unité S_2 dans $\mathbb{R}^3$. Un grand cercle est la trace d'un plan contenant le centre.

L'intérêt de S_2 réside dans le fait que $S_2 \cup \{(0,0,0)\}$ est un sous-espace vectoriel de $\mathbb{R}^3$ qui permet de mieux déplacer un observateur dans $\mathbb{R}^3$.

$S_2 \cup \{(0,0,0)\}$ est aussi appelée sphère gaussienne, et elle sera utilisée pour la détection de points de fuite.

2.2 Opérations géométriques de $\mathbb{R}^2$

2.2.1 Représentation paramétrique de droites affines de $\mathbb{R}^2$

Soient $A_1, ..., A_k$ des points de $\mathbb{R}^n$ muni de ce repère canonique. B a comme *coordonnées barycentriques* $(\lambda_1, ..., \lambda_k)$ si et seulement si

$$B = \sum_{i=1}^{k} \lambda_i A_i \text{ avec } \sum_{i=1}^{k} \lambda_i = 1.$$

Dans le cas de $\mathbb{R}^2$, soit M et N deux points, le lieu des points P_λ défini par

$$P_\lambda = \lambda N + (1 - \lambda)M, \quad -\infty \leq \lambda \leq +\infty$$

décrit la droite affine passant par M et N, que l'on note D_{MN} (voir la figure 2.5).

Cette représentation est appelée *représentation paramétrique* de D_{MN} munie du repère (M, N). On dit que λ est la *coordonnée paramétrique* (ou coordonnée barycentrique) de P dans le repère (M, N).

L'ensemble des points $\{P_\lambda, \lambda \in [0, 1]\}$ est le *segment* d'origine M et d'extrémité N et est noté $[M, N]$.

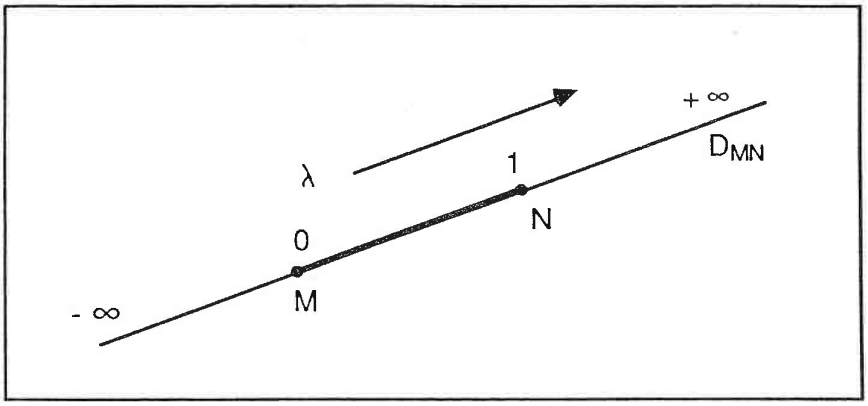


Figure 2.5. La représentation barycentrique de la droite

2.2.2 Opérations géométriques de $\mathbb{R}^2$

Opérations de base

- **Projection orthogonale** $p_\perp : \mathbb{R}^2 \times \mathbb{R}^2 \times \{\text{vecteurs de } \mathbb{R}^2\} \to \mathbb{R}^2$.

  La projection orthogonale d'un point P sur une droite passant par un point donné P_0 et parallèle à une direction donnée $\vec{z}$. (cf. figure 2.6).

$$P' = p_\perp(P, P_0, \vec{z}) = P_0 + (\vec{P_0P} \cdot \vec{z})\frac{\vec{z}}{\|\vec{z}\|^2}.$$

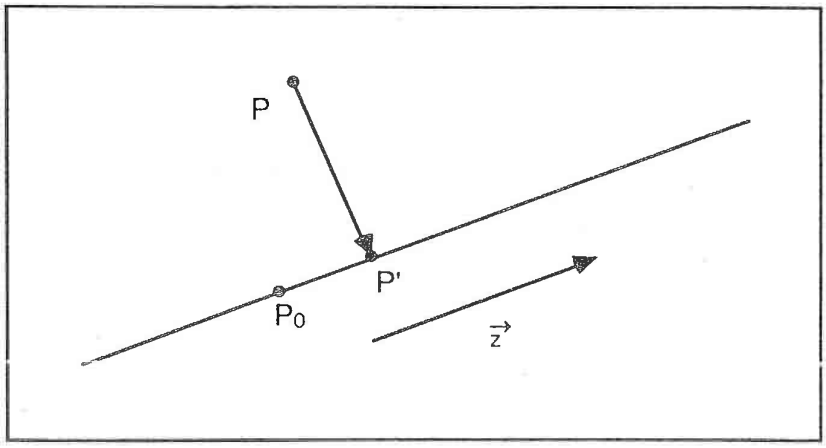


Figure 2.6. P' : la projection orthogonale de P.

- **Coordonnée paramétrique** $\Lambda : \mathbb{R}^2 \times \mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}$.

  La coordonnée paramétrique d'un point P d'une droite D_{MN} dans le repère (M, N) (cf. figure 2.7).

$$\lambda_p = \Lambda(P, M, N) = \frac{\overline{MP}}{\overline{MN}},$$

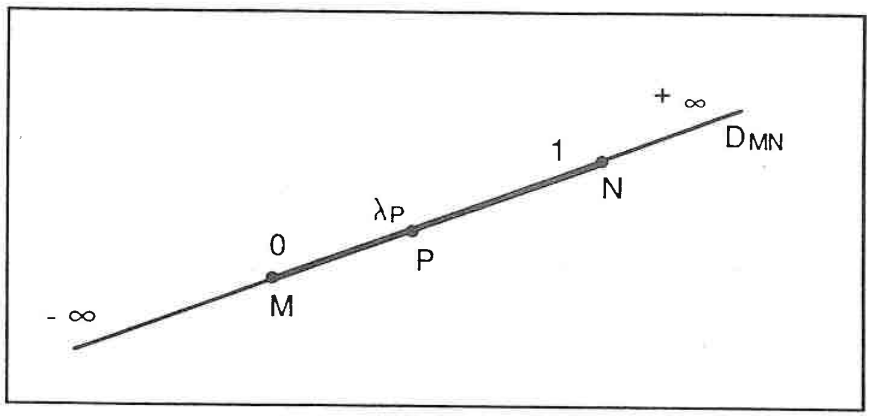


Figure 2.7. λ_P, la coordonnée paramétrique de P.

- **Recouvrement** $\bigcap : \mathbb{R} \times \mathbb{R} \times \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$.

  L'union ou le recouvrement de deux segments colinéaires $[M, N]$ et $[K, L]$ d'une droite D (cf. figure 2.8). λ_m, λ_n, λ_k et λ_l sont les coordonnées paramétriques de M, N, K et L dans un repère quelconque. Le résultat de l'opération est la longueur du recouvrement.

$$\bigcap(\lambda_m, \lambda_n, \lambda_k, \lambda_l) = \text{si } \max(\lambda_k, \lambda_l) > \min(\lambda_k, \lambda_l) \quad \text{alors} \quad \max(\lambda_k, \lambda_l) - \min(\lambda_k, \lambda_l)$$
$$\text{sinon} \quad 0.$$

- **Distance** $d : \mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}$.

  La distance euclidienne entre deux points A et B.

  On la note $d(A, B)$.

2.2.3 Opérations dérivées

- $d_{\dashv} : \mathbb{R}^2 \times \{\text{segments de } \mathbb{R}^2\} \rightarrow \mathbb{R}$.

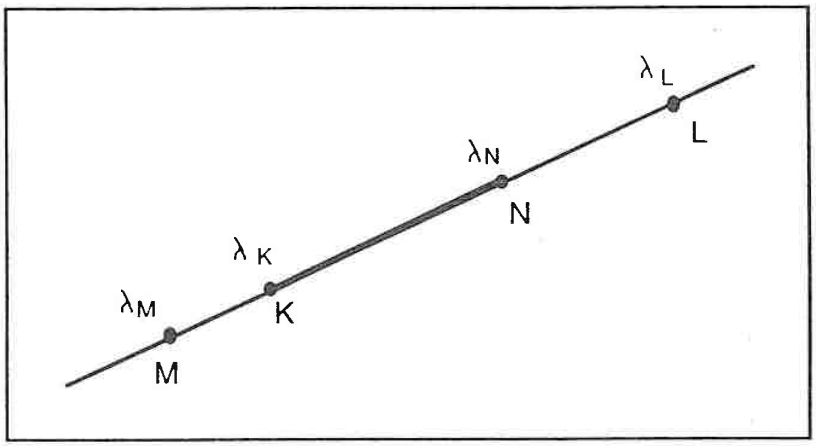


Figure 2.8. Le recouvrement de deux segments colinéaires.

La distance d'un point P à un segment $[M, N]$ (cf. figure 2.9).

$$d_{\dashv}(P,[M,N]) = \text{si } \lambda_P \in [0,1] \quad \text{alors} \quad d(P, p_\perp(P,M,\vec{MN})) \\ \text{sinon} \quad \min(d(P,M), d(P,N).$$

où

$$\lambda_P = \Lambda(p_\perp(P,M,\vec{MN}),M,N) = \frac{-((x_m - x_p)(x_n - x_m) + (y_m - y_p)(y_n - y_m))}{(x_n - x_m)^2 + (y_n - y_m)^2}.$$

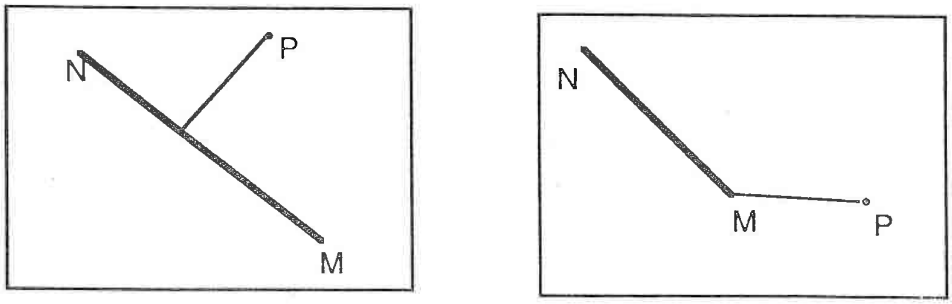


Figure 2.9. La distance d'un point à un segment.

- *dist* : {segments de $\mathbb{R}^2$} × {segments de $\mathbb{R}^2$} → $\mathbb{R}$

  La distance entre deux segment $[M, N]$ et $[K, L]$.

$$dist([M,N],[K,L]) = \min(d_{\dashv}(K,[M,N]), d_{\dashv}(L,[M,N])).$$

- $recouv_{\perp}$: {segments de $\mathbb{R}^2$} × {segments de $\mathbb{R}^2$} → $\mathbb{R}$.

  Recouvrement entre le premier segment $[M,N]$ et la projection orthogonale du deuxième $[K,L]$ sur la droite portant $[M,N]$ (cf. figure 2.10).

$$recouv_{\perp}([M,N],[K,L]) = \bigcap(0,1,\Lambda((p_{\perp}(K,M,\vec{MN}))),\Lambda((p_{\perp}(L,M,\vec{MN})))).$$

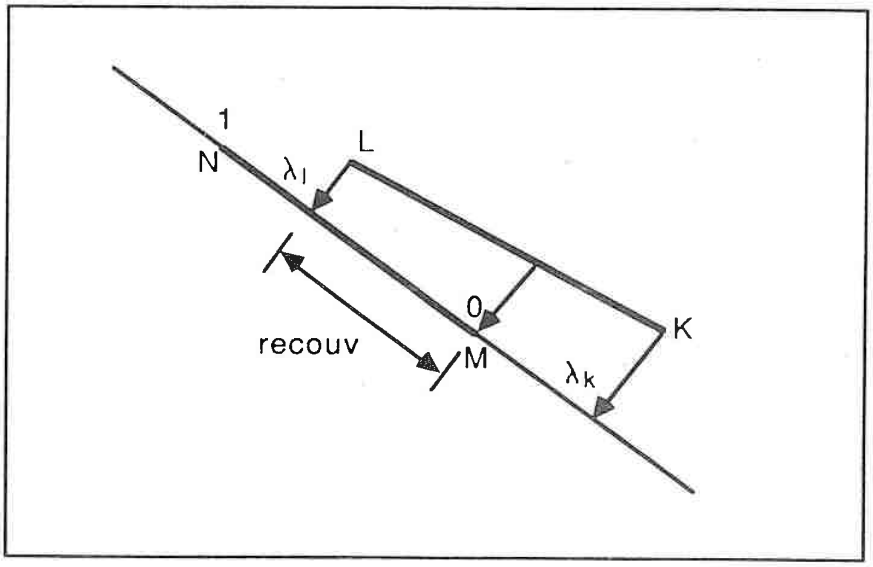


Figure 2.10. Le recouvrement orthogonal de deux segments.

- $recouv_{\parallel}$: {segments de $\mathbb{R}^2$} × {segments de $\mathbb{R}^2$} × {vecteurs de $\mathbb{R}^2$} → $\mathbb{R}$.

  Recouvrement entre les projections parallèles à une direction donnée $\vec{z}$ de deux segments $[M,N]$ et $[K,L]$ (cf. figure 2.11).

$$recouv_{\parallel}([M,N],[K,L],\vec{z}) = \bigcap(0,1,\Lambda((p_{\perp}(K,M,\vec{MN'}))),\Lambda((p_{\perp}(L,M,\vec{MN'})))).$$

  où $N' = p_{\perp}(N,M,\vec{z_{\perp}})$ et $\vec{z_{\perp}}$ est le vecteur orthogonal à $\vec{z}$.

- i : segment de $\mathbb{R}^2$ × segment de $\mathbb{R}^2$ → $\mathbb{R}$ × $\mathbb{R}$.

  L'intersection de deux droites D_{MN} et D_{KL} portant des segments $[M,N]$ et $[K,L]$. Cette intersection est sous forme d'un couple de coordonnées paramétriques $(\lambda_{MN},\lambda_{KL})$ respectivement dans les repères (M,N) et (K,L).

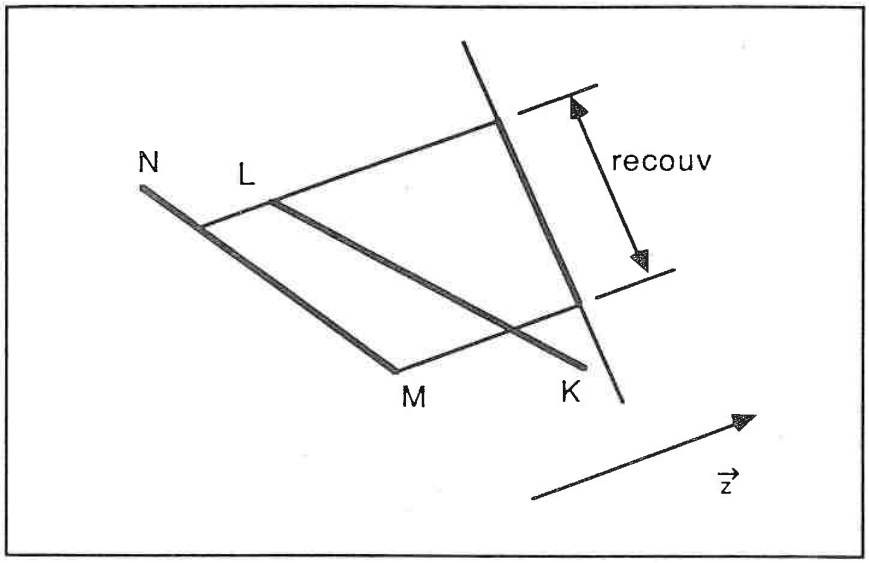


Figure 2.11. Le recouvrement de deux segments parallèle à une direction.

$$(\lambda_{MN}, \lambda_{KL}) = i([M,N],[K,L]) = (\frac{\vec{MN} \cdot \vec{KM}}{\vec{MN} \cdot \vec{KL}}, \frac{\vec{KL} \cdot \vec{KM}}{\vec{MN} \cdot \vec{KL}}).$$

Ce couple de coordonnées paramétriques permet de déterminer les positions relatives entre deux segments (cf. figures 2.12, 2.13, 2.14).

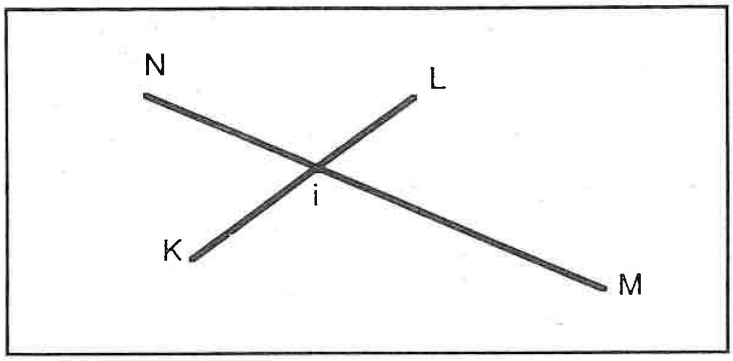


Figure 2.12. Quand $(0 \leq \lambda_{MN} \leq 1) \wedge (0 \leq \lambda_{KL} \leq 1)$, l'intersection se trouve à l'intérieur des deux segments.

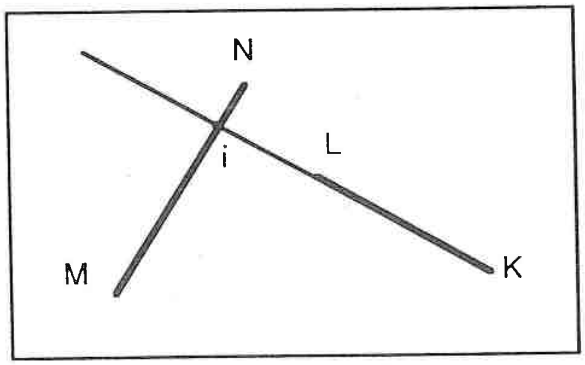


Figure 2.13. Quand $((0 < \lambda_{MN} < 1) \wedge (\lambda_{KL} < 0 \vee \lambda_{KL} > 1)) \vee ((0 < \lambda_{KL} < 1) \wedge (\lambda_{MN} < 0 \vee \lambda_{MN} > 1))$, l'intersection se trouve à l'intérieur d'un segment.

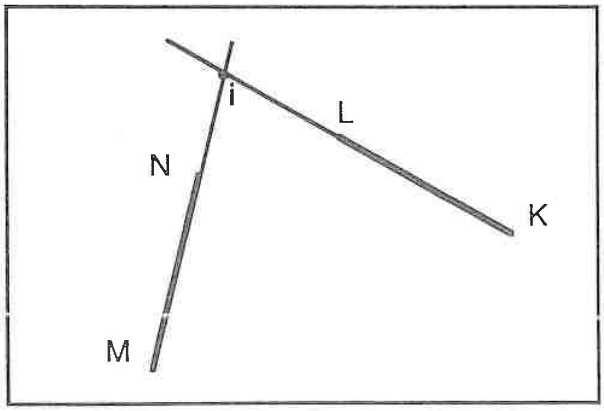


Figure 2.14. Quand $((\lambda_{MN} < 0) \wedge (\lambda_{MN} > 1)) \vee ((\lambda_{KL} < 0) \wedge (\lambda_{KL} > 1))$, l'intersection se trouve à extérieur des deux segments.

3
Extraction des informations perspectives

3.1 Introduction

Ce chapitre est consacré à l'extraction des informations perspectives, qui constitue un prétraitement pour le chapitre suivant sur le regroupement des indices visuels.

La projection perspective est une bonne modélisation de la vision humaine et de la perception des caméras, en particulier les caméras CCD offrant une faible distorsion géométrique. Certains chercheurs préfèrent une modélisation en projection parallèle, il ne s'agit là que d'une approximation de la réalité dans le but de simplifier le calcul puisque la projection parallèle est une application linéaire tandis que la projection perspective est une application homographique. De plus, l'utilisation de la projection parallèle risque de nous faire perdre la possibilité de récupérer certaines informations quantitatives.

Historiquement, la perspective a été étudiée par les artistes pour produire une meilleure représentation picturale de scènes *3D* sur un matériel plat. De nos jours, on l'étudie de façon inverse, c'est à dire en partant de la projection sur un matériel optique, afin de reconstruire la scène *3D* correspondante.

L'aspect mathématique de la perspective, telles que les notions de point de fuite et de ligne d'horizon, est illustré dans le chapitre 2. Nous étudions ici l'extraction de telles informations dans des scènes d'intérieur que l'on considère.

3.2 Recherche des points de fuite

Deux droites non-parallèles s'intersectent généralement en un point commun, nous ne pouvons donc pas en déduire le point de fuite. Alors trois droites concour-

antes sont-elles suffisantes ? On peut les interpréter comme la projection du point à l'infini de 3 segments parallèles dans l'espace ou bien comme la convergence accidentelle des projections de 3 segments dont les positions sont arbitraires dans l'espace. Cependant, plus de 3 segments de droite s'intersectent rarement s'ils ne sont pas parallèles. Donc quand il y a plus que 3 segments convergeant vers un point commun, cela donne une bonne indication de l'existence d'un point de fuite, surtout dans une scène d'intérieur bien définie ayant beaucoup de structures parallèles. En général, certaines connaissances *a priori* sur la scène sont indispensables pour obtenir une interprétation correcte.

Nous allons utiliser la méthode de la transformée de Hough pour effectuer cette recherche de points de fuite.

3.2.1 Principe de la transformée de Hough

La transformée de Hough [Duda 72] et [Muller 84] est une méthode générale permettant de détecter ou chercher certaines formes géométriques paramétrées (par exemple : droites, cercles etc.) dans un espace des paramètres. Un espace des paramètres est l'ensemble des paramètres des formes qui sont susceptibles d'exister dans l'image. Chaque indice de l'image vote pour chaque point de l'espace des paramètres. La méthode consiste donc à passer d'un espace des indices à un espace des paramètres. Cette application est souvent surjective, *i.e.* un point de l'espace des indices correspond à un ensemble des points de l'espace des paramètres. On cherche ensuite parmi l'espace des paramètres ceux qui ont obtenu *"suffisamment"* de votes, ces points là s'appelent les *points d'accumulation.* Une forme ou une propriété d'une forme est reconnue présente si, et seulement si, pour une certaine valeur du paramètre (un point de l'espace des paramètres), il y a suffisamment de points de l'espace des indices qui ont voté pour elle. Autrement dit, les points d'accumulation de l'espace des paramètres représentent les caractéristiques des points de l'espace des indices pour lesquels ils ont contribué au point d'accumulation. La recherche de formes est ainsi transformée en la recherche de points d'accumulation dans l'espace des paramètres. Cette technique est bien illustrée par le problème le plus fréquement rencontré en vision : la recherche de segments de droite dans une image (cf. [Duda 72] et [Muller 84]). Une discussion sur cette méthode sera vue en 6.

3.2.2 Choix de l'espace des paramètres

On peut directement chercher les points de fuite dans l'espace (U, V), le plan image, puisque le point de fuite est simplement le point concourant des segments de

l'image. Pourtant le problème est que, le plan image est un espace ouvert ; le point de fuite peut donc apparaître n'importe où, à un point déterminé ou à l'infini, ce dernier cas se produisant quand la direction de la caméra est perpendiculaire aux droites parallèles. Il serait néanmoins très difficile de concentrer l'espace de recherche d'une part, et la complexité du problème passe de $O(n)$ à $O(n^2)$ d'autre part car n segments produisent $O(n^2)$ intersections.

Nous pouvons prendre l'espace (r, θ) des coordonnées polaires comme celui des paramètres. Il est courament utilisé pour la détection des droites. Les segments ayant le même point de fuite sont projetés sur un cercle. Néanmoins l'espace (r, θ) reste ouvert et la détection des cercles nécessite un effort supplémentaire. Une amélioration consiste à transformer (U, V) en espace $(k/r, \theta)$ au lieu de (r, θ), où k est une constante. Le point à l'infini est projeté sur l'origine et les droites concourantes sur une droite. L'espace devient fermé mais une autre phase est nécessaire pour détecter les droites. Cette méthode à deux phases ne peut donc pas être très efficace.

La topologie de l'espace projectif que l'on a vue dans le chapitre 2 nous amène à considérer la sphère gaussienne, sphère unité centrée à l'origine du repère, comme l'espace des paramètres. Ce choix se justifie par la fermeture et la compacité de la sphère. Il a été déjà utilisé dans [Barnard 83]. L'application d'une sphère sur un plan est aussi connue sous le nom de la projection gnomonique [Coxeter 80], un point du plan produit deux points antipodaux dans la sphère, une droite produit un grand cercle.

3.2.3 Formulation du problème

Supposons que la caméra effectue une transformation perspective parfaite avec son centre optique à la distance f. Le repère lié à la caméra (x, y, z) a son origine O au centre optique. Le plan image est perpendiculaire à l'axe z à une distance f dans la direction positive et son origine à $(0, 0, f)$ comme les coordonnées dans le repère lié à la caméra. La sphère gaussienne est unitaire et centrée à l'origine (voir la figure 3.1). Nous justifions le choix de ce modèle simple et idéal plus loin.

Nous n'avons besoin que de la demi-sphère en face du plan image. Dans ce cas-là, la projection gnomonique est une application bijective entre les segments de l'image et les grands cercles de la sphère. Soit $S = [MN]$ un segment de l'image, $M = (U_1, V_1)$ et $N = (U_2, V_2)$ ses extrémités dans le repère image. $M = (U_1, V_1, f)$ et $N = (U_2, V_2, f)$ sont donc dans le repère lié à la caméra. Le *plan d'interprétation* d'un segment est le plan de $\mathbb{R}^3$ contenant le segment et l'origine (voir la figure 3.2). Soit $\vec{n} = (n_x, n_y, n_z)$ son vecteur normal, il est défini par :

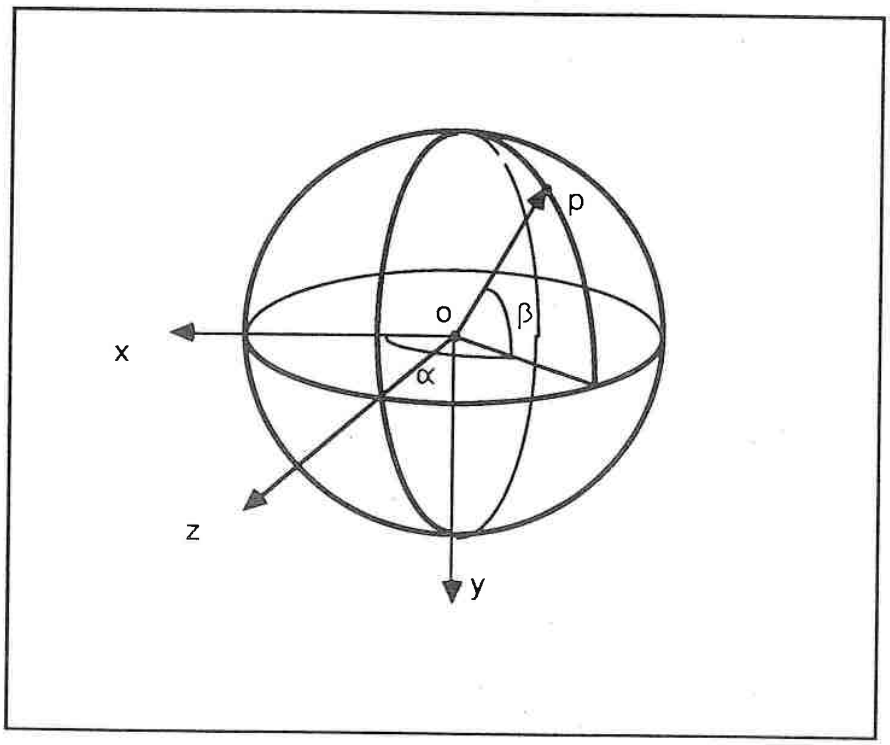


Figure 3.1. La sphère gaussienne

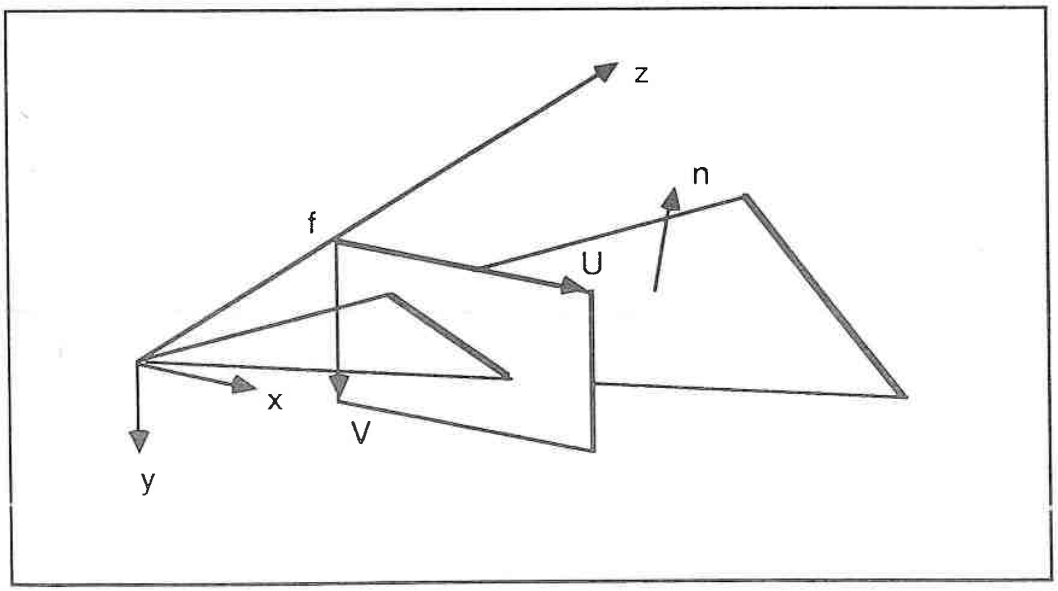


Figure 3.2. Le repère caméra, le repère image et le plan d'interprétation

$$\vec{n} = \frac{\vec{OM} \times \vec{ON}}{\| \vec{OM} \times \vec{ON} \|} \tag{3.1}$$

Un point de R^3 peut être exprimé, soit en coordonnées sphériques, soit en coordonnées cartésiennes. Supposons que (x, y, z) sont les coordonnées cartésiennes d'un point P de la sphère et (ρ, θ, ϕ) les coordonnées sphériques, où $\rho = 1$; θ est l'azimut, l'angle réalisé dans le plan xz par l'axe z et la projection de $\vec{OP}$ sur le plan xz ; ϕ est l'élevation, l'angle entre la projection de $\vec{OP}$ sur le plan xz et le vecteur $\vec{OP}$. On a la relation suivante :

$$\vec{OP} \ : \quad (x, y, z) = (\sin\theta\cos\phi, \sin\phi, \cos\theta\cos\phi).$$

Le plan d'interprétation dont le vecteur normal est $\vec{n}$ intersecte la sphère en un grand cercle. L'équation du cercle est :

$$\vec{n} \cdot \vec{OP} = 0,$$

C'est-à-dire

$$n_x \sin\theta\cos\phi + n_y \sin\phi + n_z \cos\theta\cos\phi = 0.$$

On obtient :

$$\phi = \psi(\theta) = \arctan(-\frac{n_x \sin\theta + n_z \cos\theta}{n_y}) \tag{3.2}$$

Ainsi une application de l'espace des indices (U, V), le plan image, dans l'espace des paramètres (θ, ϕ), la sphère, est établie.

Quand les points d'accumulation sont identifiés dans la sphère, pour localiser les points de fuite correspondants, cette opération inverse consiste simplement à résoudre l'équation :

$$\frac{f}{\cos\theta\cos\phi} = \frac{y}{\sin\phi} = \frac{x}{\sin\theta\cos\phi}$$

Nous obtenons :

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} f\tan\theta \\ f\frac{\tan\phi}{\cos\theta} \end{pmatrix} \tag{3.3}$$

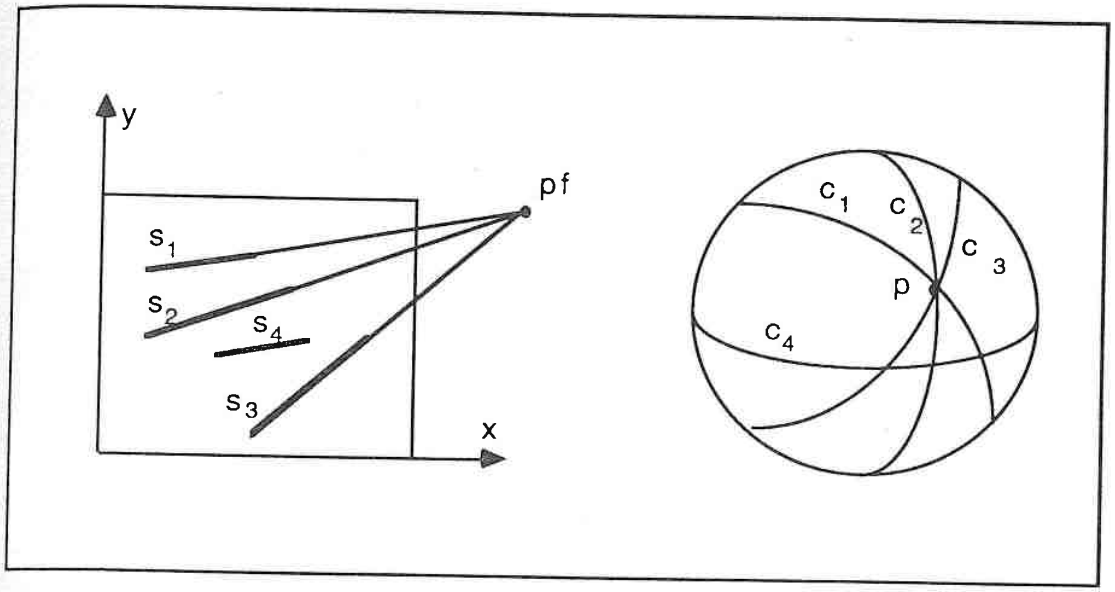


Figure 3.3. L'espace de départ et l'espace des paramètres

3.2.4 Implantation hiérarchique

Dans une implantation standard de la transformée de Hough, l'espace des paramètres se représente par un tableau. Les points d'accumulation sont identifiés par les éléments du tableau ayant le nombre maximum de votes. Barnard [Barnard 83] représente aussi la sphère comme un tableau de réels à deux dimensions. Chaque élément du tableau correspond donc à un petit morceau de la sphère. Cette partition est irrégulière : proche des pôles, le morceau élémentaire est beaucoup plus petit que celui proche de l'équateur. Une précision raisonnable oblige à diviser la sphère très finement. Ceci mène à un algorithme de recherche inefficace.

Généralement, la structure pyramidale permet d'introduire des algorithmes de type *"coarse-to-fine"* [Ballard 82] qui sont plus efficace que ceux avec un seul niveau et haute résolution.

On adapte cette structure pour la sphère, intitulée *sphère pyramidale*. Elle consiste à diviser récursivement un morceau de la sphère en 4 sous-morceaux ayant une résolution plus fine selon l'azimut et l'élevation.

La démarche de la recherche des points de fuite peut être décrite comme suit :

- Pour chaque segment, calculer la normale de son plan d'interprétation par la

formule 3.1 et lui associer un *score* qui s'estime comme la longueur relative (définie plus loin) du segment, on privilégie les segments longs car ils ont une direction plus fiable que les segments courts. Les segments trop courts dont la longueur est inférieure à la longueur minimale (typiquement 10 pixels) sont éliminés dans cette étape. La longueur relative est ainsi définie comme la longueur de ce segment modulo cette longueur minimale.

- Arrêter la division de la sphère si sa *taille* est inférieure à la taille minimale autorisée.

  Si un morceau s est caractérisé par l'ensemble convexe de l'intersection des quatre courbes $\theta_{sub}, \theta_{sup}, \phi_{sub}$ et ϕ_{sup} en coordonnées sphériques,

  $$s = (\theta_{sub}, \theta_{sup}, \phi_{sub}, \phi_{sup})$$

  La taille du morceau peut être calculée par :

  $$| \theta_{sup} - \theta_{sub} | \cdot | \sin \phi_{sup} - \sin \phi_{sup} |$$

- Diviser récursivement le morceau en 4 sous-morceaux.

  Cette division est toujours effectuée au centre du morceau père (voir la figure 3.4).

  $$s = s_1 \cup s_2 \cup s_3 \cup s_4 \tag{3.4}$$

  où

  $$s_1 = (\theta_{sub}, \theta_{mid}, \phi_{sub}, \phi_{mid}) \quad , \quad s_2 = (\theta_{sub}, \theta_{mid}, \phi_{mid}, \phi_{sup})$$
  $$s_3 = (\theta_{mid}, \theta_{sup}, \phi_{sub}, \phi_{mid}) \quad , \quad s_4 = (\theta_{mid}, \theta_{sup}, \phi_{mid}, \phi_{sup})$$

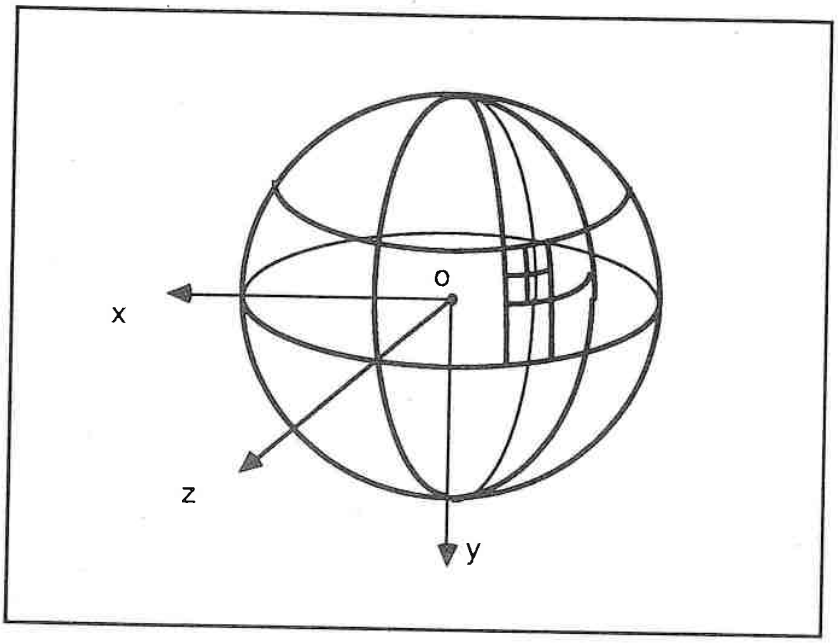
  où,

  $$\theta_{mid} = \frac{\theta_{sub} + \theta_{sup}}{2} \quad , \quad \phi_{mid} = \frac{\phi_{sub} + \phi_{sup}}{2}.$$

- Tester de quels sous-morceaux seront *membres* les segments appartenant au morceau courant. Pour ce faire, calculer

  $$\phi_1 = \psi(\theta_{sub}), \quad \phi_2 = \psi(\theta_{mid}), \quad \phi_3 = \psi(\theta_{sup}).$$

  Donc tester si un segment est un membre d'un sous-morceau revient à tester si deux segments $[\phi_1\phi_2]$ et $[\phi_2\phi_3]$ traversent ce sous-morceau (cf. figure 3.5).



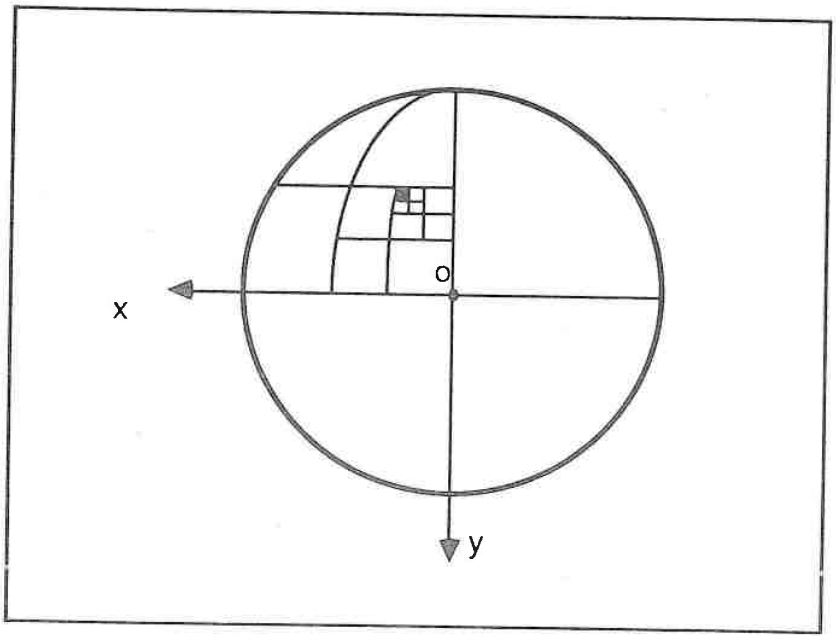


Figure 3.4. Sphère pyramidale.

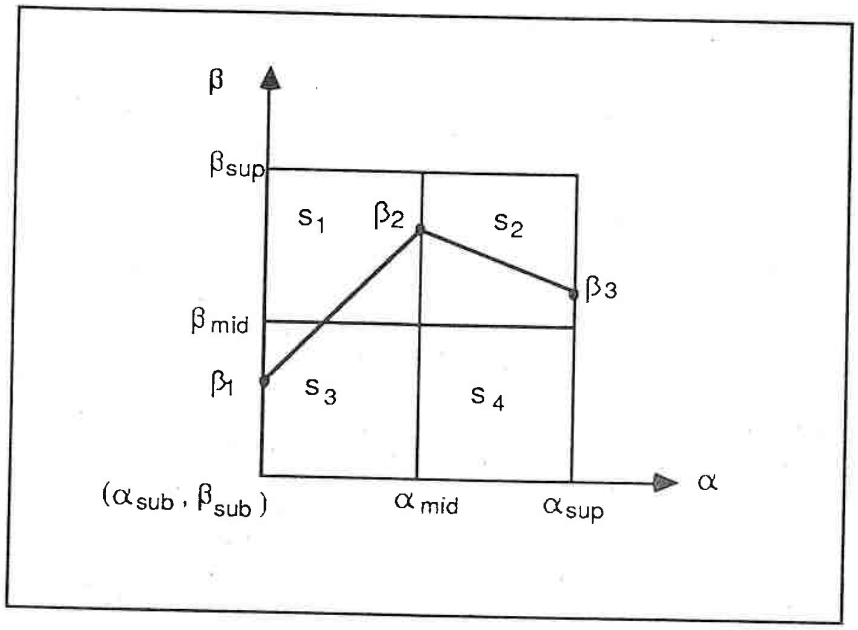


Figure 3.5. Tester si un segment est un membre d'un sous-morceau.

- Si le *score*, la somme de longueurs relatives des segments du morceau, d'un tel sous morceau est suffisant, affiner récursivement la recherche.

Tout ceci peut se résumer sous la forme d'un algorithme récursif :

/ Données : une liste de segments L et la demi-sphère S */*

/ Sortie : un point d'accumulation (S_max, L_max) */*

(S_max, L_max) = **Max** (**Score**(L_i)) pour toute la liste de candidats (S_i, L_i)
cherche (S : espace ; L : liste de segments) : liste de candidats
/ score(L) > min_score */*
si *taille*(S) < *min_taille* **alors** S et L font partie de la liste de candidats
    sinon $(s\_1, s\_2, s\_3, s\_4) := diviser(S)$
        pour $i := 1$ **a** 4 **faire**
            $L\_i = membre(S\_i, L)$
            si $score(L\_i) > min\_score$ **alors** $cherche(S\_i, L\_i)$
        finpour
finsi

On peut donc lancer cet algorithme avec l'espace de recherche de départ comme

la demi-sphère $(-\pi/2, +\pi/2; 0, \pi)$ On localise le point de fuite par la formule 3.3 avec *S_max*.

Cet algorithme donne donc un point de fuite chaque fois, pour en rechercher d'autres, il suffit de relancer cet algorithme avec une liste de segments privée de *L_max* et ainsi de suite.

3.2.5 Quelques remarques

Calibration de la caméra ?

Dans la formulation du problème, nous avons supposé que la modélisation de la caméra est une projection perspective parfaite avec son centre optique à la distance f. Pourtant ceci est loin d'être réaliste pour les caméras qui nous concernent. Le travail de Toscani [Toscani 87] a été consacré à la calibration d'une telle caméra. Alors pourquoi peut-on se passer de la calibration dans la recherche des points de fuite ? La réponse est tout simplement que la convergence est une propriété bidimensionnelle et que toutes les applications gaussiennes sont équivalentes quelque soient les paramètres de calibration en ce qui concerne la détection des points de fuite. Autrement dit, le point de fuite est une propriété invariante par rapport aux paramètres de calibration tels que les facteurs d'échelle, la distance focale, ou les coordonnées du centre de l'image (voir [Toscani 87]). Donc la valeur de la distance focale dans la formulation ci-dessus est complètement fictive, son choix peut être tout à fait arbitraire. D'ailleurs il est impossible de séparer la distance focale des facteurs d'échelle avec le modèle proposé par Toscani car ils ont toujours les mêmes effets projectifs.

La matrice de calibration ne serait nécessaire que si l'on veut inférer une direction spatiale à partir d'un point de fuite, car dans ce cas, les points de la sphère gaussienne représentent les vrais vecteurs de direction. C'est la raison pour laquelle dans la formulation de la recherche des points de fuite, nous avous pris une simple modélisation de la caméra avec une distance focale fictive au lieu de celle fournie par la matrice de calibration.

Néanmoins il est important de souligner que quand les points d'accumulation sont identifiés sur la sphère, la même distance focale fictive doit être prise pour localiser les point de fuite dans l'image.

Théoriquement la distance focale fictive est tout à fait arbitraire, mais en pratique, elle a une influence importante sur le résultat de détection. Ceci sera illustré dans les résultats expérimentaux.

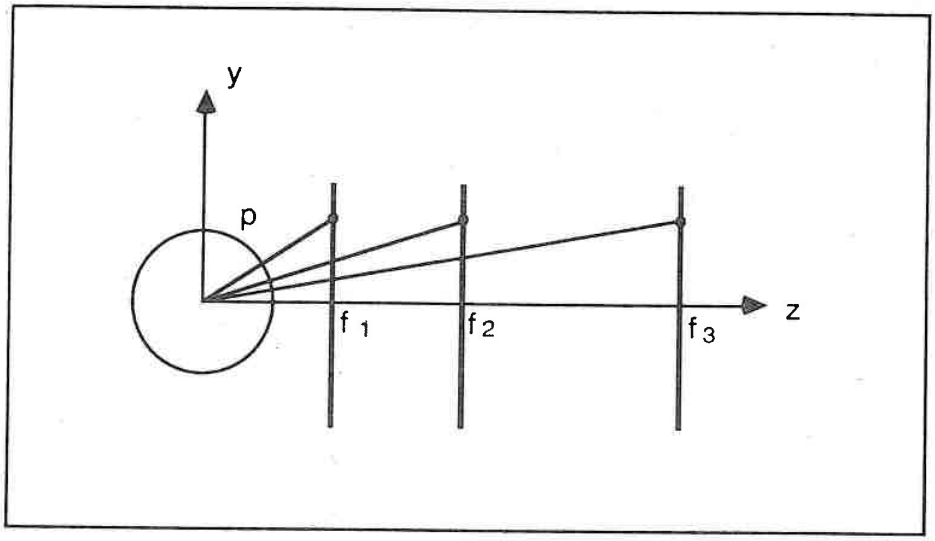


Figure 3.6. Toutes les applications gaussiennes sont équivalentes quelle que soit f

complexité

En ce qui concerne la complexité, une résolution de $1/m$ de l'espace des paramètres mène à $O(m^2)$ en temps de calcul et en espace de mémoire pour une implantation standard de transformée de Hough. Quant à la nôtre, l'arbre complet développé dans le pire cas coûte aussi $O(m^2)$ en temps et en espace. Notre approche parvient donc au même ordre de grandeur qu'une transformée de Hough. Pourtant en pratique, le pire cas est quasiment impossible ; puisque chaque nœud de l'arbre contient le nombre de votes, le développement s'arrête dès que ce nombre descend plus bas qu'un seuil fixé. Le pire cas signifie que ce seuil est nul, ce qui n'est d'ailleurs jamais vrai. D'autre part les faux points d'accumulation se dispersent très vite dans une résolution plus fine. Le développement de l'arbre se limite toujours à un petit nombre de branches à chaque étape de développement. Plus l'arbre est profond, moins il devient épais. Ceci fonctionne comme une récompense de l'explosion de l'arbre. L'analyse quantitative de la complexité de ce genre d'arbres reste un sujet difficile que nous n'avons pas abordé.

Le seuillage joue un rôle important dans le contrôle du développement de l'arbre. Il dépend aussi de la connaissance dont nous disposons *a priori* sur la scène. Le bon choix donne un algorithme quasi linéaire tandis qu'un mauvais choix en donne au pire un quadratique.

3.2.6 Détermination d'horizon

La détermination directe de l'horizon d'une surface est d'autant plus difficile que l'on ne possède pas la segmentation en régions. On ne sait pas dire qu'un segment donné appartient à une surface donnée. En revanche, étant donné un groupe de segments sur une surface, leur point de fuite doit se situer sur l'horizon de cette surface. Donc l'alignement des points de fuite spécifie la ligne d'horizon. En particulier, deux points de fuite définissent une ligne d'horizon.

La détermination de l'horizon revient ici à détecter d'abord les points de fuite de segments, ensuite lorsque ces derniers sont détectés, chaque couple (ou l'ensemble des points alignés) définit un horizon. Son interprétation juste a recours aux connaissances *a priori* sur la scène à interpréter. L'horizon terrestre est défini par deux points de fuite horizontaux.

3.2.7 D'autre travaux liés à la recherche des points de fuite

- Nakatani *et al.* [Nakatani 81] ont utilisé une méthode directe mais peu efficace qui passe de (U, V) à (r, θ).

- Barnard [Barnard 83] a proposé l'utilisation de la sphère gaussienne, pourtant sa représentation en est restée à un tableau à deux dimensions qui est une division irrégulière menant à une consommation importante en espace mémoire et en temps de calcul.

- Magee *et al.* [Magee 84] ont utilisé les produits vectoriels en tant que l'espace des paramètres. Ils ont aussi montré que la distance focale n'est pas nécessaire pour calculer des produits vectoriels. La méthode est inefficace car la sphère n'a été utilisée que pour la représentation des directions des produits vectoriels. Par conséquent, un problème en $O(n)$ se transforme systématiquement en un problème en $O(n^2)$ si n désigne le nombre des segments dans une image.

3.2.8 Résultats expérimentaux

Les figures 3.7, 3.8 et 3.9 montrent les groupes de segments convergents dans de différentes images. A chaque groupe est associé un point de fuite. Remarquons qu'une quatrième direction non-principale est également détecté dans la figure 3.8.

Les figures 3.10, 3.11 et 3.12 montrent l'influence du choix de la distance focale sur la même image.

Les figures 3.13 et 3.14 montrent les deux points de fuite horizontaux et la ligne d'horizon d'une image.

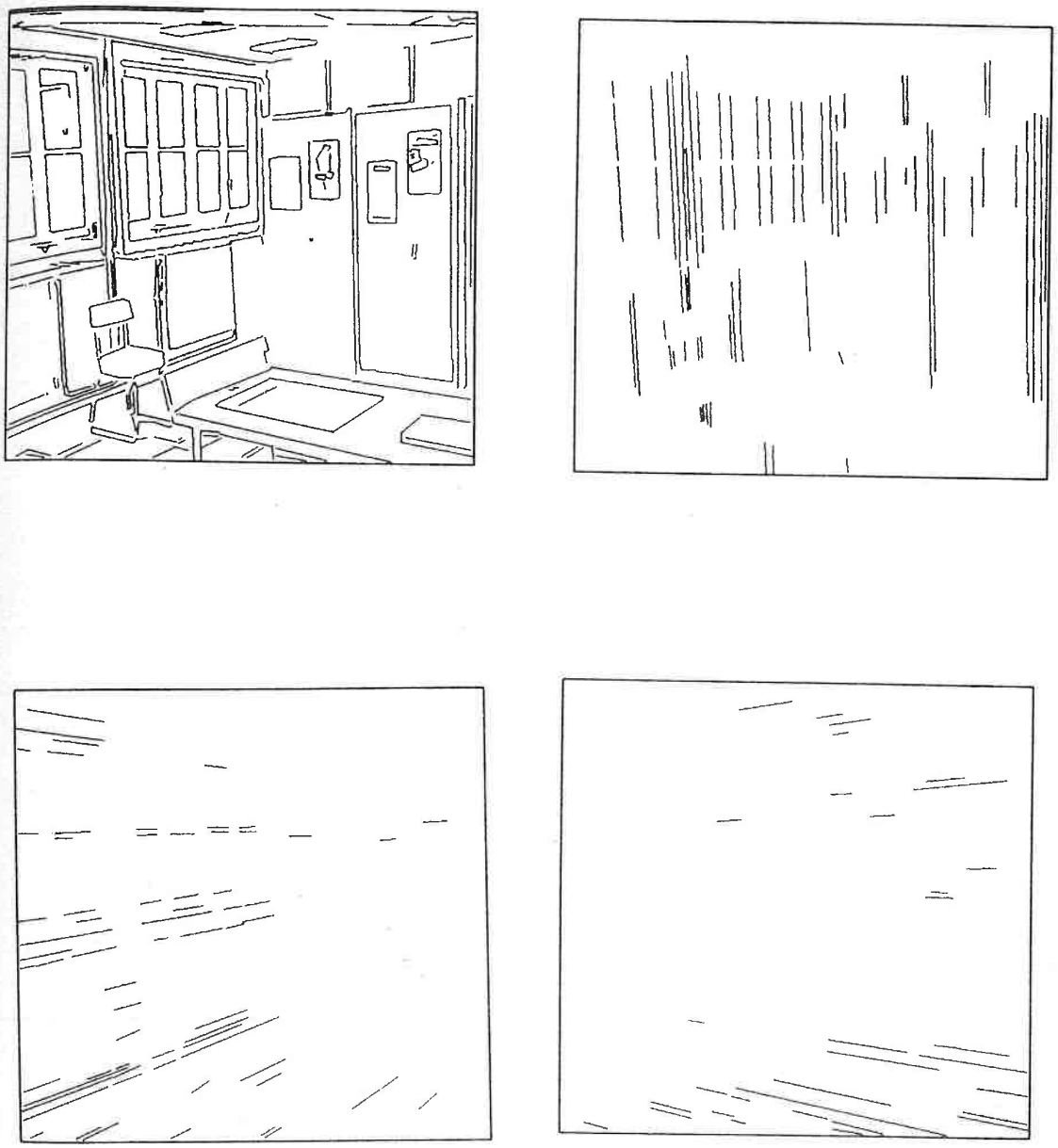


Figure 3.7. Position 75, $f = 10$.

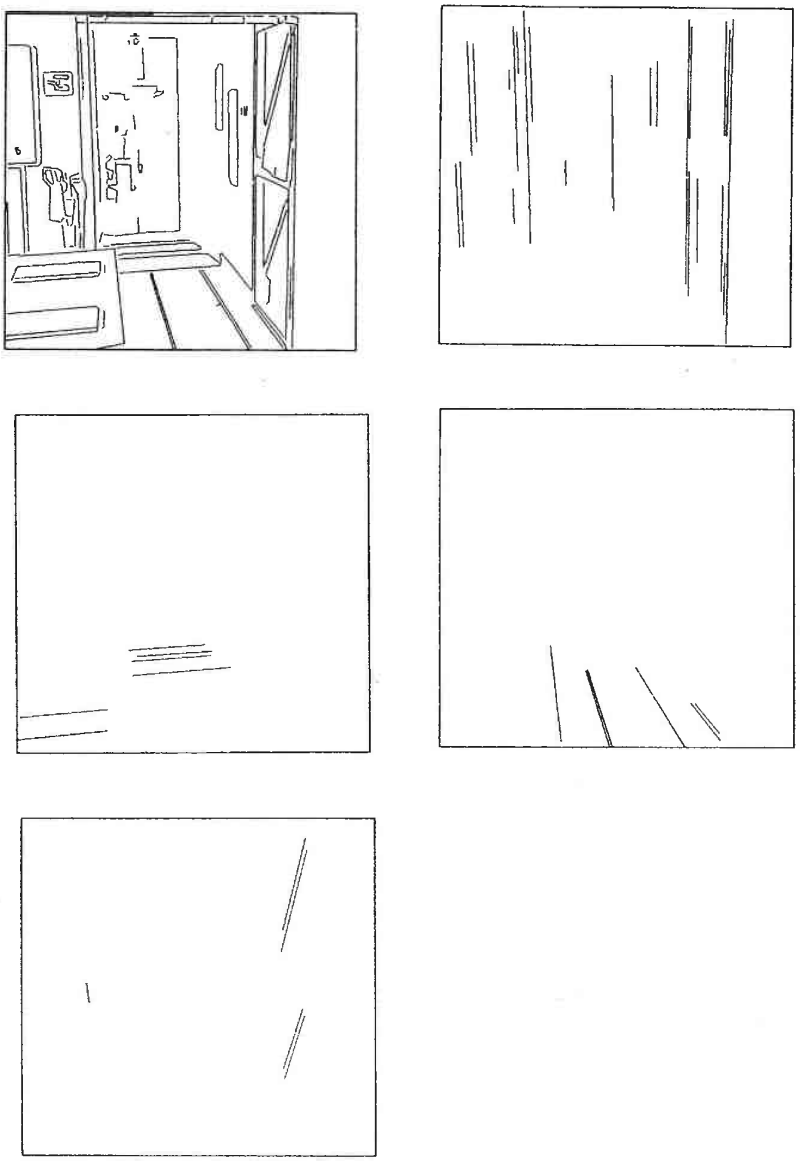


Figure 3.8. Position 30, $f = 1000$.

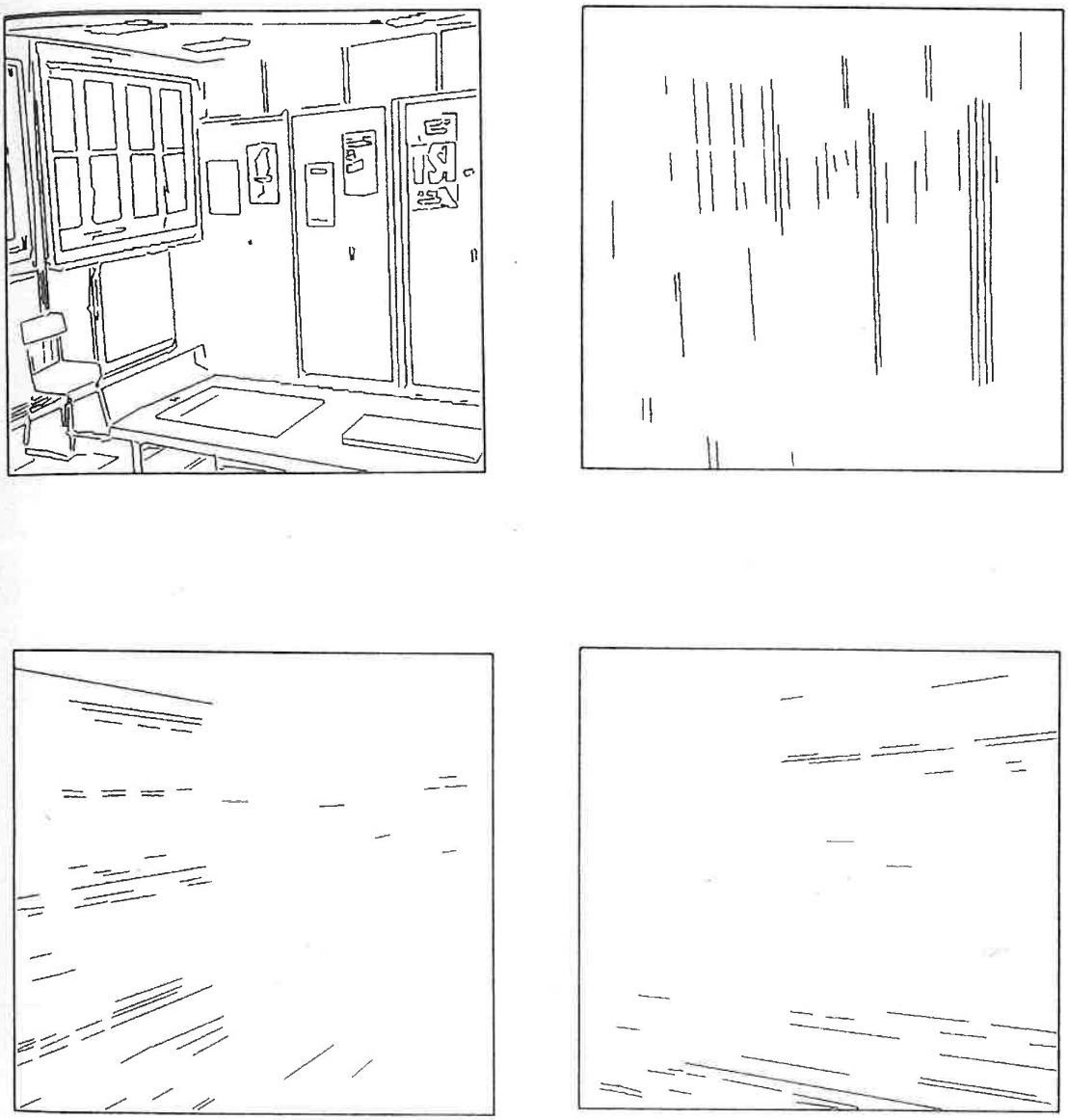


Figure 3.9. Position 70, $f = 8$.

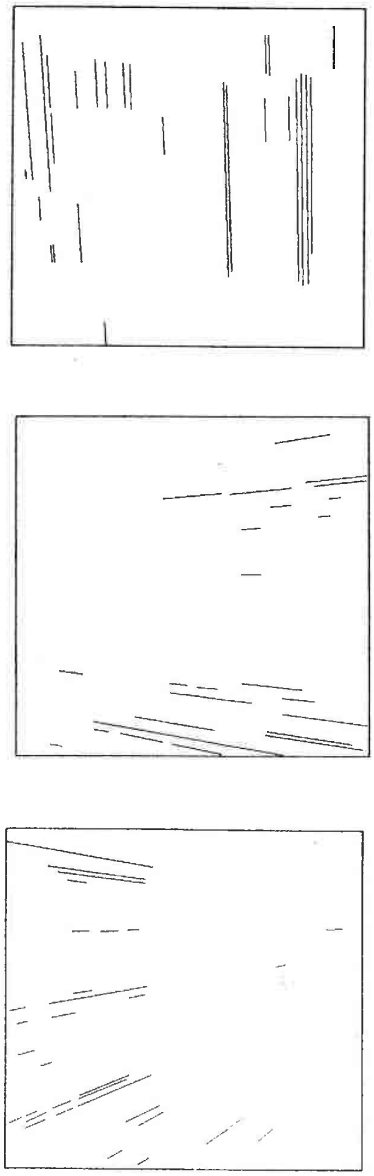


Figure 3.10. Position 70, $f = 20$.

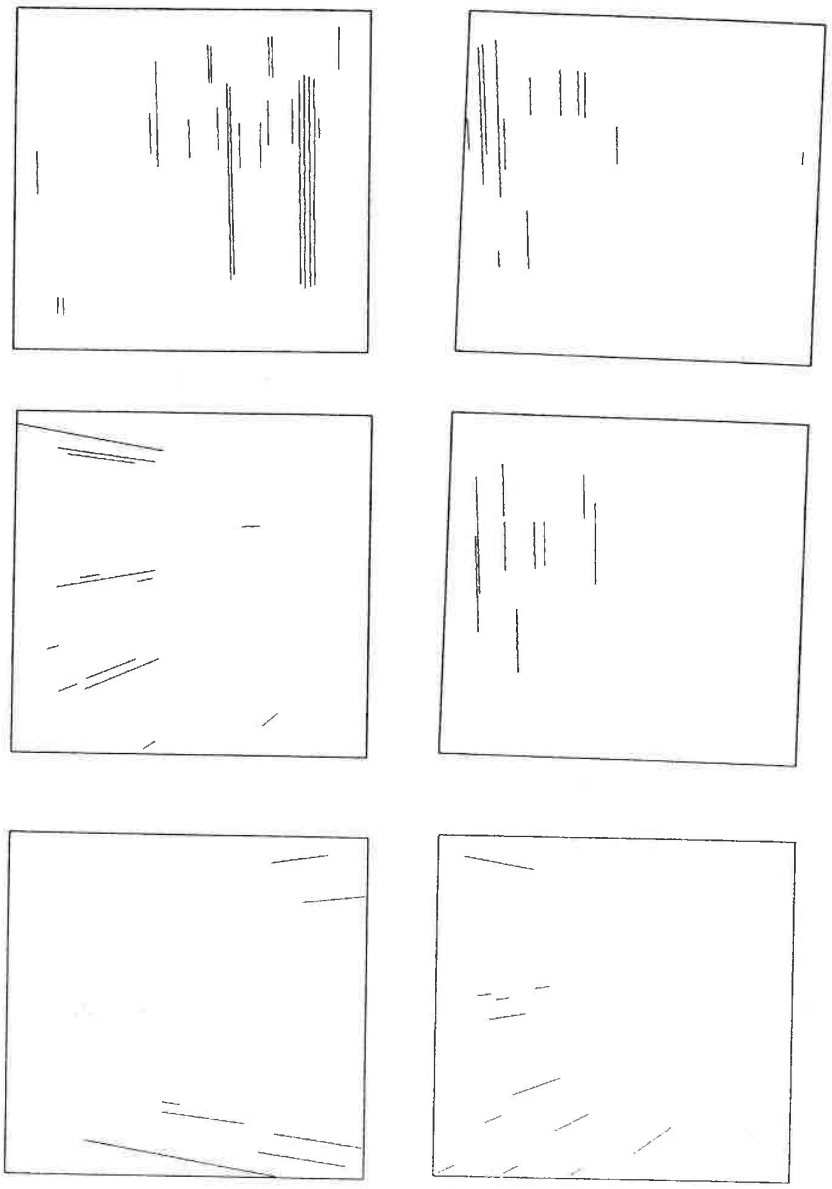


Figure 3.11. Position 70, $f = 200$.

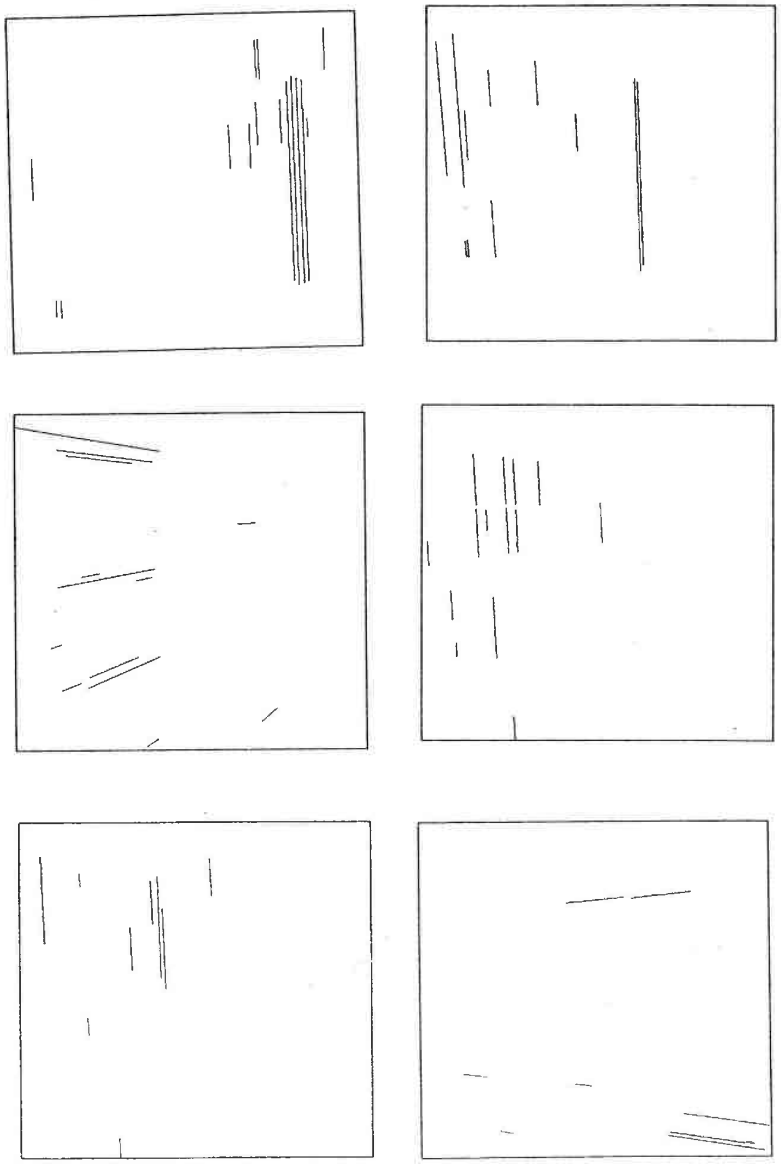


Figure 3.12. Position 70, $f = 1000$.

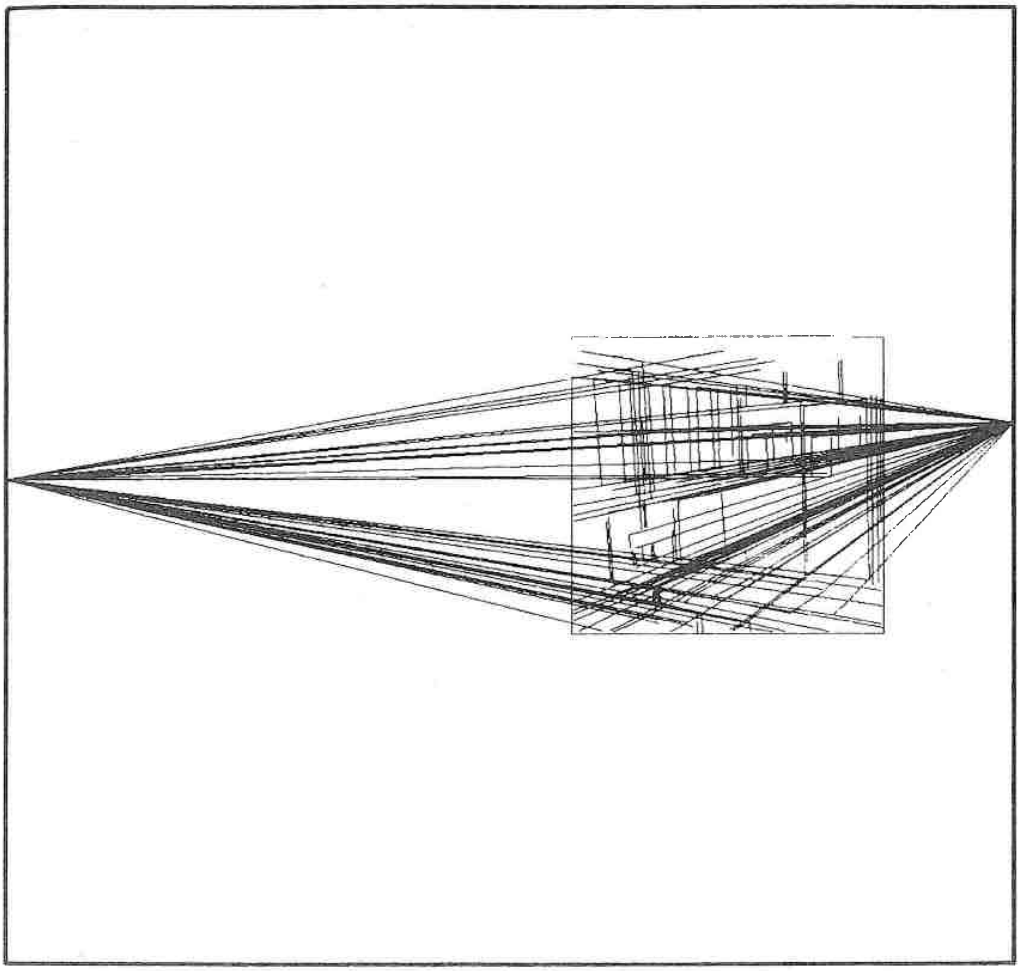


Figure 3.13. Position 75, deux points de fuites horizontaux.

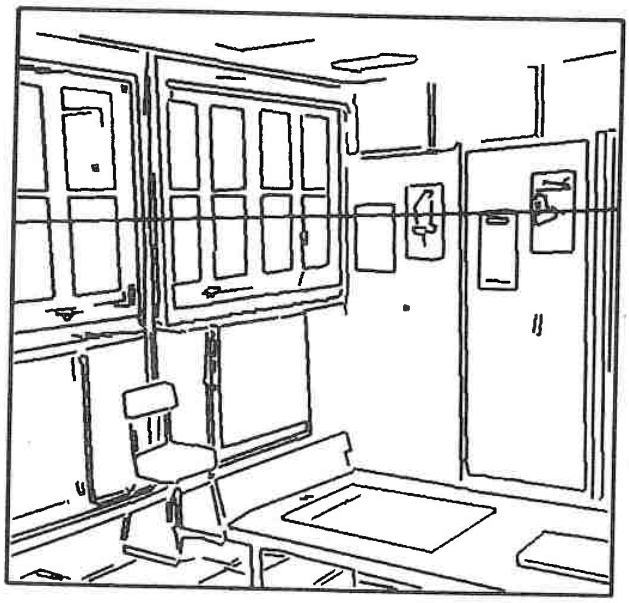


Figure 3.14. Position 75, la ligne d'horizon.

4

Regroupement perceptuel des indices visuels

4.1 Introduction

Le *regroupement perceptuel* des indices visuels est la réorganisation et la structuration des indices visuels extraits de l'image, directement effectuées dans l'image bidimensionelle. L'objectif est de réduire le nombre des indices à manipuler, et en même temps d'augmenter les informations portées par chaque indice visuel afin d'accélérer la recherche d'appariement.

Cette procédure de regroupement des indices visuels (ou *organisation perceptuelle*) correspond étroitement aux phénomènes de regroupement dans la vision humaine. Dans la figure 4.1, on détecte spontanément les groupes résultant des relations de connexité, de colinéarité et de parallélisme à partir de la répartition aléatoire de segments sur le fond, ce qui démontre la faculté de regroupement de la vision humaine.

Ces phénomènes sont étudiés par les psychologues de l'école Gestalt. Nous faisons donc d'abord un rappel sur les règles de perception étudiées par Gestaltists dans la section 4.2. Leur résultats nous serviront d'inspiration pour le choix des groupes perceptuels, conjointement avec des études des propriétés invariantes par perspective dans 4.3.

Par la suite, dans 4.4 en combinant les invariants présents et les règles de perception, nous présentons une organisation perceptuelle hiérarchique dans une scène d'intérieur sur les indices visuels de type segment fournis par la vision de bas niveau.

4.2 Rappel du principe de la théorie de Gestalt

Le groupement perceptuel repose sur la théorie de Gestalt dont le principe de base est que la perception humaine procède du tout à la partie, et interprète les

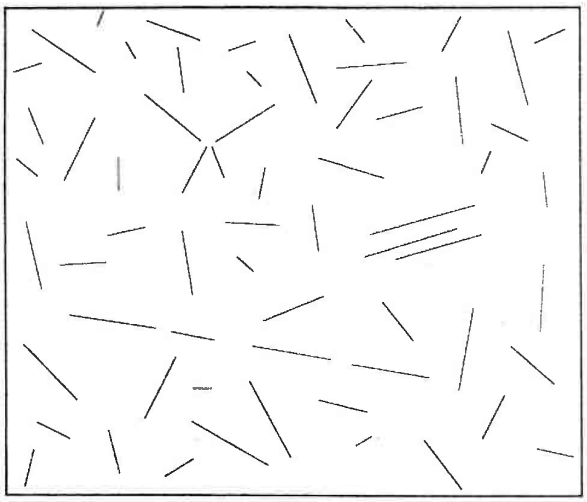


Figure 4.1. La détection spontané des groupes résultant des relations de connexité, de colinéarité et de parallélisme par la vision humaine.

données visuelles de manière la plus simple possible. L'exemple de référence, l'image d'une ellipse est toujours (ou presque) interprétée comme un cercle incliné dans l'espace. Néanmoins la simplicité est trop abstraite pour définir et quantifier. Les études plus concrètes de Gestalt mènent aux quelques règles de perception suivantes :

- *proximité* : les éléments qui sont suffisament proches ont tendance à être regroupés (voir la figure 4.2).
- *similarité* : les éléments ayant des attributs similaires tels que couleur, taille ou orientation ont tendance à être regroupés.
- *continuité* : les éléments situés le long d'une ligne ou une courbe lisse ont tendance à être regroupés.
- *fermeture* : on a tendance à percevoir les éléments bordant une région de manière que celle-ci soit une région fermée.
- *symétrie* : les éléments qui sont bilatéralement symétriques ont tendance à être perçus ensemble.
- *familiarité* : les éléments sont regroupés s'ils ont déja été perçus ensemble auparavant.

L'ambition est d'intégrer un maximum de ces règles dans un système de vision par ordinateur. Les principaux problèmes pour exploiter ces règles sont qu'elles ne sont souvent ni quantitatives ni constructives. Par exemple, que peut être la familiarité pour une machine ?

4.3 Recherche des propriétés invariantes

Etant donné que l'on n'a pas de connaissance *a priori* du point de vue de la prise d'image dans le modèle, les groupes perceptuels, détectés dans l'image pour indexer le modèle, doivent refléter des propriétés d'objets qui sont invariantes par changement de point de vue.

Il est donc inutile de regrouper les indices ayant une taille particulière, des angles particuliers ou d'autres propriétés qui dépendent fortement du point de vue. La figure 4.3 montre un contre exemple dans lequel on ne peut pas former un groupe perceptuel qui se base sur l'ordre des indices, puisque cet ordre n'est pas conservé par changement de point de vue.

Cette forte contrainte d'invariance limite le nombre de groupes perceptuels que l'on peut former, car il existe relativement peu de relations invariantes par projection. Nous examinons maintenant les invariants qui sont présents par projection perspective.

Dans la suite, l'espace et l'image désignent respectivement $\mathbb{R}^3$ et $\mathbb{P}^2$.

Connexité

La projection est une application continue. Par conséquent, si deux points sont proches l'un de l'autre dans l'espace, leurs projections le sont aussi dans l'image quelque soit le centre de projection[1] (c'est-à-dire le point de vue). Pourtant il est possible que deux points assez éloignés dans l'espace se projettent accidentellement sur l'image en des points très proches l'un de l'autre à cause d'un accident de point de vue.

Colinéarité

La projection perspective en tant que transformation homographique (cf. 2) envoyant des droites sur des droites conserve parfaitement l'alignement des points.

1. Il est le point à l'infini pour une projection parallèle.

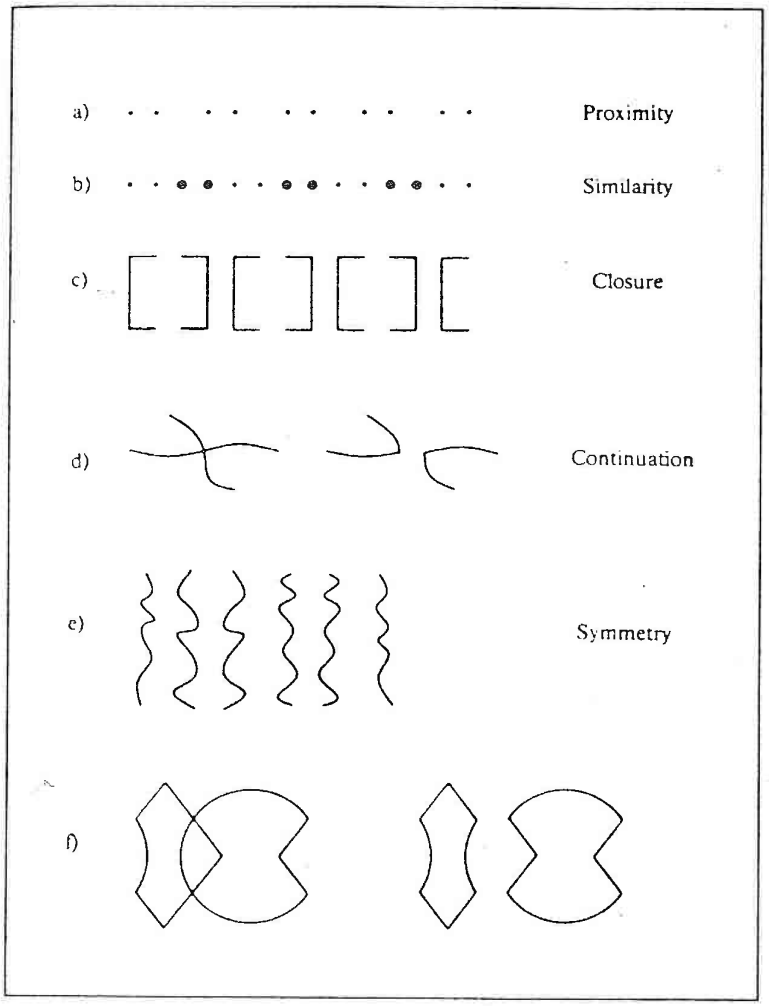


Figure 4.2. Les règles de perception.

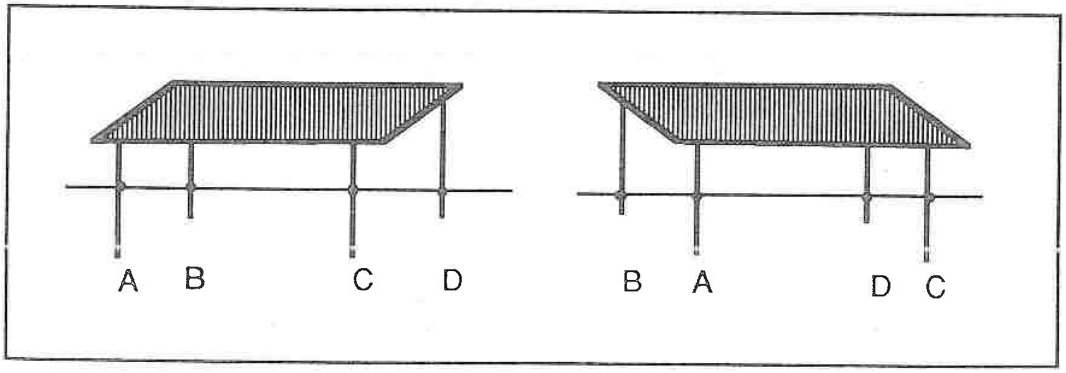


Figure 4.3. L'ordre des segments n'est pas conservé.

Un segment est un ensemble de points alignés. Si deux segments ou plus sont colinéaires dans l'espace, leurs projections le doivent aussi dans l'image. L'accident aura lieu lorsque les segments de l'espace sont coplanaires et que la caméra se trouve aussi sur ce plan (voir 4.4), ce qui n'est pas très fréquent. Cette colinéarité de segments est donc invariante.

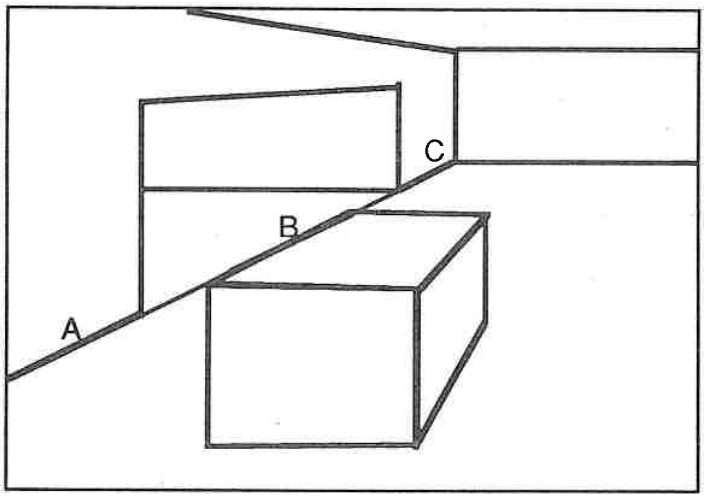


Figure 4.4. Colinéarité accidentelle : la caméra et les segments sont coplanaires.

Convexité

Par définition, un polyédre convexe dans l'espace se projette en un polygone convexe dans l'image. La *convexité* pour des objets rigides convexes est donc parfaitement invariante. Ceci est illustré par la figure 4.5.

Convergence

Les droites parallèles dans l'espace concourent vers un point commun après la projection[2]. Le parallélisme des droites dans l'espace devient donc la convergence des droites dans l'image.

2. La différence avec une jonction en Y ou en multi-Y est que le point commun pour une jonction est directement présent dans l'image tandis que le point de fuite se trouve au point de concours des droites portant les segments.

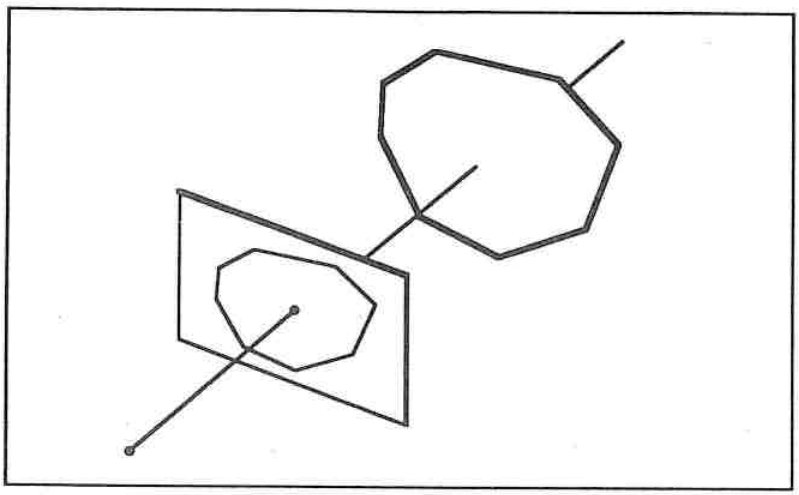


Figure 4.5. Une forme convexe de l'espace se projètte en une forme convexe dans l'image.

Birapport

Le birapport des quatre points alignés défini dans le chapitre 2 est l'invariant fondamental de la projection perspective (cf. 2). Il est totalement indépendant de points de vue, c'est à dire que pour quatres points colinéaires de l'espace M, N, A, B et leurs projections perspectives M', N', A', B' (resp. M'', N'', A'', B'') sur P' (resp. P'') de centre O' (resp. O'') (cf. figure 4.6), on a toujours :

$$[M'', N'', A'', B''] = [M', N', A', B'] = [M, N, A, B].$$

Comme nous avons discuté dans la section 2, le birapport est défini sur une droite projective ; dans le cas particulier d'un point à l'infini, on peut travailler sur trois points seulement.

Par ailleurs, la vision humaine est sensible au birapport, ce qui nous permet de sentir si des points vus en perspective sont équidistants.

Discussion

Les invariants *connexité* et *colinéarité* sont valables quelque soit la modélisation de la caméra (une projection parallèle ou une projection perspective). De plus, l'utilisation de ces deux invariants ne demande aucune hypothèse particulière vis-à-vis de la scène. Ils ont effectué les tâches les plus importantes de la vision de

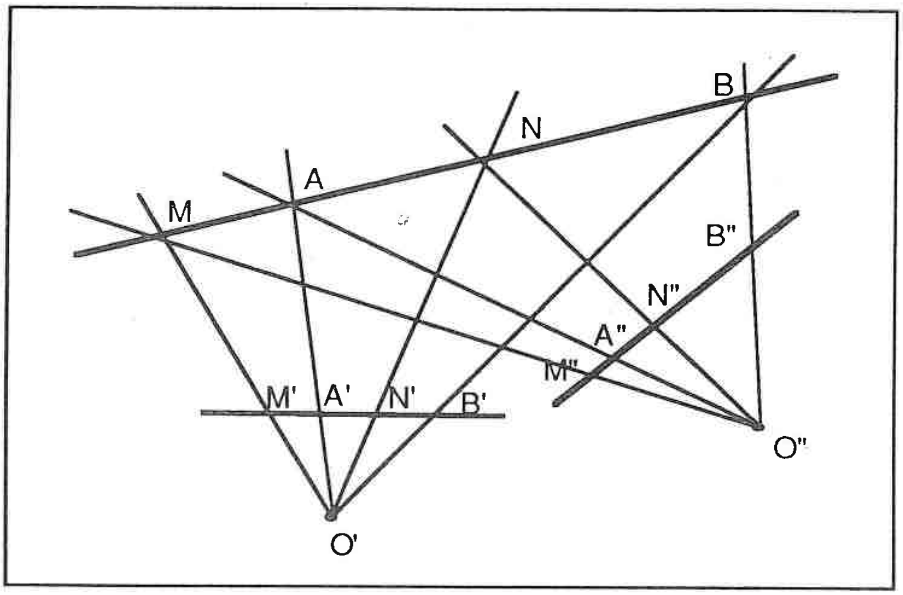


Figure 4.6. Le birapport est indépendant du point de vue.

bas niveau, par exemple, chaînage des pixels de contours et l'approximation polygonale des contours (cf. [Giraudon 87]). Ils sont aussi intégrés par quasi totalité de systèmes de vision ayant une couche de regroupement des indices [Lowe 85a, Brooks 81,Reynolds 87]. Pourtant les deux derniers, *convergence* et *birapport* ne sont présents que pour la projection perspective. Dans [Lowe 85a,?], les auteurs utilisent le parallélisme et l'équi-distance des indices. Il ne s'agit là que d'une approximation de la convergence et du birapport quand l'angle visuel des objets est petit et quand les objets ne s'étendent pas sur un champ de profondeur important par rapport à leur distance avec le point de vue. Ceci dit que l'effet de perspective n'est pas dominant. Ces deux invariants projectifs que nous pensons très important n'ont pas été suffisamment étudiés dans le passé. Ils sont encore rarement intégrés en tant que tels dans un système de vision.

Comme la contrainte d'invariance limite le nombre de groupes perceptuels possibles, on parle aussi de *quasi-invariance* : c'est-à-dire que les propriétés sont invariantes partiellement, ou bien sur un grand domaine de points de vue. C'est le cas de *parallélisme* et *équidistance* que l'on a cité au-dessus. Des maxima (minima) locaux ou des *points d'inflexion* d'une courbe peuvent être aussi utilisés comme quasi-invariants [Richetin 87], puisqu'ils sont la plupart du temps conservés.

4.4 Formation des groupes perceptuels

En combinant les règles de regroupement en perception humaine et les invariants de la projection perspective, nous proposons une approche hiérarchique de regroupement d'indices visuels de type segment. Les scènes d'application sont celles d'intérieur utilisées dans le cadre du projet ORASIS.

Le souci majeur dans une telle approche ascendante, dirigée par des données, est d'éviter la *signification accidentelle* due à l'ambiguïté massive de la projection : les éléments $3D$ n'ayant aucune relation significative peuvent conduire à une structure significative après projection (par exemple, colinéarité accidentelle des segments qui ne le sont pas dans l'espace).

Face au grand nombre d'indices visuels, les algorithmes de regroupement sont souvent assez coûteux en temps de calcul. Par exemple, pour un groupe fondé sur une relation $p-$aire, la complexité de l'algorithme correspondant est en $O(n^p)$ où n est le nombre d'indices.

L'utilisation de la technique de *baquets*[3] permet d'améliorer considérablement la complexité des algorithmes de regroupement. On peut se reporter à [Knuth 75] et [Baase 78] pour les algorithmes de base sur des baquets. Des baquets sont souvent une partition régulière de l'image, c'est-à-dire que chaque image est divisée en un certain nombre de fenêtres carrées ne se recouvrant pas, ce qui donne une structuration préalable de l'image. La partition est dans ce cas directement faite dans le plan image selon les coordonnées cartésiennes. On peut aussi faire des baquets sur d'autres paramètres correspondant aux indices selon les besoins ; par exemple, en coordonnées polaires (ρ, θ) pour la recherche de la colinéarité. Cette technique de "*bucketing*" permet un accès rapide aux indices "*proches*" d'un indice donné au sens d'une mesure prédéfinie et permet donc d'accélérer le processus de recherche. Le comportement en moyenne de ces algorithmes est pratiquement linéaire. Voir [Asano 85] pour l'analyse de la complexité en moyenne.

4.4.1 Groupe directionnel

L'invariant convergence nous conduit à regrouper les segments concourants en *groupes directionnels*. Chaque groupe directionnel est donc un ensemble de segments concourants dans l'image. Leurs homologues dans l'espace sont parallèles.

Un tel regroupement existe fort bien dans la vision humaine, ce qui fait l'objectif de nombreux dessins d'illusion (trompe-l'œil) en décorant les dessins par des fais-

3. "*bucket* en anglais

ceaux de droites concourantes.

Ce regroupement est réalisé directement à partir de l'algorithme de la détection de points de fuite que nous avons dévelopé dans le chapitre 3. Rappelons que les résultats de cet algorithme sont les coordonnées polaires du point de concours et la liste de segments qui lui sont concourant. Ainsi chaque point de fuite correspond à un groupe directionnel.

Il est intéressant de remarquer que l'on peut orienter les segments appartenant aux groupes directionnels. "Orienter" signifie que l'on peut savoir quelle extrémité du segment est la plus proche du point de vue. Comme le point de fuite est la projection du point à l'infini du groupe de segments parallèles dans l'espace, l'extrémité la plus proche du point de fuite (au sens de distance euclidienne) est nécessairement la plus éloignée du point de vue.

Chaque groupe directionnel possède les caractéristiques suivantes :

- *liste* de segments orientés ;
- *point de fuite* dont les coordonnées sont affinées par la méthode présentée dans le chapitre 5 ;
- *direction* qui est le vecteur directeur des droites parallèles dans l'espace (la perspective inverse du point de fuite).

Par la suite, lorsqu'on parlera de direction d'un segment pour un groupe directionnel, il s'agira de la direction du groupe directionnel dont le segment fait partie.

4.4.2 Groupe colinéaire : droite

L'invariant colinéarité de segments et la règle de continuité mènent naturellement à la formation des *groupes colinéaires*, appelés aussi *droites*.

Un groupe colinéaire est donc un ensemble de segments colinéaires.

Pour construire ces groupes, l'approche raisonnable consiste à

1. d'abord créer des baquets en coordonnées polaires (ρ, θ) ;
2. ensuite utiliser la représentation paramétrique (cf. 2) de segments pour déterminer définitivement la colinéarité de segments à l'intérieur de chaque baquet.

   Soient $[M, N]$ et $[K, L]$ deux segments, ils sont colinéaires (sans se recouvrir) si

$$(recouv_{\perp}([M,N],[K,L]) = 0) \wedge (max(d_{\dashv}(K,[M,N]), d_{\dashv}(L,[M,N])) < \epsilon).$$

Ce test est illustré par la figure 4.7.

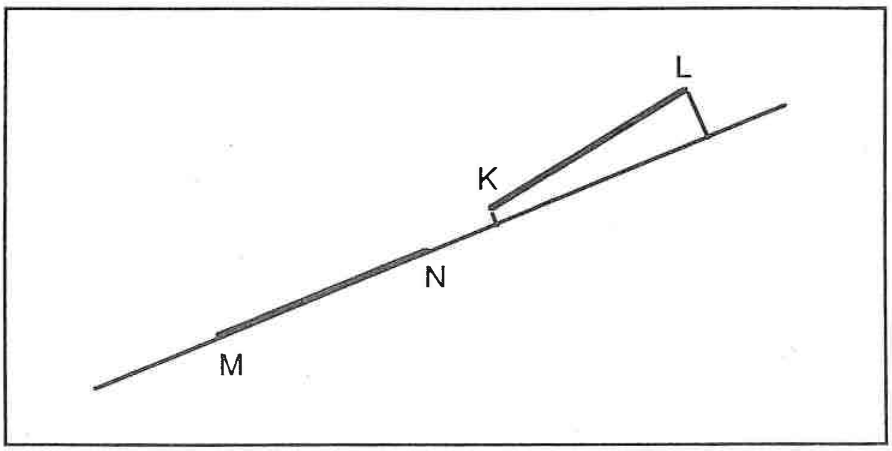


Figure 4.7. La colinéarité de $[M, N]$ et $[K, L]$.

Remarquons que dans notre cas les segments colinéaires sont forcément dans le même groupe directionnel. La recherche de segments colinéaires peut donc être restreinte à l'intérieur de chaque groupe directionnel.

A chaque point de fuite de groupe directionnel est associé un repère polaire (ρ, θ) de $\mathbb{R}^2$; l'axe de ce dernier est une ligne d'horizon[4] (cf. figure 4.8). Les segments du groupe directionnel préservent tous la même valeur nulle pour ρ, ce qui ne nous laisse plus qu'un seul paramètre : l'angle θ dans un baquet préalables de $\rho = 0$. Deux segments $[M, N]$ et $[K, L]$ sont colinéaires si la différence de leur angles est inférieure à un seuil toléré (cf fig. 4.8).

Si l'on trie les segments de chaque groupe directionnel selon θ, un seul parcours séquentiel de cette liste triée de segments suffit pour obtenir les groupes colinéaires. Toutefois, en pratique il est assez difficile de tenir compte uniquement de l'angle θ comme critère de colinéarité, surtout lorsque l'on rencontre des segments très proches et parallèles. Nous faisons donc une première partition grossière en θ (avec un seuil assez large), c'est-à-dire créer des baquets en θ, ensuite nous cherchons les segments colinéaires en utilisant la représentation paramétrique des segments présentée ci-dessus à l'intérieur d'un baquet.

La complexité de cet algorithme est de l'ordre de $O(n)$ avec un tri préalable de

4. Une ligne quelconque passant par l'origine peut être choisie comme axe. Le choix d'une ligne d'horizon se justifira par le fait que θ (angle orienté) ainsi défini permettra de déduir directement la position relative des segments horizontaux : un segment se trouve au-dessus (resp. au-dessous) de la ligne d'horizon correspond à un angle positif (resp. négatif).

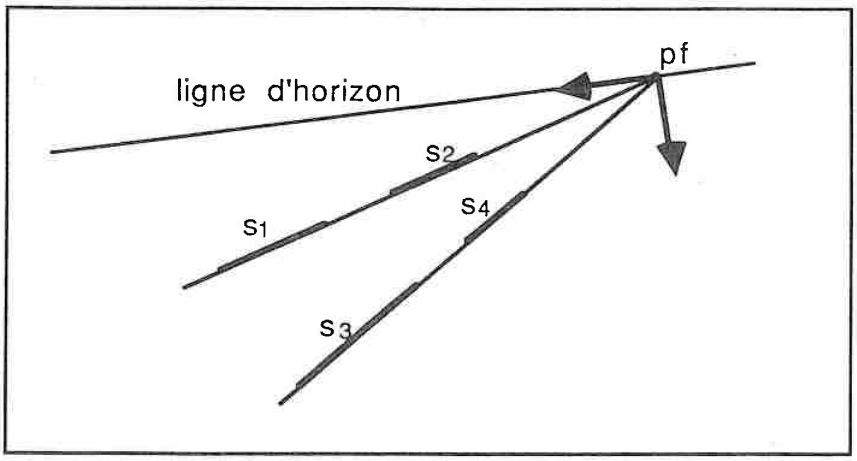


Figure 4.8. $angle(s_1) = angle(s_2)$, $angle(s_3) = angle(s_4)$

l'ordre de $O(n \log n)$.

Pour les segments qui ne sont pas dans un groupe directionnel, la méthode générale peut s'appliquer. Signalons que les segments isolés et trop éloignés des autres sont éliminés du groupe colinéaire pour éviter la colinéarité accidentelle.

Enfin chaque groupe colinéaire possède les caractéristiques suivantes :

- *liste* triée de segments colinéaires selon la distance du segment au point de fuite ;
- *direction* qui est celle du groupe directionnel dont le groupe colinéaire est issu ;
- *angle* qui est le θ dans le repère polaire associé au groupe directionnel dont le groupe est issu.
- *segment porteur* qui est le segment dont les extrémités sont les deux points les plus éloignés du groupe (cf. figure **??**). Les coordonnées des extrémités du segment porteur sont recalculés en approximant la droite passant par tous les points du groupe avec la méthode des moindres carrées (cf. figure **??**).

4.4.3 Groupe raie

Le quasi-invariant parallélisme et la règle de proximité nous mènent à la formation de groupes appelés *raies*.

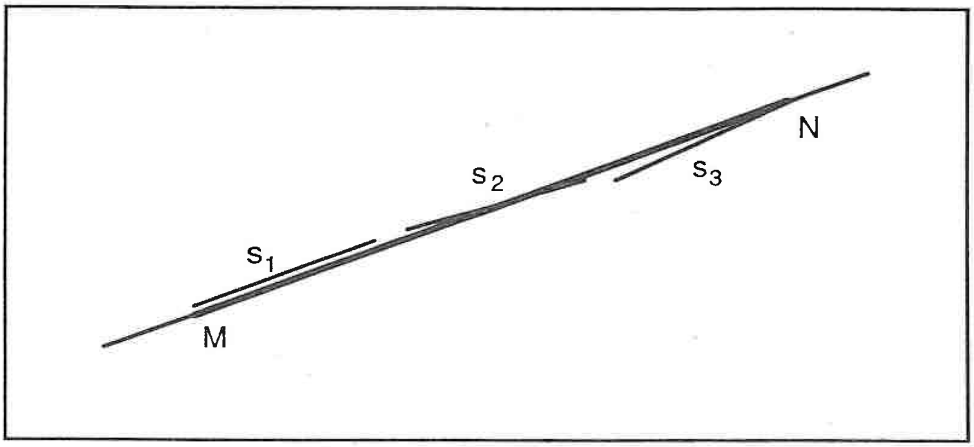


Figure 4.9. L'approximation de la droite D du groupe colinéaire et le segment porteur du groupe $[M, N]$.

Un groupe raie est composé de segments qui sont très proches et presque parallèles (voir la figure 4.10).

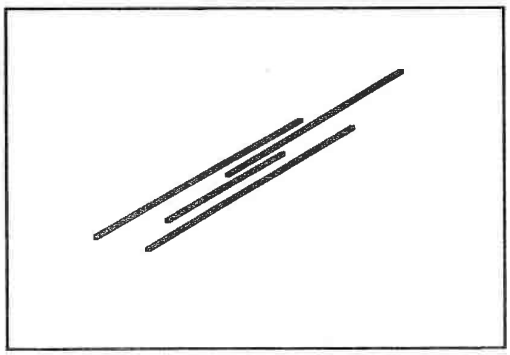


Figure 4.10. Groupe raie : segments proches et (quasi) parallèles

La détection de groupes raies est faite à l'intérieur d'un groupe directionnel. On associe toujours un repère polaire (ρ, θ) de $I\!R^2$ à chaque groupe directionnel. Chaque baquet en θ avec un seuil justifié correspond parfaitement à un groupe raie.

Dans l'implantation pratique des algorithmes, la détection de groupes raies et de groupes colinéaires est chaînée. Nous regroupons d'abord les segments directionnels en groupes raies, ensuite nous cherchons des groupes colinéaires à l'intérieur de chaque groupe raie.

Chaque groupe raie possède les caractéristiques suivantes :

- *liste* de segments ;
- *direction* qui est la direction du groupe directionnel dont le groupe raie est issu ;
- *angle* qui est le θ dans le repère polaire associé au groupe directionnel dont le groupe est issu ;
- *centre de gravité* qui est le barycentre de tous les points du groupe ;
- *segment porteur* qui est le segment dont les deux extrémités sont les deux points les plus éloignés du groupe.

4.4.4 Groupe connexe : jonction

La règle de proximité et l'invariant connexité nous mènent à former des groupes connexes de segments appelés *jonctions.*

On peut, dans la plupart des cas, classer des jonctions en deux grandes familles.

1. Quand deux ou plus de segments partagent une extrémité commune, ils forment des jonction en L (d'ordre deux), en Y (d'ordre trois) ou en multi-Y (d'ordre supérieur) (voir la figure 4.11).

   Ces relations restent très probables dans l'espace puisque le seul accident de point de vue se produit quand le centre de la caméra et les extrémités des segments sont alignés (voir la figure 4.12), ce qui est rarissime.

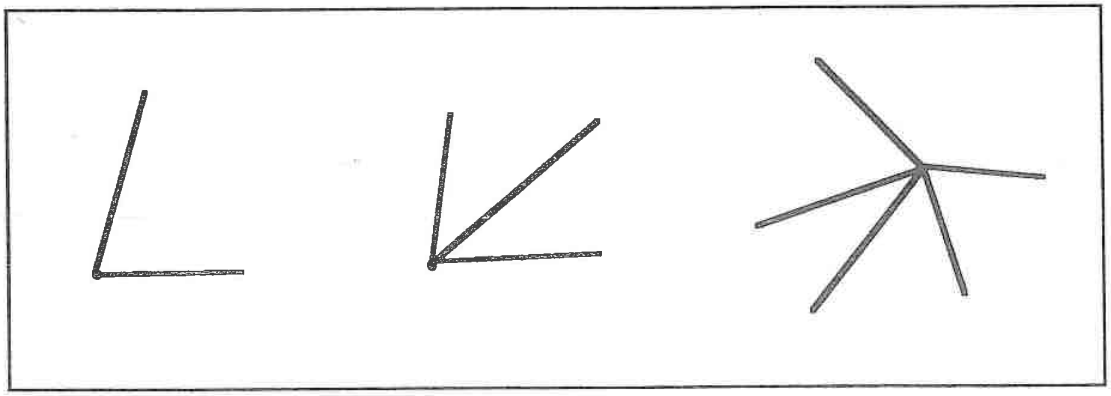


Figure 4.11. Jonctions en L, en Y et en multi-Y.

2. La deuxième famille de jonctions sont des jonctions en T, K, X, multi-K (cf. figure 4.13). Une jonction en T est souvent due à l'occultation partielle d'un

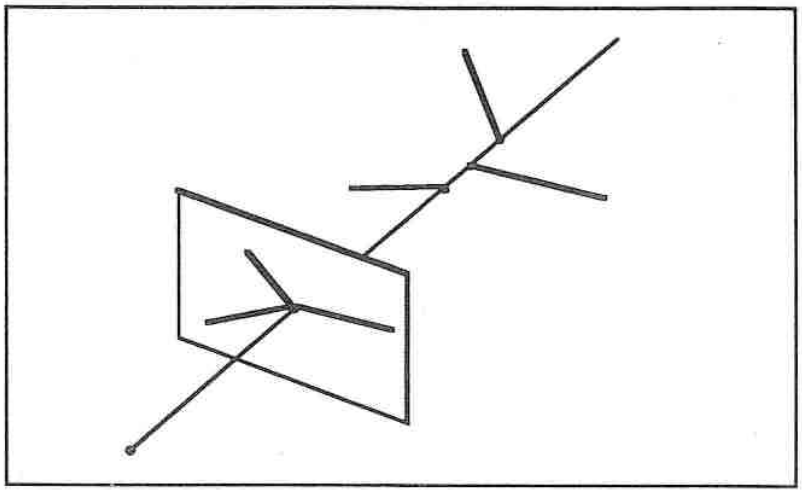


Figure 4.12. Une jonction Y accidentelle : la direction d'observation et la droite portant les extrémités des segments sont colinéaires.

objet par un autre, ou bien un contour de type "marque" se termine sur un contour physique. Une jonction en X est souvent due aux contours des objets transparents ou des "fils de fer" l'un au dessus de l'autre dans l'espace.

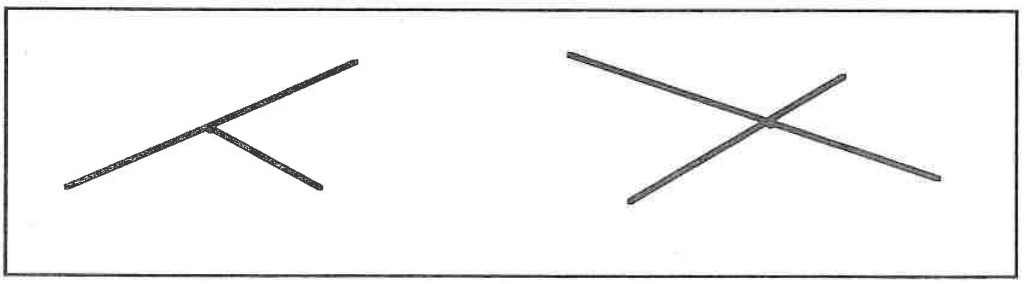


Figure 4.13. Jonctions en T et en X.

Nous ne nous intéressons pas à cette famille de jonctions , car elle ne provient pas de la connexité de segments dans l'espace. Elle est donc moins significative et les interprétations possibles ne sont pas intéressantes pour les objets rigides.

En conclusion, la première famille de jonctions significatives correspond très fortement à la connexité de segments dans l'espace. De plus l'accident de point de vue est rarissime. Nous nous intéresserons uniquement à ce type de jonctions d'ordre deux. Dans la suite, le terme *jonction*, implique que deux segments partagent une extrémité commune, sauf mention explicite.

La détection de jonctions consiste à examiner tous les couples de segments. Pour chaque couple de segments $[M, N]$ et $[K, L]$, on calcule l'intersection de droites P_i portant les deux segments (par $i([M, N], [K, L]) = (\lambda_{MN}, \lambda_{KL})$ de la section 2.2.2). Si l'intersection tombe à l'intérieur de l'un des deux segments, cela forme potentiellement une jonction en T. On peut donc éliminer définitivement la candidature de ce couple de segments pour une jonction. Sinon, on mesure la proximité de l'intersection, définie comme la somme des distances de l'intersection des segments à chacune des extrémités les plus proches de l'intersection. Alors deux segments sont candidats pour une jonction si la proximité de l'intersection est suffisante. Comme nous nous limitons à des jonctions d'ordre deux, et qu'un segment participe tout au plus à deux jonctions (une à chaque extrémité), si une extrémité d'un segment possède plusieurs segments candidats en jonction, on choisit bien entendu celui dont la proximité est la meilleure.

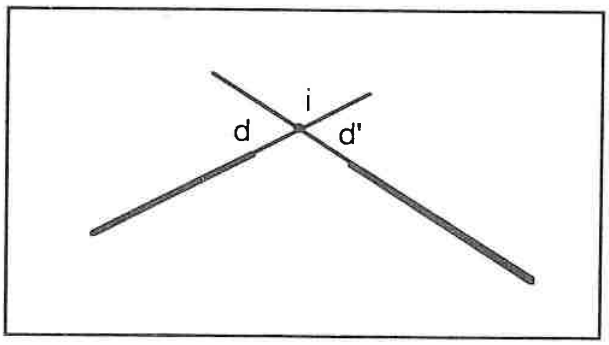


Figure 4.14. Mesure de la proximité de l'intersection : $d + d'$.

Pour la rapidité de la détection de jonctions, on fait des "baquets" en divisant l'image en des fenêtres de forme rectangulaire.

Chaque jonction possède les caractéristiques suivantes :

- *couple* de segments ;
- *direction* qui est le vecteur normale du plan déterminé par le couple de segments ;
- *position relative* à la ligne d'horizon (au-dessus, en travers ou au-dessous de la ligne d'horizon) ;
- *point d'intersection* de deux droites portant les deux segments.

4.4.5 Groupe convexe

L'invariant convexité et la règle de fermeture nous mènent à construire des *groupes convexes.*

Un groupe convexe est une chaîne convexe de segments adjacents. Une chaîne est convexe si la chaîne complétée par le segment manquant forme un polygone convexe (voir la figure 4.15). Pour en savoir plus sur le polygone convexe, on peut se reporter à [Shamos 85].

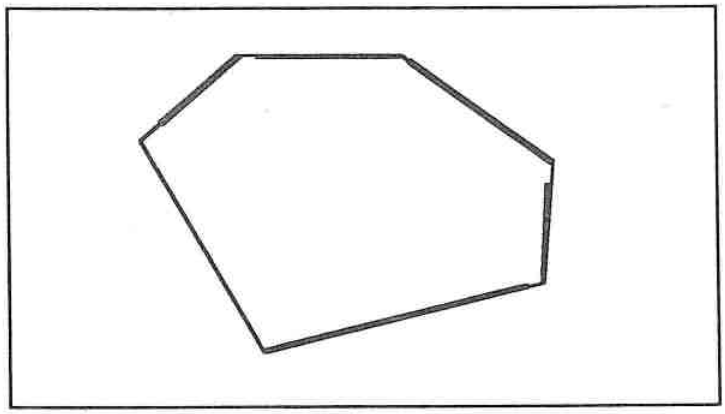


Figure 4.15. Groupe convexe

La détection d'un groupe convexe consiste donc à chercher une chaîne convexe de segments. Un couple de jonctions est dit adjacent si deux jonctions partagent un segment commun et si elles ont la même direction. Un tel couple de jonctions adjacentes est convexe si deux segments non communs se trouvent du même coté du segment commun. La fusion transitive de tous les couples de jonctions adjacentes et convexes donne lieu à une chaîne convexe de segments.

Chaque groupe convexe possède donc les caractéristiques suivantes :

- *liste* de segments ;
- *direction* qui est celle des jonctions ;
- *position relative* par rapport à la ligne d'horizon.

4.4.6 Groupe tronc

La règle de continuité et de proximité nous amène à chercher des groupes que nous appelerons *groupe tronc.*

Un groupe tronc est composé d'un tronc qui est une droite (un groupe colinéaire) et des segments qui sont en jonction avec l'un des segments du tronc (cf. figure 4.16).

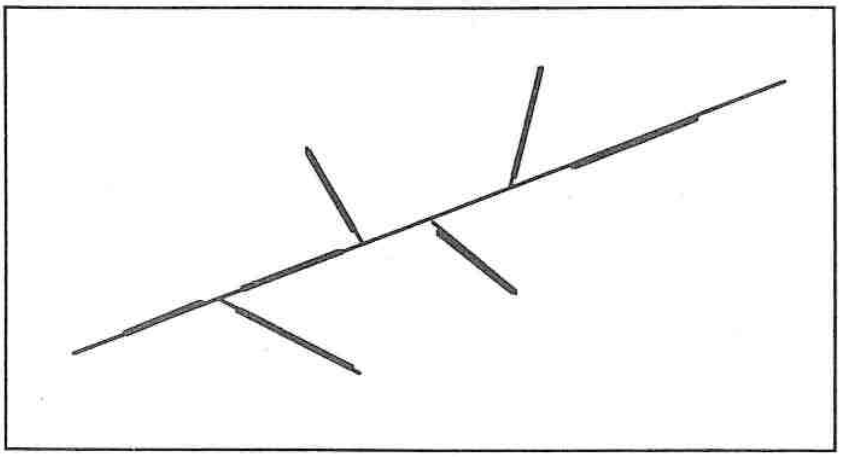


Figure 4.16. Groupe tronc

La formation de groupes troncs est immédiate à partir des groupes colinéaires et des jonctions.

Chaque groupe tronc possède les caractéristiques suivantes :

- *tronc* qui est un groupe colinéaire ;
- *liste* de segments en jonction avec l'un des segments du tronc ;
- *direction* qui est celle du tronc ;
- *angle* qui est celui du tronc.

4.4.7 Groupe coplanaire

En effectuant la fermeture transitive de la relation colinéaire et de la relation connexe, on obtient un ensemble de segments colinéaires et connexes. Si cette fermeture transitive est faite uniquement à partir des groupes colinéaires et des jonctions engendrées par deux groupes directionnels, il est probable que les homologues de cet ensemble de segments sont coplanaires dans l'espace. Nous pouvons ainsi former des groupes dits *groupe coplanaire*.

La recherche d'un tel groupe consiste simplement à faire la fermeture transitive de la relation colinéaire et de la relation connexe à partir de deux groupes directionnels.

Chaque groupe coplanaire possède les caractéristiques suivantes :

- *liste* de segments ;
- *direction* qui est le produit vectoriel de deux directions des groupes directionnels.

4.5 Résultats expérimentaux et Conclusion

Les résultats expérimentaux de regroupement sont présentés dans les figures 4.17, 4.18, 4.19 4.20, 4.21, 4.22, 4.23, 4.24 et 4.25.

Les points forts de l'approche que nous avons menée pour le regroupement d'indices visuels sont que les vrais invariants par perspective ont été intégrés dans le système et que la plupart de nos groupes perceptuels sont organisés d'une façon hiérarchique, ce qui permettera une stratégie d'appariement de "coarse-to-fine" par la suite (cf. chapitre ??).

Le point faible est qu'il n'y a pas encore une approche très rigoureuse basée sur la probabilité pour lutter contre le regroupement accidentel.

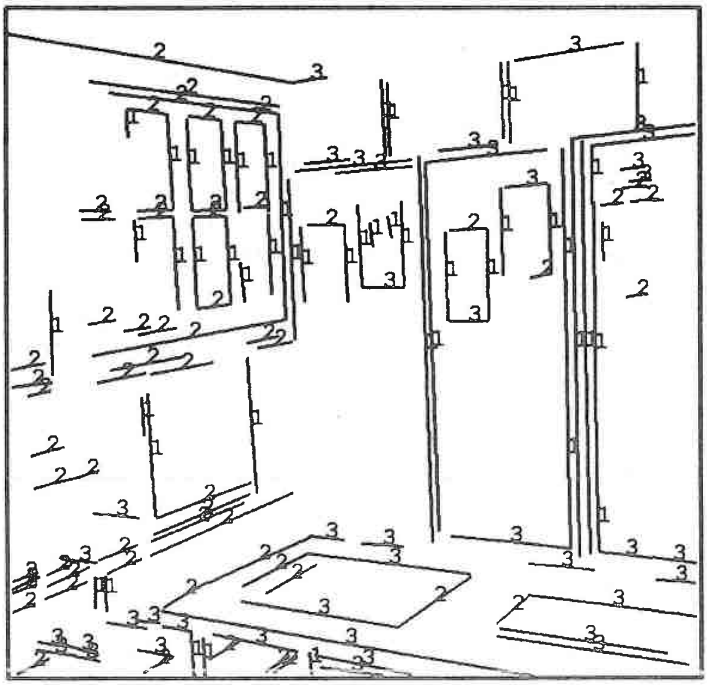


Figure 4.17. Groupes directionnels : les segments d'un même groupe directionnel est numérotés par un numéro identique.

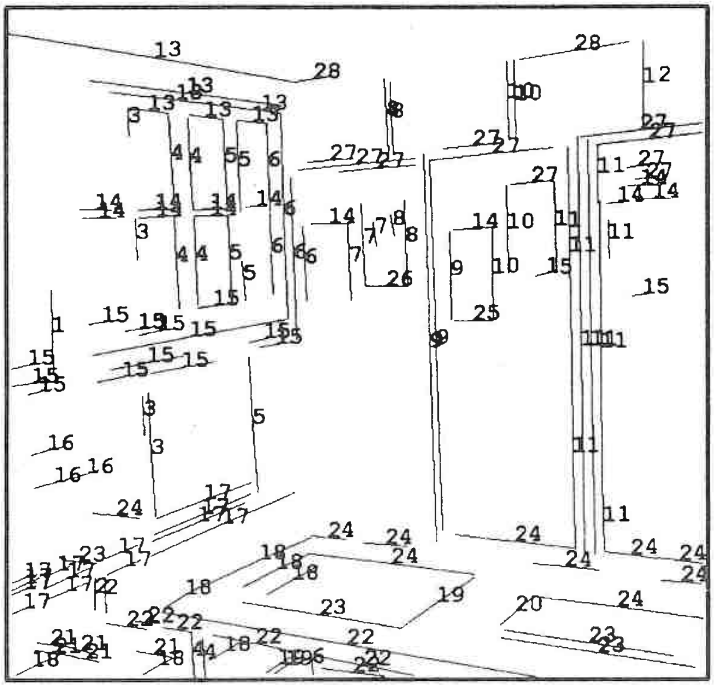


Figure 4.18. Groupes raies : les segments d'un même groupe raie est numérotés par un numéro identique.

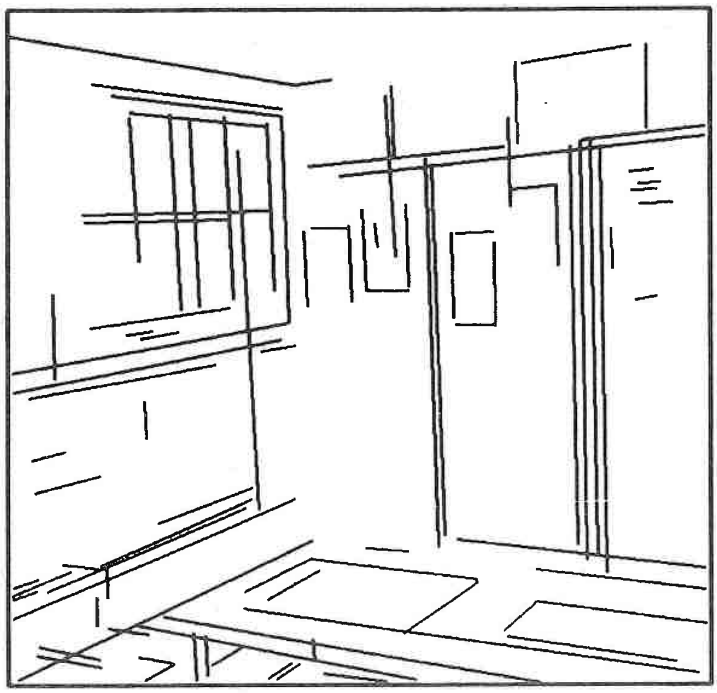


Figure 4.19. Groupes colinéaires : chaque groupe est représenté par son segment porteur.

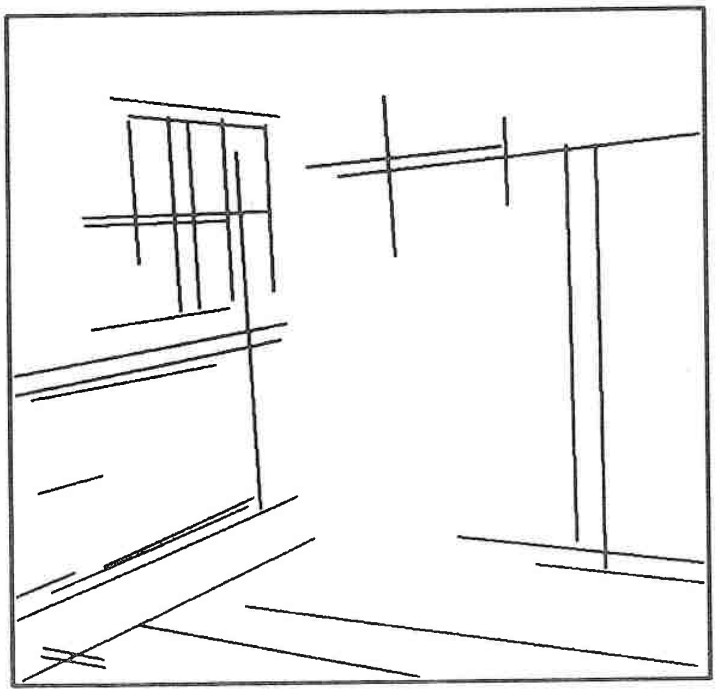


Figure 4.20. Groupes colinéaires : seuls les groupes dont le nombre de segments est supérieur à deux sont visualisés.

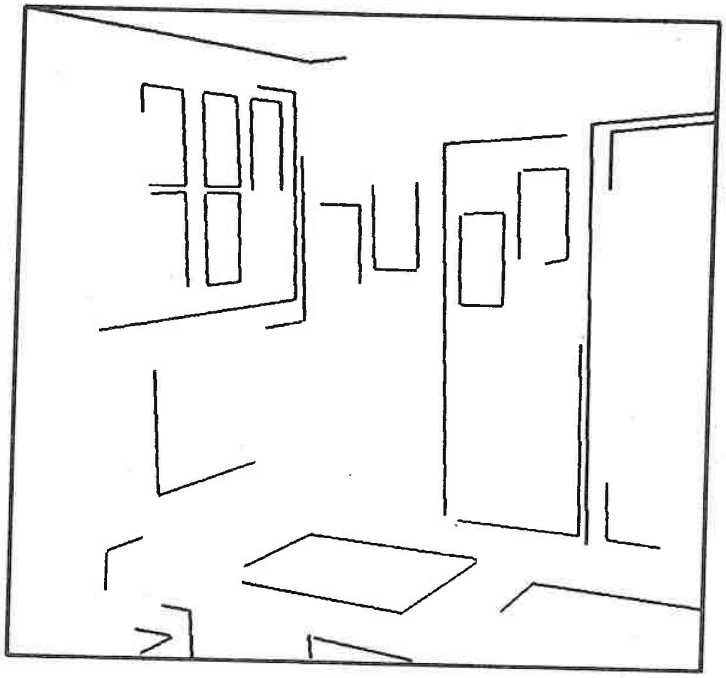


Figure 4.21. Groupes jonctions.

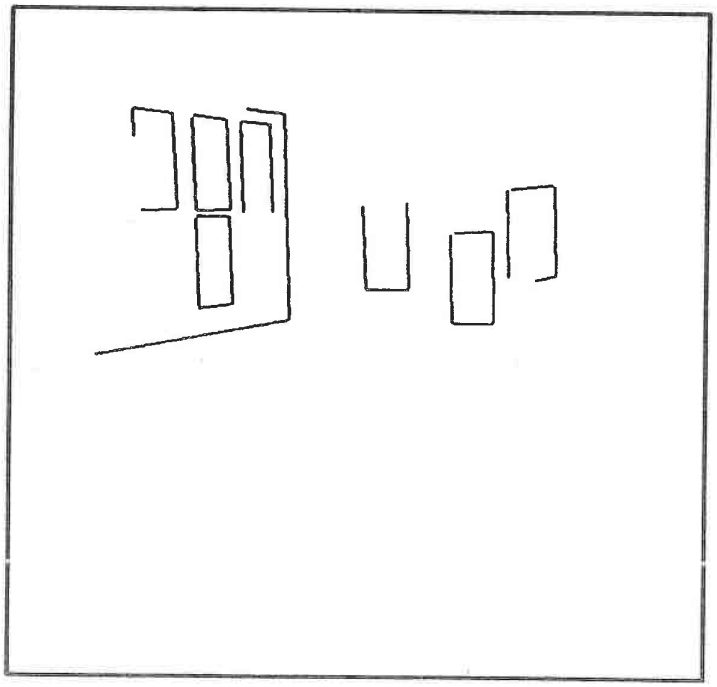


Figure 4.22. Groupes convexes.

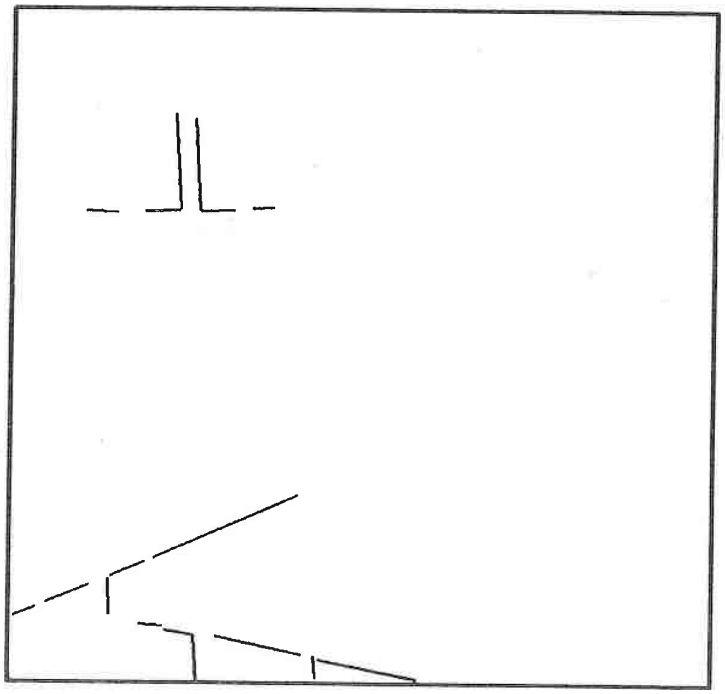


Figure 4.23. Groupes troncs : on n'a visualisé que trois d'entre eux.

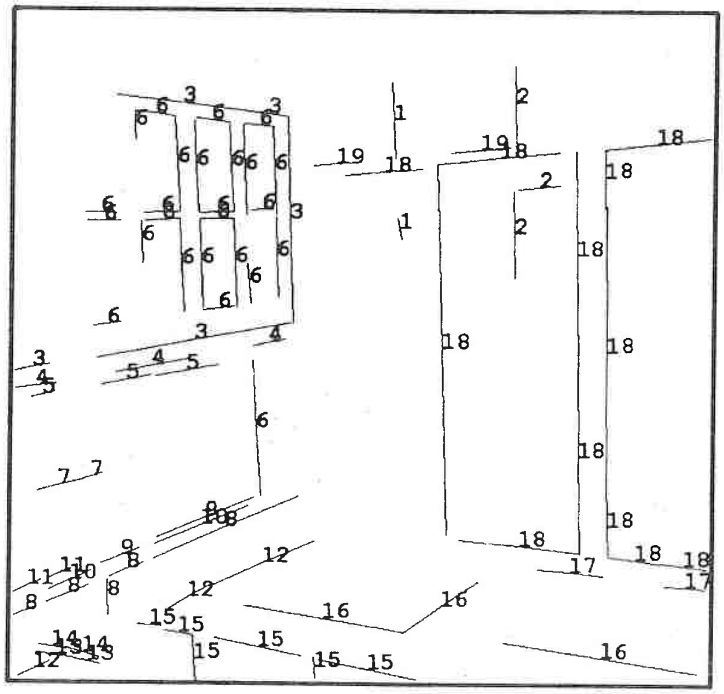


Figure 4.24. Groupes coplanaires : les segments d'un même groupe coplanaire sont numérotés par un numéro identique.

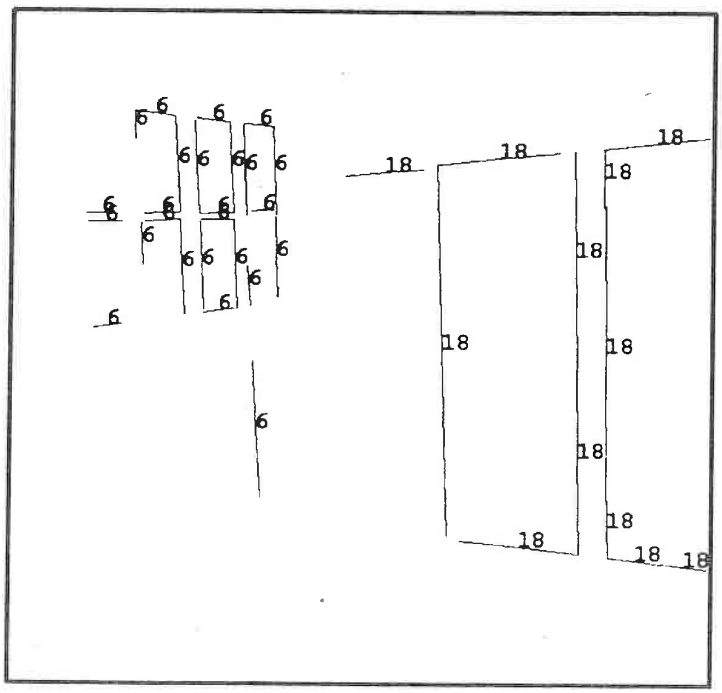


Figure 4.25. Groupes coplanaires : seule les deux groupes dont les segments sont les plus nombreux sont visualisés.

5
Détermination spatiale

Dans ce chapitre, nous présentons d'abord la modélisation de la caméra qui sera indispensable pour décrire l'algorithme. Ensuite nous énonçons le problème de la détermination spatiale et donnons par la suite un algorithme développé dans notre système. Nous terminons par la présentation des travaux connexes et une brève perspective.

5.1 Modélisation de la caméra

Dans cette thèse, le problème de la modélisation de la caméra n'a pas encore été posée rigoureusement, puisque dans le regroupement des indices, ceci se fait uniquement sur l'image sans se préoccuper du lien avec le modèle. Dans ce chapitre, il s'agit de déterminer le point de vue de la caméra dans le modèle *3D*, ce qui nécessite une modélisation rigoureuse de la caméra.

La modélisation de la caméra décrit la transformation permettant de projeter un point 3D sur une rétine de la caméra, le plan image. Cette transformation est considérée comme une projection perspective parfaite[1] dans le repère lié à la caméra (voir la figure 5.1), les paramètres qui déterminent cette projection perspective sont dits *intrinsèques*. Ensuite ce repère lié à la caméra peut subir une transformation rigide, c'est-à-dire le produit d'une translation et d'une rotation dans l'espace, par rapport au repère absolu de la scène, les paramètres qui déterminent cette transformation sont dits *extrinsèques*. La composition de cette rotation, translation et projection perspective donne une transformation complète, appelée la *matrice de calibration* de cette caméra, notée C.

Cette transformation permet donc de passer d'un point de l'espace dans le repère absolu au point de l'image en pixel. Réciproquement, à partir d'un point de l'image

1. Ce modèle de la caméra est celui *à sténopé* (voir [Toscani 87])

en pixel, il est possible de reconstituer la droite des points dans l'espace vérifiant l'opération inverse.

La matrice de calibration peut être représentée linéairement et uniformément par des matrices en coordonnées homogènes. La dimension de cette matrice est de 3×4, puisque la matrice de projection perspective est de dimension 3×4 : la matrice de rotation et celle de translation sont de 4×4.

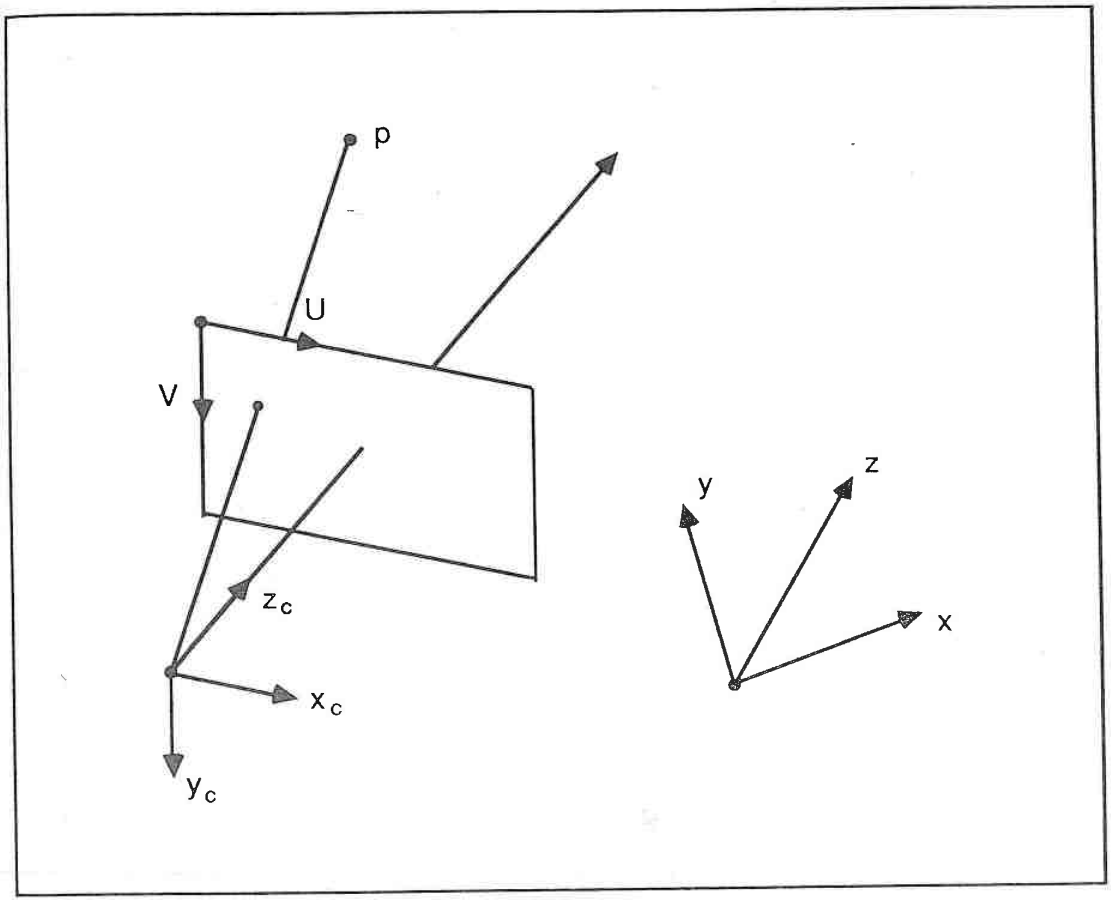


Figure 5.1. La modélisation de la caméra : repère lié à l'image, repère lié à la caméra et repère absolu.

Soient $(x, y, z, 1)^T$ un point de l'espace et sa projection dans l'image $(u, v, s)^T$ en coordonnées homogènes. Nous avons :

$$\begin{pmatrix} u \\ v \\ s \end{pmatrix} = C \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

Pour obtenir le point en coordonnées cartésiens (en pixel) $(U, V)^T$ dans le repère lié à l'image, nous avons :

$$\begin{aligned} U &= u/s \\ V &= v/s. \end{aligned}$$

L'objectif de la calibration consiste donc à déterminer *a priori* cette matrice C ou bien les paramères intrinsèques et extrinsèques de la caméra. G. Toscani (voir [Toscani 87]) a résolu ce problème en considérant la caméra comme une boîte noire sans aucune connaissance *a priori* des paramètres intrinsèques de la caméra. Dans la suite, nous supposons que telle matrice, fournie par Toscani, est disponible.

Pourtant une telle modélisation a des propriétés particulières quant à son utilisation. Il est impossible de séparer la distance focale des facteurs d'échelle car ils ont toujours les mêmes effets de perspective. Les points du plan focal de la caméra ne sont pas définis, ce qui correspond à $s = 0$. Faute de connaissance *a priori* du repère lié à la caméra par rapport au repère absolu, on ne peut pas distinguer si les objets proviennent de l'avant ou de l'arrière de la caméra.

5.2 Formulation du problème

Commençons d'abord par formuler le problème de la détermination spatiale.

Problème : *Etant donnés un ensemble de correspondances entre des indices visuels de l'image et des indices 3D du modèle et les paramètres intrinsèques de la caméra,*

- *quelle est la transformation rigide dans l'espace, composée d'une rotation et d'une translation, qui permet de projeter les indices 3D du modèle en indices visuels correspondants de l'image*
- *et quel est le nombre minimal des indices qu'il faut pour avoir cette transformation.*

Remarquons que ce problème est très similaire au problème de la calibration. La calibration nécessite l'estimation des paramètres intrinsèques de la caméra en même temps.

5.3 Algorithme

Nous proposons un algorithme linéaire qui permet de résoudre ce problème dans le cas où au moins *2 groupes directionnels* et *3 droites* sont appariés entre l'image et

le modèle.

Une transformation rigide dans l'espace représentant le point de vue peut être décomposée en le produit d'une rotation et d'une translation dans l'espace, cette décomposition peut se faire d'une infinité de façon dans le cas général. Cependant dans le cas où l'axe de rotation passe par un point donné, par exemple l'origine du repère, et on applique la rotation en premier, cette décomposition devient unique.

Le principe que nous avons développé est d'utiliser 2 groupes directionnels pour déterminer la rotation R et 3 droites pour déterminer le vecteur de la translation T. La rotation et la translation sont ainsi déterminées successivement.

5.3.1 Détermination de rotation

Avec 2 groupes directionnels qui sont mis en correspondance, la rotation peut être entièrement déterminée. La démarche est la suivante :

Relocalisation du point de fuite

Chaque groupe directionnel est associé à un point de fuite. Il peut être précisément relocalisé dans l'image à l'aide de la méthode des moindres carrés. Il est donc l'intersection des segments du groupe au sens des moindres carrés (voir la figure 5.2).

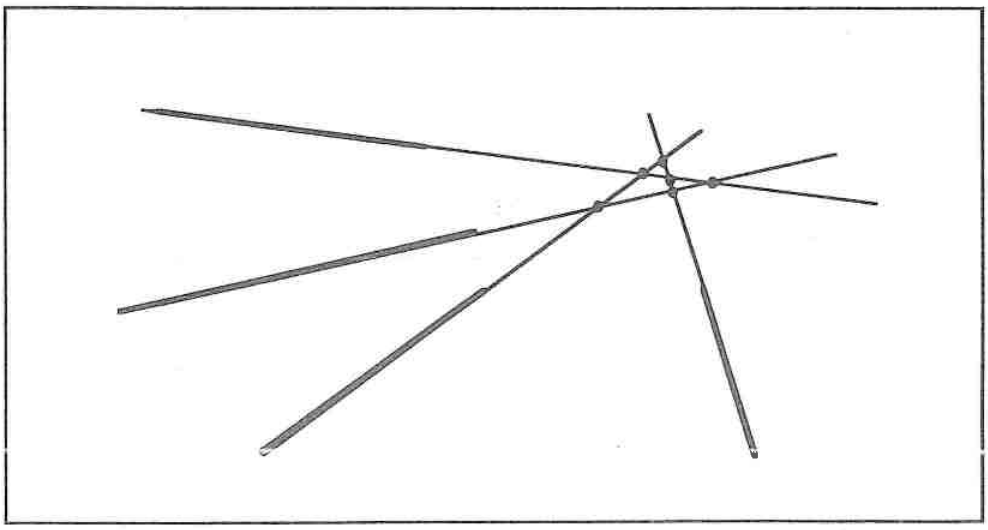


Figure 5.2. Point de fuite : l'intersection des segments du groupe directionnel au sens des moindres carrés

Chaque segment est representé par ses deux extrémités (U_1, V_1) et (U_2, V_2). La droite passant par ce segment est donc représenté par $aU + bV + c = 0$ ou bien sous forme matricielle,

$$(a \quad b)\begin{pmatrix} U \\ V \end{pmatrix} = -c$$

dont

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} = \begin{pmatrix} V_2 - V_1 \\ U_1 - U_2 \\ V_2(U_2 - U_1) - U_2(V_2 - V_1) \end{pmatrix} \text{ si } U_1 \neq U_2,$$

et

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ -U_1 \end{pmatrix} \text{ si } U_1 = U_2.$$

Tous les segments du groupe fournissent ainsi un système linéaire :

$$A\begin{pmatrix} U \\ V \end{pmatrix} = B$$

c'est-à-dire

$$A = \begin{pmatrix} \cdot \\ \cdot \\ \cdot \\ a_i \; b_i \\ \cdot \\ \cdot \\ \cdot \end{pmatrix}, B = \begin{pmatrix} \cdot \\ \cdot \\ \cdot \\ -c_i \\ \cdot \\ \cdot \\ \cdot \end{pmatrix},$$

Le système est normalement surdéterminé car le nombre des segments du groupe est strictement supérieur à 3 en utilisant la méthode de pseudo-inverse,

$$\begin{pmatrix} U \\ V \end{pmatrix} = (A^T A)^{-1} A^T B.$$

Nous obtenons donc une solution qui est le point de fuite au sens des moindres carrés.

Direction du groupe

Un point de fuite détermine la direction dans l'espace du groupe des segments parallèles de l'espace. Nous obtenons cette direction en utilisant la perspective inverse de ce point de fuite.

La perspective inverse indique qu'un point d'image correspond à une droite dans l'espace. La matrice de calibration C permet donc de déterminer cette droite dont le vecteur directeur est noté $\vec{v}$.

Écrivons la matrice sous forme :

$$C = \begin{pmatrix} C_1 \\ C_2 \\ C_3 \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & c_{34} \end{pmatrix}$$

Supposons que $X = (x, y, z, 1)^T$ et $(u, v, s)^T$ sont respectivement les coordonnées homogènes d'un point de l'espace et sa projection dans l'image. Nous avons :

$$\begin{aligned} u &= C_1 X \\ v &= C_2 X \\ s &= C_3 X \end{aligned}$$

et

$$\begin{aligned} U &= u/s \\ V &= v/s. \end{aligned}$$

donc,

$$\begin{aligned} C_3 X U &= C_1 X \\ C_3 X V &= C_2 X. \end{aligned}$$

C'est-à-dire,

$$\begin{aligned} (C_1 - C_3 U)X &= 0 \\ (C_2 - C_3 V)X &= 0 \end{aligned}$$

qui sont les équations de deux plans dont l'intersection est la droite qui est la perspective inverse du point. Nous les reécrivons comme :

$$\begin{aligned} a_1 x + b_1 y + c_1 z + d_1 &= 0 \\ a_2 x + b_2 y + c_2 z + d_2 &= 0 \end{aligned}$$

où

$$\begin{aligned} a_1 &= c_{11} - c_{31} U \quad & a_2 &= c_{21} - c_{31} V \\ b_1 &= c_{12} - c_{32} U \quad & b_2 &= c_{22} - c_{32} V \\ c_1 &= c_{13} - c_{33} U \quad & c_2 &= c_{23} - c_{33} V \\ d_1 &= c_{14} - c_{34} U \quad & d_2 &= c_{24} - c_{34} V. \end{aligned}$$

Sous forme de produit scalaire :

$$\vec{n_i} \cdot \vec{x} = -d_i$$

avec $\vec{x} = (x, y, z)$ et le vecteur normal des plans $\vec{n_i} = (a_i, b_i, c_i)$ pour $i = 1, 2$.

Le produit vectoriel des vecteurs normaux des plans conduit au vecteur directeur $\vec{v}$ de la droite.

$$\vec{v} = \vec{n_1} \times \vec{n_2} = (a_1, b_1, c_1) \times (a_2, b_2, c_2)$$

Donc pour chaque point d'image (U, V), nous obtenons un vecteur directeur $\vec{v}$ dans le repère absolu. Un point de fuite en donne aussi un.

Remarquons que ce vecteur directeur n'est pas orienté.

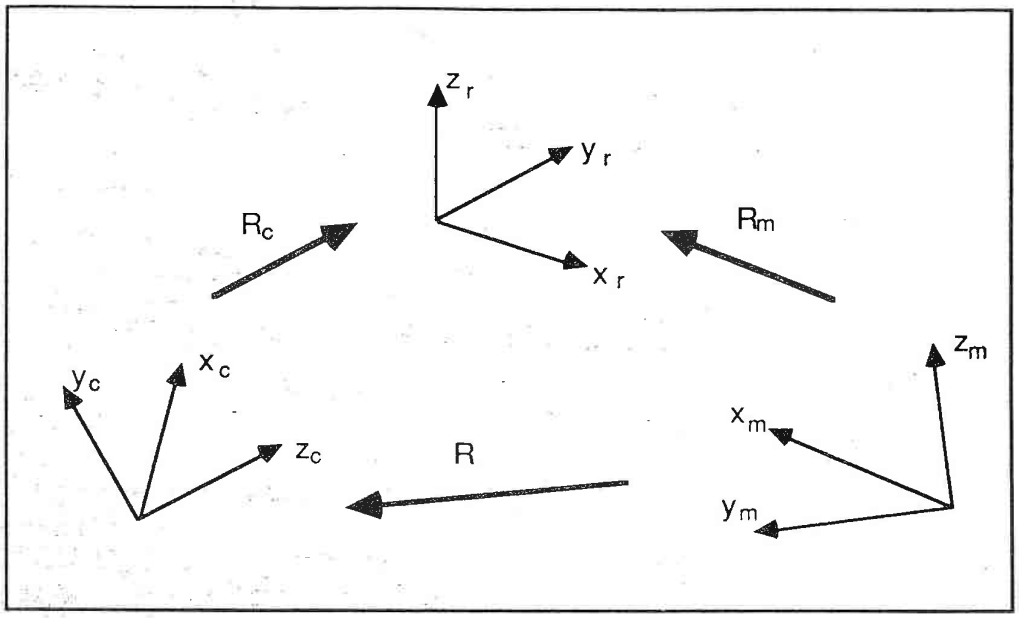


Figure 5.3. Détermination de la rotation finale par le repère auxiliaire

L'angle de rotation α autour de $\vec{a}$ est ensuite l'angle entre deux plans dont chacun passe par l'axe $\vec{a}$ et un vecteur. Notons les veceurs normaux comme suit :

$$\vec{p_1} = \vec{a} \times \vec{v_1} \; ; \; \vec{p_2} = \vec{a} \times \vec{v_{m1}}.$$

Alors,

$$\alpha = \arccos \frac{\vec{p_1} \cdot \vec{p_2}}{\| \vec{p_1} \| \| \vec{p_2} \|}.$$

Remarquons que théoriquement nous ne pouvons pas orienter le vecteur directeur de la droite dans le repère absolu. Donc sachant que 2 droites sont appariées, il existe 4 rotations possibles. En pratique comme nous le verrons en chapitre **??**, ce problème ne se pose pas.

5.3.2 Détermination de translation

Estimons le vecteur de translation $T = (t_x, t_y, t_z)$. Le principe que nous utilisons est le même que celui utilisé par Dhome [Dhome 88].

Supposons que nous avons une droite l_1 de l'image appariée avec la droite l_{m1}

du modèle. La droite l_1 peut être caractérisée par l'équation :

$$aU + bV + c = 0$$

en coordonnées homogènes,

$$l : \quad (a \quad b \quad c) \begin{pmatrix} u \\ v \\ s \end{pmatrix} = 0.$$

Prenons un point quelconque (x_0, y_0, z_0) de la droite l_{m1}, il subit d'abord la rotation R et la translation T et ensuite la projection par C sur l'image (u_0, v_0, s_0).

$$\begin{pmatrix} u_0 \\ v_0 \\ s_0 \end{pmatrix} = C \begin{pmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} R & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_0 \\ y_0 \\ z_0 \\ 1 \end{pmatrix}$$

Notons $(x_{r0}\ y_{r0}\ z_{r0})^T = R(x\ y\ z)^T$,

nous avons :

$$\begin{pmatrix} u_0 \\ v_0 \\ s_0 \end{pmatrix} = C \begin{pmatrix} x_{r0} + t_x \\ y_{r0} + t_y \\ z_{r0} + t_z \\ 1 \end{pmatrix}$$

Le point projeté doit donc être sur la droite de l'image, nous avons donc :

$$(a \quad b \quad c)\, C \begin{pmatrix} x_{r0} + t_x \\ y_{r0} + t_y \\ z_{r0} + t_z \\ 1 \end{pmatrix} = 0.$$

Nous obtenons ainsi une équation pour T avec une correspondance droite-droite :

$$A \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix} = B$$

dont

$$A = (ac_{11} + bc_{21} + cc_{31} \quad ac_{12} + bc_{22} + cc_{32} \quad ac_{13} + bc_{23} + cc_{33})$$

$$B = -(a\ b\ c)C \begin{pmatrix} x_{r0} \\ y_{r0} \\ z_{r0} \\ 1 \end{pmatrix}$$

Ce système se résout par au moins 3 appariements des segments. Quand plus de 3 correspondances sont disponibles, nous résolvons le système suivant au sens des moindres carrés :

$$T = (A^T A)^{-1} A^T B$$

Néanmoins comme l'ont indiqué Dhome *et al.* dans [Dhome 88] : dans le cas où trois segments forment une jonction en Y dans l'image, le système n'est pas indépendant, la solution n'est donc pas unique. Intuitivement, les trois plans d'interprétation se coupent sur la même droite. Cela signifie que nous ne pouvons obtenir que le vecteur directeur de la translation mais pas sa norme.

5.4 Travaux connexes et discussions

La formulation du problème est très claire, mais l'aspect mathématique du problème n'est pas facile à cause des paramètres non linéaires de rotation. De nombreux travaux ont déjà été effectués dans ce domaine.

La méthode de Church dans la littérature photogrammétrique donne une solution analytique en utilsant 3 points appariés entre le modèle et l'image [Fischler 81]. Mais l'équation non linéaire nécessite des méthodes itératives numériques pour la résoudre.

Fischler et Bolles [Fischler 81] donne une autre solution au problème. Les indices utilisés sont les points et l'équation reste non linéaire. Ce qui est intéressant sur le plan théorique est qu'ils ont explicitement analysé le cas où les solutions multiples apparaissent. Quand les 3 points sont appariés, le nombre des solutions peut atteindre 4. Les solutions multiples existent même pour 4 ou 5 points appariés. La solution devient unique quand 6 points sont appariés dans le cas général pour les 6 paramètres.

Ils existent aussi des travaux qui résolvent le problème dans le cas où la projection est orthogonale et non pas perspective (par exemple dans [Huttenlocher 87]), ce qui n'entre pas dans notre domaine d'intérêt.

Auparavant les travaux concernaient pour la plupart les appariements des points. Vue l'importance des indices segments dont la détection est beacoup plus fiable et la précision nettement meilleure que celles des points par la vision de bas niveau à l'heure actuelle, il est plus intéressant de développer des algorithmes utilisant des segments. Citons ici deux récents excellent travaux dans ce domaine :

1. *l'approche de Dhome et al.* :

   Dhome *et al.* dans [Dhome 88] résolvent analytiquement le problème dans le cas

où les indices sont des segments. La méthode consiste à décomposer la transformation en une rotation et une translation. La détermination se fait donc séparément et succesivement, une rotation suivie d'une translation. Les 3 segments appariés permettent d'établir une équation du huitième degré pour les 3 paramètres de la rotation connaissant la distance focale. Elle se résout ensuite par la méthode itérative de Bairstow, ce qui donne plusieurs solutions compatibles modulo une translation pure. La translation est ensuite déterminée par un système d'équations linéaires établies avec au moins 3 segments appariés. Les auteurs ont aussi analysé le cas dégénéré pour la translation, c'est-à-dire quand 3 segments de l'image forment une jonction en Y, ce qui donne lieu à un système non indépendant. Dans ce cas particulier, on ne peut donc récupérer que la direction du vecteur de la translation, la norme de ce vecteur restant indéterminée.

Quand il y a plus de 3 segments appariés, la méthode itérative de Lowe est utilisée pour avoir une solution plus précise au sens des moindres carrés.

Cette méthode présente un certain nombre d'avantages : les indices utilisés sont de type segment, le nombre minimal d'appariements (3) est raisonnable, elle est robuste puisqu'il n'y a qu'un seul cas dégénéré et la bonne solution est toujours présente selon les auteurs. L'inconvénient est que l'équation pour la rotation est hautement non linéaire. Il faut une méthode itérative pour la résoudre et les solutions sont multiples. Cependant quelques règles logiques suffissent (typiquement la visibilité des segments présents dans l'image) pour éliminer les mauvaises solutions d'après les auteurs.

2. *l'approche de Lowe* :

L'approche de Lowe [Lowe 87] repose sur la méthode de Newton-Raphson dans l'algorithmique numérique : méthode itérative de gradient pour résoudre un système d'équations non linéaires. C'est donc une méthode générale et rapidé : les types des indices utilisés peuvent être soit des points soit des segments. Connaissant les paramètres intrinsèques de la caméra, au moins trois indices sont nécessaires pour donner une solution des 6 paramètres de la transformation rigide. Quand il y a plus de trois appariements d'indices, la méthode mène naturellement à une solution au sens des moindres carrés. En plus, il est facile d'intégrer d'autres paramètres inconnus de la caméra ou du modèle, en augmentant bien entendu le nombre des indices appariées. Elle hérite aussi de l'avantage de la convergence rapide de la méthode de gradient (s'il converge vraiment !). La figure 5.4 montre la démarche dans le cas où les indices sont

des segments.

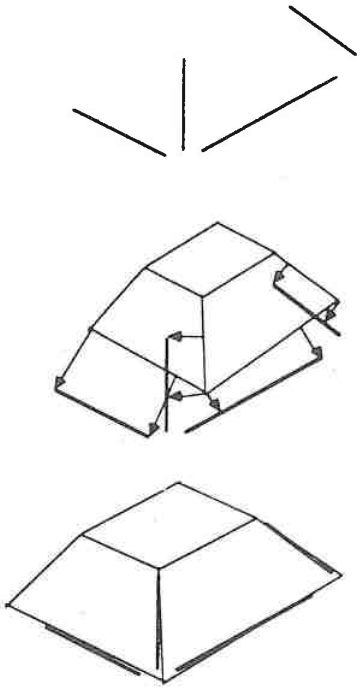


Figure 5.4. Les différentes étapes de l'application de la méthode de Newton-Raphson permettant la détermination spatiale

Néanmoins la paramétrisation du problème doit être minutieusement choisie pour que le système puisse converger correctement, surtout quand il s'agit des parmètres de la rotation. L'objectif d'une telle paramétrisation est de permettre de calculer facilement et indépendemment les dérivés partielles des paramètres inconnus par rapport aux coordonnées de l'image.

Quand la modélisation de la caméra se fait entièrement par une matrice de projection, comme l'a fait G. Toscani, au lieu d'une simple distance focale, la paramétrisation devient difficile et nous avons été confronté à un problème de convergence. Ceci est dû au fait que la transformation projective utilisée par Lowe est une apparoximation affine, non pas une vraie projection perspective. A.M. McIvor a aussi souligné ce point dans [Ivor 88].

Enfin, signalons qu'il existe aussi des travaux (cf. [Barnard 83,Horaud 87,Shakunaga 88]) dans lesquels on n'obtient pas de solutions entières mais plutôt des contraintes entre

les paramètres.

6
Appariement

6.1 Introduction

Dans un système de vision, les données visuelles et les connaissances que l'on possède sur la scène (c'est-à-dire le modèle) ont respectivement une représentation propre à chaque instant et à chaque niveau de traitement. L'*appariement* dans son sens large consiste à établir une correspondance entre les deux représentations existantes. Les termes *étiquettage* et *relaxation* sont aussi souvent utilisés dans un contexte plus précis. On peut parler d'interprétation ou de reconnaissance des données visuelles lorsqu'il s'agit d'apparier des données visuelles avec des modèles d'objets. Ce chapitre y est consacré. Le chapitre suivant sera consacré à l'appariement de deux images.

Nous présentons dans 6.2 un bref survol des approches existantes et des discussions aportées à chaque approche. La classification des méthodes est inspirée de [Mohr 88a,Mohr 88c]. Ensuite nous décrivons notre méthode de l'appariement d'une image avec un modèle. Le chapitre se termine par une discussion sur la méthode utilisée.

6.2 Différentes approches

6.2.1 Transformée de Hough

On parle de *modèle paramétré* [Mohr 88a], il s'agit là, d'une famille de modèles les plus simples dans le cas d'une fonction tabulée ou d'une fonction analytique avec quelques paramètres. Typiquement une droite peut être paramétrée par deux coordonnées polaires ρ et θ ; un cercle par son centre et son rayon etc. La transformée de Hough est particulièrement bien adoptée dans le cas où l'une des deux représentations à apparier est du type paramétré. Nous avons déjà vu le principe

de la transformée de Hough dans 3. On transforme les indices de l'image à apparier en des points dans l'espace de paramètres et ensuite on recherche les zones d'accumulation en vue d'une reconnaissance ou d'une interprétation.

C'est une méthode globale qui résiste bien aux bruits de l'image et qui est aussi facilement parallélisable. Pourtant le domaine d'applications est assez limité puisqu'elle ne convient qu'à des modèles paramétrés. De plus, lorsque le nombre des paramètres augmente, l'espace de recherche devient trop important. Une technique souvent employée consiste à diviser récursivement l'espace des paramètres afin de ne se focaliser finalement que sur des zones intéressantes (cf. [Muller 84], [Quan 89a]).

Le chapitre 3 (détection des points de fuite) fournit un exemple pour apparier un ensemble de segments de l'image avec des directions principales du modèle.

6.2.2 Approches liées aux graphes

Homomorphisme, isomorphisme

Quand la représentation des données visuelles, et celle du modèle sont relationelles, elles peuvent être considérées comme un graphe dont les nœuds sont des indices visuels de l'image ou des objets du modèle et les arêtes sont des relations entre eux. L'appariement de deux graphes se formalise donc tout naturellement comme la recherche de l'*homomorphisme* (application linéaire) entre deux graphes [Ballard 82] et [Shapiro 81]. En particulier, cet homomorphisme peut être, un *isomorphisme* (homomorphisme bijectif) entre les deux graphes, ou l'isomorphisme d'un sous-graphe du premier graphe dans le second, ou bien l'isomorphisme d'un sous-graphe du premier graphe dans un sous-graphe du second (cf. la figure 6.1).

On peut aussi considérer que certaines relations sont plus importantes que d'autres, et introduire pour cela une fonction de pondération qui associe un poids à chaque d'instance d'une relation. L'intérêt de cette pondération est de tolérer quelques instances de relation du premier graphe ne trouvent pas de correspondant dans le deuxième ; par ailleurs cela fournit une mesure de qualité de l'homomorphisme. L'homomorphisme (resp. isomorphisme) est dit *ϵ-homomorphisme* (resp. ϵ-isomorphisme), ϵ étant le seuil limite que peut atteindre le cumul des pondérations des instances des relations non vérifiées. Shapiro et Haralick [Shapiro 81] ont qualifié cet appariement d'*inexacte*.

Ce formalisme est général et élégant, mais ces algorithmes relèvent de la classe de problèmes NP-difficiles[1]. Il est donc difficile d'envisager une application pratique

1. Aussi difficile que la détermination de la valeur chromatique d'un graphe.

sans heuristiques puissantes.

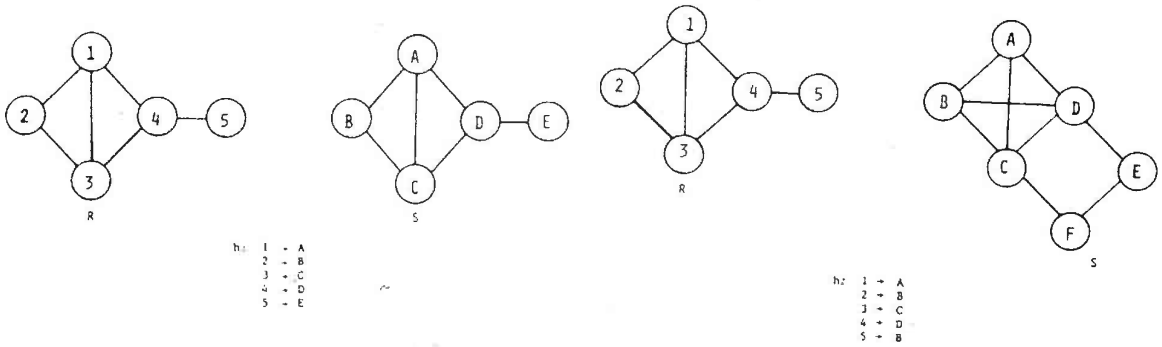


Figure 6.1. Isomorphisme, homomorphisme [Shapiro 81].

Clique maximale

Etant donnés deux graphes à apparier et des règles de compatibilité entre les relations de deux graphes, on peut construire un graphe dont les nœuds sont des couples appariés et l'arête reliant deux nœuds signifie que ces deux nœuds sont mutuellement compatibles. L'absence d'une arête entre deux nœuds signifie leur incompatibilité mutuelle. Le meilleur appariement correspond donc parfaitement au problème des cliques maximales d'un graphe qui est un problème bien connu de la théorie du graphe. (voir la figure 6.2). Une clique est un sous-graphe complet du graphe. Cette technique a été utilisée par de nombreux auteurs, notamment [Bolles 82] et [Skordas 87].

Cependant la recherche des cliques maximales est un problème combinatoire ; la seule possibilité d'y recourir est d'avoir des contraintes suffisament fortes entre indices pour qu'il n'y ait que peu d'arêtes dans le graphe construit.

6.2.3 Relaxation

Relaxation discrète

Considérons deux représentations à apparier. La première (par exemple celle de l'image) est une structure relationelle que l'on peut représenter comme un graphe dont les nœuds sont des indices visuels et les arêtes des relations entre eux. La seconde (par exemple celle de modèle) est aussi une structure relationnelle que l'on

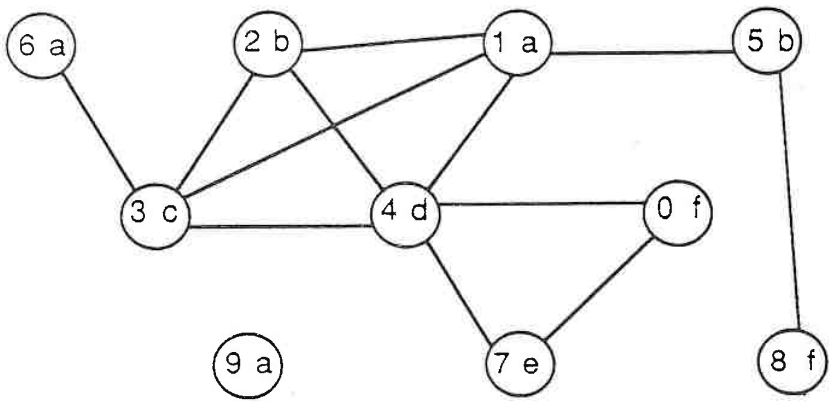


Figure 6.2. Clique maximale : chaque nœud correspond à une hypothèse de correspondance. Le groupe d'hypothèses $(1, a), (2, b), (3, c), (4, d)$ est une clique maximale - elle ne peut plus être étendue. Le groupe $(1, a), (5, b)$ est aussi une clique de taille maximale, mais plus petite [Mohr 88a].

peut représenter sous forme d'un ensemble d'étiquettes et de contraintes qui sont des relations entre étiquettes. Les contraintes sont en général binaires, on peut se reporter à [Mohr 88b] pour les extensions possibles.

Le principe est d'associer à chaque nœud du graphe l'ensemble des étiquettes qui peuvent lui correspondre. Les contraintes des étiquettes créent donc un réseau d'arcs sur ces nœuds, on transforme le réseau en un graphe. (voir la figure 6.3). Ensuite on élimine toutes les étiquettes qui ne sont pas compatibles avec les étiquettes voisines en utilisant les contraintes.

Il est évident que la recherche de la solution globale consistante est un problème NP-difficile. Cependant le problème peut se poser différemment : on parle de *consistance d'arcs* si la condition plus faible de consistance locale (au lieu de la consistance globale) est satisfaite. L'intérêt de cette consistance locale est qu'elle est suffisante pour beaucoup de problèmes réels et surtout qu'il existe des algorithmes polynomiaux pour les calculer [Mohr 88b]. Il est néanmoins difficile de manipuler des erreurs avec cette approche [Mohr 88b].

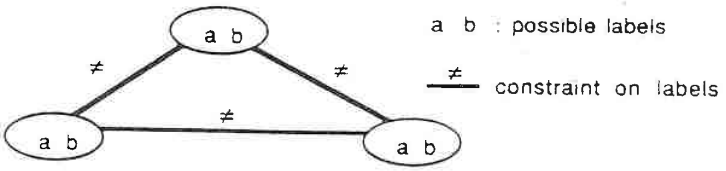


Figure 6.3. Un réseau de contraintes [Mohr 88a] : les étiquettes doivent toutes être différentes.

Relaxation continue

Contrairement à la relaxation discrète dont les contraintes sont de type booléen vrai/faux, la relaxation continue (voir [Faugeras 81] et [Bahnu 84]) manipule des contraintes portant sur des valeurs continues (depuis "impossible" jusqu'à "tout à fait admissible"). Le problème essentiel est alors de construire des valeurs continues de compatibilités entre contraintes afin de pouvoir garantir une convergence du processus de relaxation vers une solution locale satisfaisante.

6.2.4 Prédiction-Vérification

Voyons maintenant le schéma Prédiction-Vérification, qui a été utilisé dans beaucoup de systèmes avec succès (cf. [Lux 85], [Ayache 86], [Ayache 87]). La Prédiction-Vérification est l'adaptation en vision de la méthode générale d'intelligence artificielle de la recherche avec retour arrière. La prédiction consiste à générer une hypothèse d'appariement, c'est-à-dire exploiter une possibilité ; la vérification élargit par la suite les hypothèses déjà formées jusqu'à ce qu'une inconsistance apparaîsse ; on effectue alors un retour en arrière vers la solution partielle la plus récente. La solution finale est l'ensemble d'hypothèses consistant.

Evidemment, cette méthode exhaustive est dans le pire cas exponentielle. En pratique, cette démarche est satisfaisante lorsque le modèle à apparier est de type *rigide*. Si l'on prend la classification des modèles géométriques suggérée par Mohr [Mohr 88a,Mohr 88c], on parle de *modèle rigide*, quand on considère que les formes sont des assemblages de formes primitives rigides. Typiquement les formes primitives peuvent être des segments de droite [Ayache 86], [Ayache 88] ou des sommets de trièdres [Horaud 87]. Une façon de les assembler est de considérer une disposition fixe des différentes primitives modulo une transformation de l'espace de travail (par exemple $\mathbb{R}^3$ muni des déplacements). La détermination de quelques hy-

pothèses initiales (souvent une hypothèse d'appariement sur les groupes perceptuels de grande taille) permet d'estimer les paramètres de transformation rigide, et à partir de là, la vérification devient immédiate. La combinatoire est importante au début de l'appariement (l'étape de prédiction), pourtant l'estimation de la transformation va très rapidement conduire à une suite quasi-déterministe d'appariements.

Lorsqu'il s'agit de *modèles génériques*, c'est-à-dire quand les formes ne sont pas définies de façon exacte mais d'une façon qui permet des variations en proportion définies par des contraintes [Mohr 88c], ils ne sont plus adaptés à la prédiction-vérification. La prédiction ne pourra se faire que pour les parties rigides prises isolément d'une part ; et d'autre part il faut avoir plus d'hypothèses initiales non seulement pour estimer la transformation rigide mais aussi les paramètres des modèles (par exemple leurs échelles).

6.2.5 En résumé

En résumé, la transformée de Hough se limite à l'identification de formes paramétrées telles que des courbes analytiques simples. La relaxation est plutôt un outil de filtrage des interprétations possibles qu'un outil d'interprétation directe. La recherche d'homomorphismes et des cliques maximales sont des problèmes NP-difficiles, peu d'applications sont envisageables tant qu'il n'y a pas de prétraitement efficace du graphe.

L'approche de prédiction-vérification nous semble bien privilégiée et plus prometteuse lorsque le modèle à apparier est rigide. Il reste qu'elle garde du risque d'explosion combinatoire lors de la prédiction. La réussite d'une telle approche dépend donc de la manière de réduire cet espace de prédiction.

Dans notre approche, une adaptation de prédiction-vérification que nous allons présenter dans la suite, l'espace de prédiction est considérablement réduit grâce au regroupement perceptuel des indices visuels et à l'utilisation des contraintes géométriques. Ceci nous permet d'avoir un algorithme très efficace.

6.3 Appariement avec un modèle

6.3.1 Stratégie générale

L'objectif est donc de produire une liste de paires de segments image/modèle. Dans cette étape, nous disposons de trois données :

1. une image segmenté en contours, eux mêmes approchés par des segments de droite agglomérés en groupes perceptuels par les algorithmes que nous avons présentés en chapitre 4,
2. un modèle partiel, lui aussi en termes de groupes perceptuels (voir plus loin pour une brève présentation),
3. et la matrice de calibration de la caméra utilisée.

On suppose que

- les objets sont rigides ;
- il existe des structures parallèles dominantes dans la scène ;
- la caméra se déplace à peu près dans un plan horizontal.

La stratégie générale de l'appariement est :

1. *Détermination de la rotation* de la transformation en appariant au moins deux groupes directionnels.
2. *Prédiction* :
   - cette étape génère d'abord une liste d'appariements hypothétiques de groupes perceptuels.
   - A partir de chaque appariement hypothétique de groupe perceptuel, on obtient plusieurs listes d'appariements hypothétiques de droites appartenant aux groupes persceptuels hypothétiquement appariés.
   - Chaque liste d'appariements hypothétiques de droites permet d'estimer une transformation rigide de la caméra.

   La sortie de la prédiction est donc la liste des transformations possibles.
3. *Vérification de transformation* : ce module retourne la meilleure transaformation selectionnée et la liste de droites appariées par cette transformation.
4. L'*extension* consiste à affiner l'estimation de transformation et à produire la liste d'appariements des segments.

Ce principe peut se schématiser par la figure 6.4. Nous décrivons en détail chaque étape dans la partie qui suit.

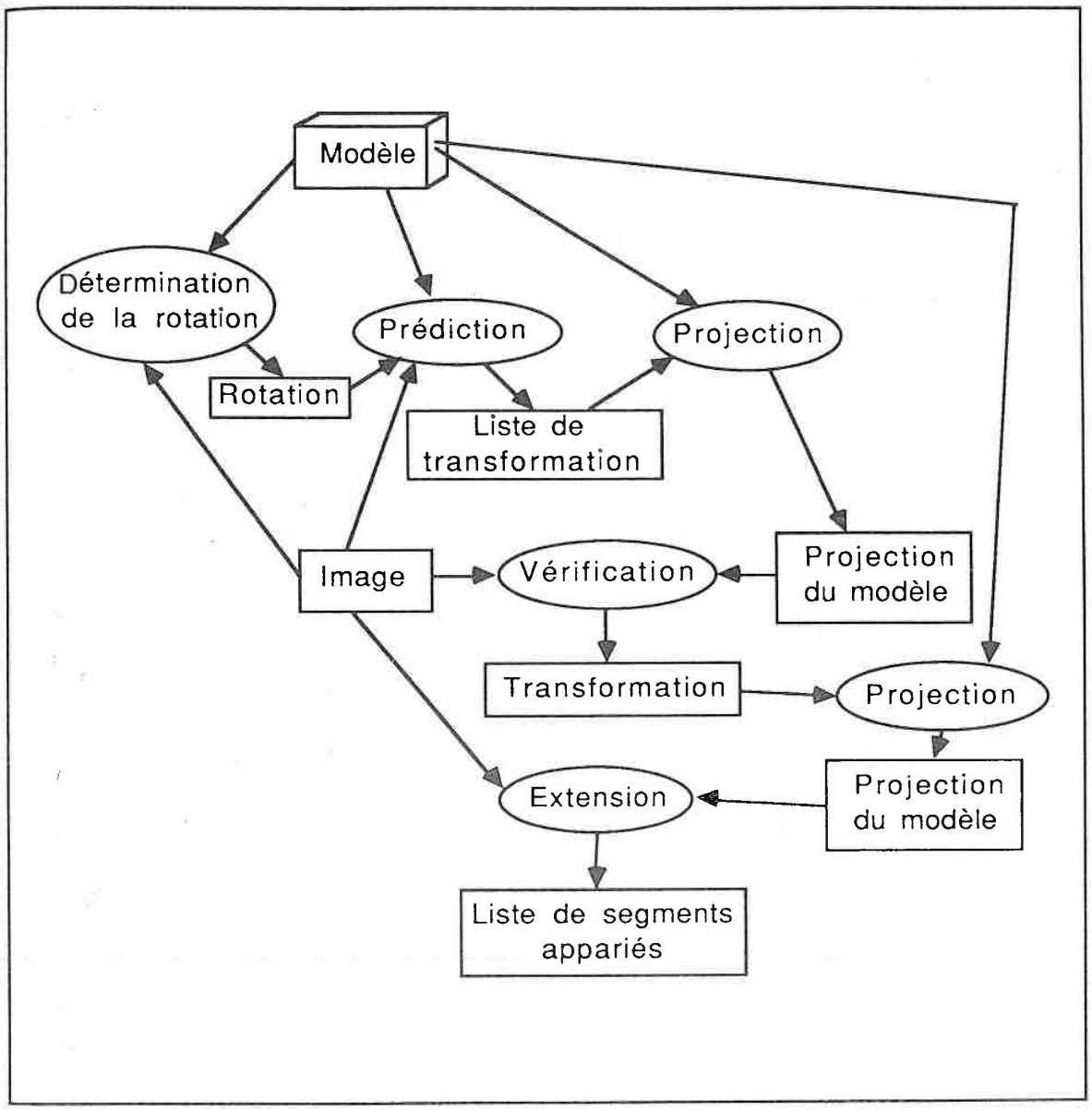


Figure 6.4. Le schéma de l'appariement.

6.3.2 Construction du modèle

La construction automatique du modèle d'un environnment inconnu est encore un sujet difficile dans le domaine de vision artificielle jusqu'à présent. De récents travaux entrepris dans le projet ORASIS semblent avoir de résultats intéressants dans ce domaine, on peut se reporter aux thèses de N. Ayache [Ayache 88] et de E. Thirion [Thirion 89]. Ici, nous nous contendons d'avoir un modèle partiel qui est construit à partir d'une perception partielle stéréoscopique de la scène. Cette perception partielle est sous forme d'un ensemble de segments *3D*. On projette cet ensemble de segments *3D* sur le plan image d'une caméra et on effectue le même traitement sur cette projection que sur une image normale, c'est-à-dire le regroupement d'indices visuels. Disposant des informations *3D*, il est possible de les utiliser par la suite pour vérifier les groupes perceptuels formés dans la projection en éliminant manuellement les faux groupes perceptuels.

Cette représentation similaire de l'image et du modèle, en terme de mêmes types de groupes perceptuels, facilitera la prédiction de l'appariement.

6.3.3 Détermination de la rotation

Tout d'abord il faut mettre en correspondance deux groupes directionnels afin de récupérer la rotation de la transformation par l'algorithme de la section 5. Ceci se fait de manière déterministe dans l'implantation actuelle en utilisant deux heuristiques suivants :

1. On peut toujours avoir au moins 2 groupes directionnels dominants selon l'hypothèse de départ. Normalement ils en sont trois dans une scène d'intérieur dont l'un vertical et les deux autres horizontaux, dans certains cas on en trouve un vertical et un horizontal et dans d'autres on peut aussi avoir un quatrième ne correspondant pas à une direction principale. (cf. les figures des résultats dans le chapitre 3) ;

2. il existe une faible rotation de l'image dans le plan horizontal par rapport au modèle actuel, c'est-à-dire que cet angle ne doit pas dépasser $\pm\pi/4$.

Deux groupes directionnels se correspondent si l'angle qu'ils forment est le plus faible (cf. figure 6.5), cet angle est défini dans le paragraphe 2.2.2.

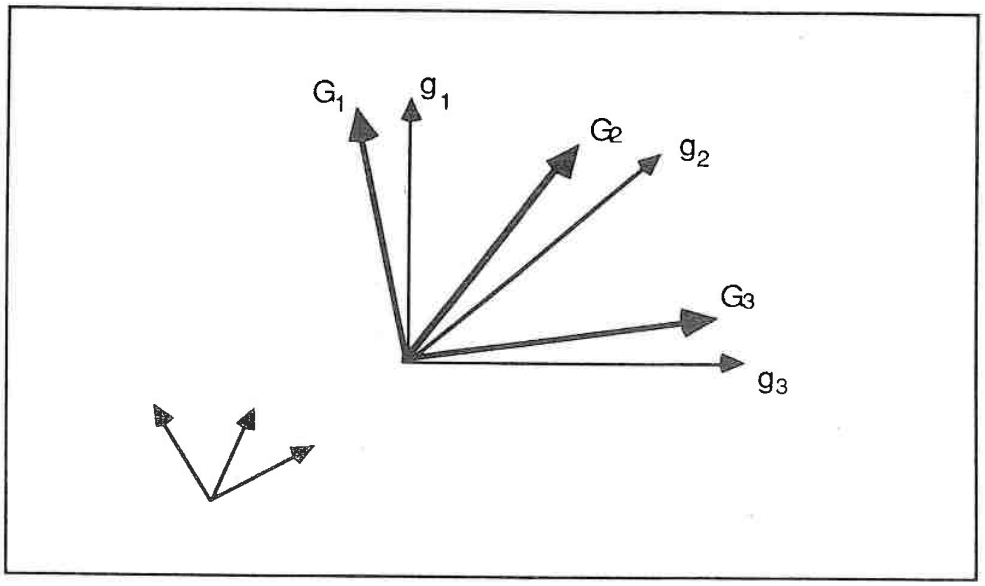


Figure 6.5. L'appariement des groupes directionnnels est déterministe.

6.3.4 Prédiction

Appariement hypothétique de groupes perceptuels

Etant donnés un ensemble de groupes perceptuels bidimensionnels qui représente une image et un deuxième ensemble de groupes perceptuels tridimensionnels représentant le modèle, la première étape de la prédiction consiste à comparer chaque groupe perceptuel de l'image avec chacun des groupes du modèle afin d'obtenir une liste des appariements hypothétiques.

A chaque groupe perceptuel, les caractéristiques suivantes lui sont associées (cf. paragraphe 4) :

- type de groupe (groupe convexe ou groupe tronc ...),
- direction de groupe (cf. paragraphe 4),
- position relative à la ligne d'horizon (audessus ou audessous).

Seuls des groupes troncs et des groupes convexes sont pris en compte dans l'implantation actuelle. On vérifie alors les caractéristiques de chaque couple de groupes perceptuels : si le type, la direction et la positions relatives sont compatibles,

alors ils forment un appariement hypothétique de groupes perceptuels. Cette étape nous fournit donc une liste d'appariements hypothétiques de groupes perceptuels.

Appariement hypothétique de droites

Les droites (groupes de segments colinéaires) appartenant à un groupe perceptuel sont celles dont l'un de leur segments colinéaires fait partie du groupe perceptuel. A partir de chaque appariement hypothétique de groupes perceptuels, on devrai ensuite mettre en correpondance des droites appartenant aux groupes perceptuels. Pour ce faire, les groupes perceptuels sont choisis de telle manière à favoriser l'appariement de droites appartenant aux leur groupes perceptuels. C'est le cas pour les groupes convexes (cf. figure 6.6) sachant que les groupes directionnels ont déjà été appariés. Pour les groupes troncs, l'appariement de droites représentant le tronc ne pose aucune ambiguïté, car le tronc est unique pour chaque groupe tronc. Il reste qu'il est encore possible de faire un décalage d'un groupe perceptuel par rapport à un autre au long du tronc apparié, même si le nombre de combinaisons est faible (cf. figure 6.7). Ainsi chaque appariement hypothétique de groupes perceptuels peut avoir plusieurs listes d'appariements de droites. Cette étape nous donne finalement plusieures listes d'appariements hypothétiques de droites.

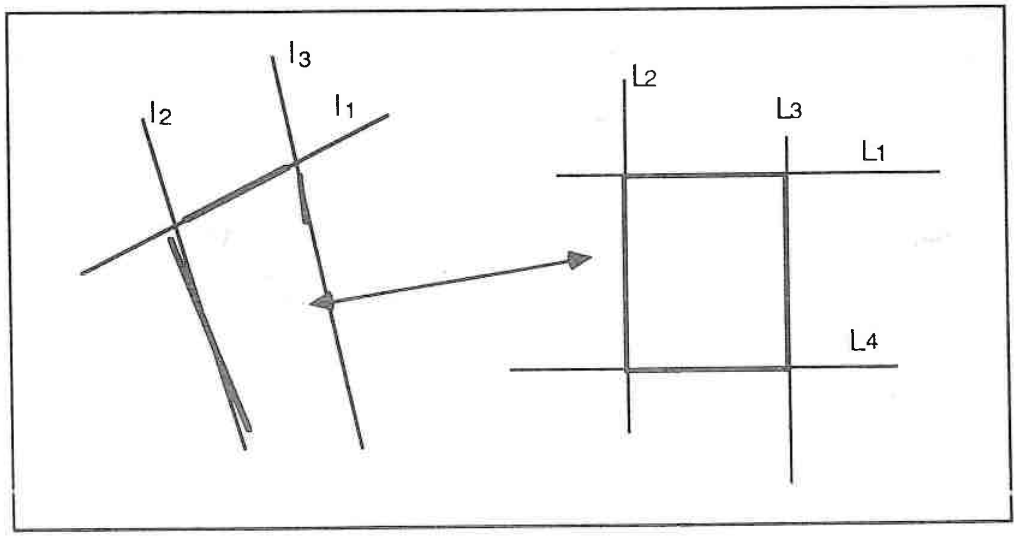


Figure 6.6. L'appariement des droites pour un groupe convexe est sans ambiguïté grâce aux relations directionnnelle et de connexité des segments.

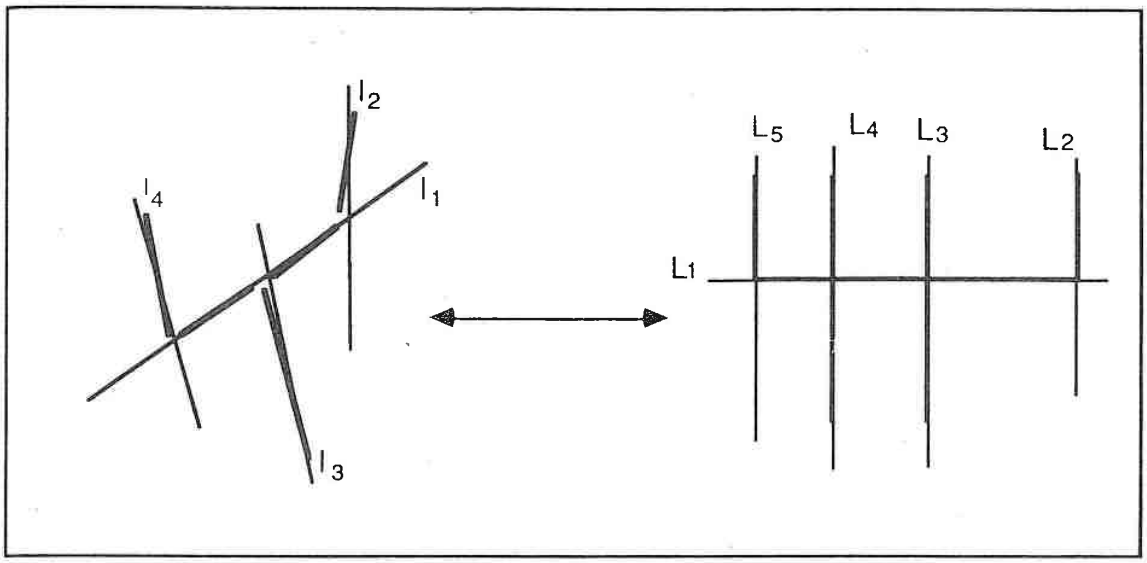


Figure 6.7. L'appariement des droites pour un groupe tronc.

Estimation d'une transformation

On peut ensuite estimer une translation de la transformation par l'algorithme de la section ?? en utilisant chaque liste d'appariements de droites. Rappelons que l'estimation d'une translation nécessite au moins trois appariements de droites et de plus ces droites de l'image ne doivent pas présenter une jonction de type Y. Donc chaque liste d'appariements de droites dont la longueur est supérieure ou égale à trois et dont les droites ne sont pas en jonction de type Y permet d'obtenir une translation possible.

Comme chaque translation composée par la rotation et la matrice de calibration donne une transformation complète de la caméra, cette dernière étape de la prédiction nous fournit une liste de transformations possibles de la caméra au modèle.

6.3.5 Vérification

L'objectif de cette étape est de vérifier chaque transformation possible issue de la prédiction et de retenir la meilleure. Comme une transformation permet de projeter le modèle dans le plan image, on fait alors la vérification dans le domaine bidimensionnel en comparant l'image et la projection du modèle.

Qualité d'une transformation

La meilleure transformation signifie qu'elle amène les indices identifiés du modèle aussi près que possible des indices visuels de l'image. Pour évaluer la qualité d'une transformation, on définit un *score* qui est la proportion des droites appariées par cette transformation.

Critères d'appariement

Soient l une droite de l'image et L une droite du modèle,

- La "*distance*" entre deux droites est la distance entre leurs segments porteurs (cf. 4), c'est-à-dire

$$dist(l, proj(L)) = dist(seg(l), seg(proj(L))).$$

- Le "*recouvrement*" de deux droites est celui de leurs segments porteurs, c'est-à-dire

$$recouv(l, proj(L)) = recouv(seg(l), seg(proj(L))).$$

Alors un couple de droites (l, L) est en correspondance si

$$recouv(l, proj(L)) > \epsilon \wedge dist(l, proj(L)) < \epsilon'.$$

Module de vérification

Ce module consiste à comparer toutes les droites de l'image avec celles de la projection du modèle. Il faut noter que, pour accélérer ce procédé, seules les droites appartenant aux groupes directionnels sont prises en considération et non pas toutes les droites ; en effet les droites appartenant au même groupe directionnel sont triées selon les angles associés (cf. 4). Ce tri des droites permet de faire la vérification en $O(n)$, puisque la comparaison successive de chaque couple de droites correspond simplement à l'interclassement de deux listes triées en $O(n)$ [Baase 78].

La meilleure transformation est la transformation dont le score est le plus grand. La vérification fournit aussi la liste des appariements de droites par la meilleure transformation.

6.3.6 Extension

Affinement de la transformation

L'estimation de la transformation retenue est ensuite affinée par la méthode de la section 5 en utilisant la liste des droites appariées. Plus précisément c'est le composant de translation qui est affiné.

Extension de la liste des appariements des droites

Rappelons que seules les droites appartenant aux groupes directionnels ont été utilisées dans la vérification. A l'aide de la transformation affinée, la recherche des appariements des droites s'étend sur toutes les droites de l'image en utilisant toujours les même critères d'appariement de droites. Notons qu'un segment qui n'est colinéaire avec aucun autre est considéré comme une droite dont le segment porteur est ce segment lui-même.

Obtension de la liste des appariements des segments

Lorsque la liste des appariments de droites a été complétée, on peut déterminer les appariements des segments faisant partie d'un appariement de droites. Ceci est immédiat, il suffit de tester le recouvrement des segments.

6.4 Travaux connexes

Nous présentons ici quelques travaux connexes en vue d'une comparaison avec la nôtre. Ces travaux nous semblent représentatifs dans ce domaine, mais cette liste n'est loin d'être exhaustive.

- D. Lowe dans son système SCERPO (cf. [Lowe 85b,Lowe 87]) étudie le problème de la reconnaissance d'objets *3D* (par exemple rasoirs) à partir d'une image monoculaire avec un point de vue inconnu.

  Les objets à reconnaître sont modélisés *a priori* géométriquement par un système CAO. L'image est segmentée en segments de droite.

  Après la segmentation, le processus d'organisation perceptuelle permet de former des groupes de segments qui sont probablement invariants par changement de point de vue. Ce regroupement est basé sur la proximité, le parallélisme et la colinéarité. L'algorithme de détermination spatiale utilisé dans le système

SCERPO est une méthode itérative de gradient (cf. ??). L'appariement est un cycle de prédiction/vérification. La prédiction initiale est obtenue en utilisant les groupes perceptuels, ce qui permet de sélectionner par un raisonnement probabiliste les modèles les plus plausibles, et de déterminer une estimation de la position spatiale de ces modèles dans l'espace. Cette estimation permet aussi de projeter des segments *3D* du modèle dans l'image pour vérifier leur présence en calculant des distances géométriques. C'est un système complet et robuste.

- Le système MOSAIC (cf. [Herman 86]) est un système de reconstruction automatique des scènes *3D* à partir d'une séquence d'images monoculaires ou d'une séquence d'images stéréoscopiques. Il a été appliqué à des images aériennes de scènes urbaines pour modéliser des bâtiments.

  Le modèle est principalement construit par le module de stéréovision du système dont nous ne parlerons pas ici. Le modèle ainsi construit utilise trois primitives géométriques : le point, la droite et le plan ainsi que des entités topologiques : segment, jonction, groupe de segments connexes et coplanaires.

  L'analyse monoculaire commence par l'extraction des segments de l'image et qui sont ensuite regroupés en jonction. La connaissance *a priori* du point de fuite vertical permet d'inférer la profondeur de jonctions supposées en contact avec le sol et celle de segments alignés avec un segment dont la profondeur est connue à un facteur près, ce facteur est la distance du point de vue au sol.

  L'appariement d'une telle description de l'image avec le modèle partiel permet la mise à jour du modèle. L'appariement devient immédiat à partir du moment où l'on suppose connu le mouvement des caméras.

  L'objectif de MOSAIC est très similaire au nôtre sauf que nous travaillons sur des images de scènes d'intérieur. Cependant, il est regrettable que certains problèmes délicats n'ont pas été abordés. En particulier, le problème de l'appariement n'est pas pris en compte puisque la transformation entre deux vues est connue.

- L'ambition du système ACRONYM (cf. [Brooks 81]), développé par R. Brooks, est d'être un système de reconnaissance d'objets indépendant du domaine d'application. Il est incontestablement le premier système de vision qui cherche à combiner des facilités générales de modélisation et des images "naturelles" sans contrainte de position ou d'éclairage. Le système a été testé avec des vues

aériennes d'un aéroport, et avec des images de tournevis électrique. Pourtant l'état exact de la réalisation du système n'est pas bien précisé.

Les modèles d'objets dans ce système sont de type générique, c'est-à-dire que les objets sont modélisés par les volumes occupés et par des transformations entre repères locaux de ces volumes. Les volumes primitifs sont représentés par des cônes généralisés. L'image est décrite par un graphe en termes de bandes et d'ellipses.

L'interprétation d'une image consiste à rechercher un isomorphisme de sous-graphes entre le graphe image et le graphe des prédictions. Ce dernier est construit par des techniques de raisonnement géométrique grâce aux groupes des indices visuels invariants.

6.5 Résultats expérimentaux et discussion

Les figures 6.8 et 6.9 montrent deux estimations initiales dont la deuxième est retenue comme la meilleure. La figure 6.10 illustre la transformation affinée par l'appariement des segments.

Les points forts de l'approche sont :

- l'aspect combinatoire de la prédiction est completement évitée grâce aux groupes perceptuels de grande taille et grâce aux fortes contraintes géométriques.
- L'aspect numérique de la méthode, l'estimation de la transformation et les critères d'appariement, sont plutôt basés sur la localisation des segment que de leurs extrémités. Ce point est important car la vision de bas niveau est nettement meilleure pour la localisation d'un segments que pour la détermination exacte de ses extrémités.
- Le module de vérification est efficace, en $O(n)$ par rapport au nombre de droites, et sans retour arrière. Ceci est possible grâce à la rigidité des objets et grâce à la bonne estimation initiale de transformation de départ.

Le point faible de l'approche est qu'elle exige l'existance de structures parallèles dominantes, ce n'est pas toujours le cas pour des scènes d'intérieur.

Enfin remarquons que l'exploitation d'heuristiques en vue d'un appariement déterministe de groupes directionnels ne réduit pas la généralité du problème. S'il en manque, il suffit d'augmenter le nombre de prédictions. Ainsi le nombre de

prédictions de rotations en appariant deux droites non orientées est quatre, ce nombre atteint huit dans le cas où on ne pent pas distinguer deux directions horizontales.

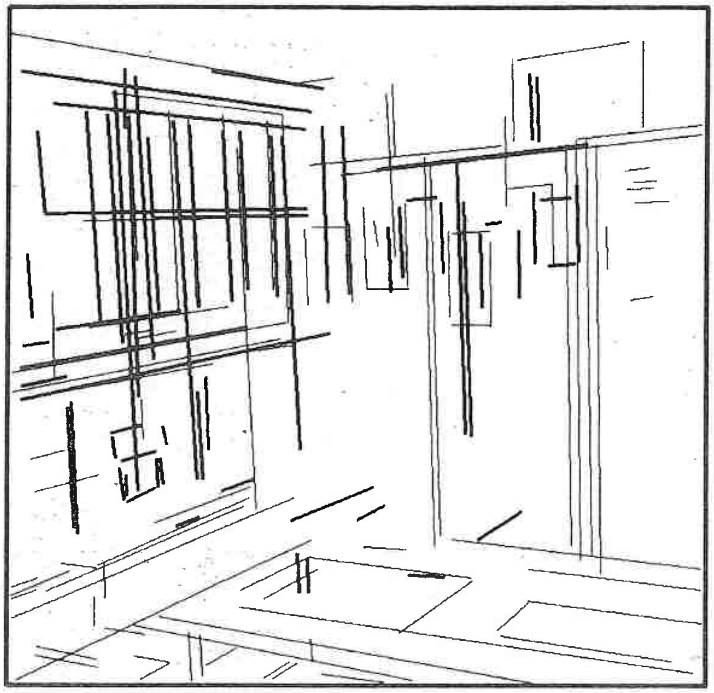


Figure 6.8. Une estimation initiale de la transformation : les groupes colinéaires du modèle (en gras) et les groupes colinéaires de l'image sont en superposition.

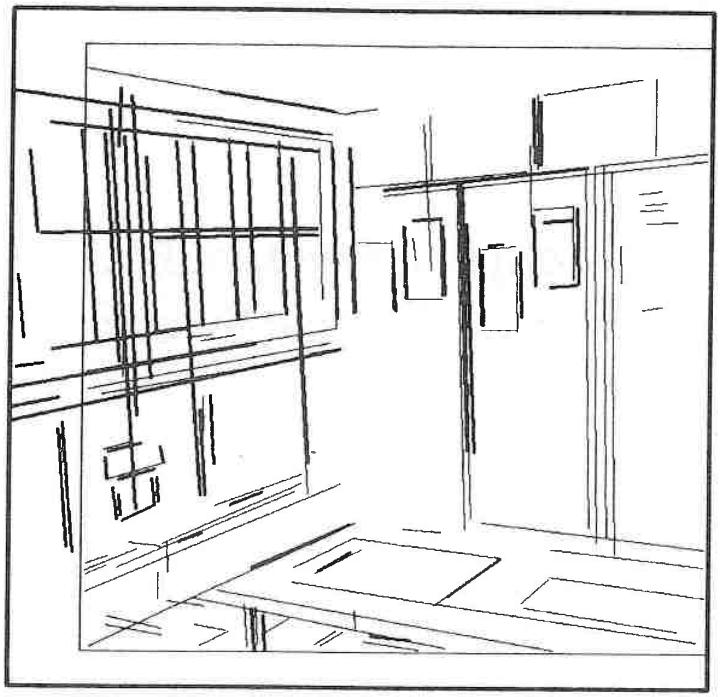


Figure 6.9. La meilleure estimation initiale de la transformation.

Figure 6.10. Le résultat final de l'estimation de la transformation.

7

Une application : appariement avec d'autres images

Le sujet abordé dans ce chapitre est l'appariement des indices visuels dans deux images. Celui ci ne se situe pas sur l'axe principal qu'a voulu aborder cette thèse, c'est-à-dire l'appariement d'une image avec un modèle. Pourtant l'appariement de deux images peut être considéré comme l'appariement d'une image avec un modèle au sens large : dans ce cas, le modèle n'est plus tridimensionnel, il n'est rien qu'une autre image bidimensionnelle. Ce chapitre est donc complémentaire. L'objectif est de démontrer encore une fois que grâce au regroupement perceptuel et à la découverte de contraintes géométriques intéressantes, l'espace de recherche est considérablement réduit. L'algorithme d'appariement devient alors quasi déterministe.

7.1 Introduction

Le problème d'appariement des images peut se poser dans le contexte du *mouvement discret* en vision par ordinateur dont l'objectif est de retrouver les données 3D en utilisant le mouvement de la caméra (cf. 1).

L'analyse du mouvement discret se décompose généralement en trois étapes principales :

1. l'extraction des indices visuels,
2. l'appariement des indices visuels,
3. et la reconstruction : estimation du mouvement et de la structure de l'environnement.

Nous n'abordons pas le problème de la reconstruction ni celui de l'extraction des indices visuels. Néanmoins nous citerons très brièvement quelques résultats

aboutis pour le problème d'estimation du mouvement dans le paragraph **??**, une fois l'appariement réalisé.

Nous nous intéressons en particulier au problème d'appariement des indices visuels. C'est un problème essentiel dans l'analyse du mouvement discret, qui a encore été peu abordé et qui est loin d'être résolu dans le cas général. Nous l'abordons dans ce chapitre en supposant que :

- les objets sont statiques et rigides ;
- il existe au moins deux directions majoritaires dans la scène ;
- la caméra se déplace dans le plan horizontal ;
- et son déplacement est faible.

7.2 Organisation perceptuelle

L'analyse monoculaire de chaque image est essentiellement le regroupement perceptuel de segments 2D. De plus, nous choisissons une organisation hiérarchique des groupes perceptuels dans cette application.

- Chaque image est un ensemble de groupes directionnels.
- Chaque groupe directionnel est un ensemble de groupes raies.
- Chaque groupe raie est un ensemble de droites.
- Chaque droite est un ensemble de segments.

Cette organisation hiérarchique est intéressante car elle permet d'avoir une stratégie d'appariement de "*coarse-to-fine*".

7.3 Contraintes géométriques

La recherche de contraintes géométriques est toujours un sujet intéressant. La contrainte de coplanarité [LonguetHiggins 81] est la contrainte fondamentale issue de la rigidité des objets pour les indices points dans le contexte du mouvement. Cette contrainte s'est traduite en contrainte épipolaire en stéréovision, ce qui permet de réduire l'espace de recherche des correspondants de bidimensionnel en monodimensionnel. Dans le mouvement dit *latéral* [Xu 87], la ligne épipolaire définie par la contrainte épipolaire se simplifie en ligne horizontale.

Nous avons déjà indiqué dans 5 que le mouvement en tant que transformation rigide peut être décomposée d'une façon unique en une rotation pure et une translation pure (cf 5) en fixant un point donné de l'axe de rotation. Nous avons parlé de l'utilisation des points de fuite pour estimer la rotation. Une fois que cette rotation a été estimée, il reste à déterminer la translation pure. Cette séparation de la translation de la rotation rend certaines relations plus explicites et plus intuitives. Faisons une remarque immédiate : *les points de fuite sont invariants sous une translation pure.* Ceci peut être illustré par la figure 7.1 dans 2D. Grâce à la rigidité des objets de la scène, les vecteurs de translation des objets rigides sont tous parallèles. Il existe donc un point de fuite de ces vecteurs parallèles dans le plan image, et en plus il doit être sur la ligne d'horizon, car les vecteurs de translation sont parallèles au sol (voir la figure 7.2). Ce point de fuite est aussi appelé *focus of expansion* (FOE) dans les litératures anglo-saxonnes concernant le flot optique [Prazdny 81] et [Ballard 82].

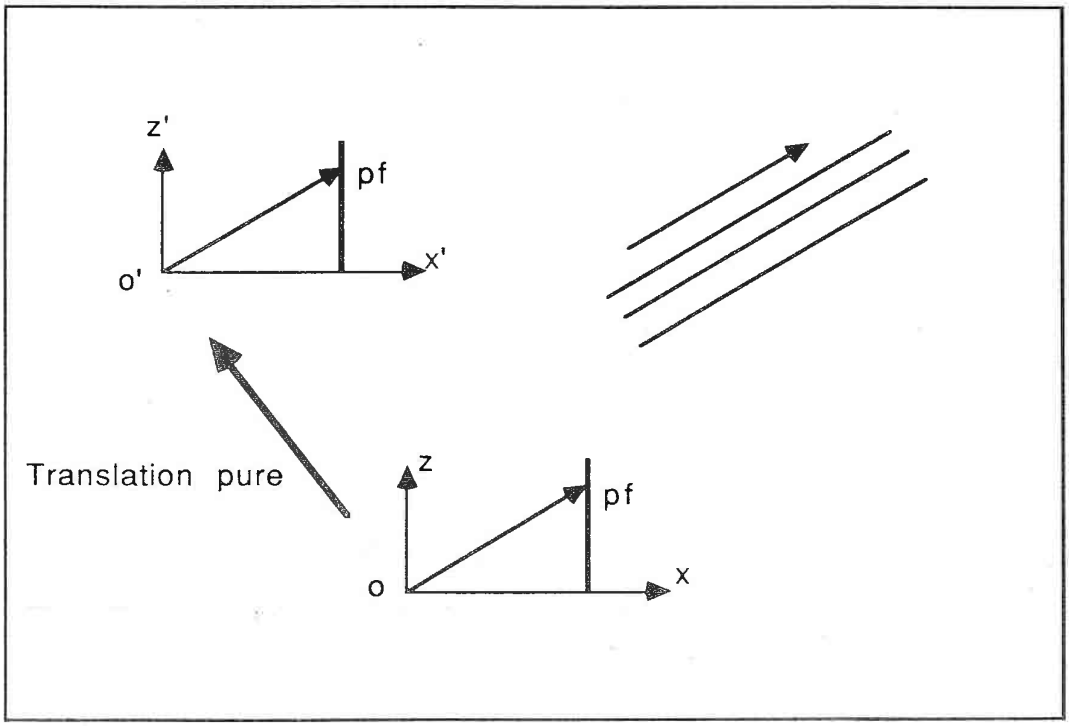


Figure 7.1. Le point de fuite est invariant par la translation.

Le vecteur défini par la perspective inverse de FOE détermine la direction de la translation (cf. 5). Il nous fournit aussi une contrainte intéressante de type épipolaire. Si l'on superpose deux images après la rotation, les points correspondants entre deux images sont alignés avec le FOE. Nous allons voir comment ces contraintes

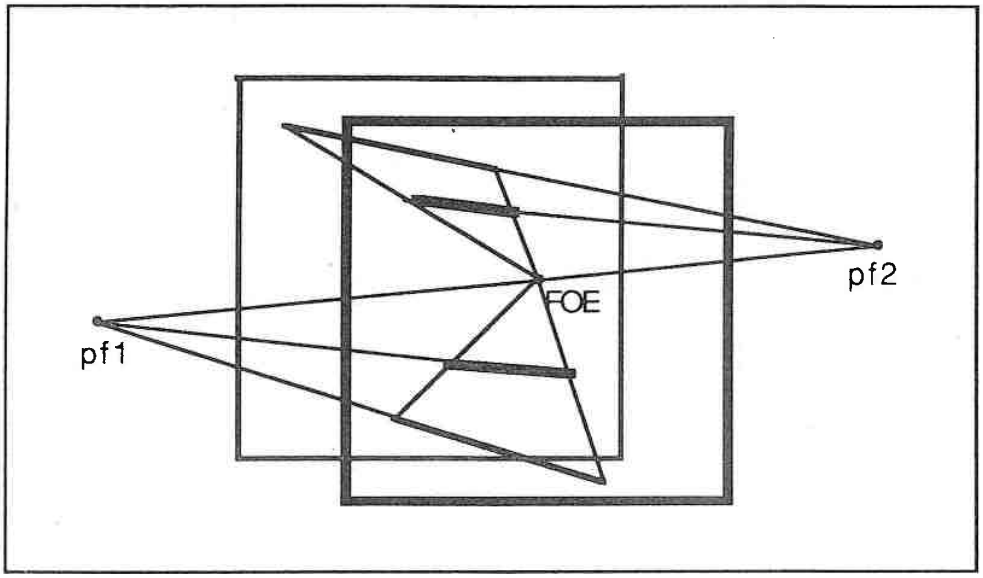


Figure 7.2. FŒ: le point de fuite des vecteurs de translation.

géométriques peuvent être intégrées dans l'appariement.

7.4 Stratégie d'appariement

Le schéma général de la stratégie d'appariement est une adaptation de la méthode prédiction-vérification (cf. la figure 7.3).

1. *Détermination de la rotation* en appariant au moins deux groupes directionnels.

2. *Prédiction* des appariements hypothétiques de segment que l'on appele *hypothéses SS*. Pour ce faire, on procède d'une façon hiérarchique grâce à l'organisation hiérarchique des groupes perceptuels, c'est-à-dire :

   - prédire d'abord des appariements hypothétiques de raies que l'on appelle *hypothèses RR* à l'intérieur de chaque appariement de groupes directionnels,
   - puis prédire des appariements hypothétiques de droites, appelé *hypothèses DD*, à l'intérieur de chaque hypothèse RR.

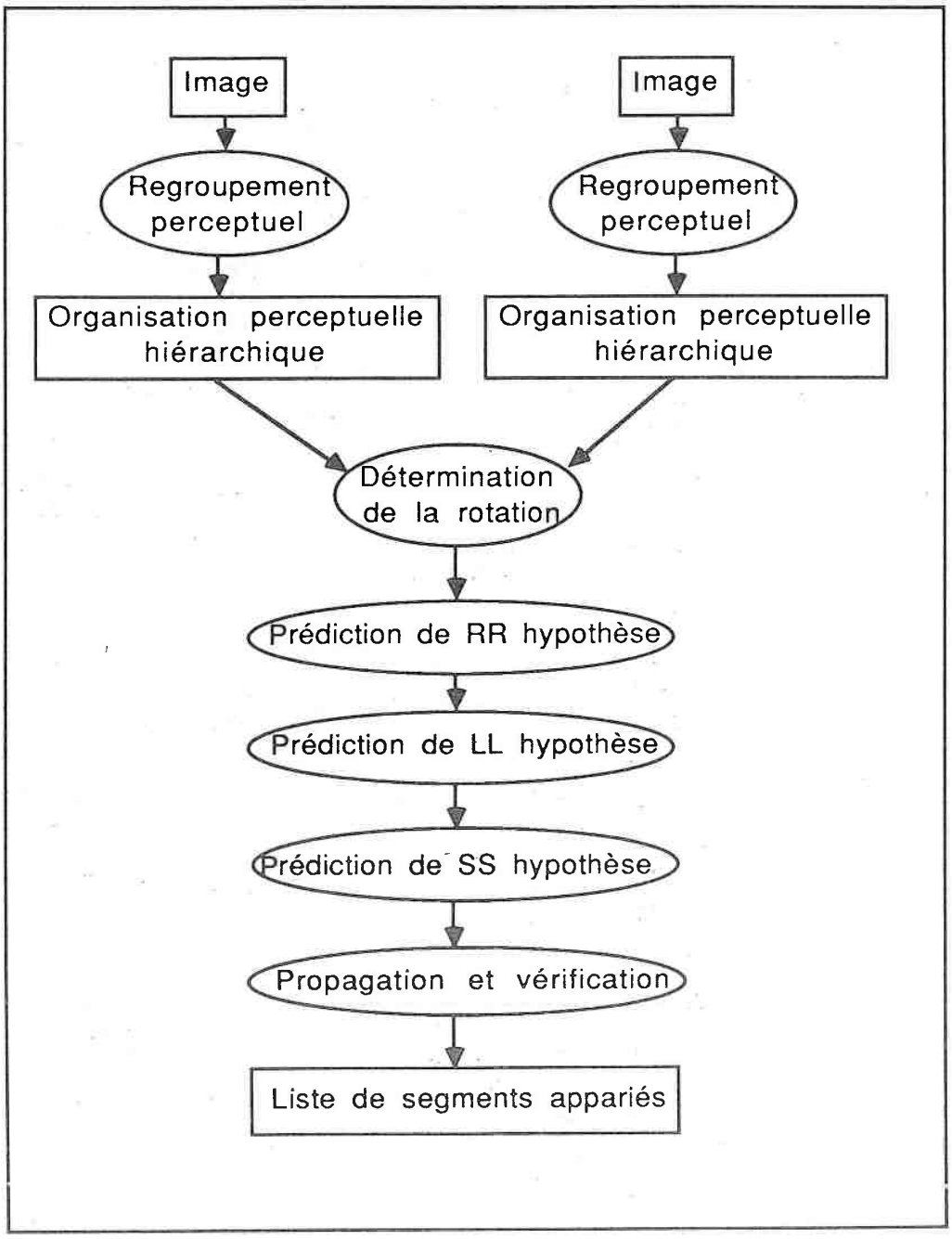


Figure 7.3. Le schéma général de l'appariement de deux images.

- Ensuite prédire des appariements hypothétiques de segments à l'intérieur de chaque hypothèse DD.

3. *Propagation et vérification.* Lorsque la prédiction est finie, on propage les hypothèses SS dans le voisinage de chacun de ses segments. La vérification consiste à selectionner les meilleurs appariements comme résultat final selon le critère que l'on va définir plus loin.

7.4.1 Détermination de la rotation

Prenons l'hypothèse de départ que le mouvement est petit, on en déduit immédiatement que l'appariement des groupes directionnels est évident : deux groupes directionnels se correspondent si l'angle qu'ils forment est le plus faible. On peut ainsi récupérer la rotation en utilisant l'algorithme de la section 5.

7.4.2 Prédiction

Dans cette étape, seuls les groupes perceptuels dont la direction est horizontale sont pris en considération, ceci pour deux raisons :

1. dans l'étape de prédiction, on n'a nul besoin de prendre en compte tous les groupes perceptuels. L'essentiel est d'avoir des hypothèses d'appariement réparties de façon homogène dans toute l'image.

2. les groupes horizontaux nous permettent d'utiliser la contrainte géométrique définie par la ligne d'horizon.

hypothèse RR

Nous utilisons deux heuristiques pour obtenir une RR hypothèse.

- L'*ordre des angles* associés à chaque raie. Cet ordre ne se conserve pas par mouvement. Pourtant si l'on tient compte du fait que le mouvement est petit et qu'une raie est un grand groupe perceptuel, cette heuristique reste raisonnable.

- Le *recouvrement* des projections parallèles à la ligne d'horizon des segments porteurs de raies. Cette heuristique provient de la contrainte géométrique FOE, nous ne l'utilisons ici qu'approximativement. En fait, quand le mouvement est petit, la norme du vecteur de la translation est petite, l'effet de perspective n'est donc pas apparent. FOE en tant que point de fuite se trouve presque

à l'infini, la région de recherche des correspondants est presque parallèle à la ligne d'horizon.

Remarquons que la position relative à la ligne d'horizon est implicitement inclu dans ces deux heuristiques. Cette prédiction peut se faire en $O(n)$, n étant le nombre de raies.

hypothèse DD

Une fois l'hypothèse RR obtenue, on cherche hypothèse DD à l'intérieur de chaque hypothèse RR. On superpose les centres de gravité des deux raies. On génère toutes les combinaisons possibles des couples de droites d'une hypothèse RR comme hypothèses DD. Cela donne donc une liste des listes de hypothèses DD. Un score est ensuite défini pour chaque liste de hypothèse DD comme la somme de distances de toutes les hypothèses DD. La distance pour une hypothèse DD est la distance entre ses deux droites, et la distance entre deux droites est la distance entre leurs segments porteurs.

La liste de hypothèses DD qui minimise cette distance est considérée comme la meilleure. La combinatoire ne peut être importante car le nombre de droites appartenant à une raie est faible.

hypothèse SS

Afin d'avoir les hypothèses SS à l'intérieur de chaque DD hypothèse, on utilise les coordonnées projectives décrites dans la section 2. Ceci se fait en deux étapes :

1. d'abord prédire un repère projectif,
2. ensuite déterminer les hypothèses SS dans le repère projectif prédit.

Prédiction d'un repère projectif Nous choisissons d'abord un repère projectif hypothétique entre deux droites à apparier. Comme on l'a vu dans la section 2, trois points alignés distincts définissent un tel repère pour une droite projective. Pour chaque DD hypothèse, nous avons déjà un point apparié, leur point de fuite. Si l'on fait hypothèse qu'un couple de segments est apparié, alors les deux extrémités d'un segment plus le point de fuite permettent donc de définir un repère projectif pour la droite portant ce segment (cf. figure 7.4). On peut ainsi générer toutes les combinaisons des couples de segments en tant que repères. Pour réduire ce nombre de

combinaisons et avoir un repère assez precis, on vérifie d'abord les caractéristiques locales des segments, la longueur et l'orientation du segment. Seuls les couples de segments réussissant le test sont sélectionné pour définir un repère projectif hypothétique.

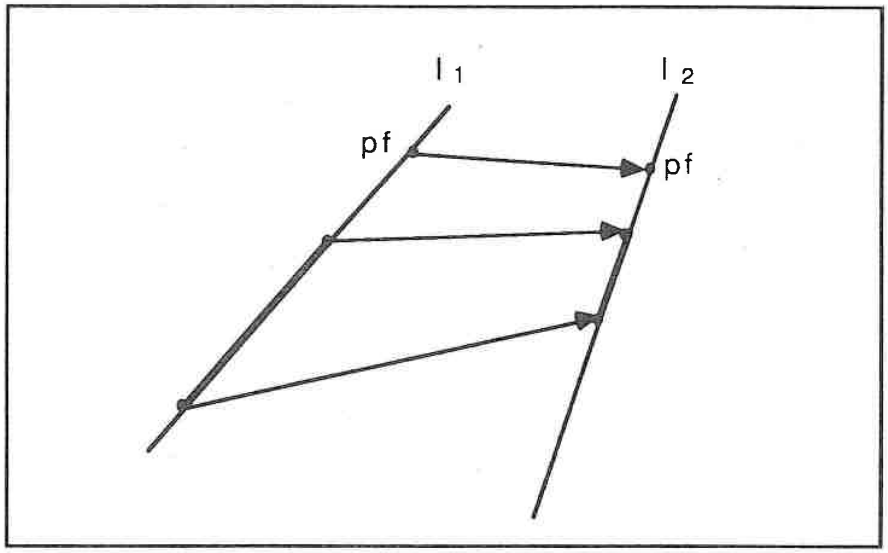


Figure 7.4. Etablir une correspondance entre des points de deux droites.

Dans un tel repère, chaque point de la droite est défini par sa coordonnée projective et chaque segment de la droite est représenté par un intervalle des coordonnées projectives. Un bon repère projectif doit permettre un bon recouvrement des intervalles des coordonnés projectives entre deux droites. Un score est donc défini pour chaque repère hypothétique comme le recouvrement des coordonnées projectives des segments faisant partie de deux droites. Le repère hypothétique dont le score est le plus grand est retenu comme le repère projectif de prédiction.

hypothèse SS Tous les segments de deux droites dont les coordonnées projectives se recouvrent suffisament dans ce repère retenu sont considérés comme hypothèses SS.

7.4.3 Propagation, Vérification et extension

Chaque segment a une liste de segments voisins. Ce voisinage a été créé à partir des segments dont les distances à ce segment sont inférieures à un seuil donné.

Chaque hypothèse SS issue de la prédiction est récursivement propagée à ses

voisins. Lorsque deux segments vérifient les caractéristiques locales qu'on a cité dans la section précédente, on dit que cet appariement est confirmé une fois. Cette propagation récursive s'arrête quand tous les segments de l'image sont visités au moins une fois. Chaque appariement de segments est associé à un score, qui est défini comme le nombre de confirmations qu'il a obtenu durant la propagation de toutes les hypothèses SS. Ensuite, les appariements de segments sont triés selon le score, on sélectionne les appariements dans l'ordre décroissant pour éviter les conflits. Les appariements qui n'ont jamais été confirmés plus de 2 fois sont éliminés définitivement.

7.5 Travaux liés à l'estimation du mouvement

Le problème de retrouver le mouvement à partir des appariements des indices visuels a été étudiés par de nombreux chercheurs dans [LonguetHiggins 81], [Liu 86], [Mitiche 86], [Faugeras 87] et [Liu 88]. Nous nous contentons de citer quelques résultats existants. On peut énoncer ce problème comme suit : *Etant donné un ensemble d'appariements des indices visuels, comment retrouver le mouvement de la caméra, c'est-à-dire une rotation et la direction de la translation qui permet de passer de la première position de la caméra à la deuxième ?*

Remarquons que l'on ne peut pas retrouver la translation en totalité (la direction et la norme), en raison de l'indétermination fondamentale sur l'échelle de la scène [Lustman 87]. Cette échelle peut en revanche être déterminée si l'on connait la dimension absolue d'un objet dans la scène.

- *Mise en correspondance des points* :

  en utilisant la contrainte de coplanarité, il est facile de montrer que 5 points appariés sont nécessaires pour obtenir un nombre fini de solutions. Malheureusement, les équations sont non linéaires et ne peuvent être résolues de manière explicite. Longuet-Higgins [LonguetHiggins 81] a proposé un algorithme linéaire des 8 points dans le cas d'une solution exacte, sans bruit. En présence de bruit, la méthode est très sensible. Faugeras *et al.* proposent une variation, plus robuste au bruit, de cette méthode dans [Faugeras 87].

- *Mise en correspondance des droites ou segments* :

  l'utilisation de droites comme indices visuels d'appariement a été récemment proposée par Mitiche *et al.* [Mitiche 86], Liu *et al.* [Liu 88] et Faugeras *et al.* [Faugeras 87], car les droites (ou les segments) sont des indices qui peuvent

être extraites sûrement d'images et sont par conséquent de bons candidats sur lesquels on peut baser l'appariement. L'appariement de deux droites dans deux images n'impose aucune contrainte sur le mouvement, trois images sont nécessaires pour déterminer le mouvement. Il faut apparier au moins 6 droites entre 3 images pour retrouver le mouvement par un algorithme non linéaire. Liu *et al.* ont linéarisé l'algorithme dans [Liu 88], mais le nombre d'appariement de segments (dans chacune des trois images) atteint alors 13, ce qui rend difficile l'application pratique de cet algorithme.

7.6 Résultats expérimentaux et discussions

La figure 7.5 montre deux images segmentées d'une scène d'intérieur. La figure 7.6 montre l'appariement des groupes directionnels. Les figures 7.7 et 7.8 montrent la détection du groupe raie et du groupe colinéaire dans deux images. Les figures 7.9, 7.10 et 7.11 montrent les hypothèses RR, les hypothèses DD et les hypothèses SS. La dernière figure 7.12 montre le résultat final de l'appariement.

Les idées à retenir dans cette approche, outre les points communs avec l'appariement du modèle, sont :

- l'utilisation des coordonnées projectives (elles sont, définies par birapport, les seules caractéristiques invariantes d'une projection perspective) que nous pensons important et qui n'ont pas été assez exploitées dans le passé ;

- l'organisation hiérarchique des groupes perceptuels permet d'obtenir une stratégie d'appariement de "coarse-to-fine".

Le point faible est que la méthode exige une scène relativement bien définie par des structures parallèles.

Le point à améliorer est que FOE n'a été utilisé qu'approximativement dans l'étape de prédiction. Une étude plus poussée doit consister à détecter FOE de façon plus rigoureuse. Comme FOE est aussi un point de fuite, il se prête bien à la transformée de Hough que l'on a utilisé dans le paragraph 5. Néanmoins en pratique la détection est plus difficile que celle des points de fuite, ceci pour trois raisons :

- un point de fuite (normal) est calculé à partir des mesures des orientations de segments. FOE doit être obtenu à partir des extrémités de segments. Les orientations de segments sont bien plus précises que leurs extrémités.

- FOE devrait être détecté après la rotation, l'incertitude causée par la rotation cumule encore l'imprécision des extrémités.
- Le mouvement est petit, la norme du vecteur de la translation est donc aussi petite, l'effet de perspective est peu apparent.

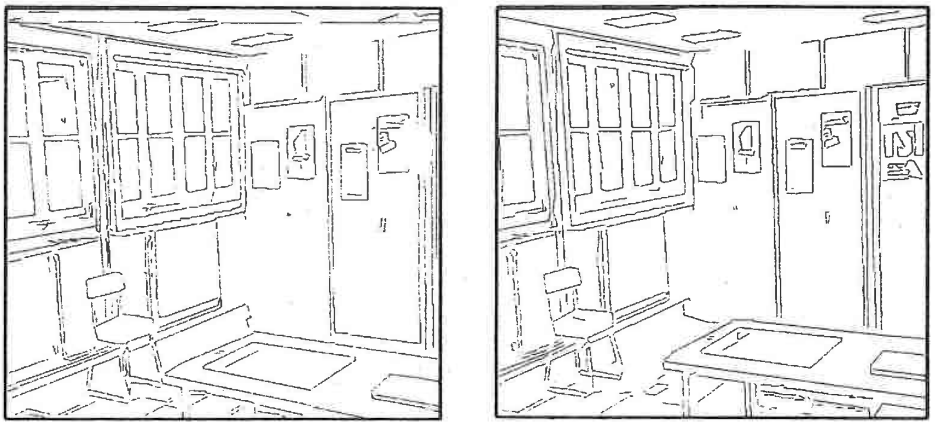


Figure 7.5. Deux images segmentées de départ.

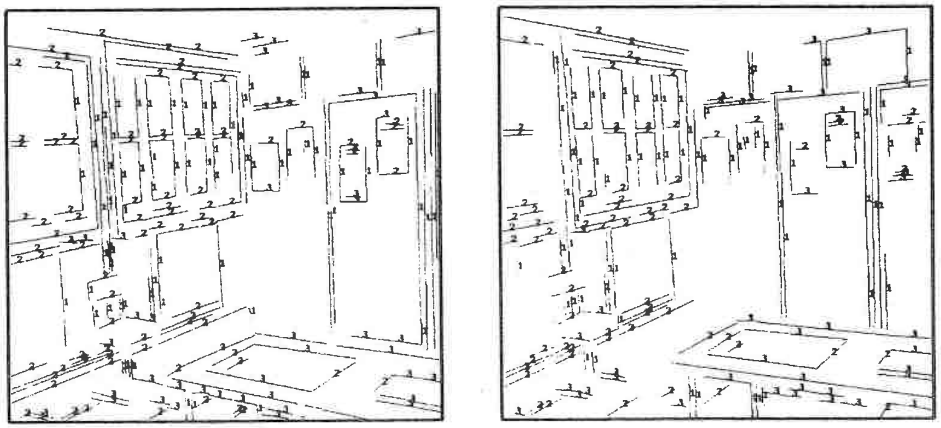


Figure 7.6. L'appariement de groupes directionnels.

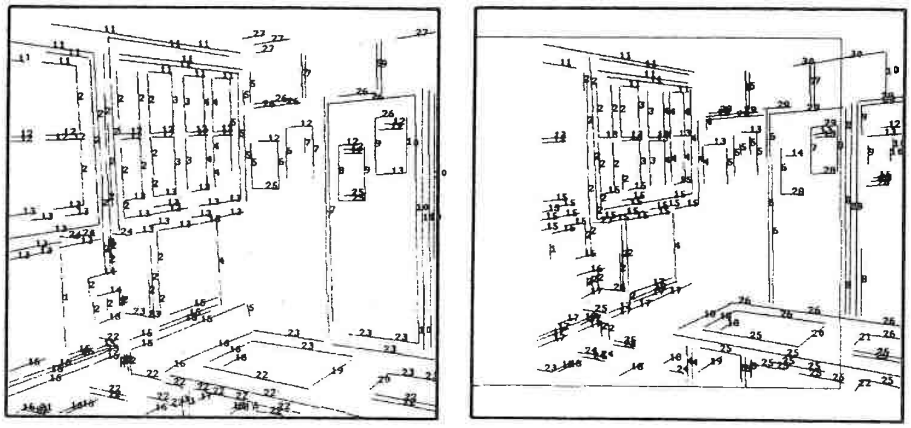


Figure 7.7. Groupes raies de deux images.

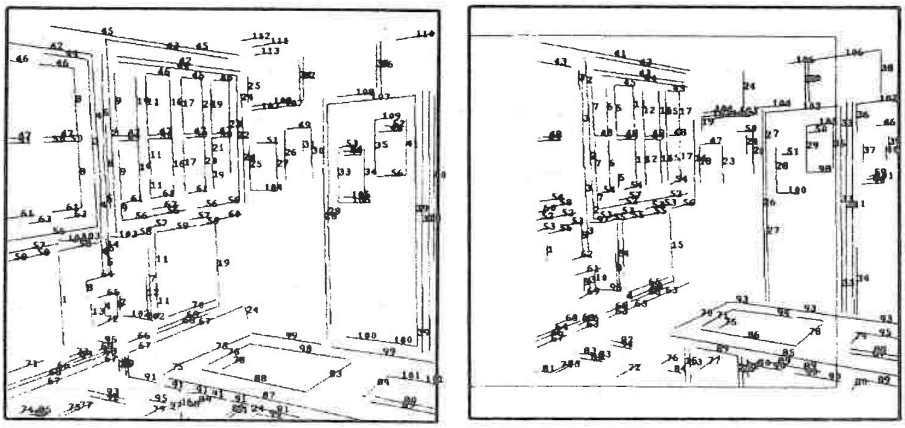


Figure 7.8. Groupes colinéaires de deux images.

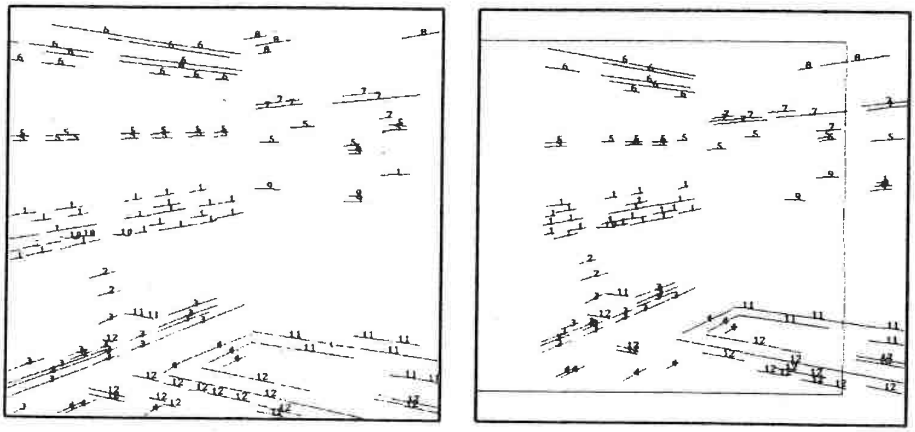


Figure 7.9. Hypothèses RR.

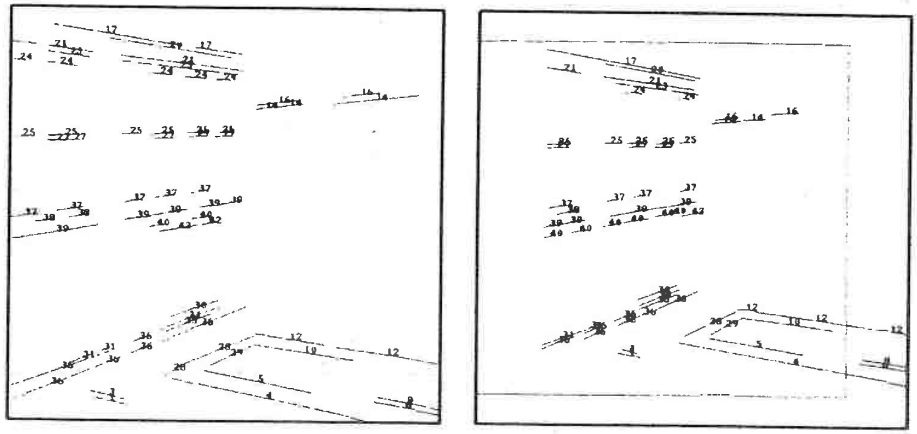


Figure 7.10. Hypothèses DD.

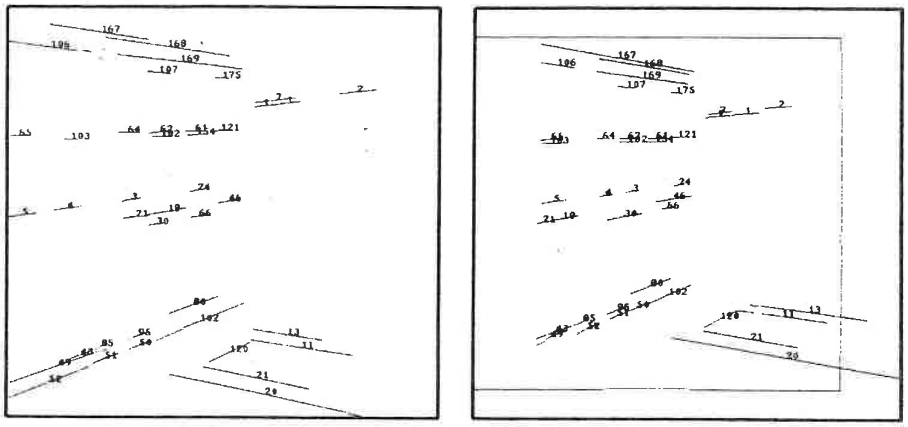


Figure 7.11. Hypothèses SS.

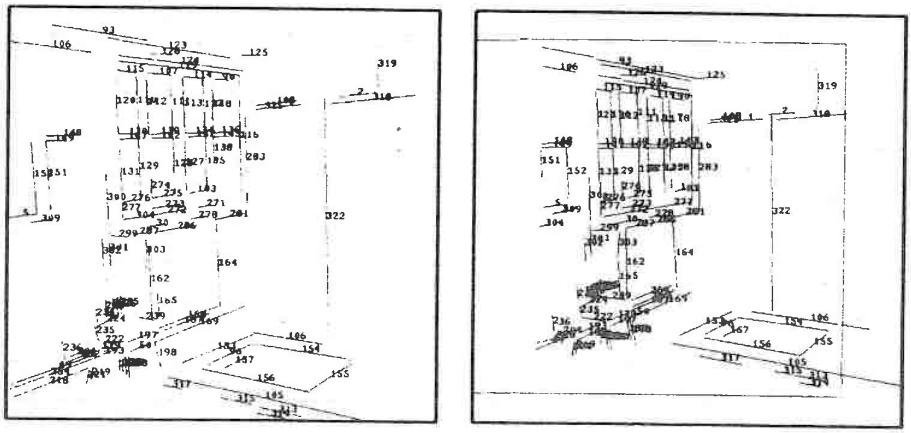


Figure 7.12. L'appariement des segments.

8
Conclusion

(À développer ...)

Nous avons présenté dans cette thèse une méthode pour la vision monoculaire qui consiste à apparier une image monoculaire avec le modèle *3D* de la scène ainsi qu'à apparier des images entre elles.

Nous avons d'abord développé un algorithme d'extraction des informations perspectives tels que les points de fuite et la ligne d'horizon. Ensuite nous avons abordé le problème de regroupement des indices visuels, et développé des algorithmes de regroupement sur les indices de type segment de droite. Les groupes utilisés sont les groupes directionnels, les groupes raies, les groupes colinéaires, les groupes convexes et les groupes coplanaires. Nous avons également développé un algorithme linéaire permettant la détermination précise du point de vue dans le modèle. Par la suite, nous avons réalisé un algorithme d'appariement image/modèle et un second destiné à l'appariement image/image. Les résultats expérimentaux sont aussi présentés.

Nous pensons que les points à retenir de notre approche sont que les vrais invariants perspectifs ont été intégrés en tant que tels dans le système. De plus le regroupement des indices visuels et la recherche des contraintes géométriques qui sont pour nous les deux aspects décisifs pour la vision passive ont été largement exploités dans le système.

Les améliorations du système actuel pourront porter d'une part sur le traitement plus systématique et plus rigoureux du problème d'incertitude des données en vision passive, qui a été souvent traité par la méthode classique, celle des moindres carrés. D'autre part, il est possible d'améliorer la modélisation de la scène : le modèle actuel est partiel et il serait bon d'utiliser soit le modèle issu de la fusion des perceptions partielles stéréoscopiques de la scène (cf. [Thirion 89] soit un modèle complet construit par un système CAO.

Le point faible de notre approche est que nous exigeons que la scène ait des

structures parallèles dominantes. La méthode est donc limité à ce type de scènes.

Néanmoins, à l'heure actuelle il est encore prématuré d'envisager un système de vision monoculaire général et indépendant du domaine d'application. En tout cas, malgré les efforts consentis durant ces vingt dernières années dans le domaine de la vision passive, vaste et passionant sujet, les résultats restent modestes en ce qui concerne un système de vision à la fois général et de haute-performance ; les succès obtenus sont pour la plupart les méthods *ad. hoc.* se limitant à des applications particulières.

La vision artificielle s'oriente actuellement vers la vision active dans laquelle on cherche à utiliser la possibilité de contrôler le mouvement de la caméra. Les techniques de ce type quoique récentes nous semblent très prometteuses et ouvrent une nouvelle perspective : de nombreux problèmes classiques de la vision passive souvent mal posés (par exemple, *shape from shading, contours ...*, flot optique) deviennent des problèmes bien posés dans le contexte de la vision active (cf. [Aloimonos 88]). Il existe déjà des travaux effectués dans ce domaine (cf. [Bolles 87,Liou 89]), mais beaucoup reste à faire tant sur le plan théorique que sur le plan pratique. La réalisation d'un système de vision à la fois général et fiable réside probablement dans l'intégration d'une bonne analyse monoculaire dans la vision active.

Bibliographie

[Aloimonos 88] J. Aloimonos, I. Weiss, et A. Bandyopadhyay. Active Vision. *International Journal of Computer Vision*, 1988.

[Asano 85] T. Asano, M. Edahiro, H. Imai, M. Iri, et K. Murota. *Practical Use of Bucketing Techniques in Computational Geometry.* in *Computational Geometry*, G.T. Toussaint Ed., Elsevier Science Publishers, North Holland, 1985.

[Ayache 86] N. Ayache et O.D. Faugeras. Hyper : A New Approach for the Recognition and Positioning of Two-Dimensional Objects. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 8(1):44–54, 1986.

[Ayache 87] N. Ayache et B. Faverjon. Efficient Registration of Stereo Images by Matching Graph Descriptions of Edge Segments. *International Journal of Computer Vision*, 1987.

[Ayache 88] N. Ayache. *Construction et fusion de représentations visuelles 3D.* Thèse d'Etat. Université de Paris-Sud, 1988.

[Baase 78] S. Baase. *Computer algorithms: Introduction to Design and Analysis.* Addison-Wesley Publishing Company, 1978.

[Bahnu 84] B. Bahnu et O.D. Faugeras. Shape Matching of Two-Dimensional Objects. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 6(2):137–155, 1984.

[Ballard 82] D. H. Ballard et C. M. Brown. *Computer Vision.* Prentice-Hall, 1982.

[Barnard 83] S. T. Barnard. Interpreting Perspective Images. *Artificial Intelligence*, 21(3):435–462, 1983.

[Barrow 81] H.G. Barrow et J.M. Tenenbaum. Interpreting Line Drawings as Three-Dimensional Surfaces. *Artificial Intelligence*, 17(1–3):75–116, 1981.

[Berger 77] M. Berger. *Géométrie 1. action de groupes, espaces affines et projectifs.* CEDIC/FERNAND NATHAN, Paris, 1977.

[Bolles 82] R.C. Bolles et R.A. Cain. Recognizing and Locating Partially Visible Objects : The LocalFeature-Focus Method. *The International Journal of Robotics Research*, 1(3):57–82, 1982.

[Bolles 87] R.C. Bolles, H. H. Baker, et D. Marimont. Epipolar-Plane Image Analysis : An Approach to Determining Structure from Motion. *International Journal of Computer Vision*, 1, 1987.

[Brady 84] M. Brady et A. Yuille. An Extremum Principle for Shape from Contour. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 6(3):288–301, 1984.

[Brooks 81] R.A. Brooks. Symbolic Reasoning Among 3-D Models and 2-D Images. *Artificial Intelligence*, 17(1–3):285–348, 1981.

[Canny 86] J. Canny. A Computational Approach to Edge Detection. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.

[Charniak 85] E. Charniak et D. McDermott. *Introduction to Artificial Intelligence.* Addison-Wesley Publishing Company, 1985.

[Coxeter 80] H. S. M. Coxeter. *Introduction to Geometry.* John Wiley and Sons, Inc., 1980.

[Deriche 87] R. Deriche. Using Canny's Criteria to derive an optimal edge detector recursively implemented. *The International Journal of Computer Vision*, 1, 1987.

[Dhome 88] M. Dhome, M. Richetin, J.T. Lapeste, et G. Rives. Determination of the attitude of 3D objects from a single perspective view. *Submitted to IEEE Transaction on Pattern Analysis and Machine Intelligence*, 1988.

[Duda 72] R.O. Duda et P.E. Hart. Use of the Hough transform to detect lines and curves in pictures. *Communication of ACM*, 15():11–15, 1972.

[Faugeras 81] O. Faugeras et M. Berthod. Improving Consistency and Reducing Ambiguity in Stochastic Labelling: an Optimization Approach. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, (4):412–423, 1981.

[Faugeras 87] O.D. Faugeras et G. Toscani. Camera calibration for 3d computer vision. *Proc. International Workshop on Machine Vision and Machine Intelligence*, page , Tokyo, 1987.

[Faugeras 88] O.D. Faugeras. *A Few Steps toward Artificial 3D Vision.* Rapports de Recherche no. 790, INRIA, Rocquencourt, France, 1988.

[Fischler 81] M.A. Fischler et R.C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[Foley 82] J.D. Foley et A. Van Dam. *Fundamentals of Interactive Computer Graphics.* Addison-Wesley, Reading Massacusetts, 1982.

[Giraudon 87] G. Giraudon. *Chaînage efficace de contour.* Rapports de Recherche no. 605, INRIA, Rocquencourt, France, 1987.

[Grimson 84] W. E. L. Grimson et Tomas Lozano-Perez. Model-based Recognition and Localization from Sparse Range or Tactile Data. *The International Journal of Robotics Research*, 3(3), 1984.

[Herman 86] M. Herman et T. Kanade. Incremental Construction of 3D Scenes from Multiple, Complex Images. *Artificial Intelligence*, 30, 1986.

[Hildreth 84] E.C. Hildreth. *The Measurement of Visual Motion.* MIT Press, Cambridge, MA, 1984.

[Hochberg 62] J.E. Hochberg et V. Brooks. Pictorial recognition as unlearned ability: A study of one child's performance. *American Journal of Psychology*, 75():624–628, 1962.

[Horaud 87] R. Horaud. New Methods for Matching 3D Objects with Single Perspective Views. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 9(3):401–412, 1987.

[Horn 81] B.K. Horn et B.G. Schunck. Determing Optical flow. *Artificial Intelligence*, 17(1–3):185–203, 1981.

[Horn 86] B.K. Horn. *Robot Vision.* McGraw Hill, New York, 1986.

[Huttenlocher 87] D.P. Huttenlocher et S. Ullman. Object recognition using alignment. *Proc. The First International Conference on Computer Vision*, pages 102–111, Londres, 1987.

[Ikeuchi 81] K. Ikeuchi et B.K.P Horn. Numerical Shape from Shading and Occluding Boundaries. *Artificial Intelligence*, 17(1–3):141–184, 1981.

[Ivor 88] A. M. Ivor. An analysis of lowe's model-based vision system. *The 4th ALVEY Vision conference*, pages 73–78, Manchester, UK, 1988.

[Knuth 75] D.E. Knuth. *Sorting and Searching.* in *The Art of Computer Programming*, Vol. 3, Addison Welsey, Reading, Ma., 1975.

[Liou 89] S. Liou et R.C. Jain. Motion Detection in Spatio-Temporal Space. *Computer Vision Graphics and Image Processing*, 45, 1989.

[Liu 86] Y. Liu et T.S. Huang. Estimation of rigid body motion using straight line correspondences: further results. *International Conference on Pattern Recognition*, pages 306–307, Paris, 1986.

[Liu 88] Y. Liu et T.S. Huang. A linear algorithm for motion estimation using straight line correspondences. *International Conference on Pattern Recognition*, pages 213–219, Rome, 1988.

[LonguetHiggins 81] H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293, 1981.

[Lowe 85a] D.G. Lowe. *Perceptual Organization and Visual Recognition.* Kluwer Academic Publishers, Norwell, Massachusetts, 1985.

[Lowe 85b] D.G. Lowe et T.O. Binford. The Recovery of Three-Dimensionnal Structure from Image Curves. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 7(3):320–326, 1985.

[Lowe 87] D.G. Lowe. Three-Dimensionnal Object Recognition from Single Two-Dimensional Images. *Artificial Intelligence*, 31, 1987.

[Lustman 87] F. Lustman. *Vision Stéréoscopique et Perception du Mouvement en Vision Artificielle.* Thèse de Doctorat. Université de Paris-Sud, 1987.

[Lux 85] A. Lux. *Algorithmique et controle en vision par ordinateur.* Thèse d'Etat, Institut National Polytechnique de Grenoble, 1985.

[Magee 84] M. J. Magee et J. K. Aggarwal. Determining Vanishing Points from Perspective Images. *Computer Vision Graphics and Image Processing*, 26():256–267, 1984.

[Marr 82] D. Marr. *Vision.* Freeman, San Francisco, 1982.

[Mitiche 86] A. Mitiche, S. Seida, et J.K. Aggarwal. Interpretation of structure and motion using line correspondences. *International Conference on Pattern Recognition*, pages 1110–1112, Paris, 1986.

[Mohr 88a] R. Mohr. Sur l'appariement modèle-perception. *Actes du 2^e Atelier Scientifique, traitement d'images : du pixel à l'interpre'tation*, pages XXIX–1–XXIX–17, Aussois, France, 1988.

[Mohr 88b] R. Mohr et G. Masini. Good old discret relaxation. *Proceedings of ECAI88*, pages 651–656, Munich, 1988.

[Mohr 88c] R. Mohr, L. Quan, et E. Thirion. Feature grouping : a way to deterministic matching. *Proc. of the Workshop on Syntactical*

and Structural Pattern Recognition, pages 213–227, Pont-a'-Mousson, France, 1988.

[Muller 84] Y. Muller. *Détection de surfaces par transforme'e de Hough en vision tridimensionnelle.* Thèse de troisième cycle en Informatique. Université de Nancy 1, 1984.

[Nakatani 81] H. Nakatani et T. Kitahashi. Determination of Vanishing Point in Outdoor Scene. *The Transactions of the IECE of Japan*, E64(5):357–358, 1981.

[Pentland 88] A.P. Pentland. Shape information from shading: a theory about human perception. *Proc. Second International Conference on Computer Vision*, pages 404–413, Tampa, Florida, 1988.

[Prazdny 81] K. Prazdny. Determining the Instantaneous Direction of Motion from Optical Flow Generated by a Curvilinearly Moving. *Computer Vision Graphics and Image Processing*, 17, 1981.

[Quan 87] L. Quan et R. Mohr. Location of a mobile robot by natural frames from a monocular image. *Proc. International Symposium on Technologies for Optoelectronis, Real Time Image Processing: Concepts and Technologies*, pages 114–120, Cannes, France, 1987.

[Quan 88a] Long Quan et Roger Mohr. Matching Perspective Images Using Geometric Constraints and Perceptual Grouping. *Proc. 2nd International Conference on Computer Vision*, pages 679–683, Tampa, Florida, 1988.

[Quan 88b] Long Quan, Roger Mohr, et Eric Thirion. Generating the initial hypothesis using perspective invariants for a 2D images and 3D model matching. *Proc. 9th International Conference on Pattern Recognition*, pages 872–874, Rome, Italy, 1988.

[Quan 88c] Long Quan, Roger Mohr, et Eric Thirion. Utilisation d"invariants par perspective en vision monoculaire. *Actes du 2^e Atelier Scientifique, traitement d'images : du pixel à l'interpre'tation*, pages XXVII–1–XXVII–17, Aussois, France, 1988.

[Quan 89a] Long Quan et Roger Mohr. Determining Perspective Structures Using Hierarchical Hough Transform. *Pattern Recognition Letters*, 9(4), 1989.

[Quan 89b] Long Quan et Roger Mohr. Using a linear viewpoint determination algorithm for a model based vision system. *To appear in Proc. 6nd SCAI*, page , Oulu, Finland, 1989.

[Quan 89c] Long Quan, Eric Thirion, Brigitte Wrobel-Daucourt, et Yolande Belaid. Vision for a Robot Moving in an Indoor Environment. *Advanced Information Processing in Automatic Control*, page , Nancy, France, 1989.

[Reynolds 87] G. Reynolds et J.R. Beveridge. Searching for geometric structure in images of natural scenes. *The DARPA Image Understanding Workshop*, pages 257–271, Los Angeles, California, 1987.

[Richetin 87] M. Richetin, M. Dhome, et J.T. Lapreste. Attitude spatiale des courbes vues en projection. *Actes du 6ème Congrès AFCET Reconnaissance des Formes et Intelligence Artificielle*, pages 671–685, Antibes, France, 1987.

[Shakunaga 88] T. Shakunaga et H. Kaneko. Shape from Angles under Perspective Projection. *Proc. 2nd International Conference on Computer Vision*, pages 671–678, Tampa, Florida, 1988.

[Shamos 85] M.I. Shamos et F.P. Preparata. *Computational Geometry - An Introduction.* Springer Verlag, 1985.

[Shapiro 81] L.J. Shapiro et R.H. Haralick. Structural Description and Inexact Matching. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, PAMI-3(5):504–519, 1981.

[Sklansky 80] J. Sklansky et V. Gonzalez. Fast Polygonal Approximation of Digitized Curves. *Pattern Recognition*, 1980.

[Skordas 87] T. Skordas et R. Horaud. *Stereo Correspondence through Feature Grouping and Maximal Cliques.* RR no. no.667-I-IMAG, LIFIA, Grenoble, France, 1987.

[Thirion 89] E. Thirion. *Interprétation et apprentissage géométrique en vision par ordinateur.* Thèse de doctorat de l'Institut National Polytechnique de Lorraine, 1989.

[Tisseron 83] C. Tisseron. *Géométries affines, projective et euclidienne.* Hermann, Paris, 1983.

[Toscani 87] G. Toscani. *Système de Calibration Optique et Perception du Mouvement en Vision.* Thèse de Doctorat. Université de Paris-Sud, 1987.

[Verri 87] A. Verri et T. Poggio. Against quantitative optical flow. *Proc. First International Conference on Computer Vision,* pages 171–180, London, 1987.

[Wertheimer 58] M. Wertheimer. *Principles of perceptual organization.* D. Beardslee et M. Wertheimer, Princeton, New York, 1958.

[WrobelDaucourt 88] B. Wrobel-Daucourt. *Perception de la distance par mise en correspondance de régions entre des images stéréoscopiques.* Thèse de doctorat de l'Institut National Polytechnique de Lorraine, 1988.

[Xu 87] G. Xu, S. Tsuji, et M. Asada. A Motion Stereo Method Based on Coarse-to-Fine Control Strategy. *IEEE Transaction on Pattern Analysis and Machine Intelligence,* PAMI-9(2):332–336, 1987.

Liste des Figures

Table des matières

INPL
INSTITUT NATIONAL POLYTECHNIQUE DE LORRAINE

AUTORISATION DE SOUTENANCE DE THESE
DU DOCTORAT DE L'INSTITUT NATIONAL POLYTECHNIQUE DE LORRAINE

VU LES RAPPORTS ETABLIS PAR :

Monsieur le Professeur CASTAN, IRIT/CERFIA Toulouse,
Monsieur le Professeur MARCHAND, CRIN Nancy.

Le Président de l'Institut National Polytechnique de Lorraine, autorise :

M onsieur Quan Long

à soutenir devant l'INSTITUT NATIONAL POLYTECHNIQUE DE LORRAINE, une thèse intitulée :

"Contribution de la vision moléculaire à la perception tridimentionnelle"

en vue de l'obtention du titre de :

DOCTEUR DE L'INSTITUT NATIONAL POLYTECHNIQUE DE LORRAINE

Spécialité : **"Informatique"**

Fait à Vandoeuvre le, **4 Octobre 1989**
Le Président de l'I.N.P.L.,

INSTITUT NATIONAL POLYTECHNIQUE DE LORRAINE — NANCY

M.GANTOIS

Par délégation
Le Secrétaire Général,
N. GAND

2, avenue de la Forêt de Haye - B.P. 3 - 54501 VANDŒUVRE CEDEX
Téléphone : 83. 57. 48. 48 - Télex : 961 715 F - Télécopie : 83. 57. 49. 55