

**THESE**

préparée par

**Yassine RUICHEK**

pour obtenir le grade de Docteur

de l'Université des Sciences et Technologies de Lille

Spécialité : Productique, Automatique et Informatique Industrielle

**STEREOVISION LINEAIRE**

**PAR RESEAU DE NEURONES DE HOPFIELD**

Application à la détection d'obstacles à l'avant des véhicules routiers

Soutenue le 04 Février 1997 devant le jury composé de :

MM.

P. VIDAL	Président	Professeur à l'USTL
I. MAGNIN	Rapporteur	Directeur de recherches à l'INSA. de Lyon
A. FAURE	Rapporteur	Professeur et Directeur du LACOS, Le Havre
M. BENJELLOUN	Examineur	Professeur à l'Université du Littoral
M. KARIM	Examineur	Maître de Conférences à l'Université de Marrakech
C. LOUIS	Examineur	Ingénieur à ALCATEL-ALSTHOM
D. HAMAD	Examineur	Maître de Conférences à l'USTL
J.-G POSTAIRE	Directeur de thèse	Professeur à l'USTL

Thèse préparée au Centre d'Automatique de Lille

## ***AVANT-PROPOS***

Le travail de ce mémoire a été effectué au sein de l'équipe "Image & Décision" du Centre d'Automatique de l'Université des Sciences et Technologies de Lille dans le cadre du programme européen Eurêka Prometheus sous la Direction de Monsieur le Professeur Jack-Gérard POSTAIRE.

Je tiens à remercier Monsieur Pierre VIDAL, Professeur à l'USTL, Directeur du Centre d'Automatique de Lille, pour m'avoir accueilli au sein de son laboratoire et fait l'honneur de présider le jury de cette thèse.

Je tiens à exprimer toute ma gratitude à Monsieur le Professeur Jack-Gérard POSTAIRE, mon Directeur de recherche, pour l'intérêt qu'il a porté à mes travaux, pour la confiance qu'il m'a accordée. Ses conseils, ses constants encouragements et sa grande disponibilité m'ont permis de mener à bien ce travail.

Je suis très honoré par la présence dans ce jury de Madame Isabelle MAGNIN, Directeur de recherches à L'INSA de Lyon, et Monsieur le Professeur Alain FAURE, Directeur du LACOS du Havre. Qu'ils trouvent ici mes sincères remerciements pour avoir accepté d'être les rapporteurs de ce travail.

Ma reconnaissance va à Monsieur Mohammed BENJELLOUN, Professeur à l'Université du Littoral, pour avoir accepté de juger mon mémoire et faire partie du jury de cette thèse.

Que Monsieur Christian LOUIS, Ingénieur à ALCATEL-ALSTHOM, trouve ici mes vifs remerciements pour l'intérêt qu'il a bien voulu porter à mon travail en participant à mon jury.

Je remercie vivement Monsieur Mohammed KARIM, Maître de Conférences à l'Université de Marrakech, d'avoir bien voulu se déplacer jusqu'à Lille pour participer au jury.

Monsieur Denis HAMAD, Maître de Conférences à l'USTL, a accepté de juger mon travail. Je lui suis très reconnaissant de faire partie des membres du jury.

Mes remerciements vont également à J.-L. BRUYELLE et J.-C. BURIE chercheurs au Centre d'Automatique de Lille pour l'aide qu'ils m'ont apportée au cours de la rédaction de ce mémoire.

J'adresse mes remerciements aux chercheurs de l'équipe "Image & Décision" ainsi qu'aux membres du Centre d'Automatique de Lille avec lesquels j'ai passé de merveilleux moments qui resteront un excellent souvenir de ma vie.

Je tiens à remercier le G.I.E. P.S.A. Peugeot Citroën et Renault pour son aide précieuse, sa contribution financière et sa confiance au cours de ce travail qui s'inscrit dans le cadre du projet Eurêka Prometheus. Je remercie également l'I.N.R.E.T.S. et la Région Nord-Pas de Calais pour leur aide et leur soutien.

**Table  
des matières**

<b>CHAPITRE I : INTRODUCTION GENERALE.....</b>	<b>1</b>
I-1 LES ACCIDENTS DE LA ROUTE.....	2
I-2 AMELIORATION DE LA SECURITE ROUTIERE .....	4
I-3 LE PROGRAMME EUROPEEN PROMETHEUS .....	5
I-3.1 Les objectifs du programme Prometheus .....	5
I-3.2 Contribution de groupe ProArt France.....	6
I-3.2.1 Le démonstrateur ProLab1 .....	7
I-3.2.2 Le démonstrateur ProLab2 .....	8
I-4 LA DETECTION D'OBSTACLES SUR ROUTE .....	9
I-4.1 Approche active.....	9
I-4.1.1 Les radars .....	9
I-4.1.2 Les sonars.....	10
I-4.1.3 Les lidars .....	10
I-4.2 Approche passive.....	11
I-5 NOUVELLE APPROCHE .....	12
I-6 LE CALCUL NEUROMIMETIQUE.....	13
I-6.1 Pourquoi des réseaux neuronaux artificiels ? .....	14
I-6.2 Historique.....	15
I-6.3 Pour quelles applications ? .....	18
I-6 PLAN DU MEMOIRE.....	20
<b>CHAPITRE II : VISION STEREOSCOPIQUE .....</b>	<b>22</b>
II-1 INTRODUCTION .....	23
II-2 PERCEPTION DU RELIEF PAR VISION ARTIFICIELLE.....	24
II-2.1 Vision monoculaire.....	24
II-2.2 Vision dynamique.....	24
II-2.2.1 Méthodes structurelles .....	25
II-2.2.2 Analyse du flux optique.....	26
II-2.2.3 Méthodes orientées mouvement.....	28
II-2.3 Méthodes K2D .....	29
II-2.4 Conclusion .....	29
II-3 STEREOVISION.....	29
II-3.1 Définitions.....	29
II-3.2 Applications de la stéréovision .....	30
II-4 STEREOVISION BINOCULAIRE .....	30
II-4.1 Décomposition du problème .....	30
II-4.2 Principes généraux .....	31
II-4.3 Aspects fondamentaux.....	31
II-4.3.1 Fonction d'appariement.....	31
II-4.3.2 Complexité du problème.....	33
II-4.3.3 Dilemme précision-appariement .....	33
II-5 SYSTEME DE PRISE DE VUE.....	33
II-5.1 Géométrie des capteurs.....	33
II-5.2 Droites épipolaires.....	34
II-5.3 Configuration géométrique idéale .....	35
II-5.4 Paramètres intrinsèques .....	36
II-5.5 Triangulation géométrique .....	36
II-5.6 Reconstruction 3D .....	38
II-6 LES PRIMITIVES .....	38
II-6.1 Les propriétés requises .....	39
II-6.2 Types de primitives.....	39
II-6.2.1 Les points d'intérêt.....	39
II-6.2.2 Les primitives de type contour.....	41
II-6.2.2.1 Points de contour .....	41
II-6.2.2.2 Modélisation des contours .....	41
II-6.2.3 Les régions.....	42
II-6.2.4 Les contours actifs .....	43
II-6.3 Conclusion .....	44
II-7 CRITERES DE RESSEMBLANCE ET CONTRAINTES.....	44

II-7.1 Critères de ressemblance .....	45
II-7.2 Contraintes stéréoscopiques .....	46
II-7.2.1 La contrainte épipolaire .....	46
II-7.2.2 La contrainte d'orientation .....	47
II-7.2.3 La contrainte d'ordre .....	48
II-7.2.4 La contrainte de continuité de la disparité .....	49
II-7.2.5 La contrainte d'unicité .....	50
II-7.2.6 La contrainte de continuité figurale .....	51
II-7.3 Conclusion .....	51
II-8 MISE EN CORRESPONDANCE .....	52
II-8.1 Méthodes corrélatives .....	52
II-8.2 Méthodes hiérarchiques .....	53
II-8.3 Méthodes basées sur la programmation dynamique .....	54
II-8.4 Méthodes d'étiquetage par relaxation .....	55
II-8.5 Appariement par isomorphisme de graphes .....	57
II-8.6 Appariement par prédiction et vérification d'hypothèses .....	58
II-8.7 Méthodes basées sur les réseaux de neurones .....	59
II-8.8 Approche par optimisation utilisant le recuit simulé .....	62
II-8.9 Autres méthodes .....	63
II-8.10 Conclusion .....	64
II-9 STEREOVISION TRINOCULAIRE .....	64
II-10 CONCLUSION .....	66
<b>CHAPITRE III : CONCEPTION D'UN STEREOSCOPE LINEAIRE .....</b>	<b>68</b>
III-1 INTRODUCTION .....	69
III-2 CAMERAS LINEAIRES .....	69
III-2.1 Constituants d'une caméra linéaire .....	69
III-2.2 Caractéristiques des capteurs linéaires .....	70
III-2.3 Exemples d'application des caméras linéaires .....	71
III-2.3.1 Reconstitution de la dimension verticale .....	71
III-2.3.2 Stéréovision .....	72
III-2.3.3 Métrologie .....	73
III-2.4 Champ de visée d'une caméra linéaire .....	73
III-3 CONCEPTION D'UN STEREOSCOPE LINEAIRE .....	74
III-3.1 Hypothèses sur les caméras .....	74
III-3.2 Calibrage du stéréoscope linéaire .....	74
III-3.2.1 Monture orientable .....	75
III-3.2.2 Mire de calibrage .....	75
III-4 INTEGRATION DU STEREOSCOPE SUR UN VEHICULE .....	77
III-4.1 Monture commune aux deux caméras .....	77
III-4.2 Installation sur le véhicule .....	78
III-4.3 Reconstruction de la profondeur et de la position .....	81
III-5 ENREGISTREMENT DE SEQUENCES D'IMAGES .....	84
III-5.1 Technique d'enregistrement .....	84
III-5.2 Exemples de séquences enregistrées .....	85
III-5.2.1 Exemple 1 : Piéton évoluant devant le stéréoscope .....	86
III-5.2.2 Exemple 2 : Route dégagée et passage d'une sortie d'autoroute .....	87
III-5.2.3 Exemple 3 : Approche d'un véhicule .....	87
III-6 CONCLUSION .....	89
<b>CHAPITRE IV : EXTRACTION DES PRIMITIVES .....</b>	<b>90</b>
IV-1 INTRODUCTION .....	91
IV-2 CHOIX DU TYPE DE PRIMITIVE .....	91
IV-3 CHOIX DU DETECTEUR DE CONTOURS .....	93
IV-3.1 Contraintes à respecter par le détecteur de contours .....	93
IV-3.2 Pretraitement de l'image .....	94
IV-3.3 Conclusion .....	95
IV-4 DETECTION DES CONTOURS .....	96
IV-4.1 Les masques de convolution .....	96

IV-4.1.1 Définitions .....	96
IV-4.1.2 Opération de lissage .....	97
IV-4.1.3 Opération de détection .....	98
IV-4.1.4 Conclusion sur les masques de convolution .....	99
<i>IV-4.2 Les détecteurs de contours optimaux</i> .....	99
IV-4.2.1 Approche de Canny .....	100
IV-4.2.1.1 Formulation mathématique des critères de performance .....	100
IV-4.2.1.2 Optimisation du détecteur de contours .....	102
IV-4.2.2 Approche de Deriche .....	104
IV-4.2.2.1 Détecteur optimal de contours .....	104
IV-4.2.2.2 Opérateur de lissage .....	105
IV-4.2.2.3 Influence du paramètre $\alpha$ .....	106
IV-4.2.3 Conclusion .....	106
<i>IV-4.3 Implantation de la détection des contours</i> .....	107
IV-4.3.1 Implantation récursive d'un filtre .....	107
IV-4.3.2 Lissage récursif .....	109
IV-4.3.3 Calcul récursif du gradient .....	111
<i>IV-4.4 Choix du paramètre <math>\alpha</math></i> .....	114
<i>IV-4.5 Conclusion</i> .....	114
<b>IV-5 EXTRACTION DES CONTOURS</b> .....	115
<i>IV-5.1 Elimination des extremums générés par le bruit</i> .....	115
<i>IV-5.2 Sélection des extremums significatifs par intervalle</i> .....	116
<i>IV-5.3 Attributs des primitives</i> .....	118
<i>IV-5.4 Conclusion</i> .....	118
<b>IV-6 RESULTATS D'EXTRACTION DES CONTOURS</b> .....	119
<i>IV-6.1 Piéton évoluant devant le stéréoscope</i> .....	119
<i>IV-6.2 Route dégagée et passage d'une sortie d'autoroute</i> .....	120
<b>IV-7 CONCLUSION</b> .....	120
<b>CHAPITRE V : APPARIEMENT STEREOSCOPIQUE SEQUENTIEL DES IMAGES LINEAIRES ...</b>	<b>122</b>
<b>V-1 INTRODUCTION</b> .....	123
<b>V-2 DESCRIPTION DE L'APPROCHE</b> .....	123
<i>V-2.1 Principes de base</i> .....	123
<i>V-2.2 Définitions</i> .....	123
<i>V-2.3 Représentation graphique</i> .....	124
<i>V-2.4 Contraintes de mise en correspondance</i> .....	125
V-2.4.1 Contrainte d'unicité .....	125
V-2.4.2 Contrainte d'ordre .....	125
V-2.4.3 Contrainte de position .....	125
V-2.4.4 Contrainte de signe du gradient .....	126
<b>V-3 ALGORITHME DE MISE EN CORRESPONDANCE</b> .....	127
<i>V-3.1 Critère de ressemblance</i> .....	127
V-3.1.1 Normalisation des intervalles .....	128
V-3.1.1.1 Recentrage des intervalles .....	128
V-3.1.1.2 Rééchantillonnage spatial .....	128
V-3.1.2 Mesure de similarité .....	129
<i>V-3.2 Module de mise en correspondance élémentaire</i> .....	129
V-3.2.1 Tests d'appariement élémentaire .....	129
V-3.2.2 Règles de décision .....	131
<i>V-3.3 Filtrage des paires incompatibles</i> .....	132
V-3.3.1 1 <sup>ère</sup> étape : vérification de la contrainte de position .....	132
V-3.3.2 2 <sup>ème</sup> étape : vérification de la contrainte de signe du gradient .....	133
V-3.3.3 Conclusion sur de filtrage .....	133
<i>V-3.4 Initialisation et arrêt de l'algorithme</i> .....	134
V-3.4.1 Condition initiale .....	134
V-3.4.2 Condition d'arrêt .....	134
<i>V-3.5 Résultats de la mise en correspondance séquentielle</i> .....	134
V-3.5.1 Piéton évoluant devant le stéréoscope .....	134
V-3.5.2 Route dégagée et passage d'une sortie d'autoroute .....	135
<b>V-4 AMELIORATIONS DE L'ALGORITHME</b> .....	136
<i>V-4.1 Mise en correspondance bidirectionnelle</i> .....	136

V-4.1.1 Principe.....	136
V-4.1.2 Résultats.....	137
V-4.1.2.1 Piéton évoluant devant le stéréoscope.....	137
V-4.1.2.2 Route dégagée et passage d'une sortie d'autoroute.....	138
V-4.1.3 Conclusion sur la méthode bidirectionnelle.....	138
V-4.2 <i>Mise en correspondance hiérarchique</i> .....	139
V-4.2.1 Principe.....	139
V-4.2.2 Sélection des points de contour de référence.....	140
V-4.2.3 Résultats.....	141
V-4.2.3.1 Piéton évoluant devant le stéréoscope.....	141
V-4.2.3.2 Route dégagée et passage d'une sortie d'autoroute.....	142
V-4.2.4 Conclusion sur l'algorithme hiérarchique.....	142
V-5 TRIANGULATION GEOMETRIQUE.....	143
V-6 CONCLUSION.....	144
<b>CHAPITRE VI : MODELE DE HOPFIELD ET OPTIMISATION COMBINATOIRE.....</b>	<b>146</b>
VI-1 INTRODUCTION.....	147
IV.2 RESEAUX NEURONAUX.....	147
VI-2.1 <i>Le neurone artificiel</i> .....	148
VI-2.2 <i>Structure d'un réseau connexionniste</i> .....	152
VI-2.2.1 Les réseaux à couches.....	152
VI-2.2.2 Les réseaux entièrement connectés.....	152
VI-2.3 <i>Dynamique d'un réseau connexionniste</i> .....	153
VI-2.4 <i>Apprentissage d'un réseau connexionniste</i> .....	154
VI-2.5 <i>Conclusion</i> .....	154
VI-3 LE MODELE DE HOPFIELD.....	154
VI-3.1 <i>Structure du réseau</i> .....	154
VI-3.2 <i>Dynamique du réseau</i> .....	155
VI-3.2.1 Mise à jour en mode synchrone.....	156
VI-3.2.2 Mise à jour en mode asynchrone.....	156
VI-3.3 <i>Apprentissage du réseau</i> .....	157
VI-3.4 <i>Stabilité des états mémorisés</i> .....	160
VI-3.5 <i>Limites des réseaux de Hopfield</i> .....	161
VI-3.6 <i>Conclusion</i> .....	162
VI-4 RESEAU DE HOPFIELD ET OPTIMISATION COMBINATOIRE.....	162
VI-4.1 <i>Structure générale du réseau</i> .....	162
VI-4.2 <i>Fonction d'énergie du réseau</i> .....	163
VI-4.3 <i>Formulation duale</i> .....	165
VI-4.4 <i>Applications</i> .....	166
VI-4.5 <i>Conclusion</i> .....	167
VI-5 RESEAUX DE HOPFIELD A RECUIT SIMULE.....	167
VI-5.1 <i>Le recuit simulé</i> .....	167
VI-5.2 <i>Utilisation dans les réseaux de Hopfield</i> .....	170
VI-5.3 <i>Conclusion</i> .....	172
VI-6 RESEAUX ANALOGIQUES DE HOPFIELD.....	172
VI-6.1 <i>Dynamique d'un réseau analogique</i> .....	173
VI-6.2 <i>Influence du terme de bruit sur l'évolution de la fonction d'énergie H</i> .....	178
VI-6.3 <i>Conclusion</i> .....	180
VI-7 CONCLUSION.....	180
<b>CHAPITRE VII : APPROCHE GLOBALE POUR L'APPARIEMENT STEREOSCOPIQUE DES IMAGES LINEAIRES PAR RESEAU DE NEURONES DE HOPFIELD.....</b>	<b>182</b>
VII-1 INTRODUCTION.....	183
VII-2 CODAGE DU PROBLEME.....	183
VII-3.1 <i>Les contraintes locales</i> .....	185
VII-3.1.1 La contrainte de position.....	186
VII-3.1.2 La contrainte de signe du gradient.....	186
VII-3.2 <i>Les contraintes globales</i> .....	187
VII-3.2.1 La contrainte d'unicité.....	188
VII-3.2.2 La contrainte d'ordre.....	188

VII-3.2.3 La contrainte de continuité de la disparité.....	190
VII-3.3 Conclusion.....	190
VII-4.1 Fonction de coût représentant la contrainte de position.....	190
VII-4.2 Fonction de coût représentant la contrainte de signe du gradient.....	191
VII-4.3 Fonction de coût représentant la contrainte d'unicité.....	192
VII-4.4 Fonction de coût représentant la contrainte d'ordre.....	193
VII-4.5 Fonction de coût représentant la contrainte de continuité.....	195
VII-4.6 Fonction de coût total.....	196
VII-5.1 Construction du réseau.....	198
VII-5.2 Fonctionnement du réseau.....	200
VII-5.2.1 Mise à jour des états des neurones du réseau.....	200
VII-5.2.2 Initialisation du réseau.....	202
VII-5.2.3 Stabilité du réseau.....	203
VII-5.3 Analyse de l'état final du réseau.....	203
VII-5.4 Procédure de mise en correspondance.....	203
VII-5.5 Conclusion.....	205
VII-6.1 Choix des paramètres.....	205
VII-6.1.1 Choix du paramètre $\theta$ .....	206
VII-6.1.2 Choix des coefficients intervenant dans la fonction de coût.....	206
VII-6.1.3 Choix des paramètres $\lambda$ et $\tau$ .....	207
VII-6.1.4 Choix du paramètre $h$ .....	208
VII-6.2 Traitement des séquences stéréoscopiques.....	208
VII-6.2.1 Piéton évoluant devant le stéréoscope.....	208
VII-6.2.2 Route dégagée et passage d'une sortie d'autoroute.....	210
VII-6.2.3 Approche d'un véhicule.....	211
VII-6.3 Traitement informatique des séquences stéréoscopiques.....	218
VII-6.4 Conclusion.....	219
VII-7.1 Calcul des paramètres du réseau.....	220
VII-7.2 Procédure de mise en correspondance.....	222
VII-7.3 Résultats de mise en correspondance.....	222
VII-7.4 Conclusion.....	222
VII-8.1 Calcul des paramètres des deux réseaux.....	224
VII-8.2 Procédure de mise en correspondance.....	225
VII-8.3 Résultats de mise en correspondance.....	226
VII-8.4 Conclusion.....	227
<b>CHAPITRE VIII : CONCLUSION GENERALE.....</b>	<b>230</b>
<b>ANNEXE 1.....</b>	<b>235</b>
<b>ANNEXE 2.....</b>	<b>238</b>
<b>PUBLICATION DANS LE CADRE DE LA THESE.....</b>	<b>242</b>
<b>BIBLIOGRAPHIE.....</b>	<b>244</b>

# **Chapitre I**

## **Introduction générale**

## I-1 LES ACCIDENTS DE LA ROUTE

Il est souvent difficile d'attribuer une cause unique à un accident, qui est, généralement, la conséquence d'une succession de circonstances dont aucune, si elle était arrivée seule, n'aurait eu de conséquences fâcheuses. De plus, les données recueillies après coup par la police ou par les constats d'accidents sont souvent insuffisantes pour tirer des conclusions sur les causes de l'accident rapporté.

Les enquêtes établies par l'INRETS<sup>1</sup> et le programme REAGIR<sup>2</sup> ont montré que les causes d'accidents les plus fréquentes sont liées aux comportements (vitesse, manoeuvre dangereuse, infraction caractérisée, mauvaise utilisation du véhicule, etc.) et aux aptitudes (alcool, fatigue, inattention, inaptitude à la conduite, problèmes physiques ou psychologiques, etc.) des conducteurs. Cependant, d'autres facteurs, liés à l'infrastructure routière (signalisation, éclairage, état des chaussées, etc.), à la conception des véhicules (résistance aux chocs, problèmes de visibilité, conception générale) et leur entretien (pneus, éclairage, freins, direction, etc.), peuvent intervenir dans les accidents.

A partir de l'analyse des causes d'accidents, de nombreux moyens ont été envisagés pour y remédier, en agissant sur les trois sujets impliqués : route, véhicule et conducteur.

Afin de modifier les zones à risques, l'action sur le réseau routier porte sur :

- La réduction de l'influence des conditions météorologiques en améliorant l'adhésion des pneumatiques, particulièrement en cas de pluie, par la mise au point de revêtements spéciaux.
- La séparation des automobiles et des autres moyens de transports (deux roues, machines agricoles et de chantier). L'absence de croisements, la douceur des virages et l'élimination des pentes importantes ont permis une facilité de circulation sur autoroute.
- L'aménagement des zones à risques où les conditions de circulation changent brutalement sur un trajet, surprenant les conducteurs imprudents ou peu habitués. Lorsqu'un tel aménagement n'est pas possible, une diminution du nombre

---

<sup>1</sup>Institut National de Recherche sur les Transports et leur Sécurité.

<sup>2</sup>Réagir par des Enquêtes sur les Accidents Graves et les Initiatives pour y Remédier.

d'accidents est généralement obtenues par l'amélioration de la signalisation et par la limitation de la vitesse aux abords de la zone dangereuse.

L'amélioration de l'infrastructure routière présente toutefois un effet pervers : les conducteurs ont tendance à adapter leur comportement à la perception du risque. Ainsi, lorsque la circulation est facilitée par un aménagement nouveau, un sentiment de sécurité trompeur pousse les conducteurs à une prise de risque accrue.

L'action sur les véhicules concerne à la fois leur conception et leur entretien. Quand il s'agit de leur entretien, cette action porte essentiellement sur l'éducation et la formation des conducteurs qui doivent être conscients du danger présenté par un véhicule en mauvais état. Quand à leur conception, les véhicules ont été dotés de certains systèmes de sécurité, permettant d'assister activement les conducteurs en cas d'urgence (éviter un accident par un freinage d'urgence, par exemple) ou passivement en cas d'accident (limiter les dégâts causés sur les passagers et sur les autres usagers éventuellement impliqués). Il ne faut pas oublier que ces dispositifs présentent certains effets pervers. D'une part, le conducteur a naturellement tendance à modifier son comportement en fonction du supplément de sécurité attendu, dans le sens d'une plus grande prise de risque : "Je ne risque rien à rouler trop vite, j'ai un ABS<sup>1</sup>". D'autre part, une panne du système d'assistance, tel que l'ABS, peut avoir des conséquences catastrophiques. De plus, des études de L'INRETS ont montré que la sécurité passive qui a pour but d'accroître la sécurité des passagers (les barres de renfort latérales situées dans les portières, les systèmes évitant l'enfoncement de la colonne de direction, les dispositifs anti-incendies, etc.) a souvent pour effet d'augmenter le poids du véhicule, sa rigidité, et par conséquent son agressivité envers les autres usagers lors d'un accident.

Enfin, l'action la plus importante est, sans doute, celle qui porte sur les conducteurs. Les efforts destinés à améliorer la sécurité routière portent donc en grande partie sur les usagers par l'éducation et la formation à ne plus prendre de risque, mais aussi par la répression pour modifier leur comportement à risque, tels que les excès de vitesse ou, plus généralement, le non respect du code de la route.

---

<sup>1</sup>Anti Bloking System, développé par Bosch.

## I-2 AMELIORATION DE LA SECURITE ROUTIERE

Depuis une dizaine d'année, de nombreux chercheurs et constructeurs automobiles ont commencé à développer une nouvelle approche pour améliorer la sécurité routière. Elle a donné naissance à plusieurs programmes de recherche dans le monde entier, d'abord au sein des constructeurs automobiles, puis dans le cadre plus vaste du projet Eurêka en Europe, I.V.H.S. (Intelligent Vehicle - Highway System) aux Etats-Unis et V.I.C.S. (Vehicle Information and Communication System) au Japon. En Europe, le programme le plus ambitieux dans ce domaine est le projet Prometheus, dans lequel nous nous sommes impliqués.

Créer une "voiture intelligente", tel est le pari de ces programmes, qui misent, pour y parvenir, sur la conjugaison des prouesses de l'informatique et des plus hautes technologies. Dotée de capteurs qui lui permettent de percevoir son environnement, de tester son propre état et celui de son conducteur, une telle voiture serait capable de prévenir l'usager de tout risque d'accident, et l'aider ainsi à les éviter.

Plus sophistiqués encore, des systèmes de sécurité interactifs permettraient de connecter l'automobiliste à l'infrastructure routière ou à un autre conducteur pour un échange d'informations portant sur l'état de la route, le trafic, les conditions de circulation, etc. Par exemple, sur une route équipée de balises émettant et recevant des informations, lorsqu'une voiture allume ses feux antibrouillard, la balise enregistre cette information et la transmet aux autres automobilistes ainsi prévenus de la présence d'un brouillard afin d'éviter les carambolages.

Certains programmes de recherche ont achevé leurs travaux par la réalisation de certains démonstrateurs qui sont testés dans des conditions de circulation réelles. Les systèmes développés assistent le conducteur et évitent les accidents grâce à l'évaluation des risques éventuels, permettant ainsi une mise en garde (prévenir le conducteur de la présence d'un danger) ou une intervention automatique (pilotage de la vitesse ou action sur la direction, par exemple). Parmi les systèmes développés, nous pouvons citer, par exemple, le régulateur de vitesse pour conduite sur autoroute : afin de conserver un intervalle de sécurité entre deux véhicules, un télémètre laser mesure la distance qui les sépare et agit sur la vitesse de la voiture prototype par l'intermédiaire d'un calculateur en pilotant l'accélération et le freinage (le conducteur n'a plus qu'à se concentrer sur la direction). Les principaux démonstrateurs réalisés dans ce domaines sont CMU-NavLab-project [Thorpe 90] aux Etats-Unis, PVS-project [Hattori 92] au Japon, VaMoRs-

project [Behringer 92], VITA-Demonstrator [Ulmer 92] en Allemagne et ProLab2 [Rombaut 93] en France.

### **I-3 LE PROGRAMME EUROPEEN PROMETHEUS**

Le 20 juin 1986 à Londres, les ministres européens des Affaires Etrangères, de l'Economie et de la Recherche avaient tenu une conférence sur la sécurité routière et les possibilités pour l'améliorer. Ils se sont mis d'accord de lancer un programme de recherche qui avait pour but de développer des systèmes destinés à améliorer la qualité et la sécurité du transport routier en Europe : il s'agit du programme Prometheus (PROgramM for an European Traffic with Highest Efficiency and Unprecedented Safety). Dans ce programme se sont impliqués la quasi-totalité des constructeurs automobiles européens (PSA, Renault, Jaguar, BMW, Daimler-Benz, Porsche, Fiat, Volvo, etc.), ainsi que de nombreux organismes de recherche publiques ou privés.

#### **I-3.1 Les objectifs du programme Prometheus**

Dans le but d'améliorer tous les aspects du trafic routier européen, différents objectifs ont été fixés. Certains d'entre eux visent à améliorer le trafic routier en matière de sécurité et de confort. D'autres ont moins de rapports avec la sécurité routière et visent plutôt des considérations écologiques. Plus précisément, les principaux objectifs visés sont les suivants :

- L'assistance du conducteur en matière de prévention dans les situations critiques, afin d'éviter que ces situations n'entraînent des accidents. Il s'agit de développer des systèmes d'aide pour prévenir le conducteur de risques éventuels, permettant ainsi une mise en garde de celui-ci.
- L'assistance du conducteur en matière d'intervention automatique en cas d'urgence. Les recherches dans ce secteur visent à développer des systèmes qui seront capable de prendre des initiatives dans le but d'éviter un accident imminent. En cas de perte de contrôle, par exemple, des systèmes automatiques de maintenance de la trajectoire du véhicule doivent se mettre en route pour agir sur la direction.
- Le confort en toute sécurité en confiant des tâches de conduite élémentaires (accélération, freinage) à des équipements intelligents et en sollicitant le moins possible l'attention du conducteur. Cet objectif doit être atteint par des systèmes intelligents qui veille à garder un intervalle de sécurité entre les véhicules par un pilotage automatique de l'accélération et le freinage.

- L'interactivité pour une bonne gestion du trafic routier. Il s'agit d'améliorer les conditions de circulation (embouteillages, saturation du réseau routier) en connectant, via des dispositifs embarqués communiquant, les automobilistes entre eux et à l'infrastructure routière. Plus sophistiquées, les recherches en sécurité interactive visent à développer des cartes routières numériques permettant d'aider le conducteur pour optimiser son trajet : comment éviter telle déviation, quel itinéraire emprunter, etc.
- La réduction de la consommation d'énergie et de l'impact sur l'environnement. Ces deux objectifs sont en rapport avec l'objectif précédent (la fluidité du trafic), car une bonne gestion de la circulation routière (aide à la recherche du meilleur trajet, évitement des embouteillages et des saturations du réseau routier) entraîne à la fois des économies d'énergie, une diminution de la pollution et des nuisances sonores.

Notons toutefois que ces différentes fonctions définies par le comité de pilotage de Prometheus peuvent avoir des effets pervers [Broughton 91][Malaterre 92] :

- Modification du comportement des conducteurs de véhicules dotés d'équipements de sécurité, dans le sens d'une plus grande prise de risque.
- L'amélioration de la sécurité de certains usagers peut augmenter le risque d'autres usagers plus vulnérables.
- Détournement de l'attention du conducteur de sa tâche prioritaire, qui reste la conduite.
- Les défaillances des dispositifs de prévention des accidents (omission de signaler un danger, génération des fausses alertes) peuvent avoir des conséquences graves.

### **I-3.2 Contribution de groupe ProArt France**

Pour réaliser ses objectifs, le programme Prometheus a été décomposé en quatre sous-programmes, ProCom, ProNet, ProChip et ProArt, auxquels quatre thèmes de recherche ont été confiés, respectivement :

- Le développement de systèmes pour la communication inter-véhicule.
- Le développement de systèmes pour la communication intra-véhicule.
- Le développement de circuits et matériels particuliers adaptés aux véhicules.
- Le développement de systèmes d'assistance pour les conducteurs.

Le programme ProArt, auquel nous avons participé, a eu comme but la conception de systèmes d'aide à la conduite, en étudiant et développant des méthodes issues du domaine de l'intelligence artificielle. Plus précisément, les travaux ont été développés autour des deux axes suivants :

- La perception de l'environnement dans lequel le véhicule évolue.
- Le copilottage pour assister le conducteur dans les situations critiques.

Le groupe français du programme ProArt (ProArt France) est composé de neuf laboratoires de recherche et du Groupement d'Intérêt Economique PSA-Renault qui a participé activement à l'élaboration de deux démonstrateurs, ProLab1 et ProLab2. C'est dans ce dernier que le travail effectué au Centre d'Automatique de Lille a été intégré. Les laboratoires participants à ProArt France sont les suivants:

- Heudiasyc, URA CNRS 817, Compiègne.
- LIFIA, URA CNRS 394, Grenoble.
- Ingénia Dialexis, Toulouse.
- INRIA Sophia Antipolis.
- IRISA Rennes.
- Centre d'Automatique de Lille.
- Laboratoire CAO et Robotique, Ecole des Mines de Paris.
- LASMEA, URA CNRS 830, Clermont-Ferrand.
- Centre de Morphologie Mathématique, Ecole des Mines de Fontainebleau.

### ***I-3.2.1 Le démonstrateur ProLab1***

Le programme Prometheus s'est déroulé en deux phases. Lors de la première phase, de 1989 à 1991, le groupe ProArt France a achevé ses travaux par la réalisation d'un véhicule prototype, une Renault 21, baptisé ProLab1. Ce prototype ne présentait aucune autonomie, mais il était doté de certaines fonctions permettant de fournir des informations au conducteur sans agir sur le véhicule. Les principaux modules réalisant ces fonctions sont :

1. La perception de l'environnement, grâce à des capteurs multisensoriels, pour avoir des données sur l'état intrinsèque du véhicule et sur l'état dynamique des obstacles.

2. Traitement et analyse des données pour prendre des décisions sur les informations à communiquer au conducteur.
3. Présentation des informations sous forme de symboles par le biais d'une interface homme-machine.

Le démonstrateur ProLab1 a été testé et présenté au public à plusieurs occasions et notamment au "Board Member Meeting Prometheus" à Turin en septembre 1991.

### ***I-3.2.2 Le démonstrateur ProLab2***

Les limitations du prototype ProLab1 en matière d'autonomie ont poussé les responsables du programme ProArt France à le développer et l'améliorer en intégrant d'autres systèmes de perception et en mettant au point des systèmes de copilote. La deuxième phase, de 1991 à 1994, a été consacrée à la réalisation du deuxième démonstrateur ProLab2, une Peugeot 605. Son objectif, comme pour ProLab1, est d'assister le conducteur lors de la réalisation des manoeuvres et de l'informer des danger potentiels ou réels lorsqu'il est en situation de conduite.

Pour réaliser ProLab2, des tâches ont été définies et confiées aux différents laboratoires participant à ProArt France. Les différentes tâches à accomplir étaient les suivantes :

- La détection de véhicules à l'avant.
- La détection de véhicules à l'arrière.
- La détection de véhicules approchant sur les côtés (à l'arrêt au stop).
- La détection de la signalisation horizontale
- La détection d'obstacles (piétons, deux-roues, etc.).
- La fusion des données provenant des différents capteurs.
- L'interprétations des situations et copilote.
- La conception d'une interface homme-machine.

Le travail mené au sein du Centre d'Automatique de Lille dans le cadre de ce projet était la conception d'un système de détection d'obstacles.

Le démonstrateur ProLab2 a été, lui aussi, présenté aux pouvoirs publiques à l'occasion du "Board Member Meeting Prometheus" qui s'est tenu à Paris, en octobre 1994.

Pour plus de précision sur les fonctionnalités et la structure générale de ProLab2, le lecteur intéressé pourra se rapporter à [Rombaut 95][Burie 95] [ProArt 95].

## **I-4 LA DETECTION D'OBSTACLES SUR ROUTE**

La détection d'obstacles est un problème qui a été abordé depuis longtemps par les roboticiens. Récemment et avec l'émergence de l'idée de créer des systèmes d'aide à la conduite, ce problème est devenu un centre d'intérêt pour de nombreux chercheurs et constructeurs automobiles : détecter un obstacle est l'une des fonctions les plus importantes qu'un véhicule intelligent doit accomplir. D'ailleurs, cette tâche a fait l'objet de plusieurs études et recherches au sein du groupe ProArt France, dans le cadre du programme Prometheus.

De nombreuses méthodes ont été développées pour résoudre ce problème. Elles relèvent, selon le type de technologie utilisé, d'une des deux approches suivantes : approche active ou passive.

### **I-4.1 Approche active**

Les méthodes qui relèvent de cette approche sont basées sur l'utilisation des systèmes qui font intervenir un émetteur et un récepteur. L'émetteur envoie vers la scène un faisceau d'ondes radio (radar), d'ondes sonores (sonar) ou de lumière (lidar), dont les éventuelles rétrodiffusions, vers le récepteur, indiquent la présence d'obstacles. En mesurant le temps aller-retour du faisceau émis, on peut alors connaître la distance des obstacles responsables de l'écho.

Ces détecteurs sont caractérisés par la puissance et l'angle d'ouverture de l'émetteur, la sensibilité et la sélectivité du récepteur. Il faut noter aussi que l'utilisation des systèmes actifs est soumise à certains problèmes qui peuvent réduire leurs performances de détection. Ils sont dus, d'une part, au milieu de propagation qui absorbe et/ou disperse les faisceaux aller et retour, d'autre part, à la surface réfléchissante de la cible qui peut absorber le faisceau, le diffuser ou le réfléchir dans d'autres directions.

Selon leur principe de fonctionnement, on distingue trois types de systèmes actifs : les radars, les sonars et les lidars.

#### ***I-4.1.1 Les radars***

Permettant une bonne précision de localisation, les radars sont bien adaptés à la détection des véhicules dont la masse métallique importante réfléchit très bien les ondes radio. De plus, la largeur du faisceau, qui définit la zone de détection, est déterminée grâce aux caractéristiques de l'antenne. En utilisant un dispositif mécanique, un balayage horizontal par un faisceau étroit

permet de construire une carte de profondeur d'une scène, mais avec une certaine lenteur et une fiabilité amoindrie [Deloof 90].

Les radars peuvent être utilisés aussi pour détecter les piétons, jusqu'à une distance de 80 m. De fausses détections peuvent subsister, en raison des réflexions parasites causées par le fond de la scène (panneaux routier, ponts, etc.) et la pluie, ce qui réduit l'intérêt du dispositif pour une telle application. Les solutions envisagées pour limiter ces fausses détections consistent à utiliser des transpondeurs sur les équipements routiers et développer des systèmes sophistiqués pour le traitement du signal.

Un système anticollision, basé sur un radar micro-onde à impulsions 60 GHz, a été développé par l'IEMN<sup>1</sup> en collaboration avec le LEOST<sup>2</sup> [Deloof 94]. D'autres applications sont également étudiées dans le cadre du projet SMILER (projet DRIVE V1002) [Drive 89].

#### ***I-4.1.2 Les sonars***

Comme tous les systèmes actifs, les sonars sont constitués de deux éléments : un émetteur (haut-parleur) et un récepteur (microphone). Comparés aux radars, ils présentent les avantages suivants : un coût, une consommation électrique et un encombrement nettement inférieurs. De plus, l'utilisation des ultrasons permet de détecter tous les obstacles acoustiquement réfléchissants, ce qui représente la majorité des objets présents dans la nature.

Sur route, la probabilité de fausses détections, pour ce type de capteur, est relativement importante à cause des problèmes de vibrations et d'écoulement de l'air qui peuvent saturer le microphone. De plus, ils ne permettent pas d'atteindre de grandes portées, même dans de bonnes conditions climatiques, ce qui diminue l'intérêt de leur utilisation pour des applications au trafic routier.

Limités généralement à la perception rapprochée, les sonars sont souvent utilisés en robotique mobile pour analyser et modéliser des scènes d'intérieurs [Wane 93], par exemple pour suivre un mur à distance constante [Briot 79].

#### ***I-4.1.3 Les lidars***

Ce type de capteur utilise le même principe que les radars, la nature physique du signal étant différente : on utilise un faisceau lumineux, généralement issu d'un laser, au lieu d'un

---

<sup>1</sup>Institut d'Electronique et Micro-électronique du Nord.

<sup>2</sup>Laboratoire d'Electronique, d'Ondes et des Signaux pour les Transports.

faisceau d'ondes radio. Notons que les systèmes basés sur l'utilisation des lidars peuvent également faire appel à d'autres principes que le temps de vol : il s'agit des télémètres optiques à déphasage ou à triangulation [Monchaud 86].

Les lidars permettent de détecter tout type d'objets, métalliques ou non, à l'exception de ceux qui possèdent des propriétés réfléchissantes inadaptées au faisceau lumineux (objets sombres, mats ou transparents). Par temps clair, ils offrent une bonne localisation des objets, mais le soleil peut parfois saturer le récepteur et le rendre inopérant. Leur portée est limitée dans certaines conditions météorologiques défavorables (brouillard, pluie, neige) en raison de dispersion et/ou absorption des faisceaux aller et retour. Il faut noter aussi que la présence de la poussière sur les lentilles frontales du lidar réduit également sa portée [Deloof 90].

Dans le cadre du projet Prometheus, Alizon et al. [Alizon 90] ont décrit une réalisation très performante de lidar destinée à être embarquée dans une voiture. Dans le même cadre, un capteur multisensoriel, constitué d'un télémètre laser et d'une caméra vidéo, a été conçu [Gallice 91] dont le but de détecter des obstacles à l'avant d'un véhicule [Gallice 94a][Gallice 94b].

#### **I-4.2 Approche passive**

Ne nécessitant l'émission d'aucun signal, les systèmes passifs ne sont constitués que d'un seul élément : il s'agit d'un système visuel composé d'une ou plusieurs caméras. Les avantages par rapport aux systèmes actifs sont un affranchissement des problèmes d'inférences entre capteurs ou provenant de l'environnement, une meilleure résolution et une exploration plus rapide de la scène. La vision passive présente néanmoins des inconvénients liés généralement à une mauvaise différenciation entre les objets et le fond. Sur route, par exemple, le principe de fonctionnement des systèmes passifs suppose un contraste suffisant entre les images des obstacles et celles de la chaussée. Par conséquent, une mauvaise visibilité dans certaines conditions climatiques (pluie, brouillard, neige) rend souvent les images inexploitable, ce qui compromet toute détection.

Le principe de cette approche consiste à traiter, analyser puis interpréter les images issues des caméras, ce qui exige un coût informatique important. Si l'aspect algorithmique est un axe de recherche fondamental en ce domaine, des travaux portent également sur l'implantation de ces méthodes en temps réel sur des architectures matérielles dédiées.

En vision passive, il existe trois grandes familles de méthodes pour la détection d'obstacles :

- Méthodes basées sur la segmentation des images monoculaires : à cause de leur principe de fonctionnement, qui est la segmentation, ce type de méthode présente l'inconvénient de ne pas différencier les obstacles réels des autres objets tels que des ombres, des marquages au sol, etc. De plus, ces méthodes ne fournissent aucune information sur le relief et le mouvement relatif des objets, ce qui les rend mal adaptées à ce genre d'application. Un exemple de cette approche est décrit dans [Graefe 90].
- Méthodes basées sur l'analyse du mouvement : ce type d'approche utilise l'information mouvement des objets présents dans la scène. Cette information est extraite à partir de l'analyse d'une séquence d'images issues d'une seule caméra. Contrairement aux méthodes basées sur la segmentation, celles-ci permettent de différencier, dans un contexte routier, les objets en mouvement, tels que les véhicules, des objets immobiles, tels que les marquages de la chaussée. Cependant ces méthodes présentent un défaut majeur. Si elles détectent effectivement les obstacles, elles ne permettent pas de les localiser. Une application basée sur l'extraction du flux optique est décrite dans [Enkelmann 90].
- Méthodes basée sur la stéréovision : elles permettent d'évaluer la profondeur des objets à partir de plusieurs images vidéo, prises sous des angles de vue légèrement différents. La connaissance de la position des objets permet donc de différencier les obstacles réels des autres objets, tels que les ombres et les marquages au sol. Il faut cependant préciser que la stéréovision présente un inconvénient majeur lié à la quantité importante de données à traiter, ce qui est incompatible avec la rapidité nécessaire pour une application temps réel. Une étude plus détaillée de cette approche fera l'objet du chapitre II.

## **I-5 NOUVELLE APPROCHE**

Les systèmes actifs de détection d'obstacles qui paraissent les plus complets sont complexes et chers. En revanche, ceux qui sont plus simples ont des performances limitées. Ainsi, nous avons opté pour une approche passive, la stéréovision, qui se contente de capteurs simples et d'unités de traitement assez petites et économes, en termes de consommation

électrique, pour pouvoir être embarqués dans une voiture particulière sans en affecter les capacités ou le fonctionnement.

C'est dans ce cadre que le Centre d'Automatique de Lille a participé au programme Prometheus, en développant un système stéréoscopique à base de caméras linéaires. Ce nouveau dispositif, dont la conception a fait l'objet de la thèse de Jean-Luc Bruyelle [Bruyelle 94], offre les avantages d'un système de détection stéréoscopique classique (utilisation des caméras matricielles), sans en avoir les inconvénients (temps de traitement) puisque la quantité d'information contenue dans une image linéaire est nettement inférieure à celle d'une image matricielle. De plus, une caméra linéaire présente une meilleure résolution horizontale qu'une caméra matricielle.

Suite à ce travail, Jean-Christophe Burie, durant sa thèse, a développé des algorithmes de traitement des images issues du stéréoscope linéaire [Burie 95]. Il s'agissait de mettre en correspondance les images linéaires afin de détecter et localiser les obstacles. Sa démarche, qui est basée sur des techniques de corrélation, a permis d'obtenir de bons résultats.

Afin d'améliorer les résultats de détection et de localisation, notre travail consistait, également, à traiter et analyser les images issues du stéréoscope linéaire. Basées sur l'utilisation des réseaux de neurones, les méthodes que nous avons développées ont été jugées très satisfaisantes.

## **I-6 LE CALCUL NEUROMIMETIQUE**

Depuis déjà très longtemps, les neurobiologistes mènent des recherches théoriques et expérimentales afin d'essayer de comprendre l'anatomie et le fonctionnement du cerveau dans son ensemble. Ils se sont trouvés devant un organe très complexe qui reste et restera pour longtemps un grand mystère que les sciences humaines se proposent de découvrir. Ce paradoxe a été merveilleusement bien exprimé par David Hubel [Hubel 79] : le cerveau peut-il comprendre le cerveau ?

Le calcul neuromimétique relève d'une approche dont l'objectif est de comprendre les principes selon lesquels les systèmes biologiques du cerveau traitent l'information et s'en inspirer pour élaborer de nouvelles techniques en sciences de l'ingénieur. C'est une double démarche, à la fois cognitive et synthétique, où le monde biologique doit être considéré comme une source de références et de connaissances. Il s'agit donc de développer des modèles dont l'intention est

d'imiter certaines des fonctions du cerveau en reproduisant certaines de ses structures de base. Ces modèles sont connus sous le terme de réseaux neuronaux artificiels, réseaux connexionnistes ou réseaux neuromimétiques. Un réseau neuronal est un ensemble d'unités de calcul, appelées neurones artificiels, interconnectées selon une architecture spécifique.

### **I-6.1 Pourquoi des réseaux neuronaux artificiels ?**

Reconnaître un visage, apprendre à parler, s'adapter à un environnement sont des exemples de tâches que tout être humain est capable d'accomplir presque instantanément, parfois dès son plus jeune âge, alors que l'ordinateur le plus puissant en est totalement incapable. Les capacités de généralisation, d'association, de classification, d'optimisation sont également caractéristiques de notre activité journalière et sont associées aux facultés d'auto-organisation et d'association de notre cerveau. Pour une machine, réaliser de telles tâches serait une véritable prouesse, et nous, nous les réalisons sans même y penser ! C'est là que réside l'attrait des réseaux neuronaux artificiels. Copier le cerveau restera pour longtemps encore une ambition exagérée, mais vouloir s'inspirer des architectures et des fonctions du système nerveux n'est pas un rêve inaccessible. Le développement continu des connaissances en biologie, l'apparition de nouvelles méthodes théoriques et l'incessante montée en puissance des outils de simulation autorisent les meilleurs espoirs.

L'intérêt porté par les chercheurs aux réseaux neuronaux a également une autre motivation, à savoir l'inadaptation de l'approche symbolique aux différents problèmes courants dans les sciences de l'ingénieur pour lesquels les solutions proposées sont souvent incomplètes. Cette approche est basée sur l'hypothèse que le raisonnement modélisant la pensée est une combinatoire de symboles soumise à des règles logiques. Malgré son énorme succès dans le développement de l'intelligence artificielle, en dégageant les fameux systèmes experts, elle semble montrer aujourd'hui ses limites, en particulier son inadéquation avec le type de raisonnement du cerveau humain, qui ne fonctionne pas totalement par une logique de manipulations de symboles de la forme "si...alors". Les principales faiblesses de cette approche symbolique sont les suivantes :

- Une première faiblesse tient à la lenteur des méthodes séquentielles utilisées, difficilement parallélisables, ce qui donne lieu à une explosion combinatoire dès que le nombre des données à traiter devient important. Cet inconvénient ne se retrouve pas dans l'approche connexionniste. En effet, la notion de parallélisme est intrinsèque

aux réseaux neuronaux, constitués d'unités de calcul, ou neurones artificiels, qui peuvent travailler en même temps.

- Une deuxième faiblesse concerne la représentation des connaissances. Elle est localisée en ce sens qu'un atome de connaissance est représenté par un objet précis qui peut être par exemple une règle de production, un octet en mémoire, etc. Par contre, les réseaux neuronaux permettent une représentation distribuée des connaissances : la mémoire d'un fait correspond à la mise en activité d'un groupe spécifique de neurones artificiels. Ce mode de représentation délocalisé permet une certaine résistance à la détérioration.
- Une troisième faiblesse de l'approche symbolique concerne la notion d'apprentissage. En effet, il semble difficile, en utilisant le calcul symbolique, de simuler l'apprentissage d'un ensemble de faits. Or, les réseaux neuronaux possèdent une telle aptitude. L'apprentissage d'un fait est réalisé par renforcement des connexions entre les neurones mémorisant ce fait, et le réseau s'auto-organise à partir d'exemples qui lui sont présentés.

En résumé, l'approche connexionniste, opposée à l'approche symbolique, privilégie les avantages suivants :

- L'activité parallèle de nombreux composants.
- La représentation distribuée des connaissances.
- La notion d'apprentissage par modification des connexions.

Ce furent ces avantages qui poussèrent les chercheurs, dans les années 80, à effectuer une ré-évaluation de l'approche connexionniste (Cf. § I-6.2).

Actuellement, les deux approches coexistent, et l'avenir verra sans doute des tentatives de synthèse, l'approche connexionniste prenant en charge les tâches de bas niveau et l'approche symbolique modélisant, intégrant et supervisant différents domaines d'auto-organisation et élaborant des synthèses selon différents niveaux d'abstraction.

## **I-6.2 Historique**

Les premiers travaux de simulation des neurones sont dus à MacCulloch et Pitts en 1943 [MacCulloch 43]. L'idée principale est de simuler de façon très simplifiée le neurone biologique. Dans leur approche, le neurone artificiel, ou formel, effectue une somme pondérée des

excitations provenant d'autres neurones. Si cette somme est supérieure à un seuil donné alors le neurone est activé, sinon il est inactif : l'information délivrée est booléenne,  $\{0,1\}$  ou  $\{-1,1\}$ . Dans ce modèle, les coefficients de pondération sont constants et il n'existe pas de procédure d'apprentissage.

En 1949, le neurophysicien Hebb [Hebb 49], dans une perspective psycho-physiologique, a établi le premier mécanisme qui met en évidence l'importance du couplage synaptique dans le processus d'apprentissage. L'interprétation de ce mécanisme pour les réseaux neuronaux artificiels est la suivante : on considère que si deux neurones connectés entre eux sont activés simultanément, la connexion qui les relie doit être renforcée. Dans les cas contraire, elle n'est pas modifiée.

Les résultats de ces travaux ont été à l'origine de modèles connexionnistes de plus en plus élaborés. En 1958, Frank Rosenblatt [Rosenblatt 62][Rosenblatt 65] a développé le concept du Perceptron pour expliquer et modéliser les facultés de reconnaissance des formes du système visuel. Ce modèle est identique au neurone formel, mais doué de la faculté d'apprentissage, c'est un séparateur linéaire. L'idée essentielle du Perceptron de Rosenblatt est de considérer la rétine de l'oeil comme une matrice de capteurs. Ces derniers sont reliés à des processeurs élémentaires intermédiaires qui réalisent des opérations prédéterminées pour reconnaître des formes visuelles particulières. Les sorties de ces processeurs sont ensuite injectées à des systèmes à seuil non linéaires binaires en vue d'une classification automatique. En parallèle, vers le début des années 60, Bernard Widrow [Widrow 60] a développé des concepts similaires pour la résolution des problèmes de filtrage adaptatif du signal en mettant au point deux modèles très connus : l'ADALINE (ADaptive Linear Neuron) et le MADALINE (Many ADALINES). Il a montré que l'ADALINE était capable de séparer linéairement des classes dans l'espace à deux dimensions. La simplicité de mise en oeuvre du Perceptron, de l'ADALINE et du MADALINE et leurs aptitudes à résoudre les problèmes simples de classification ont permis à cette époque de développer de nombreux travaux sur le calcul neuromimétique.

De leur côté, Marvin Minsky et d'autres chercheurs ont analysé, sous l'angle mathématique, les performances du Perceptron. Dans l'ouvrage publié par Minsky et Papert [Minsky 69], les auteurs démontrèrent les limites théoriques du Perceptron, en particulier son incapacité de résoudre la séparation pour l'opérateur logique "OU exclusif", et qu'en conséquence ce modèle ne présente aucun intérêt. De plus, ils affirment que même si les Perceptrons multicouches sont

capables de résoudre ce fameux problème du "OU exclusif", il est impossible de définir un mécanisme d'apprentissage basé sur la minimisation de l'erreur entre la sortie désirée et la sortie fournie par un Perceptron multicouches. L'effet de ces conclusions fut presque immédiat : chercheurs et investisseurs se désintéressèrent des réseaux neuronaux pour se tourner vers l'approche symbolique de l'intelligence artificielle, qui semblait beaucoup plus prometteuse.

Entre 1969 et 1978 des chercheurs comme Amari [Amari 71][Amari 72] ont étudié les propriétés d'un réseau de neurones formels totalement connecté pour l'utiliser en tant que mémoire associative. Pour sa part, Little [Little 74] a proposé l'utilisation des méthodes de la physique statistique pour l'étude de tels réseaux. Après la mise en évidence des sévères limitations du Perceptron, le "renouveau" mondial sur les réseaux neuronaux artificiels apparaît en 1982 avec John Hopfield [Hopfield 82] qui a persévéré dans l'étude des réseaux connexionnistes en introduisant la notion d'énergie d'un ensemble de neurones, tous interconnectés, en se basant sur la physique quantique des "verres de spin". La puissance de cette démarche a relancé les travaux sur les applications des réseaux neuromimétiques et le développement de réseaux particuliers, tels que les réseaux de Boltzmann, qui mettent en oeuvre la technique du recuit simulé.

Une autre étape majeure, contredisant les conclusions de Minsky et Pappert, a été franchie au début des années 80 avec l'algorithme de rétropropagation du gradient [Parker 85][LeCun 87][Rumelhart 86]. Le réseau associé à cet algorithme est structuré en couches interconnectées. L'apprentissage dans de tels réseaux consiste à associer à un ensemble de données d'entrée un ensemble de données de sortie désirées : on parle d'apprentissage supervisé. La mise à jour des poids des connexions se fait ensuite de façon itérative en minimisant, à l'aide de la méthode du gradient, une fonction de coût quadratique correspondant à l'écart entre les sorties désirées et celles obtenues avec les poids courants. Contrairement au Perceptron, cet algorithme permet la création de discriminateurs d'ordre supérieur.

Pendant la même période, de nombreuses équipes ont poursuivi les études menées par Hebb et ont mis au point les techniques d'apprentissage compétitif. L'ensemble de ces techniques, contrairement à l'algorithme de rétropropagation du gradient, ne nécessite pas l'utilisation d'un professeur pendant la phase d'apprentissage : on parle d'apprentissage non supervisé. Parmi les travaux les plus célèbres, on retiendra ceux de Grossberg [Grossberg 88],

Kohonen [Kohonen 82][Kohonen 84][Kohonen 88a][Kohonen 90] et Fukushima [Fukushima 75a][Fukushima 84].

Actuellement, une approche neuromimétique nouvelle est en train de se développer. Elle est connue sous le terme de réseaux modulaires [Hrycej 92][Haykin 94]. Celle-ci consiste à décomposer une tâche complexe en un ensemble de sous tâches selon le principe "diviser pour régner". Chacune de ces tâche est réalisée par un modèle neuronal. Il est important de noter que les modèles utilisés peuvent être identiques ou hybrides. Le principe consiste alors à faire coopérer ces modèles. Des travaux récents montrent, sur des cas particuliers, qu'une telle approche permet d'accroître les performances du connexionnisme [Elhor 96]. En revanche, il devient difficile de formaliser et de contrôler l'apprentissage, dit apprentissage modulaire, dans ce type de réseau.

Cet inventaire sommaire des principaux modèles exprime la richesse des différentes approches provenant de la synergie entre les biologistes, les neurobiologistes, les physiciens et les informaticiens. L'explosion, quasi-exponentielle, du nombre d'algorithmes disponibles actuellement rend malheureusement impossible leur description exhaustive dans le cadre de cette thèse. Le lecteur intéressé pourra se référer à la littérature [Wasserman 89][Pao 90][Lippmann 87][Kamp 90][Abdi 94][Amit 89][Anderson 88][Burnod 88][Freeman 92][Grossberg 88][Hecht Nielsen 90][Jodouin 94][Hinton 81].

Il est à noter enfin l'apport de la technologie VLSI, qui ne cesse d'évoluer, aux domaine des réseaux neuronaux. En effet, devant la simplicité du neurone artificiel dans la plupart des modèles, il est intuitivement tentant de penser à l'implantation d'un réseau neuronal directement en "hardware" [Choi 91][Satanarayama 92][Sheu 92]. Avec cet aspect, les réseaux neuronaux électroniques permettront non seulement d'atteindre les ordres de grandeur nécessaires pour traiter en temps réel certaines applications, mais également pour une utilisation des techniques neuronales sur des systèmes spécialisés où l'encombrement est limité [Mead 89].

### **I-6.3 Pour quelles applications ?**

L'intérêt porté ces dernières années aux réseaux de neurones tient sa justification dans les quelques propriétés fascinantes (parallélisme, adaptation, généralisation, mémoire distribuée) qu'ils possèdent et qui devraient permettre de dépasser les limitations de l'informatique traditionnelle (calcul symbolique), tant au niveau programmation qu'au niveau machine. Cet outil recouvre actuellement plusieurs applications dans de nombreux domaines :

- Reconnaissance des formes et classification des données [Knerr 92][Ben-Hanan 92][Bye 93][Flocchini 92][Fukushima 75b][Lippmann 89][Pang 90].
- Robotique et contrôle adaptatif [Jorgsen 87][Barto 83][Levin 93][Gaudiano 94].
- Traitement du signal [Hérault 94].
- Diagnostic des défaillances [Zwingelstein 95].
- Optimisation [Hérault 91][Pekergin 92][Sun 93][Peterson 89][Sudharsanan 91][Abe 95][Aiyer 90][Protzel 93][Hopfield 85][Levy 87].
- Vision artificielle et traitement des images [Mammone 94][Huang 92][Kendall 92][Pham 92][Koch 86][Shadpey 94][Zhou 92].
- Compréhension, reconnaissance et synthèse de la parole [Sejnowski 87][Mammone 94][Unnikrishnan 91].
- Codage et compression de l'information [Basso 92][Cai 92][Fang 92][Sheu 91][Walker 94][Zeng 92].
- Prévision et modélisation [Bailey 88].
- Gestion de l'information [Cochet 88].

Le lecteur intéressé trouvera d'autres applications dans [Bulsari 96][Bulsari 95][Maren 90][Bishop 95].

La vision semble être un des domaines d'application privilégié des réseaux neuronaux. Les recherches dans ce domaine ont permis de traiter de nombreux problèmes :

- Segmentation des images [Chou 93] [Graf 92][Gish 90] [Hwang 93] [Koh 93] [Moura 91][Loncelle 90][Chung 94].
- Restauration des images [Zhou 88b].
- Détection et analyse du mouvement dans une séquence d'images [Yacoubi 96][Hanssens 92][Ardizzone 90][Missler 95][Ögmen 90][Ambellouis 95][Zhou 88c][Zhou 90][Lelaurin 96].
- Stéréovision et reconstruction 3D [Nasrabadi 92b][Zhou 88a][Lee 94b][Hongo 96][Cruz 95].
- Reconnaissance des formes visuelles [Kohonen 88b][Freixinho 95][Heikkonen 95].

Le sujet de ce mémoire porte sur le problème de la stéréovision linéaire. Nous nous intéressons plus particulièrement au problème de l'appariement stéréoscopique des images linéaires par utilisation d'une approche neuromimétique.

## I-6 PLAN DU MEMOIRE

Ce mémoire présente la phase finale des travaux de recherche effectués au Centre d'Automatique de Lille dans le cadre du programme Prometheus. Le but de ces travaux était de concevoir un système de détection d'obstacles devant un véhicule routier. Les recherches ont commencé par la conception d'un stéréoscope composé de caméras linéaires qui a constitué l'essentiel de la thèse de Jean-Luc Bruyelle. Afin de développer un système de détection d'obstacles, la deuxième partie, réalisée par Jean-Christophe Burie durant sa thèse, a porté sur le développement d'algorithmes de traitement des images linéaires issues du stéréoscope linéaire. Dans le but d'améliorer les résultats de traitement, nous avons développé une approche neuronale pour la stéréovision linéaire qui constitue l'essentiel de la présente thèse.

Puisque l'ensemble des travaux fait usage d'une approche passive, le chapitre II sera consacré à une analyse des méthodes de perception du relief en vision artificielle, plus particulièrement celle basée sur la stéréovision dont nous devons nous inspirer.

Le but de ce travail est d'analyser les images issues du stéréoscope linéaire. Il nous a paru donc judicieux de rappeler, au chapitre III, les étapes essentielles de la conception de ce capteur stéréoscopique. Ceci permettra au lecteur d'avoir un aperçu général sur la stéréovision linéaire.

La première étape d'un processus stéréoscopique est le traitement bas niveau des images à appairer. Celle-ci consiste à extraire, de chaque image, les indices visuels qui seront des candidats à la mise en correspondance. Le chapitre IV expose la méthode que nous avons utilisée pour mettre en évidence les indices visuels contenus dans les images linéaires.

Dans le même cadre de travail, Jean-Christophe Burie a développé une approche basée sur des techniques de corrélation pour appairer les indices visuels extraits des images linéaires. Les principaux points de cette approche ainsi que certaines critiques sont présentés au chapitre V.

L'approche que nous avons développée pour aborder le problème de l'appariement des images linéaire est basée sur l'utilisation des réseaux de neurones. Dans un premier temps, le problème est formulé comme un problème d'optimisation où une fonction de coût doit être minimisée. A partir de cette dernière, un réseau de neurones de Hopfield est construit pour accomplir le processus de minimisation. Avant de décrire notre approche (Cf. Chapitre VII), nous exposons, au chapitre VI, le modèle neuronal de Hopfield, en particulier son utilisation pour minimiser des fonctions de coût.

Le chapitre VIII, conclusion de ce mémoire, présente les améliorations susceptibles d'être apportées à notre méthode de mise en correspondance, ainsi que les perspectives futures.

## **Chapitre II**

# **Vision stéréoscopique**

## II-1 INTRODUCTION

La vision est notre sens le plus puissant. Elle nous fournit toutes sortes d'informations sur les objets qui nous entourent et nous permet de réagir intelligemment en fonction de notre environnement. Nous sommes ainsi capables de percevoir les formes, les couleurs et les mouvements des objets. Mais nous possédons aussi la capacité de structurer le monde extérieur, d'évaluer les grandeurs et les distances grâce à notre système visuel binoculaire.

La vision est aussi notre sens le plus complexe, bien que cette faculté semble tellement naturelle qu'elle passe inaperçue. Pourtant, elle est le résultat d'un traitement cérébral incroyablement complexe, nécessitant le travail de milliards de neurones dans le cerveau. Depuis déjà très longtemps, les neurobiologistes mènent des recherches théoriques et expérimentales afin d'essayer de comprendre l'anatomie et le fonctionnement du cerveau dans son ensemble. Ils ont découvert une structure très complexe qui reste et restera pour longtemps un bastion encore inconnu que les sciences humaines se proposent de conquérir. David Hubel [Hubel 79] a merveilleusement bien exprimé ce paradoxe : le cerveau peut-il comprendre le cerveau ?

Avec l'apparition de machines de calcul de plus en plus sophistiquées, un certain nombre de scientifiques se sont attaqués au problème de la vision d'un point de vue quantitatif. Il ne s'agit pas de fournir une explication de la vision biologique, mais de créer des modèles qui peuvent reproduire le même type de performances visuelles. Grâce aux efforts des chercheurs en robotique et en intelligence artificielle, il est possible aujourd'hui de concevoir des systèmes de vision capables d'effectuer certaines tâches sous des conditions bien contrôlées, comme la reconnaissance de certains objets sur une chaîne de production, mais on est encore loin de fournir un modèle qui parviendra à rivaliser avec la vision humaine.

Un des objectifs de la vision artificielle est de percevoir le relief d'un environnement à partir d'une ou plusieurs images. La vision stéréoscopique constitue la méthode la plus utilisée dans ce domaine. Il s'agit de retrouver la géométrie spatiale d'une scène à partir de plusieurs images prises sous des angles différents. Le problème central de cette méthode est la mise en correspondance des images stéréoscopiques. Le présent chapitre décrit en détail les principes fondamentaux de la stéréovision, avec une attention particulière au problème de l'appariement qui est abordé d'un point de vue géométrique et algorithmique.

## II-2 PERCEPTION DU RELIEF PAR VISION ARTIFICIELLE

Depuis quelques années, la vision artificielle a fait des progrès considérables. Le plus remarquable d'entre eux est la possibilité d'obtenir, à partir d'images vidéo, des informations tridimensionnelles. Les recherches dans ce domaine ont permis de développer trois grandes familles de méthodes : les méthodes basées sur la vision monoculaire, les méthodes utilisant la vision dynamique et les méthodes dites K2D.

### II-2.1 Vision monoculaire

En vision monoculaire, une seule image est utilisée pour extraire des informations sur le relief de la scène observée. Il s'agit de processus d'inférence qui utilisent des connaissances spécifiques à chaque type d'analyse. Dans le cas d'une analyse des contours, les connaissances sont uniquement géométriques. Les processus d'analyse de textures utilisent à la fois des connaissances géométriques et statistiques. Les connaissances photométriques et colorimétriques sont utilisées respectivement lorsqu'on opère par analyse des ombrages et reflets. Pour plus d'information sur ces techniques, le lecteur pourra se référer à la littérature spécialisée [Bolles 82][Waltz 75][Oshima 79][Freeman 79][Ikeuchi 83].

Pour retrouver le positionnement d'un objet polyédrique et son dimensionnement, Leroi [Leroi 89] propose un système de vision monoculaire pour un robot mobile en utilisant ce qu'on appelle l'inversion de perspective. Waxman [Waxman 86] utilise aussi la vision monoculaire et propose un système permettant de faire un suivi de la route, afin de construire un modèle tridimensionnel de celle-ci.

### II-2.2 Vision dynamique

En vision dynamique, la structure spatiale d'une scène est produite par analyse d'une séquence d'images de celle-ci. Cette analyse est basée sur la mesure du mouvement apparent d'indices visuels extraits des images de la séquence.

Le mouvement apparent dans une séquence d'images est le résultat du déplacement relatif des objets par rapport à la caméra dans la scène ou vice-versa et peut être obtenu de trois façons :

- Caméra fixe, objets mobiles.
- Caméra mobile, objets fixes.
- Caméra mobile, objets mobiles.

L'extraction de l'information tridimensionnelle à partir des scènes dynamiques peut être réalisée en deux étapes [Ullman 79] :

- La première consiste à mesurer le champ des déplacements des objets par rapport à la caméra.
- En analysant le champ des déplacements, la seconde étape détermine les paramètres du mouvement et la structure 3D des objets.

Avant d'aborder les méthodes permettant d'analyser le mouvement dans une séquence d'images, citons quelques travaux basés sur la vision dynamique pour analyser des scènes 3D. Si-ahmed [Si-ahmed 86] a intégré la vision dynamique dans le système de perception 3D global du robot HILARE. Dans une première étape, il s'agit de déterminer le mouvement du robot par rapport à des entités fixes extraites du modèle de la scène. La seconde étape consiste à déterminer les mouvements des objets mobiles, estimés dans le référentiel du robot. Dans [Lawton 83] et [Chou 87], les auteurs proposent de déterminer la structure d'une scène à partir d'une séquence d'images obtenue par translation pure d'une caméra. La vision dynamique est utilisée aussi dans le domaine de navigation des véhicules routiers. Un système de détection d'obstacles par analyse du flux optique est décrit dans [Krüger 95].

L'étude du mouvement dans une séquence d'images peut être envisagée de plusieurs façons différentes. Les différentes méthodes existantes peuvent être classées en trois catégories : les méthodes structurelles, les méthodes par analyse du flux optique et les méthodes orientées mouvement.

### ***II-2.2.1 Méthodes structurelles***

Les méthodes structurelles sont basées sur l'étude de l'évolution temporelles de certains éléments caractéristiques d'objets présents dans les différentes images [Shariat 90]. Ceux-ci peuvent être définis comme des éléments de l'image (pixel, ligne, surface) possédant des propriétés caractéristiques permettant de les différencier de leurs voisinages [Aggarwal 75][Korn 80][Huntsberger 87][Blosseville 87]. La recherche de ces éléments structurels peut se faire soit sur l'image initiale, soit sur une image transformée : image de contours [Lawton 83][Maitre 84], transformée de Hough [Fennema 79].

Afin de connaître le mouvement d'un objet, il est nécessaire de définir sa position dans toutes les images successives de la séquence. Ce suivi d'objet est établi grâce à une procédure de mise en correspondance des éléments caractéristiques de l'objet. Une description des différentes

méthodes de mise en correspondance est donnée dans [Aggarwal 81]. Ce processus de mise en correspondance permet d'associer un déplacement à chaque objet lors du passage d'une image à la suivante. En analysant les images successives de la séquence, on parvient à reconstituer les trajectoires des objets appariés, et ainsi déterminer les paramètres de leurs mouvements et leurs structures tridimensionnelles.

Les méthodes basées sur ce principe d'évaluation sont en général très précises. Cependant, elles présentent le désavantage majeur de nécessiter une segmentation des images, qui n'est malheureusement pas simple quand on ne dispose pas de connaissances à priori concernant les objets.

Récemment, en se plaçant dans le cadre de la géométrie projective, Kachi [Kachi 96] a proposé des invariants projectifs pour mettre des segments de droite en correspondance entre les images successives d'une séquence. Sous l'hypothèse que la scène dans laquelle se déplace une caméra contient au moins deux structures planes statiques, Kachi est capable de retrouver le déplacement de la caméra ainsi que ses caractéristiques intrinsèques. Cette approche nécessite cependant la détermination préalable de cinq segments de droite coplanaires qui doivent être vus dans toutes les images de la séquence, ce qui limite considérablement son champ d'application.

### *II-2.2.2 Analyse du flux optique*

L'approche basée sur le flux optique comporte deux étapes distinctes :

1. Calcul du flux optique. Il s'agit, dans un premier temps, d'associer à chaque pixel d'une image de la séquence un vecteur déplacement permettant de le situer dans l'image suivante. Le champ de vecteur ainsi obtenu est appelé champ des déplacements. En divisant chacun de ces vecteurs par l'intervalle de temps séparant deux acquisitions d'images successives, on obtient un nouveau champ de vecteurs représentant cette fois la vitesse en chaque pixel : c'est le flux optique.
2. Analyse du flux optique. Ayant calculé le flux optique, cette étape consiste à calculer les paramètres cinématiques de ces vitesses et d'estimer la structure tridimensionnelle des objets. Il s'agit de mettre en évidence, sous certaines conditions, différentes caractéristiques concernant les objets observés en analysant le flux optique déduit de la séquence. On forme alors un système d'équations liant le flux optique aux paramètres décrivant le mouvement et la

structure tridimensionnelle des objets dans la scène [Webb 81][Webb 82][Ballard 83][Aggarwal 85][Mitiche 86][Shariat 90]. La résolution de ce système nécessite d'imposer des contraintes sur la nature des objets (surfaces planes) et des contraintes sur leur mouvement apparent (translation pure ou rotation pure). Une méthode itérative pour résoudre ce type de système d'équations est décrite dans [Bruss 83] et [Horn 86].

Plusieurs méthodes ont été développées pour déterminer le flux optique à partir d'une séquence d'images. Parmi celles-ci, nous pouvons citer celles basées sur les variations d'intensité de l'image [Horn 87]. Ceci nécessite l'évaluation des dérivées partielles de la fonction intensité par rapport aux variables spatiale et temporelle. De nombreuses méthodes utilisant ce principe sont décrites dans la littérature [Limb 75][Cafforio 76][Fennema 79][Thompson 81][Wohn 83][Mitiche 87][Ma 87]. Ce type de méthodes nécessite de poser les hypothèses suivantes :

- Les variations de la luminance sont lentes et continues,
- Deux points voisins d'un même objet ne peuvent pas avoir des déplacements différents. C'est l'hypothèse de continuité du mouvement.

Il faut noter que ces méthodes sont sensibles au bruit et présentent une difficulté pour détecter et gérer les discontinuités potentielles du flux optique en cas d'occlusion d'un objet par un autre. De plus, avec ces méthodes, on ne peut travailler qu'avec de faibles déplacements d'une image à l'autre.

L'estimation du flux optique peut être obtenue aussi en utilisant des techniques de corrélation [Leese 71][Bers 80][Rives 86]. Etant donné un pixel d'une image, la détermination du vecteur déplacement qui lui est associé revient à rechercher sa nouvelle position dans l'image suivante. Pour chaque pixel, plusieurs candidats sont possibles. Pour sélectionner le meilleur, une fenêtre de la première image est corrélée avec la seconde image. Cette technique est très robuste par rapport au bruit, mais est pénalisante en ce qui concerne le temps de calcul. Les méthodes basées sur la corrélation deviennent inefficaces dans le cas où il existe plus d'un objet en mouvement dans la scène observée ou lorsqu'il y a des distorsions dues au type du mouvement des objets. Le calcul des vecteurs déplacement étant réalisé sur certaines fenêtres, il s'avère nécessaire d'extrapoler les résultats afin d'obtenir des vecteurs en tous les points de l'image [Davis 79][Bouthemy 86].

D'autres auteurs [Huang 81][Haskell 74] ont proposé d'utiliser la transformée de Fourier pour calculer le flux optique. Malgré les progrès réalisés dans le domaine du calcul rapide par utilisation de circuits spécialisés, les calculs nécessaires à la détermination du flux optique par cette approche [Snyder 83][Rajala 83] restent lourds et pénalisants dans une optique temps réel.

### *II-2.2.3 Méthodes orientées mouvement*

Les méthodes orientées mouvement reposent sur des traitements ponctuels. Leur principe consiste souvent à éliminer des zones de l'image non affectées par le mouvement durant le déroulement de la séquence. Ces méthodes sont basées sur l'exploitation des différences entre images.

De nombreux chercheurs [Jain 79][Yashida 81][Yalamachili 82][Jain 85] ont utilisé l'analyse des différences d'images pixel à pixel pour détecter les objets en mouvement, les séparer du fond dans une première étape, afin d'interpréter et d'estimer leurs déplacements dans une seconde étape avec les hypothèses suivantes :

- les objets ont une forme polygonale et/ou curviligne et le niveau de gris de leur intérieur est homogène.
- la comparaison entre deux images successives d'une séquence indique les changements résultant du mouvement de ces objets dans la scène observée avec une caméra fixe.

Ces méthodes présentent des limitations car elles manquent d'efficacité dans le cas où l'on travaille sur de petits objets ou lorsque les déplacements sont faibles entre les images successives. Certains auteurs proposent d'utiliser la technique des différences cumulées [Jain 81] qui établit un historique partiel de tous les changements détectés dans la scène. Ce type d'analyse permet de pallier les inconvénients précédents.

Une description générale des méthodes utilisant la technique des différences d'images est donnée dans [Cabestaing 92].

### II-2.3 Méthodes K2D

Pour extraire l'information tridimensionnelle, les méthodes K2D utilisent plusieurs images 2D, différentes les unes les autres, saisies selon trois situations possibles :

1. Une caméra mobile, la scène étant fixe [Williams 80].
2. Une caméra fixe, les objets étant mobiles [Cheng 80].
3. Plusieurs caméras fixes [Levine 73].

La stéréovision est la méthode K2D la plus répandue. Elle permet d'évaluer l'information profondeur à partir de plusieurs images de la scène prises par des caméras fixes sous des angles de vue différents. Son principe consiste à chercher les pixels homologues dans les différentes images, puis à localiser dans l'espace le point correspondant par triangulation.

### II-2.4 Conclusion

Un des objectifs de la vision artificielle est de reconstruire la structure tridimensionnelle des scènes observées par une ou plusieurs caméras. De nombreuses études ont été réalisées dans ce sens et ont permis de développer trois types de méthodes :

- Les méthodes basées sur la vision monoculaire.
- Les méthodes basées sur l'analyse du mouvement.
- Les méthodes K2D.

Faisant partie de la famille K2D, la stéréovision est la méthode la plus utilisée. La suite de ce chapitre est consacrée à une description détaillée des démarches habituellement employées dans le cadre de cette méthode.

## II-3 STEREOVISION

### II-3.1 Définitions

On désigne par stéréovision, ou stéréoscopie, l'ensemble de tous les procédés qui permettent d'engendrer la structure tridimensionnelle d'une scène à partir de plusieurs images vidéo de celle-ci, prises sous des angles de vue différents. D'ailleurs, le mot « stéréoscopie » qui dérive du grec « stereos », solide, et « skopein », voir, signifie « voir en relief ».

Le système stéréoscopique le plus courant et le plus simple n'utilise que deux images. Il s'agit alors de la stéréovision binoculaire. Certains auteurs préconisent l'utilisation de trois caméras pour lever certaines ambiguïtés en cas d'occlusions. On parle alors de la stéréovision trinoculaire.

### II-3.2 Applications de la stéréovision

Les domaines d'application de la stéréovision sont nombreux. Nous pouvons citer par exemple :

- La reconstruction de terrains à partir d'images satellitaires.
- La modélisation de scènes d'intérieur ou d'extérieur.
- L'analyse d'images médicales et biomédicales.
- La perception dynamique pour robots mobiles.
- Le contrôle de production en 3D.
- La détection d'obstacles.

## II-4 STEREOVISION BINOCULAIRE

La stéréovision binoculaire est un processus qui, à partir de deux images représentant la même scène sous des angles différents, permet de retrouver la structure spatiale de celle-ci. Comme chez l'homme, la perception du relief par stéréovision binoculaire est principalement assurée par exploitation du décalage existant entre les deux images. Ce décalage, appelé disparité stéréoscopique, ne peut être évalué qu'en établissant des correspondances entre les deux images.

### II-4.1 Décomposition du problème

Le problème essentiel de la stéréovision binoculaire est donc de trouver dans les deux vues les images des mêmes entités physiques de la scène observée. Il faut donc apparier les deux images. Sans contraintes particulières, une mise en correspondance point à point des images n'est pas une solution idéale. En effet, pour des images de taille courante de  $512 \times 512$  pixels, on dénombre  $512^4$ , soit 68 milliards couples de points à tester. De plus, les points images ne sont pas tous significatifs. On procède donc à l'extraction de primitives telles que points remarquables, contours, régions, etc. Ces primitives-image (2D) doivent correspondre à des primitives-objet (3D) dans la scène. Ainsi, apparier les images revient à apparier les primitives-image. Constituant une description de la scène perçue, les primitives-objet peuvent être reconstruites en utilisant des calculs de triangulation géométrique sur les paires de primitives-image homologues. Ces calculs nécessitent la définition de coordonnées dans les images et dans la scène, et donc de connaître la géométrie du capteur stéréoscopique.

Le processus de stéréovision peut donc être décomposé en quatre étapes :

- Choix d'un modèle géométrique pour les capteurs.
- Extraction des primitives de chaque image.
- Appariement des primitives.
- Triangulation géométrique et reconstruction tridimensionnelle de la scène.

## II-4.2 Principes généraux

Les stratégies utilisées pour aborder le problème de la stéréovision sont nombreuses. Elles dépendent généralement des réponses qu'elles apportent aux questions suivantes :

- Quelles primitives choisir pour l'appariement ? Le choix des primitives est l'un des aspects conditionnant le problème de l'appariement. En effet, certains types de primitives sont plus aisés à appairier que d'autres. De plus, ce choix détermine la représentation finale de la scène. Les différentes primitives utilisées en stéréovision seront exposées au § II-6.
- Quelles critères faut-il utiliser ? Afin de mesurer la qualité des appariements, un ou plusieurs critères de ressemblance doivent être optimisés. Le § II-7.1 exposera les différents critères utilisés en stéréovision.
- Quelles contraintes faut-il utiliser ? Afin de lever d'éventuelles ambiguïtés dans les appariements, un certain nombre de contraintes doivent être satisfaites. Au § II-7.2, nous présenterons les contraintes employées dans la littérature.
- Quelles méthodologies faut-il adopter ? C'est un autre aspect différenciant les algorithmes proposés pour résoudre le problème de la mise en correspondance. En effet, il n'existe pas actuellement une méthode suffisamment générale pour qu'elle puisse s'appliquer à une paire d'images stéréo indépendamment du contenu et du type de ces images. Nous allons donner, au § II-8, une liste détaillée des diverses méthodes généralement utilisées en stéréovision.

## II-4.3 Aspects fondamentaux

### II-4.3.1 Fonction d'appariement

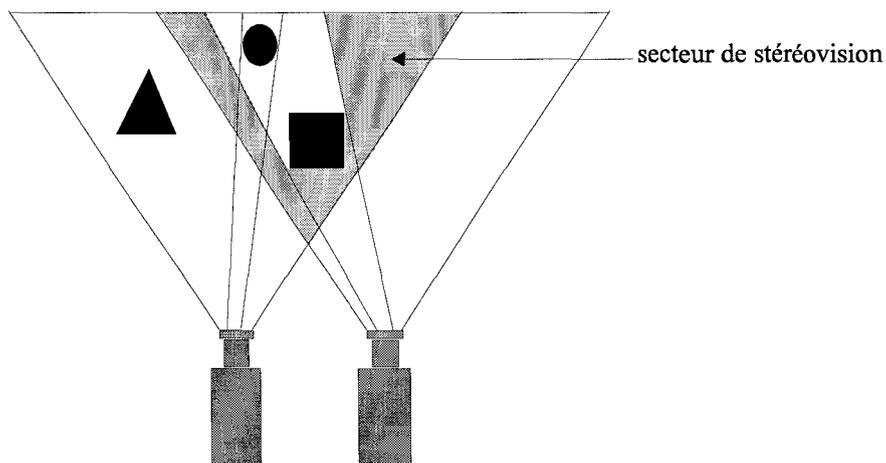
Supposons que nous disposons de deux images, gauche et droite, représentant chacune une projection différente d'une même scène 3D. Soit  $G$  (respectivement  $D$ ) l'ensemble des primitives de l'image gauche (respectivement de l'image droite). L'appariement entre ces deux ensembles de primitives peut être défini comme une fonction  $A$ , qui est une application de  $G \times D$  vers  $\{0, 1\}$  telle

que  $\mathcal{A}^{-1}(l)$  forme l'ensemble des couples de primitives appariées. La qualité de la fonction d'appariement  $\mathcal{A}$  se mesure :

- Au nombre de primitives appariées, qui doit être élevé.
- Au taux d'erreur, faux appariements, qui doit être réduit.

La recherche des appariements n'est pas la recherche d'une bijection entre les ensembles  $G$  et  $D$ . Seules certaines primitives, de chaque image, seront appariées. En effet, un élément d'une image peut ne pas être vu dans l'autre image et vice-versa, soit parce qu'il est caché par un autre élément, soit parce qu'il est dans le champ de vue d'une caméra et hors du champ de vue de l'autre, soit enfin à cause de phénomènes physiques tels que les ombres et les reflets qui dépendent des positions relatives de la source lumineuse, de l'objet observé et de la caméra.

Sur la figure II-1, l'objet "carré" qui se situe dans le secteur de stéréovision est vu simultanément par les deux caméras. Par contre, l'objet "triangle" se trouve hors du champ de vision commun, il n'est vu que par la caméra gauche. Sur la même figure, l'objet "rond" est vu par la caméra gauche, mais pas par celle de droite car il est occulté par l'objet "carré". Il y aura donc dans l'ensemble  $G$  des primitives qui n'auront pas de correspondant dans l'ensemble  $D$ .



*Figure II-1. Cas de primitives n'ayant pas de correspondant.*

De ces constatations, il découle donc une règle qui, comme on va le voir au § II-7.2, représente ce qu'on appelle la contrainte d'unicité : chaque primitive d'une image a au plus un correspondant dans l'autre image et vice versa.

### **II-4.3.2 Complexité du problème**

Le problème de mise en correspondance est de nature combinatoire : si le nombre de primitives de l'image gauche (resp. de l'image droite) est  $N_g$  (resp.  $N_d$ ), le nombre d'appariements possibles est alors  $N_g \times N_d$ . Parmi ces couples, seules  $N$  paires seront retenues avec, si l'on tient compte des remarques du paragraphe précédent,  $N \leq \min(N_g, N_d)$ . Il est donc fondamental de réduire la combinatoire, d'une part pour diminuer le nombre d'erreurs d'appariement, mais également pour limiter le temps des traitements. Cependant, la seule façon de réduire cette combinatoire est de mettre en oeuvre des contraintes, notamment la contrainte épipolaire (Cf. § II-7.2.1) qui résulte de la géométrie du capteur stéréoscopique. L'application de cette contrainte toute seule permet, comme on va le voir dans la suite, de réduire considérablement le nombre d'appariements possibles.

### **II-4.3.3 Dilemme précision-appariement**

Après la phase de mise en correspondance, les couples de primitives appariées permettent, par triangulation, de calculer les positions des primitives-objet correspondantes dans la scène réelle. Comme on va le voir plus loin, les calculs de triangulation sont liés à la géométrie du système de prise de vue, plus particulièrement à l'entre-axe, c'est à dire la distance séparant les centres optiques des deux caméras. La précision de la localisation est accrue si les points de vue sont nettement différents, c'est-à-dire lorsque l'entre-axe est important. Mais plus les caméras sont éloignées l'une de l'autre, plus les images stéréoscopiques sont dissemblables. L'apparence des surfaces et des contours des objets diffèrent alors sensiblement et le champ de vision commun est réduit. D'importantes déformations géométriques et occlusions sont également plus probables. Par conséquent, la tâche de mise en correspondance devient plus difficile à réaliser. Il faut donc trouver, selon l'application envisagée, un compromis entre précision et facilité d'appariement.

## **II-5 SYSTEME DE PRISE DE VUE**

### **II-5.1 Géométrie des capteurs**

La géométrie des capteurs est un élément important du dispositif de prise de vue stéréoscopique. Il s'agit soit de déterminer les paramètres caractérisant la disposition des caméras, soit au contraire d'imposer une configuration particulière aux capteurs en imposant les paramètres voulus (Cf. § II-5.3). Ces paramètres sont ensuite utilisés lors de la recherche des couples à apparier.

Considérons le schéma de la figure II-2 qui représente deux caméras. Pour qu'elles observent la même scène, les deux caméras doivent avoir une configuration telle que leur axes optiques convergent vers la scène observée. A chaque caméra est associé un repère. Soit  $P$  un point de la scène et soient  $p_g$  et  $p_d$  ses deux projections dans les images gauche et droite ( $p_g$  et  $p_d$  sont dits des points homologues). Nous pouvons donc écrire l'équation de la droite passant par le centre focal  $F_g$  de la caméra gauche et le point  $p_g$ . De même on peut écrire l'équation de la droite passant par le centre focal  $F_d$  de la caméra droite et le point  $p_d$ . L'intersection de ces deux droites donne, par triangulation géométrique, le point  $P$  de la scène réelle. Afin de pouvoir calculer cette intersection et donc de déterminer la position de  $P$ , il faut pouvoir exprimer les deux équations des droites  $F_g p_g$  et  $F_d p_d$  dans un même repère. On choisit en général un repère associé à la scène. Ici apparaît la nécessité de connaître les relations géométriques permettant de passer du repère lié à la caméra gauche au repère lié à la caméra droite, ainsi que celles permettant de passer des repères caméras au repère associé à la scène. La détermination de ces relations s'effectue lors de la phase de calibrage du système de prise de vue stéréoscopique.

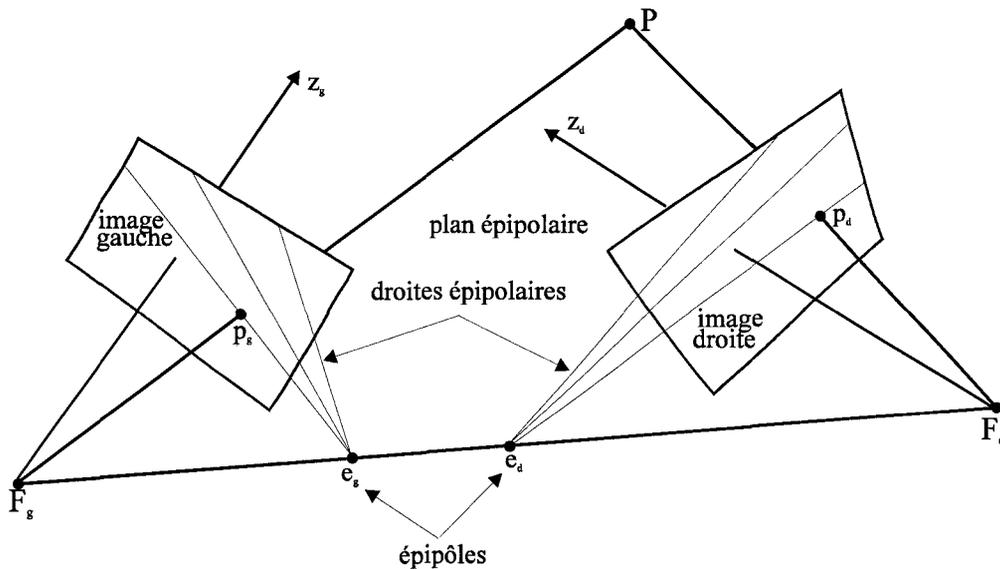


Figure II-2. Configuration générale d'un capteur stéréoscopique.

### II-5.2 Droites épipolaires

Lorsque les paramètres extrinsèques du capteur stéréoscopique qui définissent la position et l'orientation de chaque caméra par rapport à la scène sont connus, ceux-ci sont utilisés dans la phase de mise en correspondance. Ils font appel à la notion de droite épipolaire qui est une droite qui décrit le lieu des points d'une image pouvant correspondre à un même point  $p$  de l'autre image (Cf. Figure II-2). Il y a donc, pour chaque point de l'image gauche, une droite épipolaire

dans l'image droite et, réciproquement, pour chaque point de l'image droite, il y a une droite épipolaire dans l'image gauche. De plus, deux points qui se trouvent sur une même droite épipolaire d'une image ont leurs correspondants, s'ils existent, sur une même droite épipolaire de l'autre image. Ces deux droites sont appelées droites épipolaires conjuguées. Dans chaque image, les droites épipolaires forment un faisceau de droites, et leur point commun s'appelle l'épipôle (Cf. Figure II-2). En fait, l'épipôle droit  $e_d$ , par exemple, n'est que la projection du centre focal  $F_g$  de la caméra gauche dans l'image droite. De la même manière on peut définir l'épipôle gauche  $e_g$ .

### II-5.3 Configuration géométrique idéale

Dans le cas d'une configuration quelconque du capteur stéréoscopique (Cf. Figure II-2), il est nécessaire de calculer pour chaque point d'une image l'équation de la droite épipolaire correspondante dans l'autre image. Cependant, il existe une configuration particulièrement intéressante : c'est lorsque les deux caméras sont disposées de telle façon que leurs axes optiques sont parallèles et lorsque la droite  $F_g F_d$  reliant les centres optiques est confondue avec les axes horizontaux des deux caméras (Cf. Figure II-3). Autrement dit, lorsque les droites épipolaires sont parallèles et les droites épipolaires conjuguées sont confondues. Dans ce cas, les deux épipôles se trouvent à l'infini.

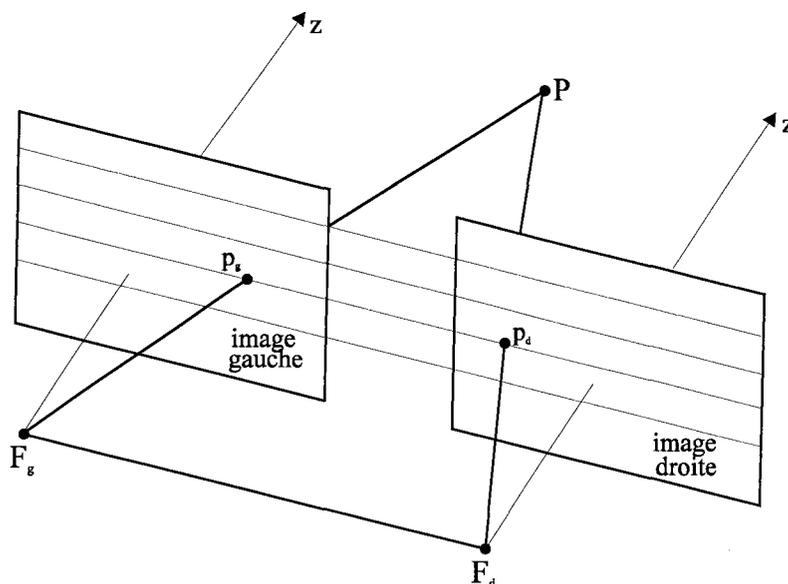


Figure II-3. Configuration particulière du système de prise de vue.

Les droites épipolaires jouent un rôle fondamental en vision stéréoscopique. En effet, lorsqu'on cherche pour un point d'une image un correspondant dans l'autre image, on peut limiter cette recherche le long de la droite épipolaire correspondante. Il est donc important d'avoir les

expressions mathématiques les plus simples pour ces épipolaires. Comme on ne peut pas toujours disposer les deux caméras pour obtenir la configuration idéale de la figure II-3, de nombreux auteurs [Heraud 93][Ayache 89] préfèrent partir d'une configuration quelconque (Cf. Figure II-2) et se ramener de façon analytique à la configuration idéale (Cf. Figure II-3) en effectuant une rectification épipolaire. Celle-ci consiste à appliquer aux images une transformation linéaire en coordonnées projectives de façon à obtenir une paire d'images stéréo coplanaires et parallèles à la droite passant par les centres de projection des deux caméras. Il faut toutefois noter que cette opération implique un accroissement de temps de calcul [Lasserre 96].

### II-5.4 Paramètres intrinsèques

Outre les paramètres extrinsèques qui définissent la géométrie du capteur stéréoscopique, en particulier la position et l'orientation de chaque caméra par rapport à la scène, d'autres paramètres doivent être choisis judicieusement afin d'optimiser, en fonction de l'application, le système de vision. Il s'agit des paramètres correspondant aux caractéristiques propres des caméras :

- Taille et résolution des capteurs.
- Distance entre les centres optiques.
- Distance focale des objectifs.

Les deux derniers paramètres interviennent lors des calculs de triangulation géométrique. Ils permettent aussi de déterminer :

- La taille de la scène observable à une distance donnée.
- La distance minimale de prise de vue permettant d'appliquer le principe de stéréovision.

### II-5.5 Triangulation géométrique

Les calculs de triangulation géométrique permettent, en fonction des positions dans les images des différentes primitives appariées, de déterminer la position des primitives-objet correspondantes dans la scène tridimensionnelle. Cette procédure utilise un modèle géométrique du système de prise de vue, décrivant la formation des images sur les capteurs, dont les caractéristiques sont décrites par les paramètres extrinsèques.

Considérons la configuration idéale du stéréoscope présentée au § II-5.3 (Cf. Figure II-4). Soient  $p_g(x_g, y_g)$  et  $p_d(x_d, y_d)$  les coordonnées des images du point  $P$  sur les capteurs gauche et droit, dans les repères relatifs aux caméras.

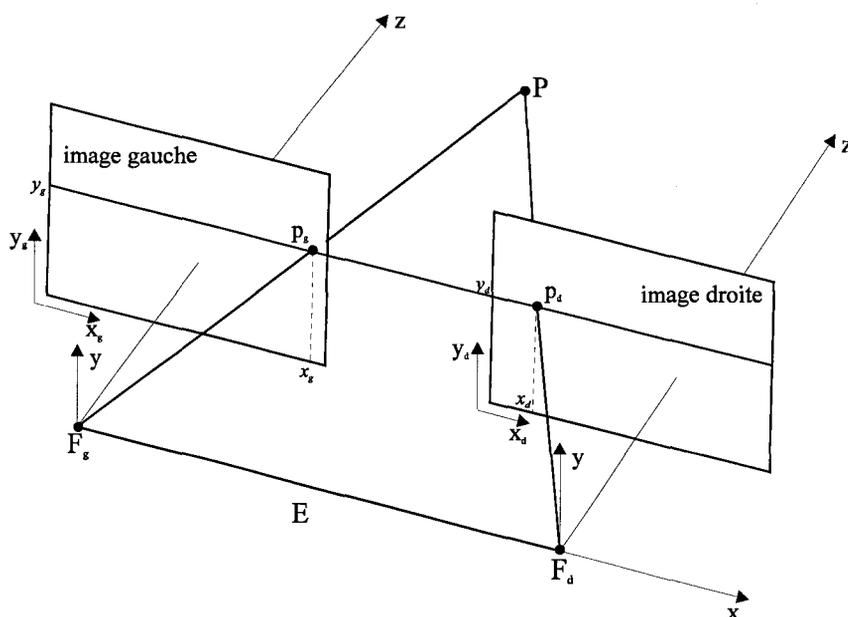


Figure II-4. Localisation tridimensionnelle.

Comme, dans ce cas, les droites épipolaires sont confondues ( $y_g = y_d$ ), la position de  $p_g$  dans l'image gauche peut se mesurer par rapport à celle de  $p_d$  avec un seul paramètre, la disparité, qui est par définition la différence des abscisses des deux points image. La disparité entre les deux points homologues  $p_g$  et  $p_d$  est notée :  $disp = x_g - x_d$ .

Ainsi, la position du point  $P$  dans le repère absolu ( $F_g, x, y, z$ ) peut être déterminée par les équations suivantes :

$$x_p = \frac{x_g \cdot E}{disp}$$

$$y_p = \frac{y_g \cdot E}{disp}$$

$$z_p = \frac{f \cdot E}{disp}$$

avec :

$E$  : l'entre axe, c'est à dire la distance entre les centres optiques  $F_g$  et  $F_d$ .

$f$  : la distance focale des objectifs, supposée la même pour les deux caméras.

### II-5.6 Reconstruction 3D

Il s'agit de fournir une représentation synthétique et formelle de la scène. Pour cela, on utilise les positions tridimensionnelles des primitives-objet calculées par triangulation. Il faut noter que la reconstruction 3D dépend du type de primitive choisi, qui détermine la représentation finale de la scène. De plus, la présence d'un grand nombre de primitives densifie la reconstruction 3D mais, en même temps, augmente la combinatoire. En revanche, lorsque les primitives ne sont pas nombreuses, la combinatoire est réduite mais la reconstruction est moins riche.

La reconstruction 3D a pour objectif de reconstituer les différents objets présents dans la scène. Sans relations entre les primitives-objet reconstruites et les objets physiques, la reconstruction est impossible. Ici apparaît la nécessité de disposer d'un modèle de la scène, c'est à dire d'un ensemble de connaissances sur les objets à reconstruire.

## II-6 LES PRIMITIVES

Avant la phase d'appariement, une représentation symbolique des images stéréoscopiques est nécessaire. Il s'agit d'extraire de chaque image des indices visuels, appelés primitives, caractérisant les entités présentes dans l'image. L'appariement des images revient donc à mettre en correspondance des primitives extraites de chacune d'elles.

La plus simple des primitives qui vient à l'esprit est sans doute le pixel lui-même. Mais un tel choix présente plusieurs inconvénients :

- L'information associée à un pixel, le niveau de gris, n'est pas suffisamment discriminante pour lever les ambiguïtés lors de l'appariement. Il y aurait donc de nombreux appariements erronés.
- L'image contient en général un nombre très important de pixels. Par conséquent, l'appariement de ces derniers conduirait à une combinatoire importante qui aurait comme effet un accroissement considérable des temps de calculs.

Il est donc essentiel de traiter préalablement les images à apparier afin d'en extraire un nombre limité de primitives, contenant d'une manière concise une information pertinente, afin de pouvoir faciliter la procédure de mise en correspondance et de diminuer les temps de traitement.

### II-6.1 Les propriétés requises

Afin d'optimiser le processus de stéréovision (qualité d'appariement, temps de traitement, bonne localisation), les primitives à apparier doivent vérifier certaines propriétés [Ayache 89]. Elles doivent être :

- Discriminantes : les primitives doivent avoir des attributs qui permettent de les discriminer afin de lever toute ambiguïté lors de la mise en correspondance.
- Denses : les primitives doivent être suffisantes en nombre pour représenter tous les objets de la scène.
- Compactes : l'image doit être représentée d'une manière concise afin de réduire la complexité algorithmique de la procédure de mise en correspondance.
- Précises : la qualité de localisation des objets physiques dépend de la position des primitives, d'où la nécessité de les extraire avec précision.
- Robustes : les primitives doivent être peu sensibles aux petites variations d'intensité dans l'image dues au bruit d'acquisition, de numérisation, etc.
- Intrinsèques : les primitives doivent correspondre à la projection, dans l'image, des objets physiques.

### II-6.2 Types de primitives

Le choix des primitives est un point fondamental en stéréovision. Il est conditionné d'une part par le type d'images à traiter mais également par l'information que l'on désire en extraire, c'est à dire la représentation finale de la scène 3D. Après la mise en correspondance, la phase de modélisation peut être immédiate si le résultat de la triangulation des primitives appariées donne directement les modèles tridimensionnels utiles. Toutefois, des étapes intermédiaires de mise en forme des données sont parfois nécessaires suivant le niveau de représentativité des primitives-objet reconstruites. La suite de ce paragraphe expose les principaux types de primitives utilisés en stéréovision.

#### II-6.2.1 Les points d'intérêt

Comme nous l'avons déjà mentionné, l'un des inconvénients lié à l'utilisation du pixel comme primitive est l'augmentation de la combinatoire de la procédure de mise en correspondance. Cependant, certains auteurs ont proposé de limiter le traitement à un certain nombre de pixels, appelés points d'intérêt, possédant des propriétés et des caractéristiques les distinguant de leurs voisins. Les attributs associés à ces points sont, comme pour les pixels, de

nature photométrique (intensité lumineuse, contraste, etc.). Les primitives de ce type sont les premières à avoir donné des résultats satisfaisants en stéréovision. Parmi les plus connus de ces points, nous pouvons citer :

- Les points de Moravec. Les points de Moravec sont les points autour desquels est observée une forte variation de la fonction niveau de gris de l'image [Moravec 77][Moravec 81]. Ces points correspondent généralement aux bords et aux coins des objets. La procédure de localisation de ces points consiste, dans un premier temps, à calculer les variances du niveau de gris dans le voisinage de chaque point, correspondant aux quatre directions verticale, horizontale et diagonales. Le minimum de ces valeurs est choisi pour caractériser le point correspondant. Ensuite, on balaye l'image et on ne conserve comme points d'intérêt que ceux dont la variance est maximale dans une fenêtre. Cette procédure sélectionne les points à fort contenu informationnel, tels que ceux correspondant à des coins. Elle permet d'éliminer les points appartenant à des régions trop homogènes ou faiblement texturées. On évite aussi les points se trouvant sur des contours lisses, c'est à dire pour lesquels la variance est faible dans le sens du contour.
- Les points de forte courbure. Pour détecter les points de forte courbure, une première approche consiste à mesurer la courbure en chaque point de l'image en travaillant directement sur la fonction niveau de gris. La mesure de courbure en un point de l'image est définie comme le produit de l'amplitude du gradient en ce point et du taux de variation du niveau de gris dans la direction de ce gradient. Un seuillage est ensuite utilisé pour ne garder que les points qui présentent une courbure importante. Les détecteurs les plus connus de ce type de points sont décrits dans la littérature [Zuniga 83][Kitchen 82][Noble 87]. Une deuxième approche consiste à extraire les contours en tant que chaînes codées pour ensuite ne chercher, dans ces chaînes, que les points présentant une forte courbure [Asada 86]. Deriche [Deriche 90] propose un nouveau concept en combinant les deux approches précédentes. Après extraction des points de contour, un algorithme de chaînage est utilisé afin d'obtenir des ensembles de points connectés. Pour chaque point des chaînes obtenues, la direction du contour, calculée à partir de la fonction niveau de gris de l'image, est utilisée pour déterminer la courbure. Les points de contour caractérisés par une forte courbure sont retenus comme candidats à l'appariement.

### *II-6.2.2 Les primitives de type contour*

En raison de leur représentativité, les primitives de type contour sont les plus utilisées en stéréovision. Elles représentent, dans l'image, les frontières des objets et les variations spatiales des propriétés géométriques et physiques de leurs surfaces (orientation, matière, texture, couleur, etc.). De cette façon, l'extraction des contours permet de mettre en évidence facilement les éléments significatifs d'une image.

Il faut cependant remarquer que la notion de contour est double. Elle englobe d'une part l'ensemble des points constituant le contour et d'autre part la modélisation des contours des objets à partir de ces points de contour.

#### *II-6.2.2.1 Points de contour*

Les points de contour s'obtiennent généralement en utilisant des techniques de filtrage (extrema locaux du gradient, passage par zéro du Laplacien, etc.) et de seuillage. Cependant, l'image des points de contour est souvent inexploitable. En effet, des points indésirables viennent la plupart du temps s'ajouter aux points de contour effectifs. D'autre part, certains points de contour ne sont pas détectés. De plus, le nombre de points de contour obtenus est souvent très important, ce qui rend les algorithmes de mise en correspondance inopérant. Les primitives de type points de contour ne sont donc pas utilisables à l'état brut et il est nécessaire de chaîner ces points afin de les rendre exploitables sous la forme de contours.

#### *II-6.2.2.2 Modélisation des contours*

La première étape pour modéliser les contours consiste à relier les points de contour voisins de proche en proche, pour former des contours définis comme des chaînes de points de contour connectés. Ces chaînes sont ensuite modélisées par approximation sous forme de segments, de polygones, d'arcs de cercles, etc. [Ayache 89][Horaud 88][Medioni 85][Chebaro 91].

Ainsi, les contours permettent de fournir une représentation relativement synthétique de l'information contenue dans l'image. Ils sont décrits par des attributs de type photométrique, mais aussi de type géométrique tels que longueur, orientation, etc. On peut leur associer également d'autres attributs plus globaux décrivant les relations structurelles qui les relient [Skordas 88a]. Toutefois, les contours présentent un inconvénient lié aux effets de perspective, les images du même contour vu par plusieurs caméras n'ont généralement pas la même forme. En pratique, l'utilisation des contours est surtout intéressante avec des objets polyédriques [Horaud 93].

### II-6.2.3 Les régions

Dans une image, une région est un ensemble de pixels adjacents, satisfaisant un prédicat qui habituellement indique une uniformité de certaines caractéristiques telles que le niveau de gris, la couleur, la texture, etc. Ce type de primitive peut être caractérisé par un ensemble d'attributs de nature photométrique (niveau de gris moyen, texture, etc.), géométrique (surface, forme) ou morphologiques (compacité, moments d'inertie, coordonnées du centre d'inertie). De plus, s'il s'agit d'images couleurs, on peut ajouter une autre information de type colorimétrique. L'utilisation des régions en stéréovision est avantageuse, car elle permet de diminuer les possibilités d'appariement et donc de réduire la combinatoire de la procédure de mise en correspondance. En contrepartie, comme pour les contours, leur appariement est souvent rendu difficile par le phénomène de distorsion sur les images saisies sous des angles différents.

Les surfaces des objets n'ont pas toujours un niveau de gris ou une couleur uniformes, mais peuvent présenter un certain motif ou texture. La segmentation d'images utilisant l'analyse de texture [Lumia 83][Mitchell 78][du Buf 90] est intéressante lorsqu'on choisit les régions comme primitives. En effet, quand il s'agit d'images fortement texturées, il est plus facile de les décrire par des régions de textures homogènes que par des contours qui, dans ce cas, seront très nombreux. En revanche, l'analyse de texture est souvent très lente en raison de l'utilisation d'outils tels que la transformée de Fourier ou les matrices de cooccurrence.

Pour extraire les régions d'une image, on distingue habituellement trois types de méthodes :

- Fusion de régions [Nagao 83][Brice 70][Horowitz 76] : les données initiales sont de très petites régions formées de pixels individuels ou de groupes de pixels adjacents ayant un niveau de gris ou une couleur identique par exemple. On regroupe alors les régions adjacentes jusqu'à ce qu'on ne puisse plus trouver deux régions adjacentes suffisamment semblables pour être regroupées.
- Partage de régions [Olhander 78][Watanabe 74][Weszka 78] : la donnée initiale est l'image entière. La méthode est récursive, on examine chaque région et on la divise éventuellement en un certain nombre de sous-régions qui sont elles-mêmes susceptibles d'être divisées de la même manière. On forme souvent un histogramme des niveaux de gris de la région pour savoir si celle-ci peut être partagée. Si l'histogramme contient plusieurs pics, la région contiendra vraisemblablement l'image de plusieurs surfaces et devra donc être partagée. On peut aussi, au lieu

d'utiliser l'histogramme, calculer la variance des niveaux de gris de la région : celle-ci sera partagée si cette variance dépasse un certain seuil.

- Fermeture de contour [Deriche 87a] : il s'agit de détecter les régions en identifiant les contours fermés constituant leurs frontières. Cette approche consiste, dans un premier temps, à extraire les points de contour dans l'image, puis à faire un suivi de ces points en les reliant. Enfin, grâce à un algorithme de fermeture de contours, les régions sont obtenues par détection des composantes connexes maximales n'incluant pas de points de contours.

Pour plus de détails sur ces techniques de segmentation d'images en régions, le lecteur intéressé pourra se rapporter à [Benamrane 94].

Notons au passage que l'utilisation de la couleur est peu fréquente, car elle nécessite du matériel coûteux, et implique de travailler sur trois plans images, ce qui rend les traitements plus lourds. Mais l'information couleur pourrait être d'un grand intérêt lors de l'appariement de primitives. L'analyse d'images en couleur est devenue, depuis quelques années, l'un des domaines sur lequel de nombreux chercheurs se sont penchés, mettant au point de nombreux outils qui pourraient servir en stéréovision [Ultré 96][Lee 91][Claxton 87][Di Zenso 86].

#### ***II-6.2.4 Les contours actifs***

Initialement introduits par Kass et al. [Kass 87], les modèles de contours actifs, ou "snakes", permettent, par rapport aux primitives précédentes, une représentation plus globale des objets. L'idée de base consiste à approcher le contour d'un objet par une courbe continue, fermée ou non. Celle-ci est constituée d'un ensemble de points de contrôle, appelés snaxels.

Le processus de recherche du contour s'effectue en deux étapes. La première étape concerne l'initialisation, souvent interactive, du contour actif au voisinage du contour à extraire. La deuxième étape consiste à faire évoluer le contour actif jusqu'à ce qu'il coïncide avec le contour recherché. L'évolution du contour actif est régie par la minimisation de son énergie, composée de deux termes, l'un correspondant à une énergie externe, l'autre à une énergie interne. L'énergie externe est destinée à attirer le contour actif vers le contour de l'objet. L'énergie interne est destinée à donner une certaine régularité au contour actif.

Dans sa thèse, Selsis [Selsis 96] a utilisé les contours actifs pour suivre et localiser, par stéréovision, des objets en mouvement. En faisant l'hypothèse que tous les objets en mouvement pénètrent dans les champs des caméras par leurs périphéries, elle a proposé une initialisation

automatique en utilisant un contour actif placé à la périphérie des images. Ainsi, tout objet pénétrant dans le champ de chaque caméra déforme ce contour actif périphérique jusqu'à sa fermeture autour de l'objet. Une procédure de scission permet alors d'initialiser un contour actif propre à cet objet, afin de le suivre durant sa traversée du champ. Le contour actif périphérique retrouve, quant à lui, sa position initiale, afin de modéliser les nouveaux objets qui pénètrent dans la scène. Afin de localiser les objets en mouvement, elle a utilisé les contours actifs comme primitives lors de la phase de mise en correspondance stéréoscopique. Trois attributs sont retenus pour caractériser ces primitives : forme, surface et les coordonnées du barycentre.

L'utilisation des contours actifs, que l'on peut qualifier de primitives de haut niveau, pour la mise en correspondance stéréoscopique présente plusieurs avantages par rapports aux primitives plus élémentaires. Nous pouvons citer :

- Le nombre réduit de primitives candidates à l'appariement.
- Le calcul direct, à partir de leur formulation mathématique, des attributs forme et surface.
- La disparition de l'étape de reconstruction, puisque ces primitives sont considérées comme des modèles des objets physiques.

### **II-6.3 Conclusion**

La première étape d'un processus de stéréovision consiste en la segmentation des images à mettre en correspondance. Il s'agit de mettre en évidence des indices visuels ou primitives dans chaque image. Les techniques de segmentation d'images permettent l'extraction d'une gamme variée de primitives telles que : points, segments, régions, etc. Ces primitives sont ensuite appariées et utilisées en vue d'une reconstruction 3D des objets dans la scène. Leur choix est cependant crucial, car il détermine la représentation finale de la scène lors de l'étape de reconstruction. Les primitives choisies doivent posséder un fort contenu informationnel pour faciliter leur mise en correspondance.

## **II-7 CRITERES DE RESSEMBLANCE ET CONTRAINTES**

La mise en correspondance des primitives nécessite de définir des critères qui permettent de fournir des informations sur leur ressemblance. Cependant, l'utilisation des critères de ressemblance est souvent insuffisante pour décider si un appariement est correct ou non. En effet, une primitive d'une image peut ressembler à plusieurs primitives de l'autre image. Pour résoudre

ce problème d'ambiguïté, on fait appel à un certain nombre de contraintes. Celles-ci sont de deux types : contraintes locales ou contraintes globales. Les contraintes locales sont généralement utilisées avant toute vérification de ressemblance des primitives. Il s'agit de simplifier l'espace de recherche des correspondants possibles et ainsi d'éviter de nombreux calculs inutiles. Les contraintes globales permettent de vérifier la compatibilité des appariements obtenus par application d'un ou plusieurs critères de ressemblance.

### II-7.1 Critères de ressemblance

La ressemblance est une notion utilisée par la plupart des algorithmes de mise en correspondance. L'hypothèse sous-jacente à la ressemblance de deux primitives est que les attributs qui leurs sont attachés doivent être semblables si ces deux primitives se correspondent. L'idée de base consiste à apparier une primitive en la comparant avec ses correspondants possibles, de telle sorte que le couple qui se ressemble le plus sera retenu comme un couple de primitives homologues. Il faut donc, dans un premier temps, établir un critère qui permet de quantifier la ressemblance. Ce critère doit tendre vers une valeur limite pour une ressemblance parfaite, mais en fait cette valeur n'est jamais atteinte car deux primitives homologues ne sont jamais parfaitement identiques. Etant donnée une primitive, la recherche de son homologue consiste à sélectionner parmi les appariements possibles celui qui optimise le critère.

Selon le type de primitives utilisé, les critères peuvent être de deux natures :

- Critères photométriques. Ce sont les critères les plus utilisés lorsqu'il s'agit des primitives de type point. Le principe consiste à examiner les niveaux de gris sur des fenêtres de même taille, centrées sur les points que l'on cherche à apparier. La ressemblance de deux points peut alors se mesurer au moyen d'un coefficient de corrélation du contenu de ces deux fenêtres. Des exemples de critères de corrélation sont présentés dans [Heraud 93][Faugeras 93]. Il faut noter que la corrélation présente l'inconvénient de nécessiter de travailler sur des fenêtres de tailles identiques. Or, dans le cas d'images stéréoscopiques, les effets du phénomène de distorsion projective conduisent deux primitives homologues à être de tailles et de formes différentes. Pour pallier cette limitation, Ohta [Ohta 85] propose, dans le cas des signaux mono-dimensionnels, d'utiliser un critère de pseudo-corrélation qui prend en compte ces effets de perspective.

- **Critères géométriques.** Ces critères s'appliquent généralement aux primitives de haut niveau (contours, régions, etc.) caractérisées par des attributs géométriques (longueurs des segments, angles entre les segments, surfaces et périmètres des régions, etc.). Il est donc possible de déterminer différents critères prenant en compte ces caractéristiques géométriques. Pour mesurer la similarité de ces paramètres, on introduit des seuils de tolérance. Ces derniers peuvent être déduits de la géométrie épipolaire du capteur stéréoscopique [Faugeras 88].

## II-7.2 Contraintes stéréoscopiques

L'utilisation des critères de ressemblance n'est en général pas suffisante pour établir des appariements sans ambiguïté, une primitive de l'image gauche peut ressembler à plusieurs primitives de l'image droite et réciproquement. Comment peut-on donc décider que l'appariement d'une paire de primitives, gauche et droite, est correct ou non ? Pour répondre à cette question, on emploie un certain nombre de contraintes. Celles-ci sont de deux types. Le premier type sert à valider les appariements des primitives. Il s'agit généralement des contraintes épipolaire et d'orientation. Ces contraintes permettent de réduire l'espace de recherche des homologues : un critère de ressemblance n'est évalué pour une paire de primitives que si elles respectent ces deux contraintes. Le deuxième type de contraintes est destiné à lever les ambiguïtés entre les appariements fournis par application d'un ou plusieurs critères de ressemblance. Il s'agit de valider la compatibilité entre les appariements satisfaisant le premier type de contraintes. Parmi les principales contraintes de ce type, nous pouvons citer les contraintes d'ordre, d'unicité, de continuité de la disparité et de continuité figurale.

### II-7.2.1 La contrainte épipolaire

C'est la seule contrainte géométrique intrinsèque au capteur stéréoscopique. Elle est très souvent utilisée car elle simplifie la recherche des correspondants lors de l'appariement des primitives. Dans le cas des primitives de type point (Cf. Figure II-5), les droites épipolaires conjuguées se caractérisent par le fait que le correspondant d'un point de l'une se trouve, s'il existe, exclusivement sur l'autre. On voit ici l'importance de cette contrainte qui permet de limiter la recherche des homologues à une droite au lieu de la totalité de l'image.

Cette contrainte peut être appliquée aussi aux primitives de type contour, modélisées généralement par des segments de droite. Soit  $S_g$  un segment de contour de l'image gauche et soit  $S_d$  un segment de contour de l'image droite. Le couple de segments  $(S_g, S_d)$  vérifie la contrainte

d'épipolarité si et seulement si la droite épipolaire conjuguée à celle passant par le milieu de  $S_g$  coupe  $S_d$ , et inversement. Dans le cas d'une configuration géométrique idéale du capteur (Cf. § II-5.3), où les droites épipolaires sont parallèles et confondues, cette contrainte consiste à vérifier que l'ordonnée du milieu de  $S_g$  est comprise entre celles des extrémités de  $S_d$ , et inversement. Ainsi, sur la figure II-6, les segments  $S_{g1}$  et  $S_d$  vérifient la contrainte épipolaire. Par contre elle n'est pas vérifiée pour les segments  $S_{g2}$  et  $S_d$ .

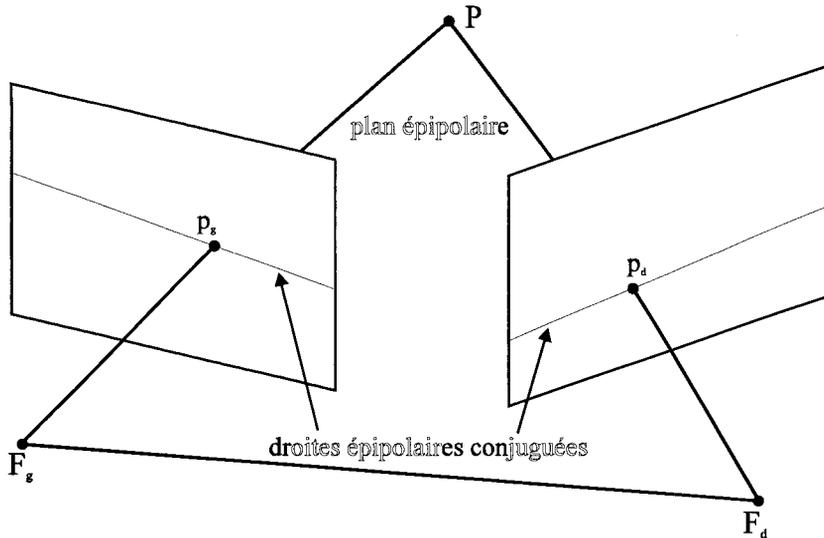


Figure II-5. Contrainte épipolaire dans le cas des primitives de type point.

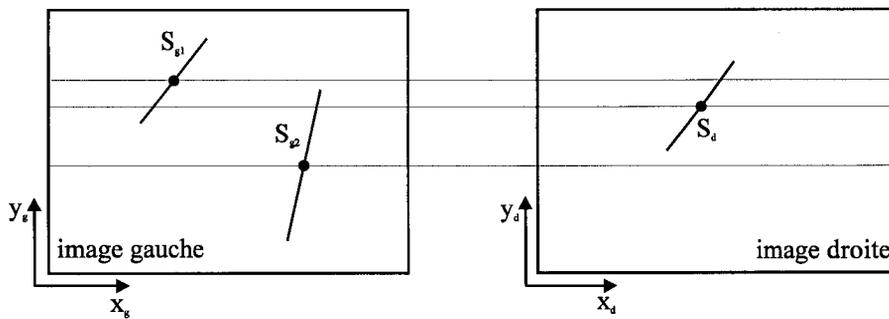


Figure II-6. Contrainte épipolaire simplifiée dans le cas des segments de droite.

**II-7.2.2 La contrainte d'orientation**

En plus de sa position, une primitive possède aussi une information relative à son orientation par rapport au repère de l'image. En stéréovision, la contrainte d'orientation impose aux primitives candidates à l'appariement de présenter des orientations semblables.

Pour les segments de droite, les orientations sont définies par l'angle entre le vecteur directeur associé au segment et un des axes du repère de l'image. Dans le cas des primitives de type point, le vecteur directeur est celui de la tangente au contour au point considéré. Lorsqu'il s'agit d'apparier les régions, cette contrainte peut être utilisée en considérant les directions des axes principaux et secondaires, mais les déformations des régions, dues aux effets de perspective, peuvent rendre cette contrainte inapplicable.

La compatibilité d'orientation de deux primitives gauche et droite se mesure par une simple différence de leurs orientations respectives. Ainsi, l'appariement de deux primitives sera rejeté si cette différence dépasse une valeur limite [Horaud 93].

Il faut remarquer qu'il s'agit d'une contrainte locale, car elle n'utilise pas de relation entre les primitives de la même image. Cependant, cette contrainte est généralement utilisée, comme la contrainte épipolaire, pour limiter l'espace de recherche des correspondants d'une primitive.

### II-7.2.3 La contrainte d'ordre

La contrainte d'ordre suppose que l'ordre des primitives appariées le long des droites épipolaires conjuguées est le même dans chacune des deux images stéréoscopiques (Cf. Figure II-7). Elle sert à valider la compatibilité des ordres des appariements. Il s'agit d'une contrainte globale, car elle est appliquée entre tous les couples de primitives. Elle permet de lever certaines ambiguïtés d'appariement.

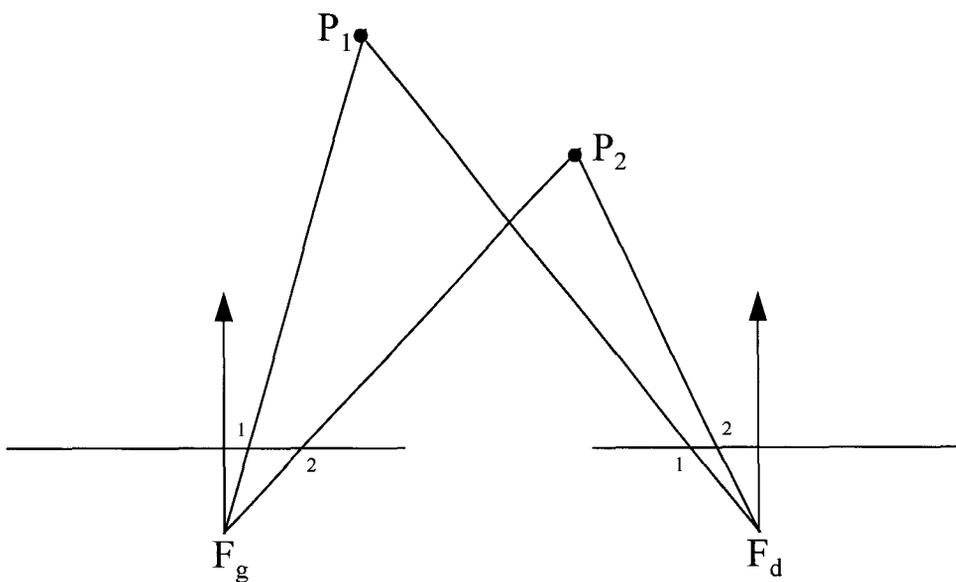


Figure II-7. Contrainte d'ordre.

Si cette contrainte est toujours vérifiée lorsqu'il s'agit d'apparier des images aériennes, elle est parfois transgressée dans les scènes d'intérieur où l'on peut rencontrer des situations telles que celle illustrée par la figure II-8 [Ayache 89][Horaud 93].

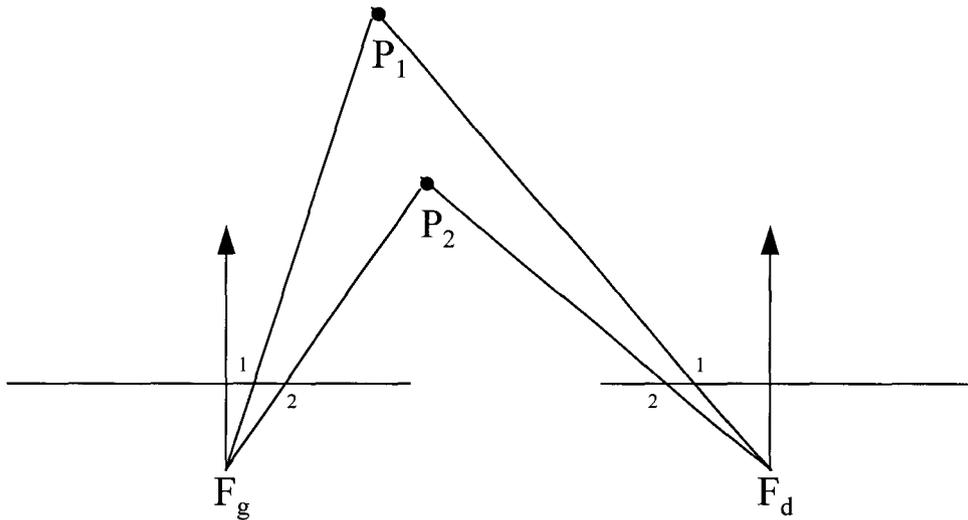


Figure II-8. Mise en défaut de la contrainte d'ordre.

#### II-7.2.4 La contrainte de continuité de la disparité

La contrainte de continuité de la disparité suppose que la scène est formée d'objets physiques dont on peut approcher la structure spatiale par des primitives géométriques, ou primitives objet, correspondant aux primitives visuelles des images [Ayache 89]. Si on suppose que la surface des objets est suffisamment régulière, et que la densité des primitives géométriques est suffisamment grande, les objets seront représentés par des sous-ensembles de primitives voisines dans l'espace que l'on peut parcourir de manière régulière, presque *continue*. De cette manière, la variation de la profondeur entre deux primitives objet voisines est faible. Comme la profondeur est directement liée à la disparité (Cf. § II-5.5), celle-ci doit varier continûment le long des primitives image. Certains auteurs ont exprimé cette contrainte à l'aide du gradient de disparité [Ayache 89][Horaud 93], d'autres ont établi une caractérisation analytique grâce à la notion du gradient de disparité limite [Pollard 85].

La contrainte de continuité de la disparité doit être utilisée dans un raisonnement global, et non pas local, afin d'obtenir, à partir des primitives appariées, un champ de disparité variant continûment. Ceci suppose, bien sûr, que la densité des primitives modélisant la scène est assez grande pour obtenir un champ exploitable.

### II-7.2.5 La contrainte d'unicité

Comme on l'a déjà vu au § II-4.3.1, la fonction d'appariement n'est pas une bijection entre l'ensemble des primitives de l'image gauche et l'ensemble des primitives de l'image droite. En d'autres termes, certaines primitives d'une image n'auront pas de correspondants dans l'autre image et vice-versa. Tenant compte de cet aspect fondamental de la stéréovision, la contrainte d'unicité exprime le fait qu'une primitive d'une image a au plus un correspondant dans l'autre image, et inversement.

Il faut cependant noter que cette contrainte suppose qu'il n'existe aucun couple de points 3D alignés avec le centre optique d'une des caméras. En effet, comme le montre la figure II-9, les points  $P_1$  et  $P_2$  ont la même image  $p_g$  sur le capteur gauche et deux images distinctes  $p_{1d}$  et  $p_{2d}$  sur le capteur droit. Dans ce cas, le point  $p_g$  pourra être apparié avec soit  $p_{1d}$ , soit avec  $p_{2d}$ .

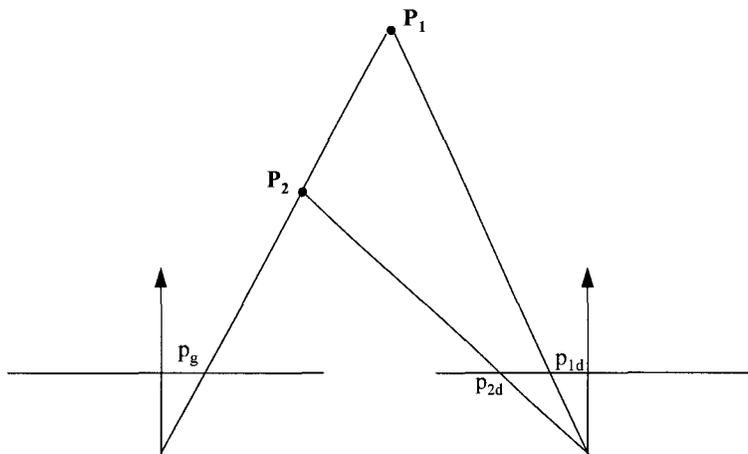


Figure II-9. Transgression de la contrainte d'unicité.

Si l'on considère ces deux appariements comme compatibles, on augmente la difficulté de la mise en correspondance, puisque, excepté pour des alignements tels que celui de la figure II-9, un point d'une image devrait avoir au plus un correspondant dans l'autre image. En pratique, pour imposer la contrainte d'unicité qui facilite le processus d'appariement, on exclut ce type d'alignement accidentel. Ceci est justifié en supposant l'opacité des points objets, ce qui conduit à ne considérer lors d'un alignement avec le centre optique que le point physique le plus proche de la caméra. Dans l'exemple de la figure II-9, on considère que le point objet  $P_2$ , le plus proche du capteur gauche, cache le point objet  $P_1$ . Ainsi le point  $p_g$  est l'image du point objet  $P_2$ . Le point objet  $P_1$  n'a pas d'image sur le capteur gauche, et alors le point  $p_{1d}$  n'a pas de correspondant.

### **II-7.2.6 La contrainte de continuité figurale**

L'utilisation de la contrainte de continuité figurale est intéressante lorsque les primitives à mettre en correspondance se trouvent le long des contours des objets. Si on suppose que la scène est constituée d'objets continus par morceaux, l'ensemble des primitives qui les modélisent doivent présenter une continuité identique. L'utilisation de cette contrainte impose donc un traitement régulier des droites épipolaires du couple d'images stéréoscopiques à apparier. La contrainte de continuité figurale permet alors de propager des hypothèses de mise en correspondance d'une droite épipolaire à l'autre : les primitives appariées sur une droite épipolaire sont utilisées pour confirmer les appariements de la droite épipolaire suivante. Il s'agit alors de privilégier, sur la droite épipolaire courante, les appariements qui prolongent un contour mis en évidence par les appariements établis durant le traitement des droites épipolaires précédentes.

La contrainte de continuité figurale doit être utilisée dans un contexte global, car, localement, elle peut générer des appariements erronés [Horaud 93]. Elle doit donc prendre en compte l'ensemble des appariements de la droite épipolaire courante et l'ensemble des contours déjà formés à partir des appariements des droites épipolaires précédentes.

### **II-7.3 Conclusion**

Nous avons vu que la mise en correspondance des primitives nécessite de définir un ou plusieurs critères afin d'évaluer la ressemblance entre les primitives. Il s'agit d'apparier une primitive en la comparant avec ses correspondants possibles, de telle sorte que le couple qui se ressemble le plus soit retenu comme un couple de primitives homologues. Cependant, l'usage des seuls critères de ressemblance pour apparier les primitives de deux images stéréoscopiques n'est pas suffisant et laisse subsister des ambiguïtés d'appariement. Il est donc nécessaire d'introduire un certain nombre de contraintes qui doivent coopérer entre elles pour lever certaines ambiguïtés. Nous avons vu qu'il existe deux types de contraintes : les contraintes locales et les contraintes globales. Appliquées avant toute vérification de ressemblance, les contraintes locales servent à valider les appariements afin de simplifier l'espace de recherche des homologues. Dans le but de lever certaines ambiguïtés, les contraintes globales sont destinées à valider la compatibilité des appariements obtenus par application d'un ou plusieurs critères de ressemblance.

Il faut toutefois noter que ces contraintes, mise à part la contrainte épipolaire qui est intrinsèque au stéréoscope, représentent des hypothèses sur la scène observée. Cependant, elles doivent être utilisées avec précaution et être sélectionnées en fonction du type d'images traitées.

## II-8 MISE EN CORRESPONDANCE

Dans un processus stéréoscopique, la tâche la plus difficile et la plus délicate est celle concernant la mise en correspondance des primitives extraites de deux images stéréo. Cependant, il n'existe pas, jusqu'à présent, une méthode assez générale pour résoudre ce problème et qui puisse s'appliquer à une paire d'images stéréoscopiques sans tenir compte de leur type, de leur contenu et de la façon dont elles sont traitées. D'un point de vue algorithmique, les méthodes développées comportent généralement une forme ou une autre de recherche des meilleurs appariements dans un ensemble fini. La qualité d'un algorithme de mise en correspondance se mesure au nombre de primitives appariées, qui doit être le plus élevé possible, et au taux d'erreur, dû à de faux appariements, qui doit être aussi faible que possible (Cf. § II-4.3.1).

Nous allons décrire les principales méthodes de mise en correspondance proposées dans la littérature.

### II-8.1 Méthodes corrélatives

Les méthodes corrélatives sont utilisées généralement pour mettre en correspondance les primitives de type point. La recherche des appariements est basée sur un critère de ressemblance photométrique : elle consiste à considérer une fenêtre autour du point que l'on veut apparier, puis à chercher dans l'autre image le point dont la fenêtre, de même taille que la précédente, est la mieux corrélée. Avant chaque mesure de ressemblance, la contrainte épipolaire est utilisée afin de réduire le nombre de candidats à l'appariement. Après optimisation du critère de ressemblance sur l'ensemble des appariements possibles, certaines contraintes globales (ordre, unicité, continuité de la disparité) sont ensuite utilisées pour lever certaines ambiguïtés. Plusieurs auteurs [Moravec 80][Gennery 80][Tsai 86][Castan 84] ont utilisé les algorithmes de corrélation sur des primitives particulières (points de Moravec, points correspondant aux passages par zéro du Laplacien, etc.).

Le choix de la taille des fenêtres est crucial. Trop petites, elles ne contiennent pas suffisamment d'information pour permettre un appariement fiable. Trop grandes, elles peuvent

contenir des éléments appartenant à des objets différents situés à des distances différentes. Afin de pallier ce défaut, Kanade [Kanade 94] propose une solution en utilisant une fenêtre adaptative.

Les méthodes corrélatives sont simples à mettre en oeuvre. Elles présentent toutefois deux limites importantes :

- Sensibilité aux distorsions de perspective : un effet de perspective peut influencer les coefficients de corrélation soit par la différence de luminosité apparente des surfaces suivant l'angle de prise de vue, soit par le fait qu'on corrèle des fenêtres de même taille alors qu'en réalité deux zones qui se correspondent n'ont pas les mêmes tailles. Pour contourner ce problème, Ohta [Ohta 85] utilise un critère de pseudo-corrélation.
- Impossibilité à détecter les occlusions : une méthode corrélative fournit toujours, pour chaque point, un correspondant qui présente la meilleure corrélation, même si elle est faible. Or, lors d'une occlusion, un point d'une image peut ne pas avoir de correspondant dans l'autre image. L'utilisation d'un seuil pour ne retenir que les coefficients de corrélation suffisamment élevés peut conduire à éliminer en même temps certains bons appariements.

### II-8.2 Méthodes hiérarchiques

Ces méthodes, dites aussi pyramidales, consistent, dans un premier temps, à représenter les images à plusieurs niveaux de résolution, du niveau le plus "fin", correspondant à la résolution du capteur, au niveau le plus "grossier", correspondant à une basse résolution. Il s'agit d'établir, à partir d'une paire d'images stéréoscopiques, deux pyramides d'images. Chaque pyramide contient l'image initiale ainsi que ses différents niveaux de résolution. Pour construire une pyramide, on réduit une image en moyennant par exemple la valeur des niveaux de gris des pixels dans une zone  $N \times N$ . Ainsi pour  $N = 2$ , une zone  $2 \times 2$  d'une image est remplacée par un pixel dans l'image de résolution inférieure, et ainsi de suite. Supposons qu'on dispose d'une paire d'images stéréo de taille  $1024 \times 1024$ . Chaque pyramide, gauche et droite, contient donc six niveaux, du niveau de haute résolution  $1024 \times 1024$  au niveau de basse résolution désiré, par exemple  $32 \times 32$  :  $1024 \times 1024$ ,  $512 \times 512$ ,  $256 \times 256$ ,  $128 \times 128$ ,  $64 \times 64$  et  $32 \times 32$ .

Avant la phase de mise en correspondance, on procède à l'extraction des primitives, généralement de type point, qui seront les candidats à l'appariement. Soit un point de l'image gauche haute résolution que l'on veut apparier. Les coordonnées de ce point sont calculées dans chaque image de la pyramide. On établit ensuite une correspondance de ce point au niveau de

résolution le plus faible à l'aide de la contrainte épipolaire et d'un critère de ressemblance. La prochaine étape consiste à parcourir la pyramide droite de l'image basse résolution vers l'image initiale. Le point sélectionné dans l'image basse résolution correspond à une zone  $2 \times 2$  dans l'image de niveau supérieur. L'intersection de cette zone avec la ligne épipolaire correspondante à ce niveau constitue la zone d'initialisation pour la recherche du point le plus ressemblant. Ce processus est itéré jusqu'au niveau de résolution le plus "fin" qui correspond à l'image initiale.

Ce type de méthode a été utilisé par Moravec [Moravec 81] pour la perception de l'environnement d'un robot mobile. Shen [Shen 86] a proposé d'apparier les passages par zéro du Laplacien en utilisant les images binaires du Laplacien (BLI) à trois niveaux de résolution ( $256 \times 256$ ,  $128 \times 128$ ,  $64 \times 64$ ). Il faut noter que les méthodes pyramidales sont très bien adaptées pour la reconstruction de terrain à partir de deux images aériennes [Hannah 84][Hannah 89] dont la résolution peut aller jusqu'à  $2048 \times 2048$ .

### II-8.3 Méthodes basées sur la programmation dynamique

Le problème de l'appariement peut être posé sous forme de recherche d'un chemin optimal dans un graphe. Il s'agit au départ d'utiliser la contrainte épipolaire et de considérer à chaque fois deux droites épipolaires conjuguées. On établit ensuite un graphe représentant tous les appariements possibles. Ce graphe peut être matérialisé par une matrice à deux dimensions, où chaque dimension est définie par les primitives de chaque droite épipolaire (gauche et droite). A chaque sommet du graphe correspond un élément de la matrice qui représente un coût, déterminé à partir d'une mesure de ressemblance, indiquant la qualité de l'appariement de deux primitives. Un chemin du graphe consiste en une séquence de paires de primitives (gauche et droite). Le coût d'un chemin est calculé en fonction de celui affecté à chaque paire qui le constitue. L'objectif sera donc de trouver un chemin optimal, c'est à dire de coût minimum, dans ce graphe. La nature de la recherche d'un chemin optimal, qui est liée au principe de la programmation dynamique, impose deux contraintes :

- L'ordre des primitives homologues doit être respecté le long des deux droites épipolaires (contrainte d'ordre).
- Chaque primitive n'intervient qu'une seule fois (contrainte d'unicité).

Il est possible qu'une primitive d'une image n'ait aucun correspondant dans l'autre image, situation qui peut se produire en cas d'occlusion. Dans ce cas, le coût associé à une paire contenant une telle primitive est très faible voire nul, ce qui permet de traiter les cas d'occlusions.

De telles situations se traduisent par la présence, dans le chemin optimal, de transitions horizontales et verticales entre les éléments de la matrice. Par contre, les transitions entre deux couples de primitives appariées sont diagonales.

Plusieurs auteurs [Benard 84][Baker 81][Ohta 85] ont utilisé la programmation dynamique pour appairer des images stéréoscopiques en choisissant les points de contour comme primitives. La recherche des correspondances est faite entre droites épipolaires conjuguées. Outre les contraintes d'ordre et d'unicité, Ohta [Ohta 85] complète cette recherche en prenant en compte la contrainte de continuité figurale des contours d'une ligne image à l'autre (Cf. § II-7.2.6). Récemment, Lee [Lee 94a] a utilisé la programmation dynamique pour appairer des segments de droite. Dans [Oksenhendler 94], les auteurs ont proposé un processus auto-adaptatif de stéréovision. Dans les différentes phases du traitement, une nouvelle primitive, la déclivité, est utilisée. Celle-ci est définie, au niveau d'une ligne image, par un ensemble de pixels limité par deux extrema locaux adjacents de la fonction niveau de gris.

L'utilisation de la programmation dynamique fournit généralement de bons résultats, mais au prix d'un grand nombre de calculs qui rend les algorithmes plus ou moins lents suivant la complexité des images et la puissance du matériel disponible. Il faut également noter que la contrainte d'ordre, imposée impérativement par la programmation dynamique, limite son utilisation à des images où l'ordre des primitives est respecté dans la plupart des cas, comme les images aériennes par exemple.

#### **II-8.4 Méthodes d'étiquetage par relaxation**

Le principe de ce type de méthode consiste à établir un étiquetage de l'ensemble des données à traiter, c'est-à-dire à associer des étiquettes à des objets avec des coefficients de confiance. On part souvent d'un étiquetage ambigu, ce qui signifie que plusieurs étiquettes peuvent correspondre a priori à un même objet. Pour réduire l'ambiguïté, on est amené à faire "coopérer" les objets entre eux pour modifier les coefficients de confiance attribués aux associations objet-étiquette à l'aide d'un algorithme itératif de relaxation. On suppose que l'étiquetage d'un objet est d'autant plus satisfaisant et fiable qu'il est cohérent avec l'étiquetage de ses voisins. Dans ce but, on définit un critère global que l'on doit optimiser à chaque itération. Le lecteur intéressé trouvera dans [Davis 81][Hancock 90] un aperçu sur les techniques de relaxation.

Le problème de mise en correspondance peut être vu comme un problème d'étiquetage. Deux formulations peuvent être utilisées :

- Les primitives de l'image de référence sont les objets, les étiquettes sont les primitives de l'autre image.
- Les objets sont les paires de primitives appartenant aux deux images, et il n'y a que deux étiquettes :  $1$  (appariement) ou  $0$  (pas d'appariement).

Supposons qu'on utilise la première formulation. Initialement, un ensemble d'appariements entre les primitives gauches et droites qui définissent des associations objet-étiquette est sélectionné en respectant certaines contraintes, comme la contrainte épipolaire. Une mesure de confiance est ensuite affectée à chaque appariement. L'étape suivante est destinée à comparer les appariements de primitives voisines en utilisant des relations découlant des contraintes géométriques et/ou physiques. Les appariements ainsi que leurs coefficients de confiance sont ensuite modifiés itérativement pour réduire les inconsistances d'appariement entre les primitives voisines. Le processus itératif s'arrête quand les coefficients de confiance n'évoluent plus.

De nombreux auteurs ont proposé la relaxation comme méthode de mise en correspondance. Un algorithme "générique" est présenté dans [Ballard 82]. L'algorithme de Marr et Poggio [Marr 76] est initialisé par tous les appariements qui satisfont la contrainte épipolaire, auxquels sont associés des poids. A chaque itération et pour chaque appariement, un processus inhibiteur réduit le poids des appariements conflictuels, et un processus excitateur augmente le poids des appariements de disparités voisines. Au cours de la phase itérative, les appariements de poids insuffisants sont éliminés. Grimson [Grimson 81] a proposé d'apparier les points correspondant aux passages par zéro du Laplacien, en combinant la relaxation et une approche multirésolution : les appariements obtenus à une échelle grossière sont utilisés pour résoudre les conflits d'appariement à une échelle plus fine. Dans [Pollard 85][Pollard 86], Pollard et al. ont proposé un algorithme pour apparier les points de contour. L'algorithme est initialisé en utilisant les contraintes épipolaire et d'orientation, puis, pendant la phase itérative, les contraintes d'unicité et de continuité de la disparité sont imposées progressivement. L'originalité de leur approche réside dans la formulation de la contrainte de continuité de la disparité qui fait intervenir la notion de gradient de disparité limite entre deux couples de points appariés. D'autres auteurs ont proposé d'apparier les segments de droite provenant de l'approximation polygonale des contours [Medioni 85][Long 86]. Les contraintes épipolaire et d'orientation sont imposées initialement.

L'algorithme de relaxation utilise progressivement les contraintes d'unicité et de continuité de la disparité pour réduire les ambiguïtés. Plus récemment, Nasrabadi [Nasrabadi 92a] a utilisé la relaxation pour mettre en correspondance des segments incurvés. Pour initialiser son algorithme, il a appliqué la contrainte épipolaire aux centres de courbure des segments.

De manière générale, les méthodes basées sur la relaxation donnent de bons résultats grâce à la prise en compte globale de l'environnement sous forme de contraintes et en autorisant que certaines primitives n'aient pas d'homologues par des associations de très faibles coûts. La mise en correspondance est d'autant plus fiable que les primitives sont de haut niveau [Medioni 84], car elles sont moins sensibles, par exemple, aux variations d'éclairage. En revanche, la fiabilité des appariements est conditionnée par la manière dont les primitives sont obtenues. L'inconvénient majeur de la relaxation est lié à son caractère exhaustif qui nécessite un grand nombre de calculs pour déterminer les coefficients de confiance et de compatibilité, et pour remettre à jour, à chaque itération, l'ensemble de ces valeurs.

### **II-8.5 Appariement par isomorphisme de graphes**

Il s'agit de méthodes de description structurelle. Chaque image est décrite au moyen d'un graphe relationnel, où les noeuds correspondent aux primitives extraites de l'image et les arcs, ou arêtes, aux relations entre ces primitives. Ainsi, le problème de mise en correspondance peut être ramené à un problème d'isomorphisme de graphes. En d'autres termes, on est amené à trouver le meilleur appariement entre les noeuds et les arcs de deux graphes. Dans la pratique, les deux graphes ne sont pas identiques, et ceci pour un certain nombre de raisons parmi lesquelles on peut citer les occlusions, les effets de perspective, le non respect de la contrainte d'ordre, etc. [Horaud 93]. Cependant, le problème d'appariement des deux graphes n'est pas simplement un problème d'isomorphisme de graphes, mais plutôt un problème d'isomorphisme de sous-graphes maximal. Autrement dit, il faut chercher tous les sous-graphes qui se correspondent et sélectionner ensuite le couple de sous-graphes qui met en correspondance le plus grand nombre de noeuds.

Skordas [Skordas 88b] a utilisé cette approche pour appairer des segments de droite entre lesquels différentes relations sont définies, de type : "à gauche de", "à droite de", "colinéaire avec" ou "a une jonction commune avec". Pour chercher le meilleur appariement entre les deux graphes représentant les deux images à appairer, un graphe d'association, ou de correspondance, est construit de la manière suivante. Un certain nombre d'appariements potentiels est établi en

utilisant les contraintes épipolaire et d'orientation. Les noeuds du graphe d'association, représentant ces appariements, sont reliés entre eux dès qu'il y a compatibilité entre les appariements correspondants. Cette compatibilité est établie à partir des contraintes d'ordre, d'unicité et de continuité figurale ainsi que d'autres propriétés projectives. L'étape suivante consiste à chercher les ensembles de noeuds mutuellement compatibles, formant ainsi des cliques. La plus grande clique maximale, correspondant au plus grand nombre d'appariements, est ensuite retenue comme solution [Horaud 89]. Une clique est dite maximale si elle ne peut être incluse dans une autre clique.

D'autres travaux ont utilisé l'isomorphisme de graphes comme méthodologie de mise en correspondance. Shen [Shen 87a] et Zhao [Zhao 89] ont proposé d'apparier les graphes pyramidaux des images binaires du Laplacien. Dans [Cocquerez 87a] et [Cocquerez 87b], Cocquerez représente les images à apparier par des graphes relationnels où les noeuds correspondent aux régions extraites des deux images et les arcs aux relations d'adjacence. Après l'établissement d'un ensemble d'appariements potentiels, en utilisant des critères géométrique et photométrique, l'isomorphisme est optimisé en propageant les appariements initiaux à l'aide des relations de voisinage existantes entre les deux graphes. La même approche a été utilisée par Benamrane dans sa thèse [Benamrane 94] pour apparier les régions extraites d'images stéréoscopiques. Pour modéliser les images sous forme de graphes, elle a proposé d'utiliser des relations topologiques définies à partir des positions des rectangles exinscrits aux régions. Ces relations sont du type : "au dessus de", "au dessous de", "à gauche de", "à droite de", "masquée par" ou "adjacente à".

### II-8.6 Appariement par prédiction et vérification d'hypothèses

Le processus de mise en correspondance utilisant ce type de méthode se décompose en trois étapes indépendantes :

1. Prédiction d'hypothèses. Après avoir extrait de chaque image les primitives à apparier, cette première étape consiste à établir un ensemble d'appariements à partir des contraintes locales exclusivement, telles que la contrainte épipolaire et la contrainte d'orientation, et de critères de ressemblance géométriques et/ou photométriques.
2. Propagation d'hypothèses. Après avoir généré un certain nombre d'appariements hypothétiques lors de la phase de prédiction, la phase de

propagation a pour but d'étendre le nombre d'appariements. Il s'agit de propager chacune des hypothèses aux voisines des primitives appariées. Supposons qu'on émette l'hypothèse d'appariement des primitives  $(P_g P_d)$ . La propagation de cette hypothèse consiste à chercher pour les  $P_{g_i}$  au voisinage de  $P_g$  les appariements à des  $P_{d_j}$  au voisinage de  $P_d$ , en utilisant les contraintes de continuité de la disparité et d'unicité par exemple. D'une manière récursive, les nouvelles hypothèses générées sont elles aussi propagées.

3. Vérification d'hypothèses. Cette phase est destinée à comparer les appariements obtenus par propagation des hypothèses prédites. Les appariements retenus sont ceux qui correspondent au plus grand nombre d'hypothèses compatibles vis à vis de la contrainte de continuité de la disparité par exemple.

Cette approche a été utilisée par Ayache [Ayache 85] qui propose d'apparier des segments de droite. Une transformation géométrique rigide permettant de passer d'une image à l'autre est déduite d'un certain nombre d'hypothèses générées en appariant, par ressemblance, un ou plusieurs segments, sélectionnés parmi les plus longs d'une image, à des segments de l'autre image. Par propagation, on teste si cette transformation permet aussi d'apparier les segments voisins des premiers. Le nombre de segments ainsi appariés lors de chaque propagation confirme ou infirme la validité de cette transformation. D'autres méthodes semblables sont décrites dans [Ayache 89][Robert 86]. Nous pouvons citer aussi les travaux de Krotkov [Krotkov 90] qui intègre cette approche en appariant des segments dans un système stéréoscopique actif, dont on peut commander de nombreux paramètres (mise au point, orientation, zoom, etc.). Wrobel [Wrobel 88] procède, elle aussi, par prédiction et vérification d'hypothèses pour apparier les graphes d'adjacence des régions extraites des images stéréoscopiques. Pour guider la mise en correspondance des régions, elle apparie leurs contours en utilisant la programmation dynamique. Benamrane [Benamrane 94] a proposé aussi d'apparier les régions. Les images sont dans un premier temps modélisées sous forme de graphes relationnels. Après génération d'hypothèses, leurs propagation et leurs validation sont réalisées par la définition d'une compatibilité sur des relations topologiques entre les régions.

### II-8.7 Méthodes basées sur les réseaux de neurones

Les réseaux de neurones sont des modèles qui tentent d'imiter quelques fonctions du cerveau humain en reproduisant certaines de ses structures de base. Un réseau de neurones est un

ensemble de cellules, appelées neurones formels, reliées par des arcs orientés et pondérés, définissant des liens synaptiques entre les neurones, selon une architecture spécifique. On appelle poids de connexion, ou poids synaptique, la pondération d'un lien synaptique. Afin de déterminer son état, chaque neurone réalise un prétraitement sur les potentiels d'action qui lui parviennent. Chacun de ces potentiels d'action est une valeur numérique qui représente l'état du neurone qui l'a émis. Cet état est ensuite transmis, sous forme de potentiel d'action, aux neurones concernés via leurs liens synaptiques avec le neurone émetteur.

Dans un système connexionniste, la connaissance réside dans les poids synaptiques, calculés par apprentissage du réseau. L'apprentissage est l'ensemble des règles qui tentent de déterminer les valeurs optimales des poids synaptiques pour que le réseau accomplisse une certaine tâche. Le type d'apprentissage à appliquer à un réseau dépend de son architecture. Il existe deux grands types d'architecture :

- Les réseaux à couches. Dans une organisation par couche, les signaux arrivent sur des cellules sensibles, les neurones d'entrées, puis sont transmis vers un système de neurones cachés, puis aboutissent aux neurones de sorties qui déclenchent un comportement. L'apprentissage de ce type de réseau est effectué généralement par essais et erreurs selon un schéma de rétropropagation.
- Les réseaux entièrement connecté. Ils sont aussi appelés réseaux de Hopfield, en référence à l'auteur de ce modèle. Ils sont structurés de telle manière que chaque neurone soit relié à tous les autres et possède même un retour sur lui-même. Leur apprentissage, statique, est qualifié d'apprentissage par coeur. En effet, les poids des connexions peuvent être calculés directement à partir de l'ensemble des données que l'on veut faire apprendre au réseau. Au cours de son évolution, le réseau est décrit, à tout moment, par un état qui correspond à l'ensemble des réponses, ou sorties, de ses neurones. Le fonctionnement du réseau consiste à le faire évoluer, à partir d'un état initial, vers un état stable correspondant à une donnée mémorisée. Cet état est atteint lorsque les réponses des neurones du réseau n'évoluent plus. Dans le chapitre VI, nous décrirons plus en détail ce type de réseau, et plus particulièrement son utilisation pour minimiser une fonction quadratique.

Pour plus de détails sur les différentes architectures des réseaux et leurs techniques d'apprentissage, le lecteur intéressé pourra consulter [Jodouin 94].

De nombreux auteurs ont utilisé les réseaux de neurones pour résoudre le problème de mise en correspondance stéréoscopique.

Dès 1976, Marr et Poggio [Marr 76] ont développé un algorithme itératif faisant coopérer les contraintes de continuité de la disparité et d'unicité. Ils ont montré que cet algorithme pouvait être implanté en utilisant un réseau de neurones sélectifs à différentes combinaisons de coordonnées spatiales et de disparités rétiniennes, mais sans aucun apprentissage incrusté. Les neurones du réseau sont connectés de la manière suivante : excitation entre les neurones correspondant à des appariements qui représentent des disparités voisines, et inhibition entre les neurones correspondant à des appariements conflictuels.

Khotanzad et al. [Khotanzad 90] [Khotanzad 91][Khotanzad 93] ont considéré, dans un premier temps, le problème de mise en correspondance comme une application d'un ensemble d'appariements initiaux, vérifiant la contrainte épipolaire, vers l'ensemble des appariements corrects. Les primitives utilisées sont les pixels des deux images stéréoscopiques. Cette application est ensuite implantée sur un réseau de neurones multicouche (couche d'entrée, couche cachée, couche de sortie) en effectuant l'apprentissage par l'algorithme de rétropropagation. Cette technique présente un aspect important : seule la contrainte épipolaire est imposée explicitement. Les autres contraintes (unicité et continuité de la disparité) sont codées sur le réseau par l'apprentissage.

Récemment, Cruz et al. ont proposé d'apparier des segments de droite en utilisant un réseau de neurones multicouche basé sur une technique d'auto-organisation [Cruz 95]. Tout d'abord, le problème est ramené à un problème de classification. Il s'agit de classer toutes les paires de primitives, gauches et droites, dans deux classes : "appariement correct" ou "appariement incorrect". Un apprentissage non supervisé, compétitif, est ensuite appliqué au réseau pour assigner chaque appariement possible à l'une des deux classes.

D'autres travaux, utilisant la même approche, ont été publiés dans la littérature [O'Toole 89][Qian 88][Sun 87].

Une autre approche consiste à considérer le problème d'appariement comme un problème d'optimisation. Dans un premier temps, une fonction de coût, représentant les contraintes du problème (unicité, continuité de la disparité, ordre, etc.), est construite de telle façon que ses minima correspondent à des appariements acceptables. Un réseau de neurones de Hopfield, entièrement connecté, est ensuite utilisé pour minimiser cette fonction de coût.

Nasrabadi et Choo [Nasrabadi 92b] ont utilisé cette approche pour appairer les points d'intérêt extraits par l'opérateur de Moravec (Cf. § II-6.2.1). La fonction de coût est formulée à partir des contraintes d'unicité et de continuité de la disparité. Chaque neurone du réseau bidimensionnel correspond à un appariement possible entre un point de l'image gauche et un autre de l'image droite.

Dans [Zhou 88a], la fonction de coût est construite à partir de la contrainte de continuité de la disparité et d'une contrainte de similarité. La contrainte d'unicité n'est pas codée dans la fonction de coût. Le processus de minimisation utilise un réseau tridimensionnel où chaque neurone représente une disparité possible d'un point de l'image de référence. Du fait qu'il ne respecte pas certaines conditions, ce réseau présente un problème de convergence. Cependant, Lee et al. [Lee 94b] ont proposé une amélioration en intégrant la contrainte d'unicité dans la fonction de coût.

D'autres méthodes très semblables sont décrites dans [Hu 93a][Hu 93b][Mousavi 90][Nichani 94].

### **II-8.8 Approche par optimisation utilisant le recuit simulé**

Nous avons vu au § II-8.7 que le problème de mise en correspondance peut être ramené à un problème d'optimisation. Après avoir construit une fonction de coût, représentant les contraintes du problème, un réseau de Hopfield est ensuite utilisé pour la minimiser.

L'inconvénient majeur de cette approche est que le processus de minimisation associé au réseau est local, c'est à dire qu'il tend à proposer une solution qui est le premier minimum local rencontré. Pour pallier cette limitation, certains auteurs ont proposé d'utiliser l'algorithme du recuit simulé [Kirkpatrick 83][Kirkpatrick 84]. L'idée de base consiste à considérer le système à optimiser comme un système physique défini par ses états et par son énergie. Le minimum absolu de cette énergie est atteint en simulant un processus de recuit en partant d'une température élevée pour terminer à une température voisine de zéro.

La qualité des résultats produits par l'algorithme du recuit simulé est excellente car le processus du recuit force l'état du système dans des régions de basse énergie tout en lui évitant d'être piégé dans des états correspondant à des minima locaux où l'énergie est élevée. De plus, son implantation sur des architectures neuronales de type réseau de Hopfield est naturelle. Il suffit d'associer à chaque état élémentaire du système la sortie d'un neurone et d'identifier

l'énergie à minimiser à celle du réseau (Cf. Chapitre VI). Néanmoins, cette approche nécessite une grande quantité de calculs et de tests, à cause de sa nature intrinsèquement stochastique.

De nombreux auteurs ont proposé d'utiliser cette approche pour aborder le problème de l'appariement stéréoscopique. Nous pouvons citer, par exemple, les travaux de Vinet et Gagalowicz [Vinet 91] qui ont développé un algorithme d'appariement de régions basé sur la contrainte de cohérence spatiale des adjacences : un couple  $(R_g, R_d)$  de régions similaires au regard de critères géométrique et photométrique est accepté si les régions adjacentes à  $R_g$  sont mises en correspondance avec des régions adjacentes à  $R_d$  et vice-versa. Dans un premier temps, cette contrainte est exprimée sous la forme d'une fonction de coût. La recherche des meilleurs appariements consiste à minimiser cette fonction en utilisant l'algorithme du recuit simulé. Dans [Martinez 93], les auteurs ont proposé de mettre en correspondance les pixels des images stéréoscopiques. Le problème d'appariement est modélisé sous la forme d'un problème de minimisation d'une fonction de coût. Celle-ci code plusieurs propriétés du monde physique, du capteur et de la scène : unicité-occlusions, ressemblance locale, continuité-régularité locale, point de fixation. Le processus de minimisation combine à la fois l'algorithme du recuit simulé et la méthode de recherche Tabou [Glover 93]. Celle-ci est une méthode d'optimisation basée sur une recherche par voisinage, caractérisée par l'exploitation de l'historique de la recherche.

### II-8.9 Autres méthodes

Il existe d'autres méthodes de mise en correspondance qui n'entrent dans aucune des catégories que nous venons d'évoquer. Nous pouvons citer par exemple les méthodes d'appariement utilisant directement les contraintes. Il s'agit, dans un premier temps, d'établir un certain nombre d'appariements vérifiant une contrainte donnée, comme la contrainte épipolaire par exemple. Les meilleurs appariements sont ensuite obtenus par application successive de plusieurs contraintes. Dans [Deriche 90], Deriche et Faugeras ont utilisé ce principe pour appairer les points de courbure. En partant d'un ensemble d'appariements potentiels, obtenu par application de la contrainte épipolaire, ils appliquent certaines contraintes (profondeur, direction du gradient, courbure, continuité figurale) pour réduire l'ensemble des possibilités d'appariement. Une approche très semblable est proposée dans [Zagrouba 92], où les primitives à appairer sont les régions. D'autres approches originales sont publiées dans la littérature, dans [Adjouadi 94], par exemple, les auteurs proposent une technique de mise en correspondance en exploitant la transformation de Walsh.

### II-8.10 Conclusion

La mise en correspondance constitue la partie la plus difficile et la plus délicate d'un traitement d'images stéréoscopiques, et ceci pour deux raisons principales :

- La nature combinatoire du problème. Malgré les contraintes mises en oeuvre pour réduire au maximum l'espace de recherche des homologues d'une primitive, la complexité des méthodes d'appariement reste prohibitive pour certaines applications où la puissance de calcul est limitée.
- La présence d'ambiguïtés. Celles-ci sont causées par le phénomène d'occlusion ainsi que celui de distorsion projective lié au capteur stéréoscopique. Pour lever certaines ambiguïtés d'appariement, on fait appel à des contraintes qui correspondent, en fait, à certaines hypothèses sur la scène observée. Cependant, la nature heuristique de ces contraintes rend parfois ces méthodes moins fiables, suivant la complexité des scènes à analyser.

Devant ces difficultés, de nombreux auteurs ont suggéré l'utilisation d'une troisième caméra pour améliorer le processus stéréoscopique.

## II-9 STEREOVISION TRINOCULAIRE

Introduite par Yashida [Yashida 86], la stéréovision trinoculaire désigne tout processus stéréoscopique utilisant trois caméras. Cette approche a rapidement séduit de nombreux chercheurs et a donné naissance à plusieurs systèmes de perception du relief [Ayache 89][Shen 89][Shen 87b][Ito 86][Pietikainen 86][Lustman 87].

Avec une troisième image, la recherche des appariements devient une simple vérification. En effet, lorsqu'on apparie deux primitives de deux images, la validité de cet appariement est testée en vérifiant sur la troisième image la présence de la primitive correspondante, à l'aide de la contrainte épipolaire.

Considérons la figure II-10 qui représente un système stéréoscopique trinoculaire. Soit  $P$  un point de la scène et soient  $p_1, p_2, p_3$  ses trois projections sur les trois plans image. Ce système peut être vu comme trois systèmes binoculaires. Ainsi, à chaque couple de primitives  $(p_i, p_j)$  sont associées deux droites épipolaires conjuguées,  $L_{ij}$  et  $L_{ji}$  ( $i, j = 1, 2, 3$  et  $i \neq j, i \neq k, j \neq k$ ). Géométriquement, lorsque  $p_1, p_2$  et  $p_3$  forment un appariement correct, le point  $p_i$  se trouve nécessairement à l'intersection des droites épipolaires  $L_{ij}$  et  $L_{ik}$  ( $i, j = 1, 2, 3$  et  $i \neq j, i \neq k, j \neq k$ ).

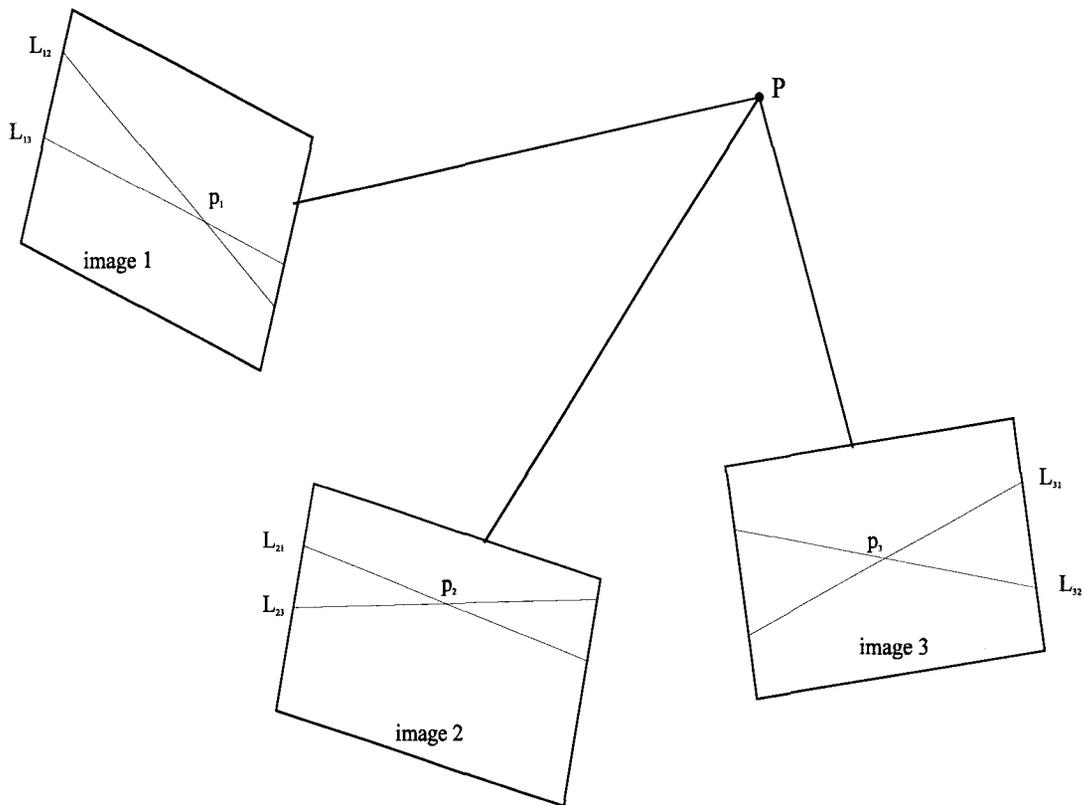


Figure II-10. Géométrie trinoculaire.

Cependant, en se basant sur la géométrie épipolaire trinoculaire, on peut développer un algorithme simple de mise en correspondance. Soit  $p_1$  un point de l'image 1. La recherche du correspondant de ce point peut être effectuée en trois étapes :

- Sélectionner tous les points, de l'image 2 par exemple, susceptibles d'être appariés à  $p_1$ . Soit  $\{p_2\}$  l'ensemble de ces points. Ceux-ci se trouvent sur la droite épipolaire  $L_{21}$  conjuguée à celle contenant  $p_1$ .
- Par un critère de ressemblance, déterminer le point qui ressemble le plus à  $p_1$ . Soit  $(p_1, p_2)$  le couple retenu.
- Dans l'image 3, à partir des droites épipolaires  $L_{31}$  et  $L_{32}$ , correspondant aux points  $p_1$  et  $p_2$ , vérifier la présence d'un point  $p_3$  à leur intersection. Ainsi, l'appariement  $(p_1, p_2)$  est validé si  $p_3$  est présent. Le couple est rejeté dans le cas contraire et on continue alors la recherche en éliminant le point  $p_2$  de la liste  $\{p_2\}$ .

Par rapport aux algorithmes binoculaires, celui-ci montre que l'utilisation d'une troisième caméra apporte trois avantages :

- La simplicité. La mise en oeuvre de l'algorithme trinoculaire est moins coûteuse en puissance de calcul et en temps d'exécution.
- La fiabilité. Aucune contrainte liée à la scène n'est utilisée au cours de l'appariement. Seule la contrainte épipolaire, qui est une contrainte géométrique non heuristique, est appliquée.
- La précision. L'influence du bruit est réduite, car on utilise trois mesures au lieu de deux. Il en résulte une meilleure localisation pour la reconstruction 3D.

En revanche, l'utilisation d'une troisième caméra peut entraîner certains inconvénients :

- Réduction du champs de vision, ce qui réduit la taille de la scène reconstruite.
- Nécessité de mettre en oeuvre trois traitements au lieu de deux pour extraire les primitives.

## II-10 CONCLUSION

Un des objectifs de la vision artificielle est de percevoir le relief d'un environnement à partir d'une ou plusieurs images vidéo. Les recherches dans ce domaine ont permis de développer de nombreuses approches. La stéréovision constitue la méthode la plus répandue et la plus utilisée pour aborder ce problème. Il s'agit d'extraire des informations tridimensionnelles à partir de plusieurs images de la même scène, prises sous des angles différents. En fait, la stéréovision permet de retrouver l'information de profondeur perdue lors de la projection de la scène sur les capteurs.

Le problème central de la stéréovision est celui de la mise en correspondance des indices visuels, ou primitives, extraits de chacune des images à apparier. Ce problème a une nature combinatoire et la seule façon de la réduire est d'utiliser des contraintes, telle que la contrainte épipolaire, qui permettent de limiter la recherche à certaines zones des images. De plus, au cours du processus de mise en correspondance, de nombreuses ambiguïtés d'appariement doivent être levées par application de certaines contraintes qui ont trait à la structure de la scène et des objets observés.

Cependant, la complexité des algorithmes de mise en correspondance reste, malgré les contraintes mises en oeuvre pour la réduire, incompatible avec la rapidité nécessaire à la

détection, en temps réel, d'obstacles devant les véhicules routiers. En plus de la nature combinatoire du problème, cette complexité algorithmique est liée à la complexité des scènes naturelles, où, contrairement aux scènes d'intérieur, on ne peut pas imposer assez d'hypothèses pour faciliter la procédure de mise en correspondance.

Pour conserver les avantages d'un système de détection stéréoscopique pour lequel la localisation est certainement le plus important, sans en avoir les inconvénients au niveau des temps de traitement, il est possible d'abandonner les caméras matricielles, employées habituellement, afin d'opter pour un système stéréoscopique à base de caméras linéaires. Dans le chapitre suivant, nous présentons le principe de fonctionnement de ce nouveau capteur, dont la conception a fait l'essentiel de la thèse de Jean-Luc Bruyelle dans le cadre du programme Prometheus.

## **Chapitre III**

# **Conception d'un stéréoscope linéaire**

## III-1 INTRODUCTION

Le travail confié au Centre d'Automatique de Lille, dans le cadre du programme Prometheus, est destiné à la réalisation d'un système stéréoscopique de détection d'obstacles sur route. L'étude a commencé par la conception d'un capteur stéréoscopique à base de caméras linéaires. Comme la partie du programme qui nous a été confiée consiste à analyser les images linéaires issues de ce capteur, il nous a paru judicieux de rappeler les grands axes de sa conception.

Après une description générale des caméras linéaires, ce chapitre expose le principe de fonctionnement du stéréoscope linéaire dont la conception a fait l'essentiel de la thèse de Jean-Luc Bruyelle [Bruyelle 94]. Nous nous contenterons de n'en reprendre que les principaux éléments. A la fin de ce chapitre, sont présentés quelques séquences d'images enregistrées à l'aide de ce capteur stéréoscopique linéaire.

## III-2 CAMERAS LINEAIRES

La meilleure façon d'imaginer une caméra linéaire est de penser à une caméra matricielle pour laquelle une seule ligne serait active. Physiquement, une caméra linéaire comporte une surface sensible mono-dimensionnelle constituée d'une barrette CCD. Munie de son optique, elle fournit une image linéaire qui est une vue en coupe de la scène observée.

### III-2.1 Constituants d'une caméra linéaire

Comme tout capteur d'images, une caméra linéaire permet de transformer la lumière de chaque point d'un objet observé en un signal électrique qui matérialise l'image acquise. Elle est constituée :

- d'un capteur linéaire : c'est une version simplifiée des capteurs CCD matriciels utilisés dans les caméras vidéo standards. Il comporte essentiellement un système d'éléments photosensibles, organisés sous la forme d'une barrette rectiligne, qui reçoivent les rayonnements pour les transformer en une image électrique. Il faut noter également la présence de registres à décalage, d'amplificateurs de sortie et de circuits annexes destinés à la gestion de ces registres à décalage.
- d'un objectif : c'est un dispositif optique qui focalise l'image d'une scène dans le plan de la barrette. On utilise généralement des objectifs photographiques.

- d'un dispositif électronique : c'est un système électronique qui permet la synchronisation et le conditionnement du signal de sortie du capteur.

### III-2.2 Caractéristiques des capteurs linéaires

Les capteurs linéaires présentent certaines caractéristiques qui les rendent particulièrement intéressants dans de nombreuses applications :

- La grande résolution horizontale : leur structure très simple permet d'intégrer un grand nombre de photosites en augmentant la longueur de la barrette. La résolution typique est de 2048 points, contre environ 750 points pour les capteurs matriciels courants, soit un gain d'un facteur de l'ordre de 3 en moyenne.
- La faible quantité d'information : un capteur matriciel typique délivre 582 lignes de 750 points, soit 436.500 pixels. En comparaison avec un capteur linéaire de 2048 pixels sur sa ligne unique, on obtient 200 fois moins d'information à traiter, tout en disposant d'une résolution horizontale supérieure.
- La compatibilité avec les optiques photographiques : grâce à la grande longueur du capteur, généralement comprise entre 20 et 28 mm, on peut utiliser des objectifs photographiques dont les caractéristiques de résolution, de distorsion et de vignetage sont très supérieures à celles des objectifs vidéo. La grande résolution des objectifs photographiques est indispensable pour tirer partie de celle des capteurs linéaires. Leur faible distorsion peut généralement être négligée, ce qui évite de nombreux calculs de correction d'images. Enfin, leur faible vignetage peut éviter des problèmes lors de l'appariement des images en stéréovision.
- La souplesse des paramètres de prise de vue : l'opérateur peut ajuster, selon les besoins et les conditions de prise de vue, la fréquence d'acquisition et le temps d'exposition. Il peut aussi synchroniser chaque prise de vue sur un événement extérieur, même à intervalles irréguliers. Il est même possible de définir la fréquence de transfert des pixels en fonction des performances de la chaîne d'acquisition et de traitement des images.

Il faut noter que les capteurs linéaires posent cependant certains problèmes qui sont liés à leur structure et au type de signal qu'ils fournissent :

- La difficulté pour le calibrage stéréoscopique : lors de l'utilisation des caméras linéaires en stéréovision, il est nécessaire d'avoir un positionnement précis et rigide des deux caméras afin qu'elles oeuvrent dans une zone de vision commune.
- La pauvreté relative du signal mono-dimensionnel : à cause de la perte de l'information verticale, il est difficile d'interpréter une image linéaire. La dimension manquante peut être reconstruite alors, dans certains cas, grâce à un balayage mécanique de la scène (Cf. § III-2.3.1).

### **III-2.3 Exemples d'application des caméras linéaires**

Dans beaucoup d'applications de la vision par ordinateur, les caméras linéaires peuvent remplacer avantageusement les caméras matricielles.

#### ***III-2.3.1 Reconstitution de la dimension verticale***

L'utilisation des caméras linéaires est très appréciée dans des applications pour lesquelles la résolution des caméras matricielles est insuffisante. Un dispositif de balayage mécanique permet alors de reconstituer la dimension manquante, c'est à dire l'information verticale, au prix d'une certaine lenteur. Parmi ces applications, nous pouvons citer :

- La numérisation de documents (ou scanner) : filmé avec une très grande résolution, un document peut être intégré en tant qu'image dans la composition d'un autre document. Afin de reconstituer le document à intégrer, on effectue alors un balayage mécanique de celui-ci, en déplaçant le capteur.
- Le contrôle de qualité : les caméras linéaires sont très utilisées dans le domaine de contrôle de qualité, notamment dans la fabrication de tissus, de tôles, et plus généralement de tout produit plan fabriqué en continu. Les caméras sont disposées perpendiculairement au sens de défilement du produit, et c'est le mouvement de ce dernier qui permet de reconstituer l'information verticale et d'obtenir ainsi des images bidimensionnelles des matériaux contrôlés.
- La télédétection : utilisant des capteurs linéaires de grande résolution, les satellites d'observation terrestre Spot, pour lesquels le balayage est fourni grâce au mouvement orbital du satellite, sont plus fiables et plus rapides que les radiomètres à balayage mécanique qui équipaient les satellites plus anciens tels que les Landsat.

### III-2.3.2 Stéréovision

Bien que la stéréovision linéaire soit rarement utilisée, plusieurs auteurs ont commencé à s'y intéresser. En robotique mobile, certains travaux utilisent des caméras linéaires pour percevoir l'environnement d'un robot. A cet effet, deux caméras sont montées en couple stéréoscopique pour pouvoir déterminer les positions des objets constituant l'environnement du robot.

Dans [Colle 90], Colle propose un dispositif de stéréovision linéaire destiné à guider un robot mobile. Les caméras, fixées sur un support rigide afin de conserver un champ de vision commun, sont posées sur des rails permettant ainsi le déplacement du dispositif stéréoscopique. Les axes optiques des caméras sont parallèles et horizontaux par rapport au sol. Cette disposition, permet de voir à n'importe quelle distance des objets placés à la hauteur des deux caméras, mais, en revanche, elle empêche de voir les objets de faible hauteur. L'étude a été faite sur des scènes d'intérieur simples, constituées d'objets aux contours verticaux.

Contrairement au dispositif de Colle, Inigo [Inigo 85][Inigo 87] utilise un système stéréoscopique linéaire incliné vers le sol. Ce système est destiné à guider un robot mobile le long d'une bande de peinture tracée au sol, tout en détectant les obstacles placés sur son chemin. Cette approche présente l'avantage de fournir un fond relativement homogène, et donc des images faciles à traiter, mais limite la portée du dispositif. Comme dans beaucoup d'autres travaux en robotique mobile, l'environnement d'expérimentation envisagé est très simple, et suppose un fond uniforme avec des objets bien contrastés. Les problèmes causés par les ombres et les occlusions ne sont pas pris en compte.

Le dispositif que nous allons décrire au § III-3 a la même configuration que celui réalisé par Inigo, avec des caméras inclinées vers la chaussée. Ce dispositif est destiné à la détection d'obstacles sur route. Cependant, l'environnement routier impose des contraintes supplémentaires de précision et de robustesse dues aux formes complexes rencontrées, aux fonds hétérogènes, au faible contraste, à l'éclairage variable, etc. Ces conditions difficiles interdisent certaines hypothèses simplificatrices courantes en robotique mobile [Colle 90][Inigo 85][Inigo 87][Kriegman 89]. Par ailleurs, les méthodes de traitement et d'analyse des images doivent être en mesure d'éliminer les fausses détections, de détecter des objets de faible contraste et/ou de petites dimensions, tout en opérant en temps réel.

### III-2.3.3 Métrologie

Dans le domaine de la métrologie, l'intérêt de la vision linéaire réside dans sa précision ( $\pm 0.05\%$  dans le cas d'un capteur de 2048 points), alliée à une absence de tout contact mécanique susceptible de perturber la mesure. Comme exemple de ce type d'application, Duvieubourg et al. [Duvieubourg 93] ont utilisé un capteur linéaire pour suivre les déformations d'une éprouvette soumise à des contraintes mécaniques. La caméra linéaire vise la longueur de l'éprouvette, sur laquelle ont été tracés deux repères dont on mesure l'écartement au niveau subpixel afin de suivre l'évolution du matériau au fur et à mesure qu'il se déforme.

### III-2.4 Champ de visée d'une caméra linéaire

Le champ de visée d'une caméra linéaire est réduit à un plan, nommé plan de visée. De ce fait, la notion de verticale est absente. Le plan de visée est caractérisé géométriquement par (Cf. Figure III-1) :

- L'ouverture angulaire du champ de visée : 
$$\alpha = 2 \arctan \frac{l}{2 \cdot f}$$
- La largeur du champ de visée à une distance  $D$  : 
$$L = \frac{l \cdot D}{f}$$

où :

- $l$  est la longueur du capteur.
- $f$  est la distance focale des objectifs.
- $D$  est la distance entre la caméra et la ligne objet.

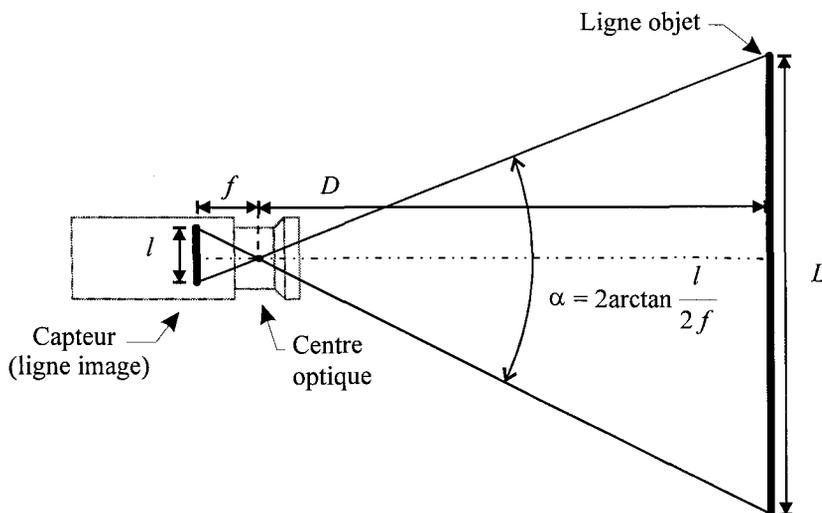


Figure III-1. Champ de visée d'une caméra linéaire.

### III-3 CONCEPTION D'UN STEREOSCOPE LINEAIRE

Nous allons à présent décrire le dispositif de prise de vue réalisé dans le but de détecter les obstacles sur route [Bruyelle 94]. Composé de deux caméras linéaires, ce dispositif n'est pas différent dans son principe de tous les systèmes de prise de vue stéréoscopique à deux caméras. Cependant, afin de rendre possible la reconstruction tridimensionnelle, la structure particulière des caméras linéaires, qui ne restituent pas l'information verticale, impose une configuration particulière pour que les deux caméras du stéréoscope visent la même scène.

#### III-3.1 Hypothèses sur les caméras

Afin d'obtenir la configuration désirée du stéréoscope, certaines hypothèses concernant les caméras doivent être vérifiées :

- Les plans de visée des deux caméras sont confondus. Pour cela, il faut, d'une part, que les deux capteurs soient dans le prolongement l'un de l'autre. D'autre part, il faut que la ligne joignant les deux capteurs soit parallèle à celle joignant les centres optiques des deux objectifs.
- Les objectifs sont des sténopés. Il est courant d'assimiler un objectif à un sténopé, mot grec signifiant trou étroit, c'est à dire un trou de diamètre infiniment petit, confondu avec le centre optique de l'objectif. Dans la pratique, cela signifie que les points image et les points objet sont reliés par des lignes droites passant par le centre optique, et que chaque point de la scène est projeté par l'objectif sur un photosite unique du capteur quand la scène est parfaitement focalisée.
- Les deux objectifs ont la même distance focale. Il est toujours possible de les choisir différentes, mais, dans ce cas, la complexité des algorithmes de reconstruction augmente, ce qui entraîne un accroissement des temps de calcul.

#### III-3.2 Calibrage du stéréoscope linéaire

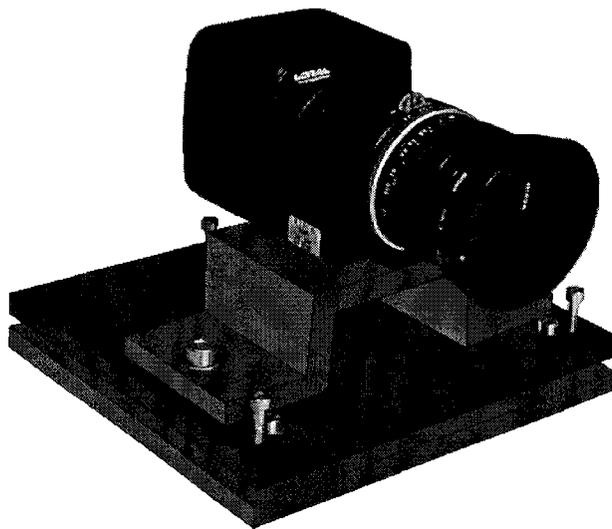
Le stéréoscope est composé de deux caméras linéaires qui ne peuvent constituer un couple stéréoscopique que si leurs plans de visée sont confondus en un même plan. Ce dernier est appelé plan de visée du stéréoscope. En effet, l'absence de l'information verticale ne permet pas la compensation passive d'un écart de position ou d'orientation de l'une des deux caméras par une procédure logicielle.

Une méthode de calibrage a été mise au point afin d'assurer la coplanéité des plans de visée des deux caméras et de mettre celles-ci dans une configuration géométrique prédéterminée.

### **III-3.2.1 Monture orientable**

Pour pouvoir mettre en oeuvre la procédure de calibrage, un système mécanique de positionnement et d'orientation des caméras a été développé. Ce système permet un réglage extrêmement précis de la coplanéité des plans de visée par un ajustement simultané des angles d'inclinaison et de roulis, ainsi que de la hauteur relative des centres optiques des objectifs.

Chaque caméra est donc montée sur un support indépendant (Cf. Figure III-2), orientable avec une grande précision selon les angles de site, d'azimut et de roulis. La hauteur du support est également ajustable. Chaque monture est constituée de trois platines métalliques superposées, visibles sur la figure III-2 : la platine inférieure fixe, la platine supérieure orientable en site et en roulis et la platine azimutale.



*Figure III-2. Caméra linéaire sur sa monture ajustable.*

### **III-3.2.2 Mire de calibrage**

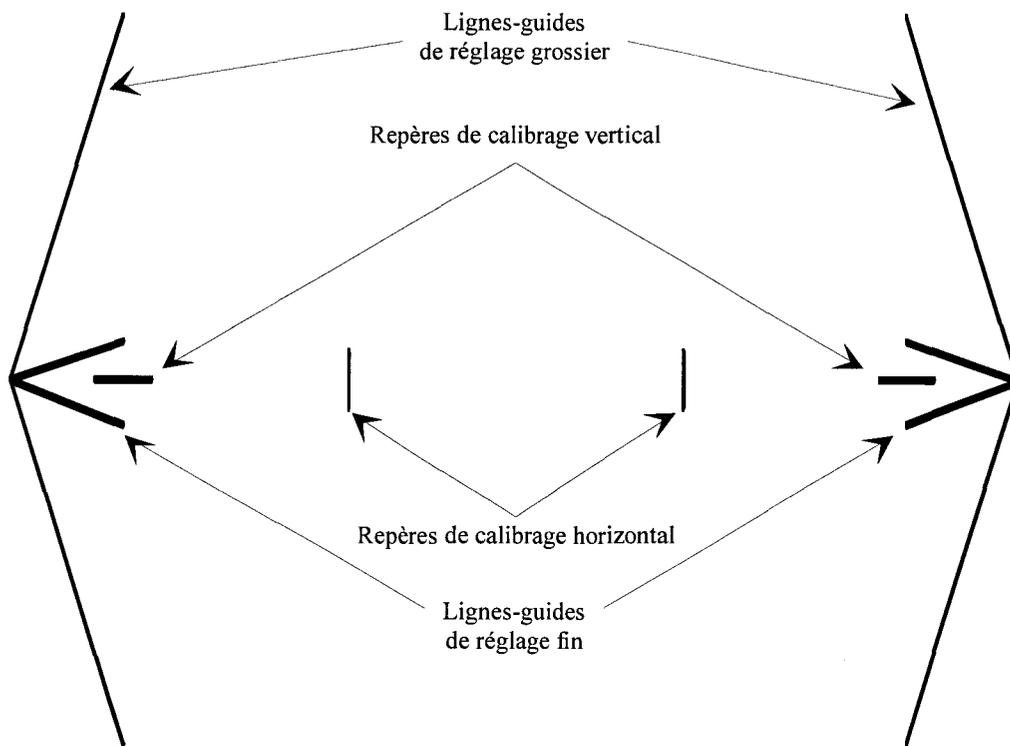
Avec une caméra linéaire, où l'image est réduite à une ligne, un objet est soit vu ou ne l'est pas du tout. Dans le cas où l'objet n'est pas vu par la caméra, il est impossible de le situer par rapport au plan de visée car il se situe soit au-dessus soit au-dessous du plan de visée. La mire de calibrage doit donc fournir le moyen de déterminer la position du plan de visée par rapport aux repères tracés sur la mire.

La mire de calibrage, plane, tracée en noir sur un fond blanc, est constituée d'un certain nombre de repères permettant de situer le plan de visée au cours de la procédure de calibrage (Cf. Figure III-3) :

- Repères de calibrage vertical : ils sont destinés à mettre les deux capteurs dans le prolongement l'un de l'autre (Cf. § III-3.1). Ils correspondent, dans la figure III-3, aux deux segments de droite horizontaux, placés dans le prolongement l'un de l'autre. Le plan de visée d'une caméra est correctement ajusté lorsque cette dernière "voit" ces deux repères simultanément. La coplanéité des deux plans de visée est obtenue lorsqu'ils sont tous les deux parfaitement ajustés.
- Repères de calibrage horizontal : ils permettent d'établir le parallélisme des deux axes optiques des deux caméras (Cf. § III-3.1). Ils sont représentés, dans la figure III-3, par les deux segments de droite verticaux, situés de part et d'autre du centre de la mire et placés de façon à couper la droite qui porte les repères de calibrage vertical. La distance entre ces deux segments est égale à l'entre axe séparant les centres optiques des objectifs des deux caméras. Les deux axes optiques des deux caméras sont parallèles lorsque l'image du repère gauche (resp. du repère droit) est au centre du capteur de la caméra gauche (resp. de la caméra droite).
- Lignes-guides : limitée à une ligne, l'image fournie par une caméra ne permet pas, si les repères de calibrage vertical et horizontal ne sont pas visibles, de situer le plan de visée par rapport à ces derniers. Afin d'obtenir cette information, des lignes obliques, situées de part et d'autre des repères de calibrage, ont été dessinées sur la mire (Cf. Figure III-3). Leur rôle est d'indiquer la bonne direction d'ajustage du plan de visée. Deux jeux de lignes sont prévus, l'un pour un réglage grossier, utile au début de la procédure de calibrage, l'autre pour un réglage fin lorsque le plan de visée est proche des repères de calibrage.

Notons au passage que les dimensions de la mire sont fixées en fonction des caractéristiques des caméras et des dimensions de la scène à observer.

Le lecteur intéressé trouvera dans la thèse de Jean-Luc Bruyelle [Bruyelle 94] la procédure de calibrage complète, présentée sous la forme d'un guide pratique.



*Figure III-3. Mire de calibrage.*

Après calibrage, le stéréoscope linéaire a été soumis à un étalonnage destiné à mesurer la précision de localisation des objets, afin d'évaluer son degré de fiabilité. Les positions déterminées avec le stéréoscope ont été comparées à celles mesurées au moyen de deux théodolites de chantier. Pour plus de détails sur cette opération, le lecteur pourra consulter également [Bruyelle 94].

### III-4 INTEGRATION DU STEREOSCOPE SUR UN VEHICULE

Le stéréoscope linéaire a été intégré sur un véhicule expérimental dans le but de détecter les obstacles à l'avant de ce dernier.

#### III-4.1 Monture commune aux deux caméras

Les deux montures des deux caméras, décrites au § III-3.2.1, ne sont pas montées directement sur le véhicule. En effet, la rigidité de celui-ci est insuffisante pour assurer le maintien du calibrage des caméras durant les prises de vue. Les deux caméras ont donc été montées, avec leurs supports, sur une poutre d'acier très rigide, dont la structure mécanique évite des vibrations parasites durant les prises de vue sur route (Cf. Figure III-4).

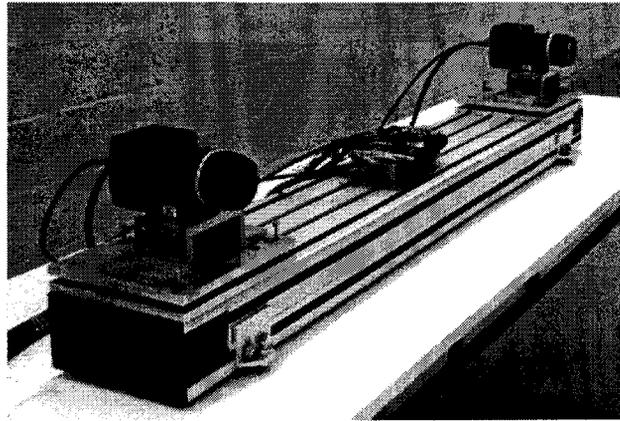


Figure III-4. Stéréoscope linéaire sur sa monture rigide.

### III-4.2 Installation sur le véhicule

Pour les premiers essais, le stéréoscope linéaire a été placé sur le toit d'une voiture de tourisme aimablement prêtée par P.S.A. (Cf. Figure III-5). Le plan de visée est légèrement incliné par rapport à l'horizontale d'un angle  $\theta$ , de manière à couper la chaussée à une distance  $D_{max}$  à l'avant du véhicule, qui définit la portée du dispositif (Cf. Figure III-6).



Figure III-5. La voiture utilisée pour les essais de prise de vue. Noter le stéréoscope linéaire sur le toit, et le moniteur VGA à la place du passager avant.

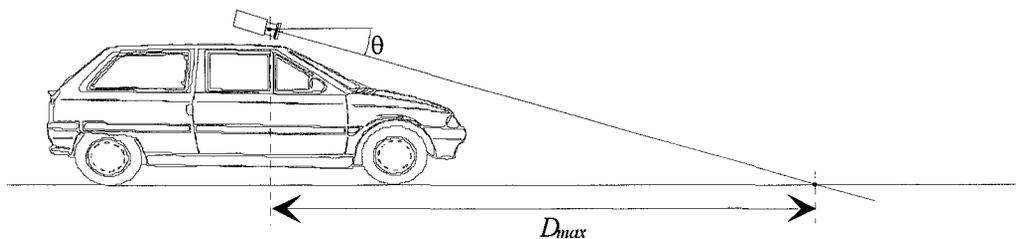


Figure III-6. Plan de visée incliné par rapport à l'horizontale.

Il faut noter que la chaussée est supposée plane. Cette hypothèse constitue la base de nombreuses méthodes de détection d'obstacles sur route [Carlsson 90][Enkelmann 90].

Les axes optiques des deux caméras sont parallèles et écartés d'une distance  $E$  (Cf. Figure III-7). Sur la même figure, la partie commune aux champs de visée des caméras définit ce qu'on appelle le secteur de stéréovision. Du fait de l'inclinaison du plan de visée, tout obstacle, même de faible hauteur, qui entre par l'avant dans ce secteur est vu simultanément par les caméras gauche et droite. Il peut ainsi être détecté et localisé par triangulation (Cf. § III-4.3).

Les caractéristiques du stéréoscope linéaire sont les suivantes :

- Définition des caméras :  $1728 \text{ pixels}$ .
- Largeur des capteurs :  $22,1 \text{ mm}$ .
- Distance focale des objectifs :  $100 \text{ mm}$ .
- Entre axe :  $1 \text{ m}$ .

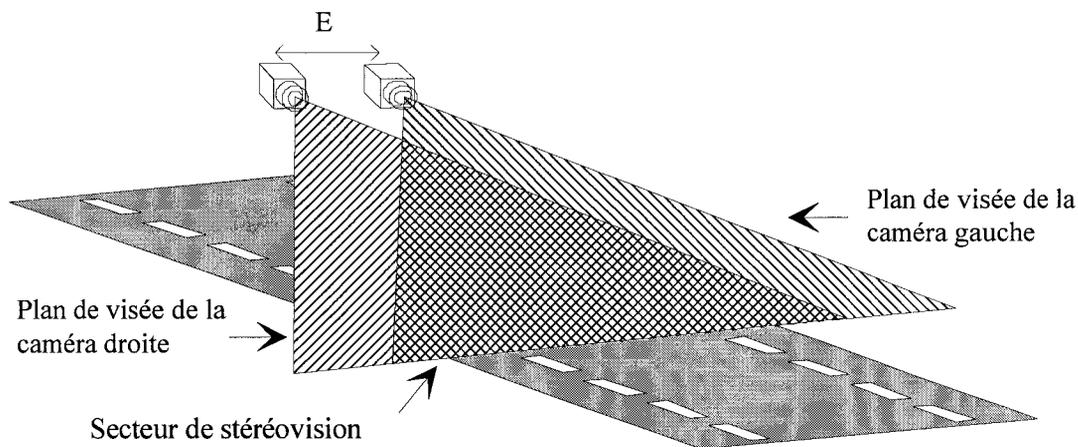


Figure III-7. Configuration du stéréoscope linéaire.

Les caractéristiques géométriques du secteur de stéréovision sont les suivantes (Cf. Figure III-8) :

- L'axe est confondu avec l'axe du stéréoscope.
- Distance minimale au centre optique :
- Ouverture angulaire :
- Largeur couverte par un pixel à une distance  $D$  :

$$D_{min} = \frac{E \cdot f}{l}$$

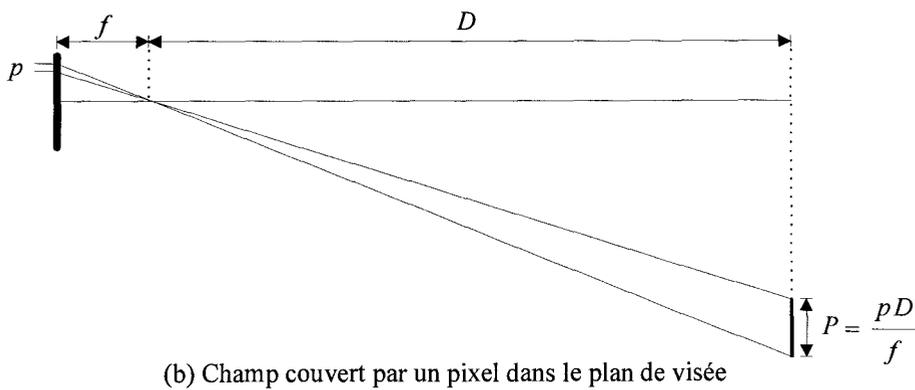
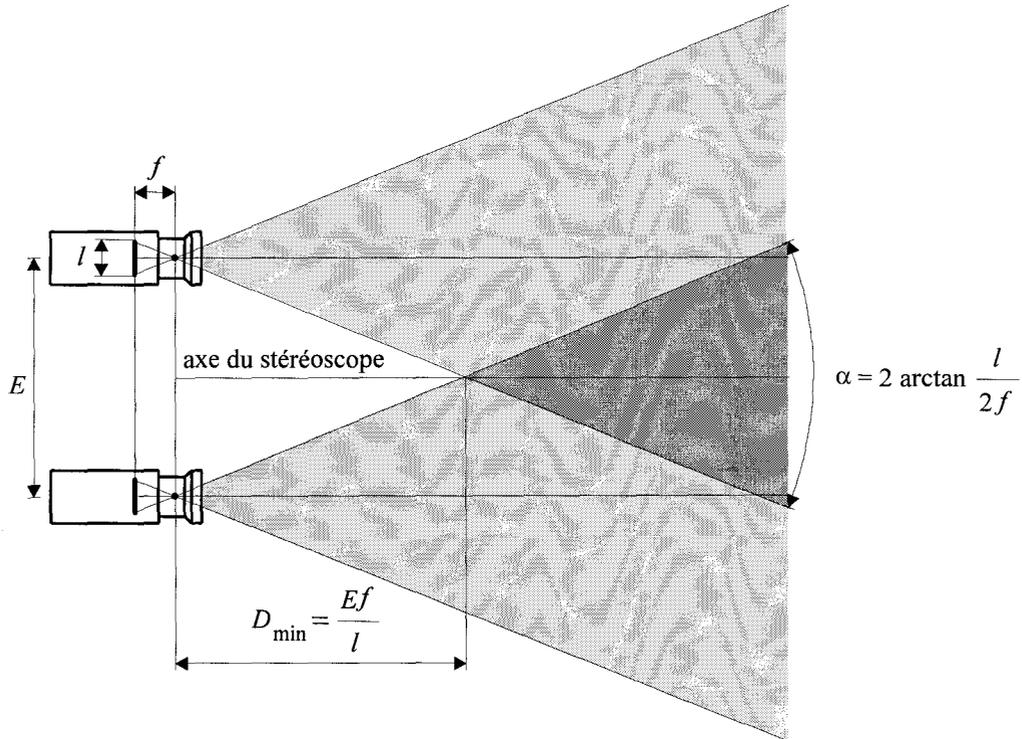
$$\alpha = 2 \arctan \frac{l}{2 \cdot f}$$

$$P = \frac{p \cdot D}{f}$$

où :

- $l$  est la longueur du capteur CCD.
- $N$  est le nombre de pixels du capteur.
- $p$  est la largeur d'un photosite :
- $f$  est la distance focale des objectifs.
- $E$  est l'entre axe entre les deux caméras.

$$p = \frac{l}{N}$$

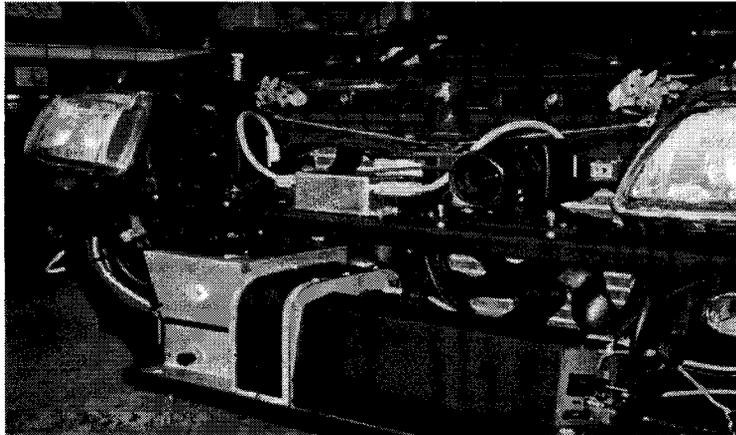


**Figure III-8.** Géométrie du stéréoscope linéaire.

Un second prototype du stéréoscope linéaire a été réalisé afin d'être intégré dans le véhicule expérimental Prolab 2. Le stéréoscope a été installé entre les phares du véhicule (Cf. Figure III-

9). Sa configuration est la même que celle du premier prototype (Cf. Figure III-7), mais avec des caractéristiques légèrement différentes :

- Définition des caméras : 2592 pixels.
- Largeur des capteurs : 25,92 mm.
- Distance focale des objectifs : 35 mm.
- Entre axe : 0,465 m.



*Figure III-9. Le second prototype du stéréoscope linéaire installé sur Prolab 2.*

La figure III-10 montre le stéréoscope linéaire après son installation complète. Seuls les objectifs des deux caméras sont visibles.



*Figure III-10. Le stéréoscope linéaire après installation complète.*

### III-4.3 Reconstruction de la profondeur et de la position

Considérons un point objet  $P$  observé par les deux caméras du stéréoscope linéaire (Cf. Figure III-11). Chaque capteur perçoit de ce point une image qui est sa projection à travers le centre optique de l'objectif associé.

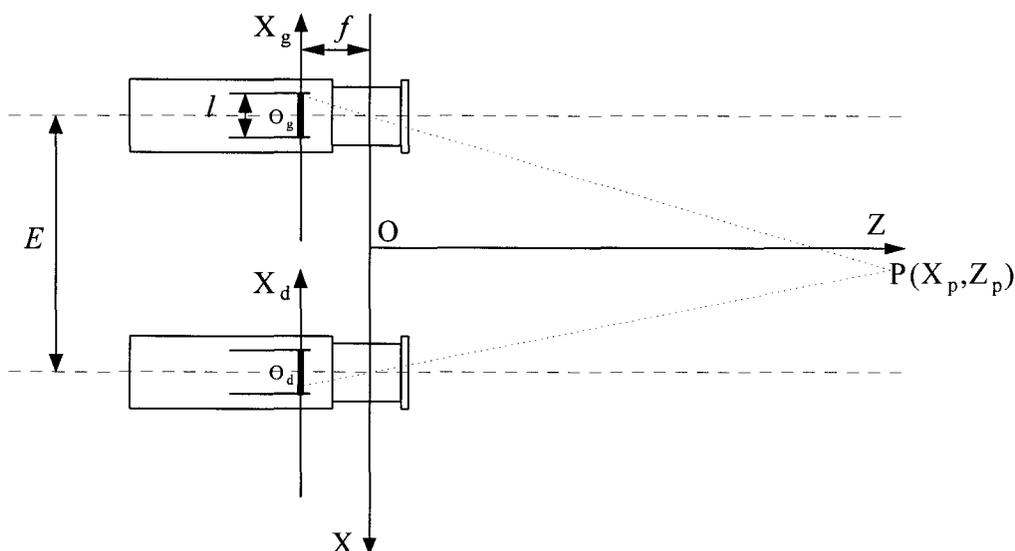


Figure III-11. Projection d'un point-objet sur les capteurs du stéréoscope linéaire.

Soit  $\{O, X, Z\}$  un repère lié au dispositif de prise de vue, défini comme suit :

- L'origine  $O$  est au milieu du segment de droite joignant les centres optiques des deux objectifs.
- L'axe  $OX$  (des abscisses) est la droite joignant les centres optiques des deux objectifs. Il est dirigé vers la droite quand on regarde vers la scène.
- L'axe  $OZ$  (des profondeurs), confondu avec l'axe du stéréoscope, est contenu dans le plan du secteur de stéréovision, normal à l'axe  $OX$ , et est dirigé vers la scène.

Soit  $P(X_p, Z_p)$  la position du point objet  $P$  dans ce repère, où  $X_p$  est son abscisse selon l'axe  $OX$  et  $Z_p$  sa profondeur selon l'axe  $OZ$ .

A chaque capteur est associé un repère :

- L'origine est au centre du capteur,  $O_g$  pour le capteur gauche,  $O_d$  pour le capteur droit.
- L'axe des abscisses,  $O_g X_g$  ou  $O_d X_d$ , est dirigé vers la gauche quand on regarde vers la scène.

Soit  $x_g$  (resp.  $x_d$ ) la position du point image sur le capteur gauche (resp. sur le capteur droit). Les positions des deux points image dans leurs repères locaux respectifs s'expriment sous la forme suivante :

$$x_g = \frac{f \cdot \left( \frac{E}{2} + X_p \right)}{D} \quad \text{Equation III-1}$$

$$x_d = \frac{f \cdot \left( X_p - \frac{E}{2} \right)}{D} \quad \text{Equation III-2}$$

Le calcul de la profondeur et de la position s'effectue de la manière suivante :

### **Calcul de la disparité**

La disparité du point  $P$  est donnée par la différence de position entre les images de  $P$  sur les deux capteurs (Cf. Figure III-11) :

$$\delta = |x_g - x_d| \quad \text{Equation III-3}$$

A partir des équations III-1 et III-2, elle prend la forme suivante :

$$\delta = \frac{E \cdot f}{D} \quad \text{Equation III-4}$$

Ce résultat indique que la disparité est la même pour les images de tous les points situés à une même profondeur, indépendamment de leurs abscisses.

### **Calcul de la profondeur**

La profondeur  $Z_p$  du point  $P$  se déduit directement de la disparité et des paramètres du stéréoscope à partir de la relation précédente :

$$D = \frac{E \cdot f}{\delta} \quad \text{Equation III-5}$$

### **Calcul de la position**

Une fois la profondeur calculée, nous pouvons utiliser les équations III-1 et III-2 pour déterminer la position  $X_p$  du point  $P$  :

$$X_p = \frac{x_g \cdot D}{f} - \frac{E}{2} \quad \text{Equation III-6}$$

$$X_p = \frac{x_d \cdot D}{f} + \frac{E}{2} \quad \text{Equation III-7}$$

L'une ou l'autre de ces équations permet de calculer la position  $X_p$  du point objet  $P$  à partir de sa profondeur  $D$  et de la position du point image sur le capteur gauche ( $x_g$ ) ou sur le capteur droit ( $x_d$ ).

### III-5 ENREGISTREMENT DE SEQUENCES D'IMAGES

Pour les essais du stéréoscope décrit précédemment et la validation des méthodes de traitement des séquences d'images qu'il fournit, il est nécessaire d'enregistrer ces séquences afin d'obtenir des données de référence.

#### III-5.1 Technique d'enregistrement

Une image linéaire seule est illisible. C'est un problème spécifique à la vision linéaire. Afin d'améliorer la lisibilité, les images linéaires sont enregistrées sous forme de séquences. Même si elle est parfois difficile à interpréter, la visualisation d'une séquence permet, avec une certaine expérience, d'avoir une idée de la scène qui a été filmée. De plus, une séquence stockée sous la forme d'un fichier est plus facile à manipuler que plusieurs fichiers contenant chacun une seule image linéaire de la séquence enregistrée.

Pour réaliser les enregistrements, nous utilisons le disque dur d'un *PC* comme un magnétoscope, et le stéréoscope comme une caméra vidéo. La visualisation est, quant à elle, réalisée grâce au moniteur *VGA* du *PC* hôte pour vérifier la qualité des enregistrements. Le système d'acquisition et d'enregistrement est représenté sur la figure III-12.

Notons que l'acquisition des images gauche et droite est réalisée en interfaçant les deux caméras avec une seule carte d'acquisition linéaire, par l'intermédiaire d'un multiplexeur développé spécifiquement pour garantir la simultanéité des prises de vue stéréoscopiques (Cf. Figure III-12). Ceci permet d'économiser le coût, mais aussi les interconnexions et la consommation électrique d'une deuxième carte d'acquisition. Le rôle du multiplexeur est de gérer les signaux de synchronisation et de commande des deux caméras et de la carte d'acquisition, ainsi que de contrôler le transfert des pixels vers cette dernière.

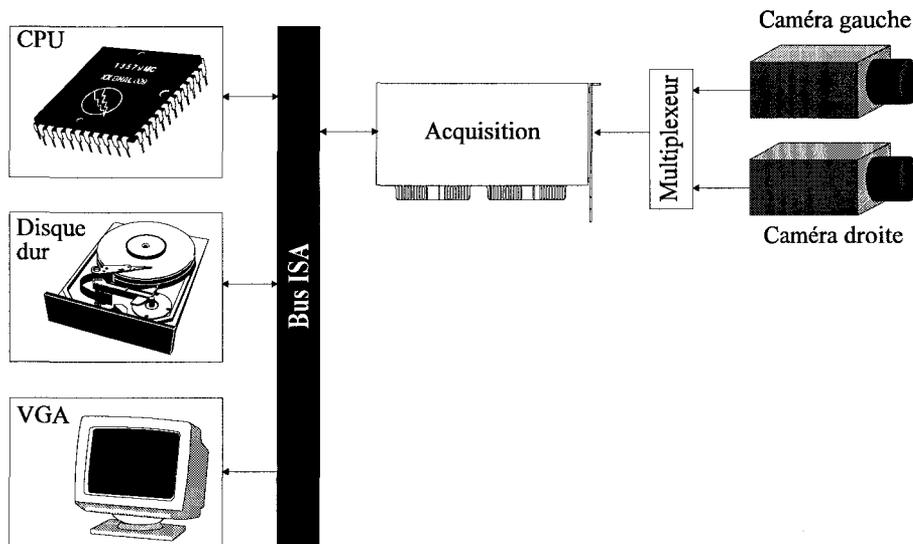


Figure III-12. Architecture électronique du stéréoscope linéaire.

La mémoire interne de la carte d'acquisition contient un tableau de  $2 \times 2592$  octets, où sont stockées les deux images linéaires. Après chaque prise de vue, les éléments du tableau d'indices pairs correspondent aux pixels de la caméra gauche, tandis que les éléments d'indices impairs correspondent aux pixels de la caméra droite (Cf. Figure III-13).

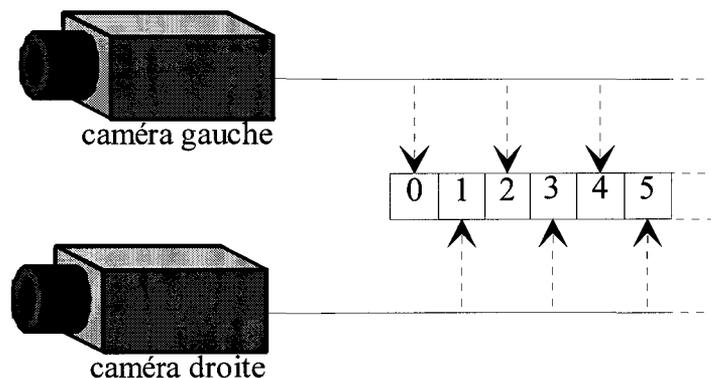


Figure III-13. Multiplexage des signaux et organisation des pixels en mémoire.

### III-5.2 Exemples de séquences enregistrées

Dans cette section, nous donnons quelques exemples de séquences enregistrées avec le stéréoscope linéaire. Chaque séquence stéréoscopique est constituée de deux séquences issues des deux caméras, gauche et droite, du stéréoscope. Les séquences sont représentées sous la forme d'images bidimensionnelles en "empilant" les différentes images-ligne acquises. Les séquences se lisent comme suit :

- L'axe horizontal représente le plan de visée du stéréoscope. Chaque ligne correspond donc à une image linéaire.
- L'axe vertical représente l'axe du temps, dirigé vers le bas de l'image. Toutes les séquences ont la même longueur de 200 lignes. Afin d'enregistrer l'intégralité d'une action sur une séquence, la durée totale de cette dernière est fixée à 20 secondes. L'intervalle de temps entre deux prises de vue consécutives est donc de 100 ms.

Les exemples de séquences présentés ci-après ont été enregistrés avec le premier prototype du stéréoscope linéaire (Cf. Figure III-5).

### III-5.2.1 Exemple 1 : Piéton évoluant devant le stéréoscope

Dans ce premier exemple, la scène enregistrée représente le déplacement d'un piéton devant le stéréoscope linéaire (Cf. Figure III-14), monté sur un support fixe et visant la chaussée à une distance de 50 mètres.

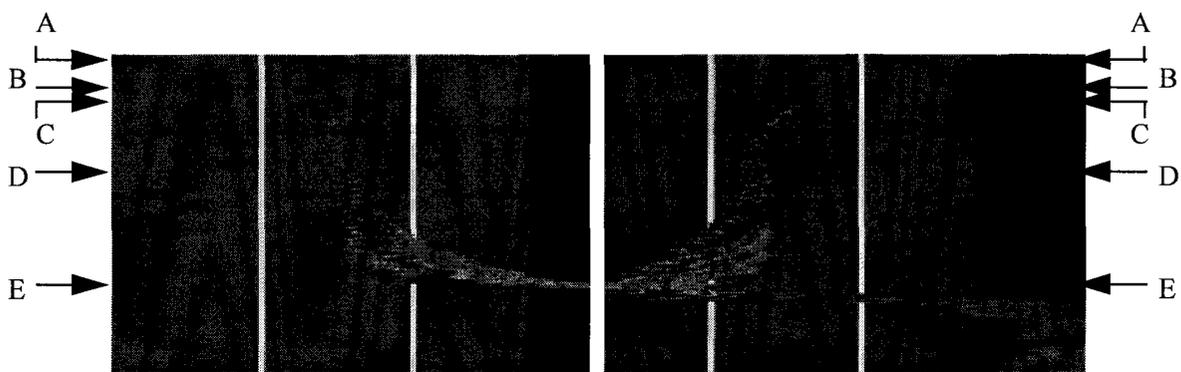


Figure III-14. Déplacement d'un piéton vu par le stéréoscope linéaire.

Sur les images de la séquence, on voit clairement les lignes blanches de la chaussée. Une voiture située hors du champ du stéréoscope (Cf. Figure III-15) projette une ombre, visible sur la droite des images comme une zone entièrement noire.

La figure III-15 décrit schématiquement les déplacements du piéton au cours du temps. On le voit d'abord se diriger vers son point de départ, situé un peu au-delà de l'intersection du plan de visée et de la chaussée (A). Ce faisant, il passe hors de portée du stéréoscope linéaire (B) et, en conséquence, disparaît de l'image. Quelques secondes plus tard, il commence à se diriger vers le stéréoscope. Il réapparaît alors dans le champ de visée, en se dirigeant vers la caméra gauche (C), avant d'obliquer légèrement vers la caméra droite (D). Puis il part vers sa gauche et sort du secteur de vision stéréoscopique (E).

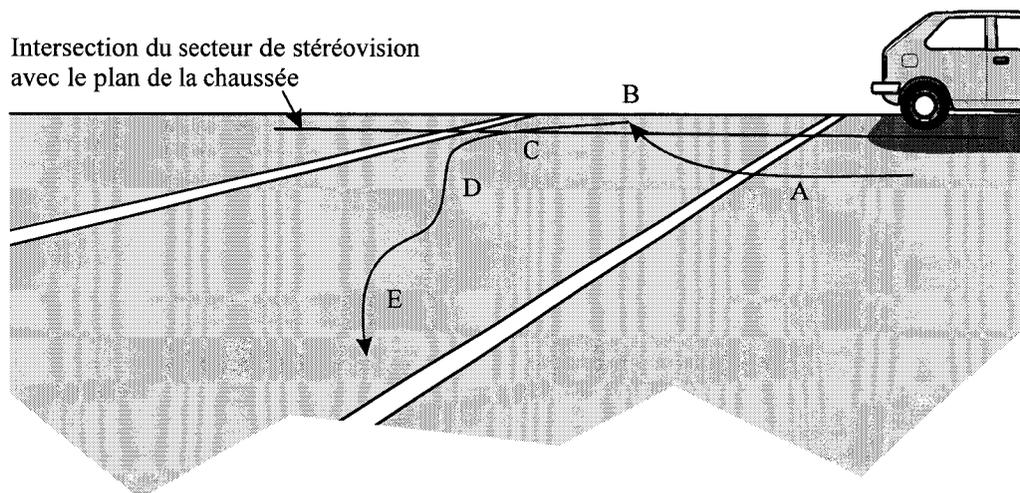


Figure III-15. Description de la scène filmée de la figure III-14.

### III-5.2.2 Exemple 2 : Route dégagée et passage d'une sortie d'autoroute

Cette séquence a été prise par beau temps sur autoroute avec le stéréoscope linéaire installé sur un véhicule. Elle représente la chaussée sans obstacle (Cf. Figure III-16). Le triangle gris que l'on voit sur la droite est l'image du marquage au sol qui délimite une sortie d'autoroute. Le ruban gris est l'image de la voie qui quitte l'autoroute. Aucun relief n'apparaît, tout ce que l'on voit sur cette séquence est situé dans le plan de la chaussée.

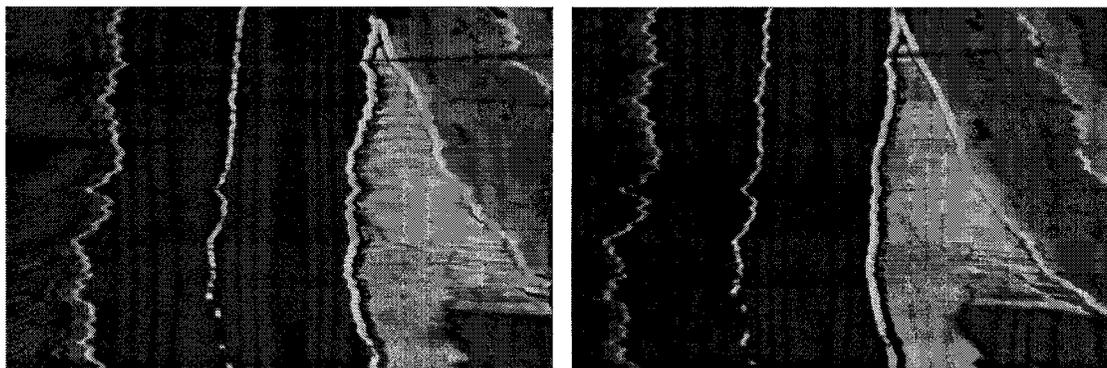
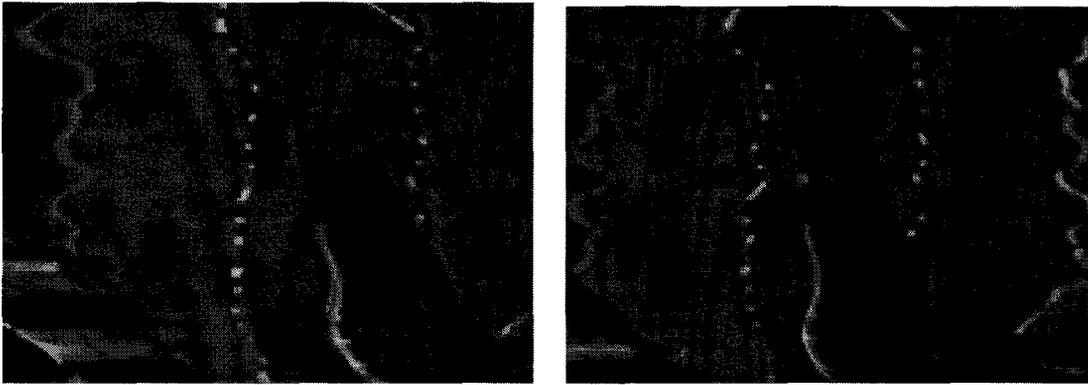


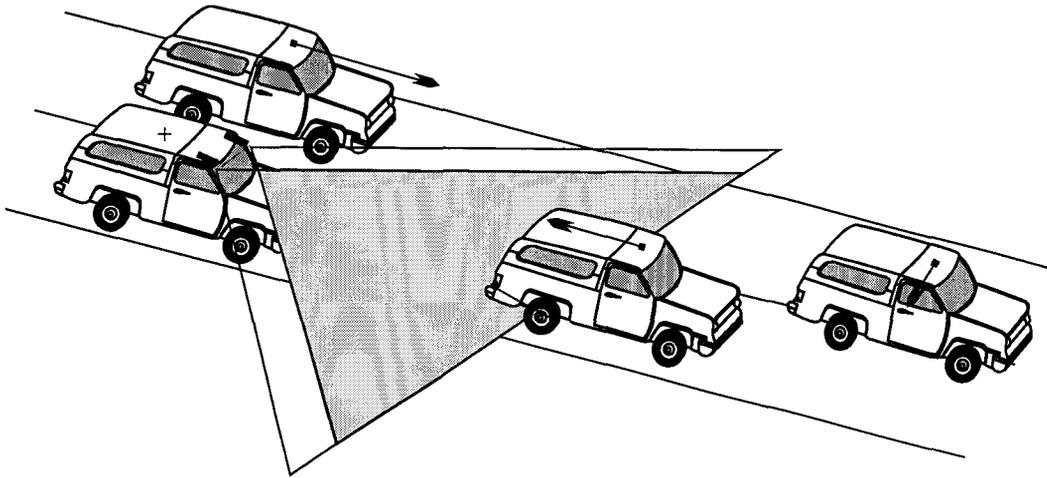
Figure III-16. Passage d'une sortie d'autoroute.

### III-5.2.3 Exemple 3 : Approche d'un véhicule

Le troisième exemple concerne une scène un peu plus complexe que les deux précédentes. La séquence correspondante est représentée sur la figure III-17. Afin d'aider le lecteur à l'interprétation, la figure III-18 donne une description schématique de la scène filmée.



*Figure III-17. Approche d'un véhicule.*



*Figure III-18. Les déplacements des différents véhicules au cours de la séquence. Les flèches indiquent les mouvements relatifs des véhicules par rapport au véhicule prototype (marqué d'une croix).*

Le véhicule prototype roule derrière une *R4* en se rapprochant d'elle. Le plan de visée croise l'arrière de la *R4* à une hauteur croissante à mesure que la distance décroît, ce qui explique l'inversion de l'image de la *R4*, dont on commence à voir les roues pour remonter jusqu'au toit. Noter également son aspect évasé, dû à la décroissance de la distance.

Devant la *R4* se trouve un autre véhicule, en train de se rabattre après l'avoir dépassée. Par ailleurs, le véhicule prototype est lui-même dépassé par une voiture qui apparaît à la fin de la séquence, et qui n'entre pas entièrement dans le secteur de stéréovision.

Enfin, on remarquera le mouvement de tangage du véhicule prototype, visible au début de la séquence sous la forme d'une variation de la largeur de l'image de la chaussée.

### III-6 CONCLUSION

Au cours de ce chapitre, nous avons présenté les différentes étapes de la réalisation du stéréoscope linéaire développé par Jean-Luc Bruyelle. A ce stade, nous disposons d'un système de prise de vue stéréoscopique avec les éléments suivants :

- Une méthode de calibrage permettant de fixer les caractéristiques géométriques des caméras du stéréoscope.
- Des relations géométriques permettant de déterminer la position des objets à partir de leurs images sur les capteurs gauche et droit.
- Un dispositif permettant l'enregistrement de séquences d'images linéaires qui serviront de base de tests pour le développement de nos algorithmes de traitement et d'analyse.

Comme tous les systèmes stéréoscopiques, le traitement commence par l'extraction des indices visuels, appelés primitives, dans chaque image du couple stéréoscopique. Ces primitives serviront comme entrée à la procédure de mise en correspondance. Dans le chapitre qui suit, nous décrivons la méthode que nous avons retenue pour extraire les primitives des images linéaires.

## **Chapitre IV**

# **Extraction des primitives**

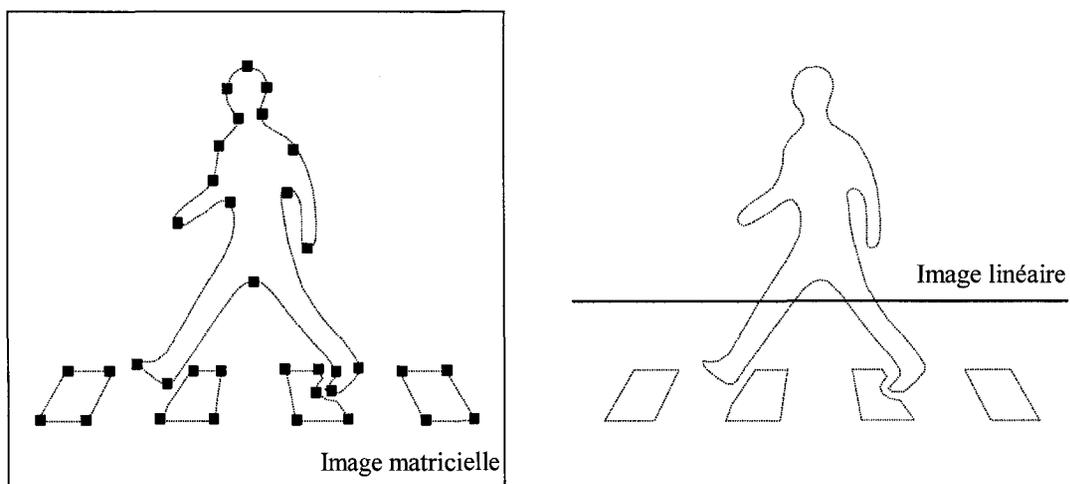
## IV-1 INTRODUCTION

Bien que le stéréoscope linéaire que nous avons présenté dans le chapitre précédent soit différent des systèmes stéréoscopiques usuels matriciels, il a été créé dans le même but, à savoir percevoir la structure tridimensionnelle de l'environnement dans lequel il évolue. La démarche pour aboutir à ce but est la même que celle utilisée en stéréovision classique matricielle. Etant donné un couple d'images linéaires stéréoscopiques, la première étape consiste à mettre en évidence, dans chaque image, les primitives à appairer. Nous présentons, dans ce chapitre, la méthode que nous avons utilisée pour cette phase de traitement préalable.

## IV-2 CHOIX DU TYPE DE PRIMITIVE

La première étape d'un processus de stéréovision consiste à extraire, dans chaque image, les primitives à appairer. La nature de ces primitives et les différents attributs qu'on peut leur associer jouent un grand rôle lors de la phase d'appariement, ainsi que pour la reconstruction de la scène. Nous avons présenté, au chapitre II, les différentes primitives utilisées en vision matricielle. Dans le cas d'images linéaires, le choix des primitives se trouve restreint en raison de la nature mono-dimensionnelle du signal.

La première possibilité envisageable est la recherche de primitives de type point. Mais cette recherche se trouve limitée puisque les notions de point de courbure ou de coin n'ont plus aucune existence dans un espace à une dimension (Cf. Figure IV-1).

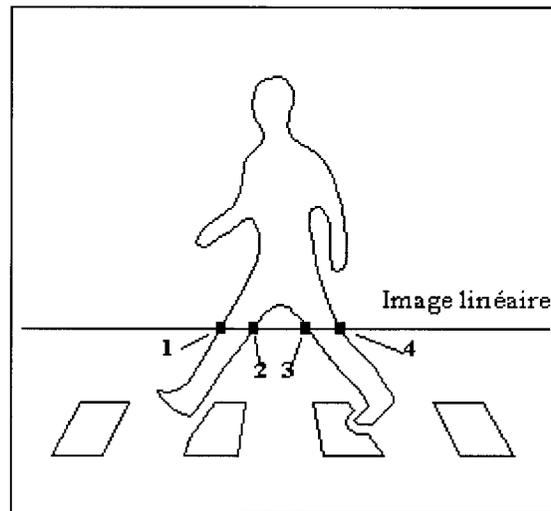


a) Exemple de points d'intérêts :  
les points de fortes courbures

b) Aucune primitive de type point d'intérêt  
dans un espace monodimensionnel

*Figure IV-1. Limitation de la recherche de primitives de type point.*

L'emploi d'images linéaires restreint donc considérablement les points particuliers à extraire. Ainsi, seuls les points de contour, correspondant aux frontières des différents objets présents dans l'image, peuvent être extraits (Cf. Figure IV-2).



*Figure IV-2. Primitives de type point de contour.*

La recherche des points de contour consiste à détecter les variations locales de l'intensité de l'image. Il faut toutefois noter que la définition des points de contour en vision linéaire est différente de celle donnée en vision matricielle, puisqu'un point de contour est considéré comme un point unique et non comme une partie d'un contour. Aussi, les propriétés géométriques couramment attribuées au point de contour "matriciel", comme la direction du contour, ne peuvent être définies en vision linéaire.

Nous pouvons aussi envisager un autre type de primitive : les régions. Elles sont définies comme des parties de l'image linéaire, donc des segments de droite, où le niveau de gris est uniforme. Deux régions sont distinctes si leurs niveaux de gris moyens diffèrent notablement. Ainsi, toute région pourra être limitée par deux points de contour et chaque point de contour servira de frontière à deux régions adjacentes. Il en résulte alors une dualité entre régions et contours.

En conclusion, quelque soit le type de primitive utilisé, il nécessite la détection des points de contour présents dans l'image, c'est à dire des variations locales importantes de l'intensité lumineuse. Dans les deux cas, les points de contour seront les éléments clefs de la procédure d'appariement. De plus, il est logique de choisir ces points comme des éléments très importants, car ils correspondent aux frontières des différents objets présents dans l'image.

## IV-3 CHOIX DU DETECTEUR DE CONTOURS

Il existe de nombreux algorithmes de détection de contours qui peuvent être aussi bien utilisés sur des images matricielles que sur des images linéaires. L'adaptation de ces algorithmes d'un espace à deux dimensions à un espace à une dimension ne pose généralement aucun problème. Il faut également préciser que toutes ces procédures de détection ne fournissent pas les mêmes résultats. Certaines sont plus performantes en termes de localisation, de détection ou de robustesse par rapport au bruit, d'autres sont plus rapides. Devant tant d'outils disponibles, il est donc nécessaire de faire un choix, ou plutôt de définir des critères de sélection permettant de réaliser ce choix.

### IV-3.1 Contraintes à respecter par le détecteur de contours

Le détecteur de contours doit se plier à certaines contraintes :

- Détecter tous les contours : la notion de contours se confond en général avec celle de discontinuité de la fonction image. Cependant, ces discontinuités ne sont pas toujours nettes, et il existe plusieurs types de modèles de contours (Cf. Figure IV-3). Le détecteur choisi doit donc être capable de tous les mettre en évidence. En effet, dans le cadre de notre application, nous ne pouvons nous permettre de négliger un obstacle dangereux, simplement parce que ses contours n'ont pas été détectés.

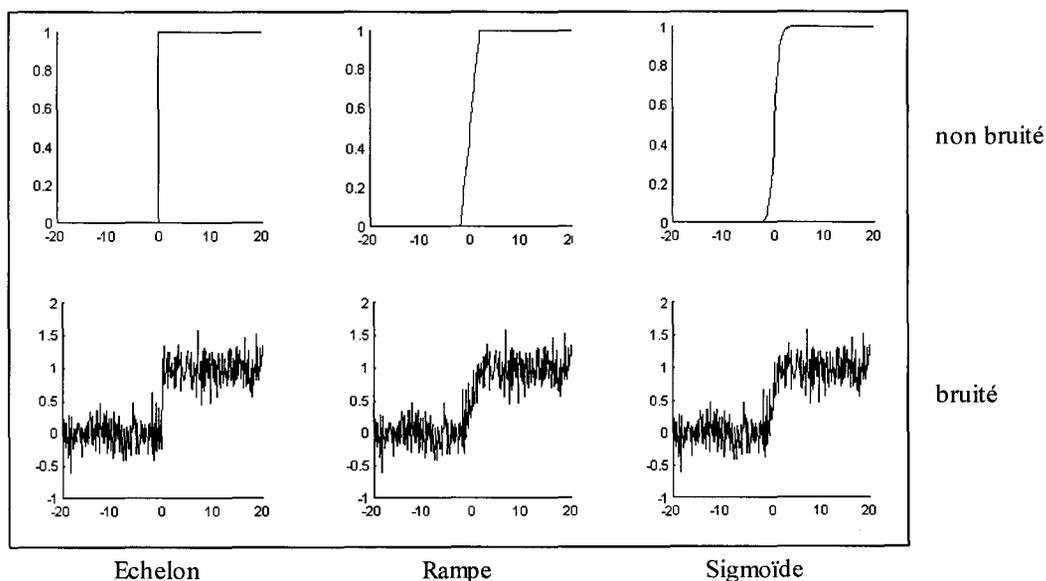


Figure IV-3. Profil de différents types de contours.

- Localiser les contours avec précision : dans le chapitre précédent, nous avons vu comment il était possible, par simple triangulation, de déterminer la distance et la

position des objets grâce au stéréoscope linéaire (Cf. § III-4.3). Aussi, la position d'un contour dans une image, c'est à dire la position du pixel où est détecté ce contour, influe directement sur le calcul de la distance, c'est à dire la profondeur des objets par rapport au stéréoscope. Ainsi, toute erreur de localisation d'un contour aura des conséquences sur la reconstruction 3D de la scène observée. Ce type d'erreur conduirait à une mauvaise interprétation de l'environnement du stéréoscope, dont les conséquences pourraient être graves dans le cas de l'application à la détection d'obstacles. Le détecteur utilisé devra donc localiser aussi précisément que possible les contours afin d'éviter des erreurs de localisation des objets dans la scène tridimensionnelle.

- Ne pas créer de faux contours : le détecteur devra être aussi peu sensible au bruit que possible. En effet, de fausses détections pourraient conduire à localiser des objets là où il n'y en a pas, ce qui diminuerait la fiabilité du système de détection. D'autre part, l'ajout de faux contours à la liste des vrais points de contour augmenterait le nombre d'appariements possibles. Il en résulterait un accroissement du temps de traitement avec, certainement, une augmentation des possibilités d'erreur et de faux appariements.
- Etre implantable en temps réel : le dernier critère à considérer concerne le temps de calcul. Il faudra que l'extraction des points de contour puisse se faire le plus rapidement possible pour ne pas affecter les performances du système de détection d'obstacles, qui devra opérer en temps réel.

### IV-3.2 Pretraitement de l'image

La détection des contours dans une image fait appel à des opérateurs spatiaux qui répondent à des variations locales de l'intensité. Des discontinuités brutales peuvent être détectées en marquant les points de l'image correspondant à des valeurs élevées d'un estimateur des dérivées spatiales de l'image [Roberts 65][Prewitt 70]. La différentiation d'une image numérique peut s'avérer une tâche délicate dans le sens où ses performances se détériorent rapidement en présence de bruit. En effet, nous avons vu au § IV-3.1 que le bruit peut générer des faux contours. De même, il peut être la cause d'une mauvaise localisation des contours puisqu'il peut les déplacer légèrement lorsqu'il s'ajoute au signal. Ces deux constatations vont à l'encontre de la deuxième et la troisième contraintes auxquelles doit se plier le détecteur de

contours (Cf. § IV-3.1). Il est donc impératif d'effectuer un pré-traitement de l'image pour diminuer au maximum le niveau de bruit.

Pour rendre la différentiation robuste, la technique dite de régularisation est utilisée [Ziou 91]. Il s'agit d'imposer de nouveaux critères au problème de différentiation pour restreindre l'ensemble des solutions possibles et choisir celle qui minimise une fonctionnelle adéquate. Tikhonov et Arsenin, cités dans [Ziou 91], montrent que la régularisation numérique du problème de différentiation peut être effectuée simplement en lissant l'image [Tikhonov 77].

Ainsi, avant de différencier l'image, celle-ci est préalablement lissée afin de diminuer l'influence du bruit sur la détection des contours. Ce pré-traitement est souvent effectué par une simple convolution de l'image avec un filtre passe-bas.

Il faut cependant noter que cette opération de lissage a pour effet indésirable de modifier la structure de l'image : élimination et déplacement de certains contours et création de nouveaux contours. Il faudra alors établir un compromis entre l'élimination du bruit et la préservation des contours.

### IV-3.3 Conclusion

Un détecteur de contours opère sur une image discrète pour mettre en évidence les contours présents dans cette image, c'est à dire les variations locales de l'intensité de l'image. Nous avons vu au § IV-3.1 que le détecteur de contours le plus adapté doit se plier aux contraintes suivantes :

- Détecter tous les contours.
- Localiser les contours avec précision.
- Ne pas créer de faux contours.
- Etre implantable en temps réel.

Basée sur la différentiation de l'image, la détection des contours peut s'avérer impossible en présence de bruit puisque les opérateurs différentiels, tout en rehaussant les contours, augmentent aussi le niveau de bruit. Pour rendre cette opération robuste, nous avons vu qu'il est nécessaire d'effectuer un lissage préalable afin de diminuer l'influence du bruit. Le détecteur de contours s'organise ainsi de la façon suivante :

- Le lissage : il a pour objectif la réduction de la quantité de bruit contenu dans l'image. Cette phase de pré-traitement est réalisée souvent par une simple convolution de l'image avec un filtre passe-bas.

- La différentiation : il s'agit de calculer la dérivée de la fonction niveaux de gris de l'image afin de mettre en évidence les contours.

La présentation de ces deux opérations, nécessaires à la détection des contours, fait l'objet du paragraphe suivant.

## IV-4 DETECTION DES CONTOURS

La procédure de détection des contours consiste à chercher les variations locales et significatives de la fonction intensité de l'image. Pour mettre en évidence ces variations, on utilise des opérateurs différentiels.

L'aspect monodimensionnel des signaux représentant les images linéaires nous amène à ne décrire les détecteurs que dans leur version monodimensionnelle. Le lecteur intéressé par une utilisation bidimensionnelle pourra trouver de plus amples informations dans les références citées.

### IV-4.1 Les masques de convolution

Les masques de convolution permettent de détecter les contours par approximation du gradient tout en réduisant le bruit. Comme les performances des différentes méthodes dépendent de la nature des images, leur sélection se fait généralement par essais successifs.

#### IV-4.1.1 Définitions

Dans le cas d'un signal monodimensionnel, un masque prend la forme d'une matrice de coefficients à une seule ligne. L'opération de convolution d'un masque et d'une image s'effectue de la manière suivante :

Soit  $Q$  un masque de dimension  $2w+1$  :

$$Q = [q_{-w} \quad \cdots \quad q_{-1} \quad q_0 \quad q_1 \quad \cdots \quad q_w]$$

où les  $q_i$  sont les coefficients du masque. Le nombre  $w$ , qui permet de déterminer la taille du masque, est choisi supérieur ou égal à 1.

Soit  $V$  un ensemble de  $2w+1$  pixels de l'image, appelé voisinage, centré sur le pixel considéré  $p_0$  :

$$V = [p_{-w} \quad \cdots \quad p_{-1} \quad p_0 \quad p_1 \quad \cdots \quad p_w] \quad .$$

Soit  $G$  l'ensemble des niveaux de gris correspondant à ces pixels:

$$G = [g_{-w} \quad \cdots \quad g_{-1} \quad g_0 \quad g_1 \quad \cdots \quad g_w] ,$$

où les  $g_i$  sont les niveaux de gris des pixels  $p_i$ .

Le filtrage par convolution consiste à remplacer le niveau de gris  $g_0$  du pixel  $p_0$  de l'image par le niveau de gris  $g'_0$ , calculé de la façon suivante :

$$g'_0 = \sum_{i=-w}^w q_i \cdot g_i .$$

Cette opération est effectuée pour tout les pixels de l'image pour obtenir l'image filtrée.

On constate que le masque doit être de même taille que le voisinage. Afin que le filtrage soit symétrique par rapport au pixel  $P$  considéré, le masque est constitué d'un nombre impair d'éléments. Généralement, le choix se porte sur des masques de taille assez petite, d'environ 3 à 5 pixels.

Selon les valeurs des coefficients  $q_i$ , les masques de convolution permettent d'effectuer diverses opérations de filtrage.

#### ***IV-4.1.2 Opération de lissage***

Comme nous l'avons déjà mentionné au § IV-3.2, les images doivent être préalablement traitées afin de diminuer l'influence du bruit sur la détection des contours. Il existe de nombreux opérateurs de lissage basés sur des masques de convolution. Le filtre moyenneur est certainement le plus connu des filtres de convolution. Les coefficients du masque correspondant, pour  $w = 1$ , prennent les valeurs suivantes :

$$Q = \frac{1}{3} [1 \quad 1 \quad 1] ,$$

où le coefficient  $1/3$  sert à normaliser le résultat.

Si ce filtre a pour effet de moyennier le bruit sur plusieurs pixels, il atténue également les variations brusques de l'intensité. De ce fait, les contours sont moins nets, ce qui se traduit visuellement par un effet de flou.

Des variantes de ce filtre sont possibles. On peut pondérer les coefficients du masque de façon à donner plus d'importance au pixel central. On utilise par exemple le masque :

$$Q = \frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \end{bmatrix} ,$$

où le coefficient  $1/4$  sert à normaliser le résultat.

L'opération de lissage peut être effectuée aussi par un filtrage médian. Ce type de filtrage s'applique sur un voisinage donné mais sans opération de convolution. Il s'agit d'un filtre non linéaire.

Soit un voisinage de dimension  $2w+1$  d'un pixel  $p_0$  de l'image. Le filtrage médian consiste à ordonner l'ensemble des  $2w+1$  pixels, constitué du pixel  $p_0$  et ses  $2w$  voisins, selon l'ordre croissant de leurs niveaux de gris. On affecte alors au pixel  $p_0$  considéré la valeur médiane de l'ensemble des niveaux de gris ainsi ordonnés.

Le filtre médian produit un effet voisin de celui du filtre moyenneur, mais il respecte beaucoup mieux les variations brutales du niveau de gris, c'est à dire les contours.

Il est important de noter que plus la taille de ces filtres est élevée, plus l'effet du filtrage passe-bas sur le bruit est efficace.

#### ***IV-4.1.3 Opération de détection***

Le filtre de détection le plus simple est de taille 3. Son masque de convolution présente les coefficients suivants :

$$Q_1 = \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} .$$

On peut étendre ce filtre sur 5 pixels. On obtient alors :

$$Q_2 = \begin{bmatrix} -1 & 0 & 0 & 0 & 1 \end{bmatrix} .$$

Si on utilise le filtre :

$$Q_3 = \begin{bmatrix} -1 & -1 & 0 & 1 & 1 \end{bmatrix}$$

on introduit également un effet de moyennage.

Le filtre  $Q_1$  donne d'excellents résultats pour des contours où la zone de transition entre deux niveaux de gris est étroite. Par contre si la largeur du contour dépasse la taille du masque, le filtre sera nettement moins efficace. On constate alors que ce type de filtre est très sélectif, et ne

met en évidence qu'un certain type de contour. Il en est de même pour le filtre  $Q_2$  dont la détection est efficace pour des contours de largeur inférieure à 5 pixels.

Pour résoudre ce problème, Rosenfeld propose d'utiliser plusieurs filtres de tailles croissantes, chacun mettant en évidence des contours d'une certaine largeur [Rosenfeld 71][Rosenfeld 72]. Il propose ensuite de retenir le contour correspondant à la valeur maximale des réponses des différents filtres utilisés. Les résultats fournis par cette technique sont excellents, mais sa complexité algorithmique dépend du nombre de filtres utilisés. Si on choisit un grand nombre de filtres, les contours de grande largeur seront mieux mis en évidence, mais au prix d'une grande quantité de calculs.

#### ***IV-4.1.4 Conclusion sur les masques de convolution***

Nous venons de voir qu'un moyen très simple pour lisser une image ou calculer son gradient consiste à utiliser des masques de convolution. Cependant, l'efficacité de ces filtres dépend de la taille du masque utilisé.

Dans le cas d'une opération de lissage, plus la taille du filtre est grande, plus il atténue le bruit. Mais ce résultat est obtenu au détriment des contours qui sont alors moins nets, et qui risquent, en conséquence, de ne plus être détectés. Dans le cas du calcul du gradient, un second paramètre entre en jeu, à savoir le nombre de masques à utiliser, puisque tous les contours ne peuvent être détectés avec un seul type de masque. Il faut également rappeler que plus la taille des masques est importante, plus le temps de calcul augmente, ce qui va à l'encontre de la dernière contrainte citée au § IV-3.1.

Le choix arbitraire de la taille des masques et le temps de calcul important qui peut en résulter nous ont conduit à ne pas retenir ce type de filtre.

#### **IV-4.2 Les détecteurs de contours optimaux**

Plusieurs travaux ont été consacrés au problème de la recherche d'un détecteur optimal et différents critères ont été proposés pour estimer, sinon quantifier, les performances des détecteurs de contours. Les procédures associées à ces critères visent à améliorer la détection des contours en définissant des filtres qui optimisent ces critères de qualité. Ce type d'approche semble le plus adéquat et connaît un succès croissant. Parmi les plus connus de ces travaux, nous citons, au cours de cet exposé, ceux de Canny et de Deriche.

### IV-4.2.1 Approche de Canny

Canny a étudié les propriétés souhaitées pour un détecteur de contours performant [Canny 83][Canny 86]. Il a proposé une méthode pour déterminer un détecteur de contours en se basant sur les trois critères suivants :

1. **La qualité de détection.** Le critère de détection permet de vérifier l'efficacité du détecteur de contours en tant que filtre appliqué sur un signal. La qualité de détection est d'autant plus grande que le filtre est insensible au bruit présent dans l'image traitée.
2. **La précision de localisation.** Le critère de localisation permet de quantifier la précision de la position du contour détecté dans une image bruitée par rapport à la position réelle de celui-ci.
3. **L'obtention d'une réponse unique pour un contour donné.** Le critère de réponses multiples sert à déterminer si le filtre est susceptible de fournir plusieurs réponses pour un seul contour présent dans l'image.

Canny fut le premier à spécifier à la fois la qualité de détection et la précision de localisation d'un contour sous une forme mathématique (Cf. § IV-4.2.1.1). Il a proposé, dans [Canny 83], de quantifier la qualité de détection par la probabilité de ne pas omettre un vrai contour et de ne pas marquer *contour* un point image qui ne l'est pas. La précision de la localisation est quantifiée par la variance de la distance entre le vrai contour et le point image marqué *contour*. Un troisième critère mathématique a été ajouté aux deux premiers, afin de quantifier les possibilités d'apparition de réponses multiples à un contour unique.

#### IV-4.2.1.1 Formulation mathématique des critères de performance

Dans [Canny 86], le problème de la détection de contours est formulé dans le cas monodimensionnel de la façon suivante : un signal est convolué avec un filtre de telle sorte qu'un contour puisse être marqué au niveau d'un maximum local du résultat de la convolution.

Soit  $f(x)$  la réponse impulsionnelle du filtre et soit  $G(x)$  le signal contenant le contour, sur lequel est superposé un bruit gaussien  $N(x)$  de moyenne nulle et de variance  $N_0$ . Il est supposé que le contour est centré en  $0$ . La réponse  $y(x)$  du filtre au signal en un point  $x_e$  est donnée par l'opération de convolution suivante :

$$y(x_e) = \int_{-\infty}^{+\infty} G(x_e - x) f(x) dx \quad \text{Equation IV-1}$$

Selon Canny, le problème de la détection de contours peut être assimilé à la recherche du filtre qui présente les meilleures performances, par rapport aux trois critères définis ci-après :

**Critère de détection :**

Pour quantifier la qualité de détection du contour, Canny considère le rapport signal sur bruit, noté  $S/B$ , comme le rapport de la valeur absolue de la réponse du filtre  $f(x)$  au signal  $G(x)$  en  $x_e = 0$  sur la réponse moyenne du filtre au bruit  $N(x)$ . Ce rapport peut être alors défini par :

$$S/B = \frac{\left| \int_{-\infty}^{+\infty} G(-x) f(x) dx \right|}{N_0 \sqrt{\int_{-\infty}^{+\infty} f^2(x) dx}} \quad \text{Equation IV-2}$$

Puisque les probabilités de mauvaise détection et de détection manquée diminuent lorsque le rapport  $S/B$  augmente, le critère de détection correspond à la maximisation de ce rapport.

Pour un contour idéal de type échelon défini par :

$$G(x) = \begin{cases} A & \text{si } x \geq 0 \\ 0 & \text{sinon} \end{cases} \quad \text{Equation IV-3}$$

le critère de détection devient :

$$\Sigma = \frac{\left| \int_{-\infty}^0 f(x) dx \right|}{\sqrt{\int_{-\infty}^{+\infty} f^2(x) dx}} \quad \text{Equation IV-4}$$

**Critère de localisation :**

Comme nous l'avons déjà mentionné au § IV-4.2.1, le critère de localisation indique la capacité du détecteur à extraire un contour du signal bruité qui soit situé aussi près que possible du contour réel. Pour ce critère, Canny propose de maximiser l'inverse de la variance de la distance entre le contour détecté et le contour réel. Cette mesure de localisation est définie par :

$$L = \frac{\left| \int_{-\infty}^{+\infty} G'(-x) f'(x) dx \right|}{N_0 \sqrt{\int_{-\infty}^{+\infty} f'^2(x) dx}}$$

Equation IV-5

Pour le contour en échelon idéal défini précédemment (Cf. Equation IV-3), le critère de localisation devient :

$$\Lambda = \frac{|f'(0)|}{\sqrt{\int_{-\infty}^{+\infty} f'^2(x) dx}}$$

Equation IV-6

### **Critère de réponses multiples :**

Le filtre  $f(x)$  ne doit fournir en réponse qu'un seul point marqué lorsqu'il n'existe qu'un seul contour effectif. Afin de minimiser le nombre de réponses multiples du filtre à un contour unique, Canny propose d'utiliser la distance moyenne entre les passages par zéro de sa dérivée  $f'(x)$ , donnée par :

$$M = \pi \sqrt{\frac{\int_{-\infty}^{+\infty} f'^2(x) dx}{\int_{-\infty}^{+\infty} f''^2(x) dx}}$$

Equation IV-7

Ainsi, plus cette distance moyenne est importante, moins le détecteur fournit de maxima incorrects dus au bruit.

#### *IV-4.2.1.2 Optimisation du détecteur de contours*

Afin de simplifier le problème, Canny a imposé que le détecteur  $f(x)$  soit à support fini, c'est à dire qu'il doit vérifier la contrainte :

$$f(x) = 0 \quad \text{si } x \notin [-W, W]$$

où  $W$  est une constante réelle positive marquant la limite de l'intervalle où  $f(x)$  est défini. Il a aussi imposé une contrainte d'anti-symétrie au filtre  $f(x)$ , c'est à dire que  $f(-x) = -f(x)$ . Cette contrainte est due simplement au choix de la dérivée première comme opérateur de différentiation.

Selon Canny, un "bon" détecteur de contours doit maximiser les trois critères définis au § IV-4.2.1.1. Afin de simplifier les calculs nécessaires à la détermination d'un détecteur optimal dans le cas de contours en échelon, il a réduit la procédure de maximisation des deux premiers critères à la maximisation de leur produit  $\Sigma \cdot \Lambda$ . Le critère de réponses multiples est utilisé en tant que contrainte dans la procédure d'optimisation en imposant que la distance maximale entre deux maxima adjacents dans la réponse du filtre au bruit soit une constante, égale à  $k$  fois l'étendue  $W$  du filtre.

Cette procédure d'optimisation sous contrainte a permis de montrer que le filtre maximisant le produit  $\Sigma \cdot \Lambda$  en respectant la contrainte concernant les réponses multiples a une réponse impulsionnelle relativement bien approchée par la dérivée d'une Gaussienne :

$$f_{can} = Axe^{-\alpha^2 x^2} \quad \text{Equation IV-8}$$

où  $\alpha$  est un paramètre qui permet de régler l'étendue efficace du filtre. Le coefficient de normalisation  $A$  est calculé de telle sorte que la réponse du filtre à un signal d'amplitude unité soit elle-même d'amplitude unité.

Notons au passage qu'il a été montré que ce type de filtre est optimal selon un autre critère, basé sur l'analyse stochastique du nombre de passages par zéro de la dérivée de la réponse [Tagare 86].

La figure IV-4 présente le profil de la réponse impulsionnelle du filtre optimal de Canny.

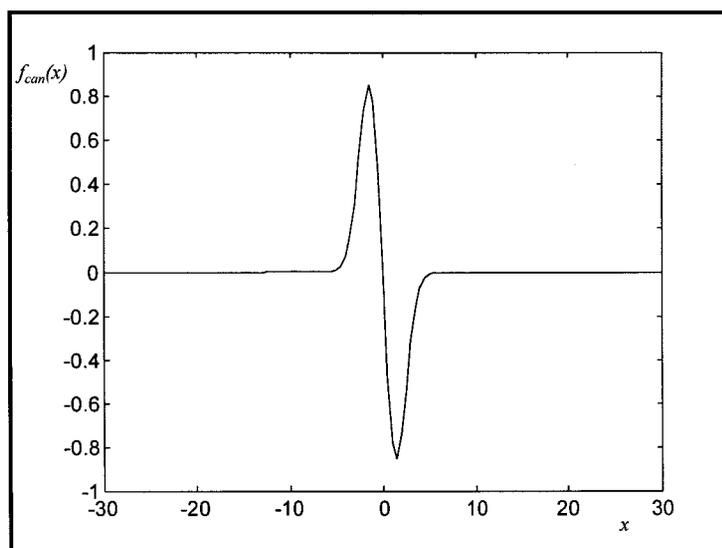


Figure IV-4. Réponse impulsionnelle du filtre optimal de Canny.

### IV-4.2.2 Approche de Deriche

Afin d'améliorer les résultats obtenus par Canny, Deriche [Deriche 90b] a proposé une famille de filtres, basés sur la fonction sinusoïdale amortie, qui peuvent être utilisés pour lisser les signaux ou calculer leurs dérivées. Tout comme le filtre de Canny, celui de Deriche est conçu afin de présenter un indice de performance optimal en termes de détection et de localisation des contours, tout en satisfaisant la contrainte de réponses multiples.

#### IV-4.2.2.1 Détecteur optimal de contours

Deriche a repris les trois critères de performance de Canny [Deriche 87b], mais propose d'étendre le filtre à l'infini, c'est à dire  $W \rightarrow \infty$ . En utilisant les conditions limites suivantes :

$$f(0) = 0, \quad f(\infty) = 0, \quad f'(0) = S, \quad f'(\infty) = 0, \quad \text{où } S \text{ est une constante,}$$

Deriche obtient le filtre de détection de contour dont la réponse impulsionnelle est une sinusoïde amortie :

$$f_{der} = Ae^{-\alpha|x|} \sin(\omega x), \quad \text{Equation IV-9}$$

où  $\omega$  est une constante positive. Le coefficient de normalisation  $A$  permet de garantir une réponse égale à 1 pour un contour en forme d'échelon unitaire. Le paramètre  $\alpha$  permet de régler l'étendue efficace du filtre.

Dans son étude, Deriche a constaté que le filtre présentant les meilleures performances en termes de détection et de localisation du contour est obtenu en faisant tendre  $\omega$  vers zéro, ce qui correspond à  $\sin(\omega x) \approx \omega x$ . Il obtient alors le détecteur optimal :

$$d(x) = Bxe^{-\alpha|x|}, \quad \text{Equation IV-10}$$

où  $B$  est un coefficient de normalisation et  $\alpha$  est un paramètre qui permet de régler l'étendue efficace du filtre.

La figure IV-5 représente le profil de la réponse impulsionnelle du filtre optimal de Deriche.

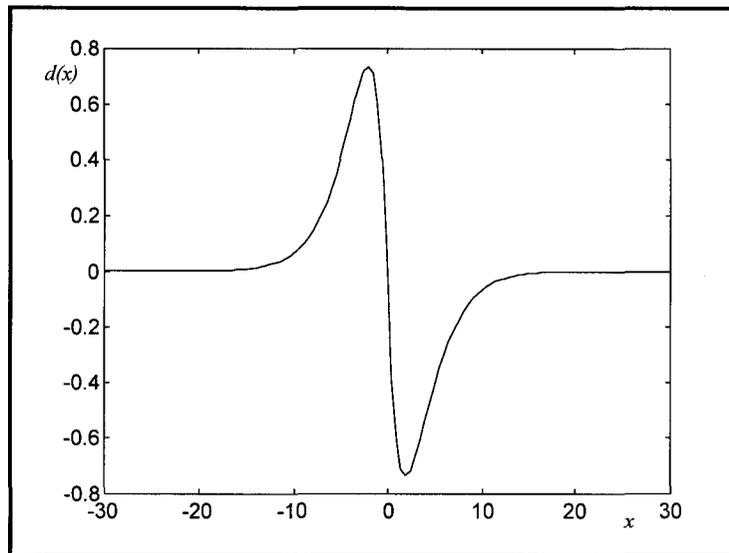


Figure IV-5. Profil du détecteur optimal de Deriche.

IV-4.2.2.2 Opérateur de lissage

Il s'agit maintenant de déterminer un opérateur de lissage qui permet de limiter les effets du bruit. A partir du détecteur  $d(x)$ , Deriche propose un opérateur de lissage  $l(x)$  qui n'est autre que l'intégrale de l'opérateur de dérivation  $d(x)$  :

$$l(x) = C(\alpha|x| + 1)e^{-\alpha|x|}, \tag{Equation IV-11}$$

où  $\alpha$  est un paramètre qui permet de régler l'étendue efficace de l'opérateur de lissage. La constante réelle  $C$  représente le coefficient de normalisation. L'allure de l'opérateur de lissage est représentée sur la figure IV-6.

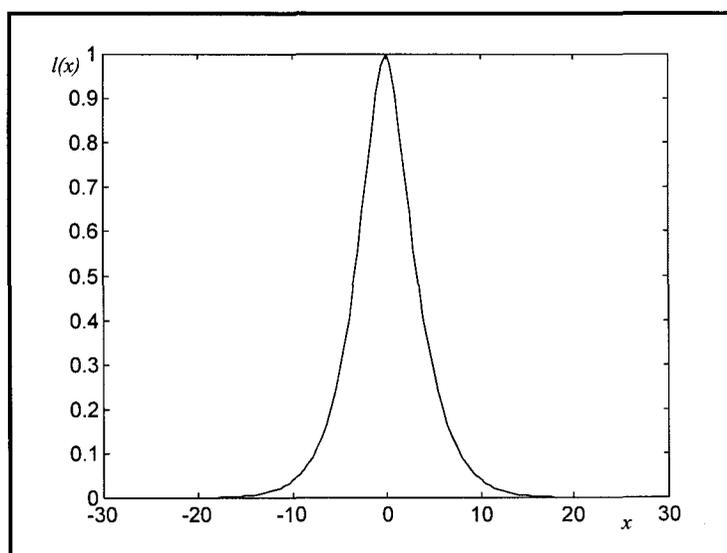


Figure IV-6. Profil de l'opérateur de lissage  $l(x)$ .

### IV-4.2.2.3 Influence du paramètre $\alpha$

La figure IV-7 illustre l'effet de l'ajustement du paramètre  $\alpha$  sur le comportement des deux filtres  $l(x)$  et  $d(x)$ . On constate que plus  $\alpha$  est petit, plus l'étendue des opérateurs est importante.

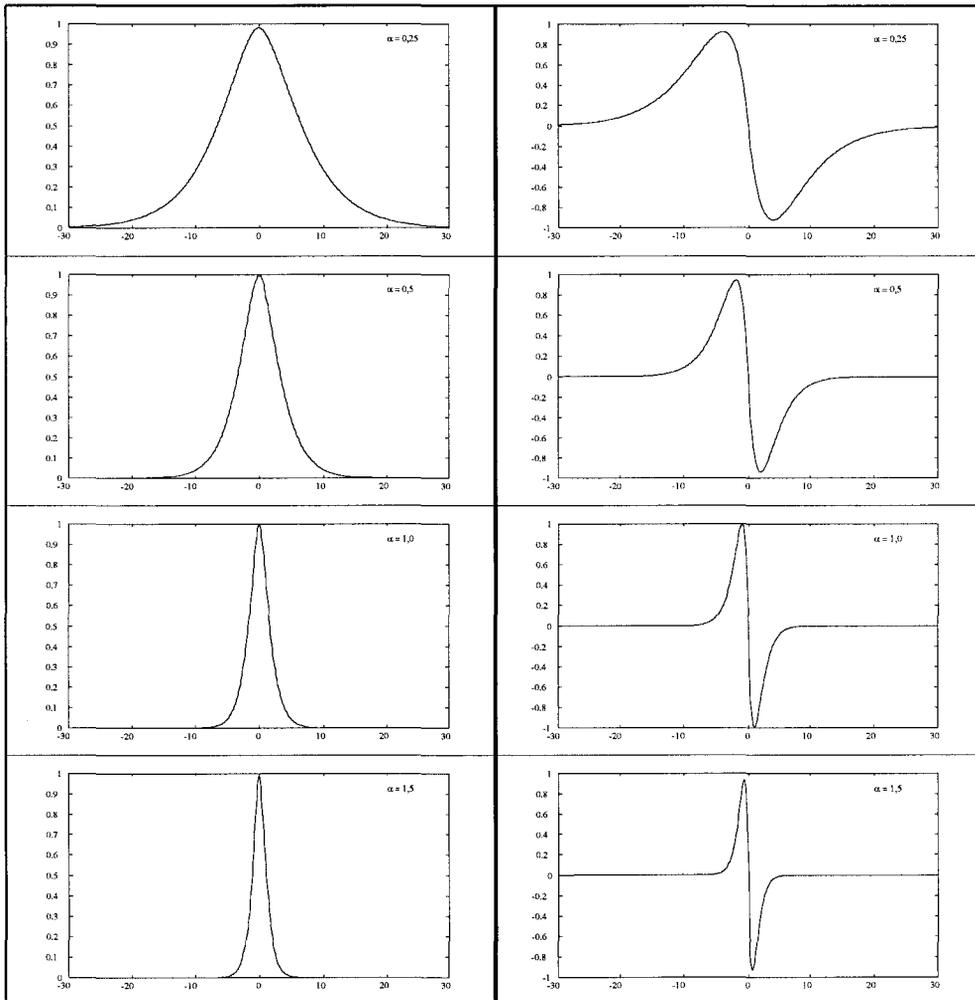


Figure IV-7. Influence de  $\alpha$  sur l'étendue des opérateurs  $l(x)$  (à gauche) et  $d(x)$  (à droite).

### IV-4.2.3 Conclusion

Pour notre module de détection de contours dans les images linéaires, nous avons retenu les filtres  $l(x)$  et  $d(x)$  de Deriche. Ce choix nous permet de satisfaire les contraintes mentionnées au § IV-3. En effet, ces filtres ont été développés en se basant sur les critères de performance de Canny. De plus, Deriche a prouvé que son détecteur localise mieux les contours que celui de Canny.

Avant de mettre en oeuvre la détection des contours, il reste un point important à prendre en considération : le temps de calcul. L'opérateur, bien que performant, doit être également rapide pour les raisons citées au § IV-3.1. Afin que les opérations de lissage et de dérivation

soient faites le plus rapidement possible, nous avons implanté les filtres  $l(x)$  et  $d(x)$  selon le schéma récursif proposé par Deriche. Ceci fera l'objet du § IV-4.3.

### IV-4.3 Implantation de la détection des contours

Deriche propose dans [Deriche 90b] une implantation récursive de ses opérateurs  $d(x)$  et  $l(x)$ . Le principal avantage d'un filtre récursif réside dans la diminution du nombre total d'opérations élémentaires nécessaires à l'évaluation de la réponse par rapport à un filtre non récursif de propriétés équivalentes.

#### IV-4.3.1 Implantation récursive d'un filtre

Soit  $x(n)$  un signal discret placé en entrée d'un filtre non causal de réponse impulsionnelle discrète  $h(i)$ . La réponse discrète  $y(n)$  du filtre, donnée par :

$$y(n) = \sum_{i=-\infty}^{i=+\infty} x(n-i)h(i), \quad \text{Equation IV-12}$$

peut être considérée comme la superposition de deux réponses :

$$y(n) = y^+(n) + y^-(n). \quad \text{Equation IV-13}$$

La réponse  $y^+(n)$  résulte de la convolution discrète d'un filtre causal  $h^+(i)$  avec le signal d'entrée  $x(n)$  :

$$y^+(n) = \sum_{i=-\infty}^{i=+\infty} x(n-i)h^+(i). \quad \text{Equation IV-14}$$

$y^-(n)$  est la réponse d'un filtre anticausal  $h^-(i)$  à ce même signal d'entrée :

$$y^-(n) = \sum_{i=-\infty}^{i=+\infty} x(n-i)h^-(i). \quad \text{Equation IV-15}$$

Les filtres  $h^+(i)$  et  $h^-(i)$  sont donnés par les expressions suivantes :

$$h^+(i) = \begin{cases} h(i) & \text{si } i \geq 0 \\ 0 & \text{sinon} \end{cases} \quad \text{Equation IV-16}$$

$$h^-(i) = \begin{cases} h(i) & \text{si } i < 0 \\ 0 & \text{sinon} \end{cases} \quad \text{Equation IV-17}$$

Quand la transformée en  $Z$ ,  $F(z)$ , d'un filtre causal est une fraction rationnelle de la forme :

$$F(z) = \frac{\sum_{k=0}^{k=u} a_k \cdot z^{-k}}{1 - \sum_{k=1}^{k=v} b_k \cdot z^{-k}}, \text{ avec } 0 < u < v, \quad \text{Equation IV-18}$$

la réponse  $r(n)$  à de ce filtre au signal d'entrée  $x(n)$  peut être exprimée par l'équation récurrente suivante :

$$r(n) = \sum_{k=0}^{k=u} a_k \cdot x(n-k) + \sum_{k=1}^{k=v} b_k \cdot r(n-k) \quad \text{Equation IV-19}$$

où  $u$  et  $v$  désignent respectivement les degrés du numérateur et du dénominateur de la transformée en  $Z$ ,  $F(z)$ , du filtre.

Si les deux transformées en  $Z$  des filtres  $h^+(i)$  et  $h^-(i)$ , notées respectivement  $F^+(z)$  et  $F^-(z)$ , sont des fractions rationnelles, leurs réponses  $y^+(n)$  et  $y^-(n)$  peuvent être calculées de façon récursive en utilisant des expressions similaires à l'équation IV-19 :

$$y^+(n) = \sum_{k=0}^{k=u^+} a_k^+ \cdot x(n-k) + \sum_{k=1}^{k=v^+} b_k^+ \cdot y^+(n-k) \quad \text{Equation IV-20}$$

$$y^-(n) = \sum_{k=0}^{k=u^-} a_k^- \cdot x(n-k) + \sum_{k=1}^{k=v^-} b_k^- \cdot y^-(n-k) \quad \text{Equation IV-21}$$

Les paramètres  $a_k^+$  (resp.  $a_k^-$ ) désignent les coefficients du polynôme de degré  $u^+$  (resp.  $u^-$ ) constituant le numérateur de  $F^+(z)$  (resp.  $F^-(z)$ ), alors que les paramètres  $b_k^+$  (resp.  $b_k^-$ ) sont les coefficients de son dénominateur de degré  $v^+$  (resp.  $v^-$ ).

Le problème de la conception d'un filtre récursif revient alors à déterminer les coefficients des deux fonctions de transfert  $F^+(z)$  et  $F^-(z)$ . Nous présentons, par la suite, les équations récurrentes permettant d'implanter les deux filtres  $l(x)$  et  $d(x)$  que nous avons retenus pour mettre en évidence les points de contour dans les images linéaires.

### IV-4.3.2 Lissage récursif

L'opérateur de lissage que nous avons retenu dans le cadre de notre application s'exprime sous la forme :

$$l(x) = C(\alpha|x|+1)e^{-\alpha|x|} ,$$

où la constante réelle  $C$  désigne le coefficient de normalisation. Le paramètre  $\alpha$  permet de régler l'étendue efficace du filtre.

En suivant la démarche présentée au § IV-4.3.1, Deriche montre que le lissage d'un signal discret  $x(n)$ , avec  $n = 1, \dots, M$ , s'obtient à l'aide des équations récurrentes suivantes :

$$y^+(n) = K \left[ x(n) + e^{-\alpha} (\alpha - 1)x(n-1) \right] + 2e^{-\alpha} y^+(n-1) - e^{-2\alpha} y^+(n-2)$$

pour  $n = 1, \dots, M$

avec les conditions initiales :

$$x(0) = 0, y^+(0) = y^+(-1) = 0 ,$$

et :

$$y^-(n) = K \left[ e^{-\alpha} (\alpha + 1)x(n+1) - e^{-2\alpha} x(n+2) \right] + 2e^{-\alpha} y^-(n+1) - e^{-2\alpha} y^-(n+2)$$

pour  $n = M, \dots, 1$

avec les conditions initiales :

$$x(M+1) = x(M+2) = 0, y^-(M+1) = y^-(M+2) = 0.$$

Pour chaque équation, le coefficient  $K$  de normalisation est donné par :

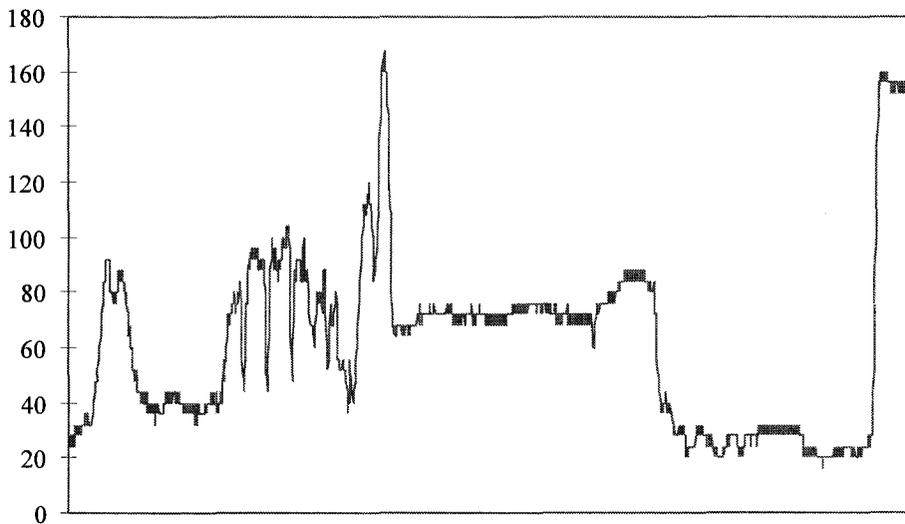
$$K = \frac{(1 - e^{-\alpha})^2}{1 + 2\alpha e^{-\alpha} - e^{-2\alpha}} .$$

On calcule d'abord  $y^+(n)$  pour  $n$  variant de 1 à  $M$ , puis  $y^-(n)$  pour  $n$  variant de  $M$  à 1. Le signal lissé  $y(n)$  s'obtient alors en prenant en compte ces deux résultats de la manière suivante :

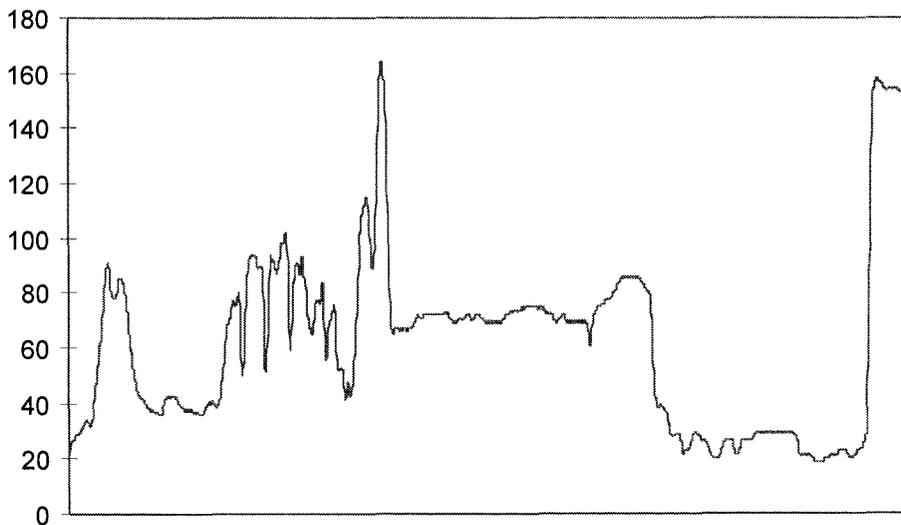
$$y(n) = y^+(n) + y^-(n) \quad \text{pour } n = 1, \dots, M .$$

On remarque, d'après les équations ci-dessus, que la valeur du paramètre  $\alpha$ , qui permet d'ajuster la taille de l'opérateur de lissage, peut être modifiée sans que cela augmente la quantité de calculs. Ceci montre l'avantage considérable qu'offre une implantation récursive du lissage.

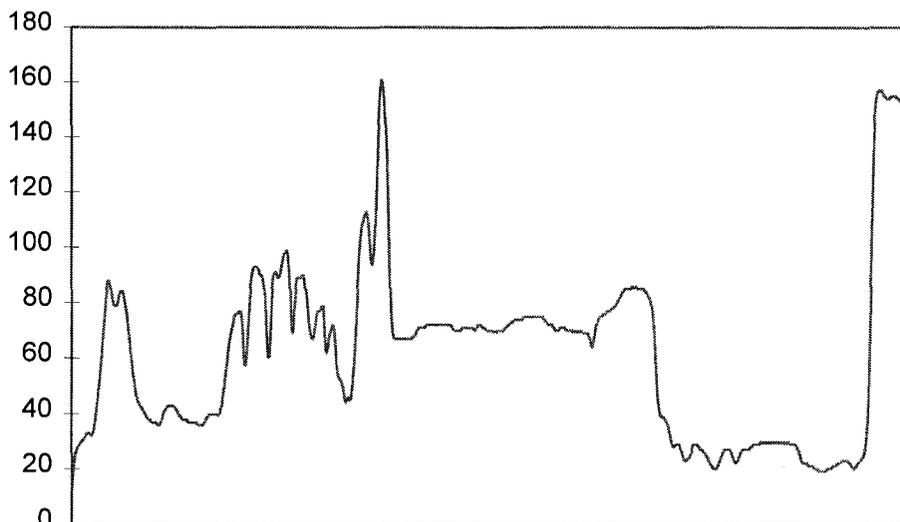
La figure IV-8 montre les résultats du lissage d'un signal monodimensionnel représentant une image linéaire et l'influence du paramètre  $\alpha$  sur ce lissage. On constate que plus  $\alpha$  est petit, c'est à dire que plus l'étendue du filtre est importante, mieux les petits détails non significatifs dus au bruit sont éliminés.



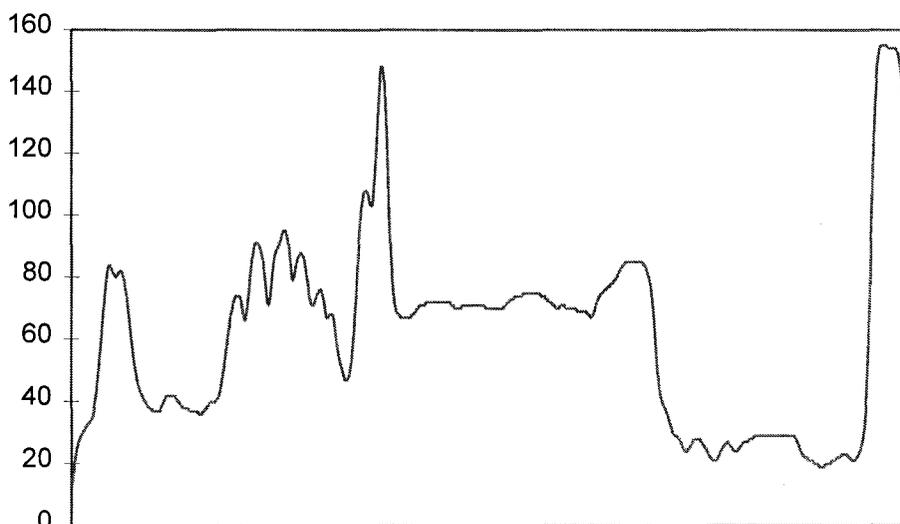
a) Image originale.



b) Lissage pour  $\alpha = 1,0$ .



c) Lissage pour  $\alpha = 0,5$ .



d) Lissage pour  $\alpha = 0,25$ .

**Figure IV-8.** Résultats du lissage d'une image linéaire.

#### IV-4.3.3 Calcul récursif du gradient

Le détecteur de contour que nous avons retenu s'exprime sous la forme :

$$d(x) = Bxe^{-\alpha|x|},$$

où  $B$  est le coefficient de normalisation et  $\alpha$  est un paramètre qui permet de régler l'étendue efficace du filtre.

De la même manière que pour le lissage, le calcul du gradient d'un signal discret  $x(n)$ , avec  $n = 1, \dots, M$ , s'obtient à l'aide des équations récurrentes suivantes :

$$y^+(n) = x(n-1) + 2e^{-\alpha} y^+(n-1) - e^{-2\alpha} y^+(n-2)$$

pour  $n = 1, \dots, M$

avec les conditions initiales :

$$x(0) = 0, y^+(0) = y^+(-1) = 0,$$

et :

$$y^-(n) = x(n+1) + 2e^{-\alpha} y^-(n+1) - e^{-2\alpha} y^-(n+2)$$

pour  $n = M, \dots, 1$

avec les conditions initiales :

$$x(M+1) = 0, y^-(M+1) = y^-(M+2) = 0.$$

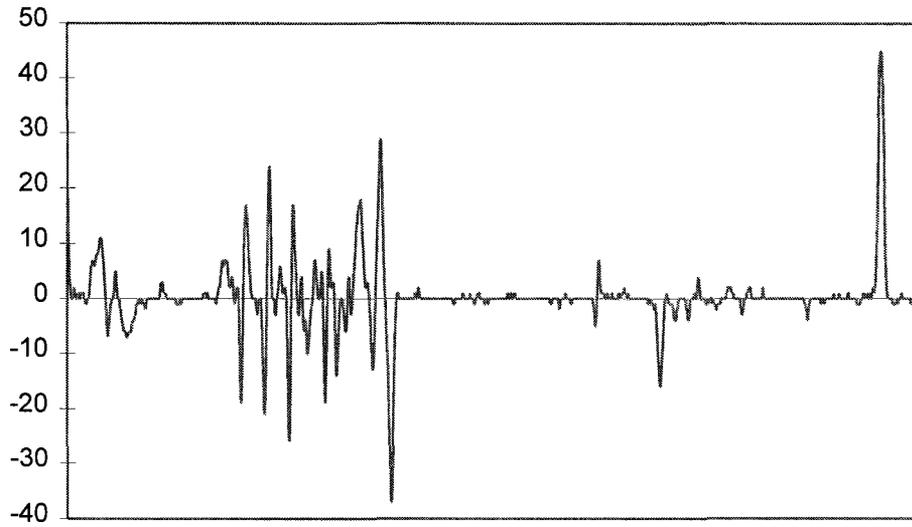
Comme pour le lissage, on calcule d'abord  $y^+(n)$  pour  $n$  variant de 1 à  $M$ , puis  $y^-(n)$  pour  $n$  variant de  $M$  à 1. Le gradient du signal s'obtient sous la forme :

$$y(n) = Ke^{-\alpha} \left[ y^+(n) - y^-(n) \right] \quad \text{pour } n = 1, \dots, M$$

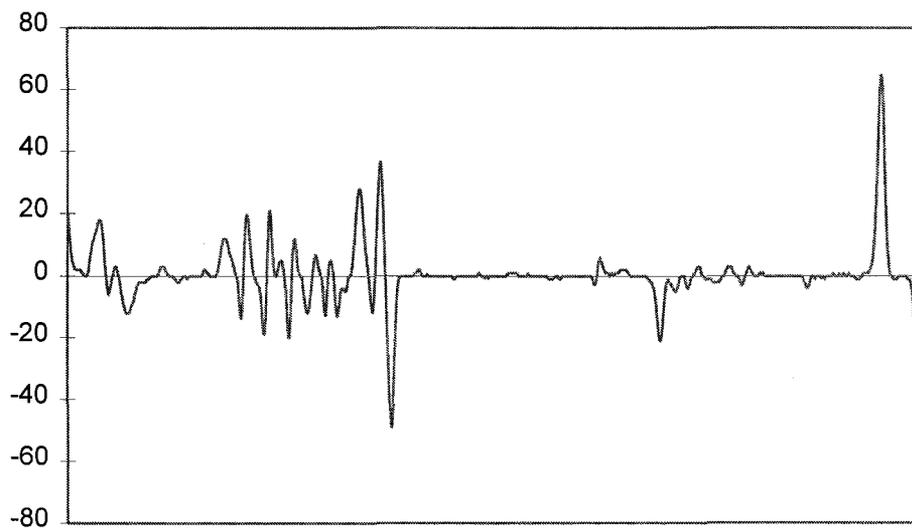
avec :  $K = -\frac{(1 - e^{-\alpha})^2}{e^{-\alpha}}$

Nous pouvons faire la même remarque que pour le lissage. Le paramètre  $\alpha$ , qui permet d'ajuster la largeur du filtre, n'influe pas sur le temps de calcul.

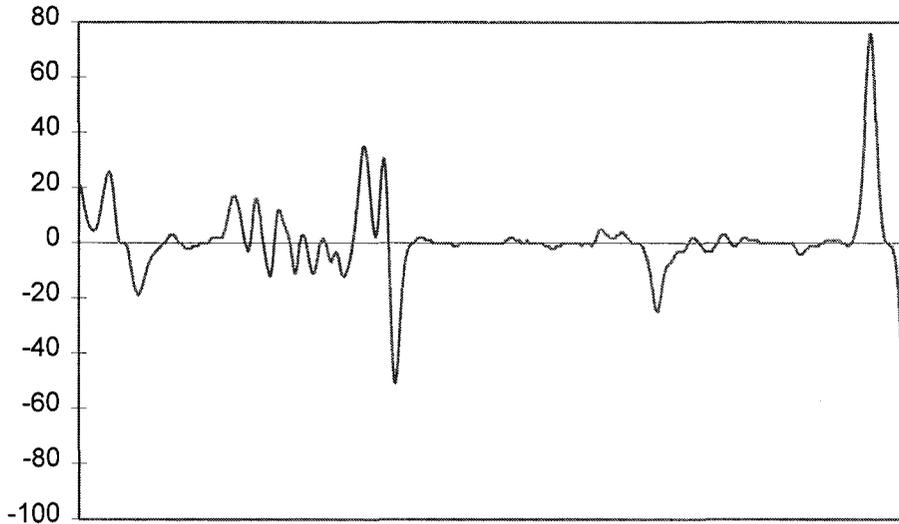
La figure IV-9 montre les résultats de la dérivation du signal de la figure IV-8-a et l'influence du paramètre  $\alpha$  sur le calcul de la dérivée. Plus  $\alpha$  est grand, plus les contours fins sont mis en évidence.



a) Dérivation pour  $\alpha = 1,0$ .



b) Dérivation pour  $\alpha = 0,5$ .



c) Dérivation pour  $\alpha = 0,25$ .

Figure IV-9. Résultats de la dérivation d'une image linéaire.

#### IV-4.4 Choix du paramètre $\alpha$

Nous avons vu que pour mettre en évidence les points de contour dans une image, il faut effectuer une opération de lissage de celle-ci afin de réduire l'influence du bruit. On doit néanmoins prendre certaines précautions car l'ajustement de  $\alpha$  a des effets contradictoires sur l'élimination des petites variations dues au bruit et sur la sensibilité de l'opérateur aux contours présents dans l'image. Il faudra donc établir, en jouant sur le paramètre  $\alpha$ , un compromis entre l'élimination du bruit et la préservation des points de contour.

Deriche [Deriche 90b] propose d'ajuster ce paramètre à une valeur comprise entre 0,5 et 1,5 pour des images moyennement bruitées. Dans notre application, l'expérience a montré que la valeur  $\alpha = 1$  donne des résultats satisfaisants.

#### IV-4.5 Conclusion

Le détecteur que nous avons utilisé pour mettre en évidence les contours dans une image linéaire est celui proposé par Deriche. La procédure de détection se fait en deux étapes :

- L'image linéaire brute est d'abord lissée afin de réduire l'influence du bruit sur la détection des points de contour. L'image lissée est obtenue en appliquant les équations récursives du filtre  $l(x)$  aux pixels de l'image originale (Cf. § IV-4.3.2).

- L'image gradient est ensuite obtenue en appliquant les équations récursives du filtre  $d(x)$  aux pixels de l'image lissée (Cf. § IV-4.3.3).

## IV-5 EXTRACTION DES CONTOURS

Les résultats obtenus avec un détecteur de contours sont souvent inexploitable, car on constate toujours la présence de faux points de contour et l'absence de certains points de contour due aux détections manquées, et ceci malgré l'utilisation d'un détecteur aussi éprouvé que celui de Deriche. Cependant, il est très important d'extraire de tous les extrema locaux de la fonction dérivée ceux qui correspondent effectivement à des points de contour significatifs de l'image. La recherche des extrema locaux significatifs du gradient de l'image peut se faire par un seuillage de l'image gradient.

Les techniques de seuillage les plus utilisées sont le seuillage simple et le seuillage par hystérésis. Cette dernière [Canny 83] fait appel à la notion de continuité du contour, ce qui la rend inutilisable dans le cas des images linéaires où cette notion n'est pas disponible.

Il ne reste donc plus que la méthode de seuillage la plus simple, qui consiste à conserver tous les points de l'image dont l'amplitude du gradient est supérieure à un seuil donné. Cependant, lors d'une opération de seuillage, le choix du seuil pose généralement un problème délicat à résoudre. Il est difficile, voire impossible, de choisir le seuil adéquat permettant à la fois de sélectionner les points de contour significatifs et d'éliminer tous ceux qui ne le sont pas. Nous n'avons donc pas retenu cette approche pour notre application, car les résultats obtenus sur nos images n'étaient pas satisfaisants.

Pour sélectionner les extremums significatifs, nous avons fait appel à une nouvelle technique, baptisée extraction par intervalle [Burie 95]. Celle-ci procède tout d'abord par une élimination des extremums générés par le bruit. L'étape de sélection des extrema significatifs est ensuite appliquée sur les extrema restants, supposés correspondre aux contours des objets présents dans l'image.

### IV-5.1 Elimination des extremums générés par le bruit

Dans le cas de nos images, l'expérience nous a montré que les extrema locaux dus au bruit sont toujours majoritaires. Par contre, leur amplitude est toujours très faible. La solution proposée consiste à utiliser un seuil bas  $S_b$  pour éliminer tous les extremums dont l'amplitude du gradient est comprise entre  $-S_b$  et  $S_b$ . De cette manière, seuls les extremums correspondant à des

contours significatifs sont conservés. L'expérience a montré que pour un seuil  $S_b$  ajusté à 10 % de l'amplitude maximale de la réponse de l'opérateur de différentiation, tous les extrema locaux générés par le bruit présent dans nos images saisies sur route sont éliminés. La figure IV-10 montre les résultats de cette première étape.

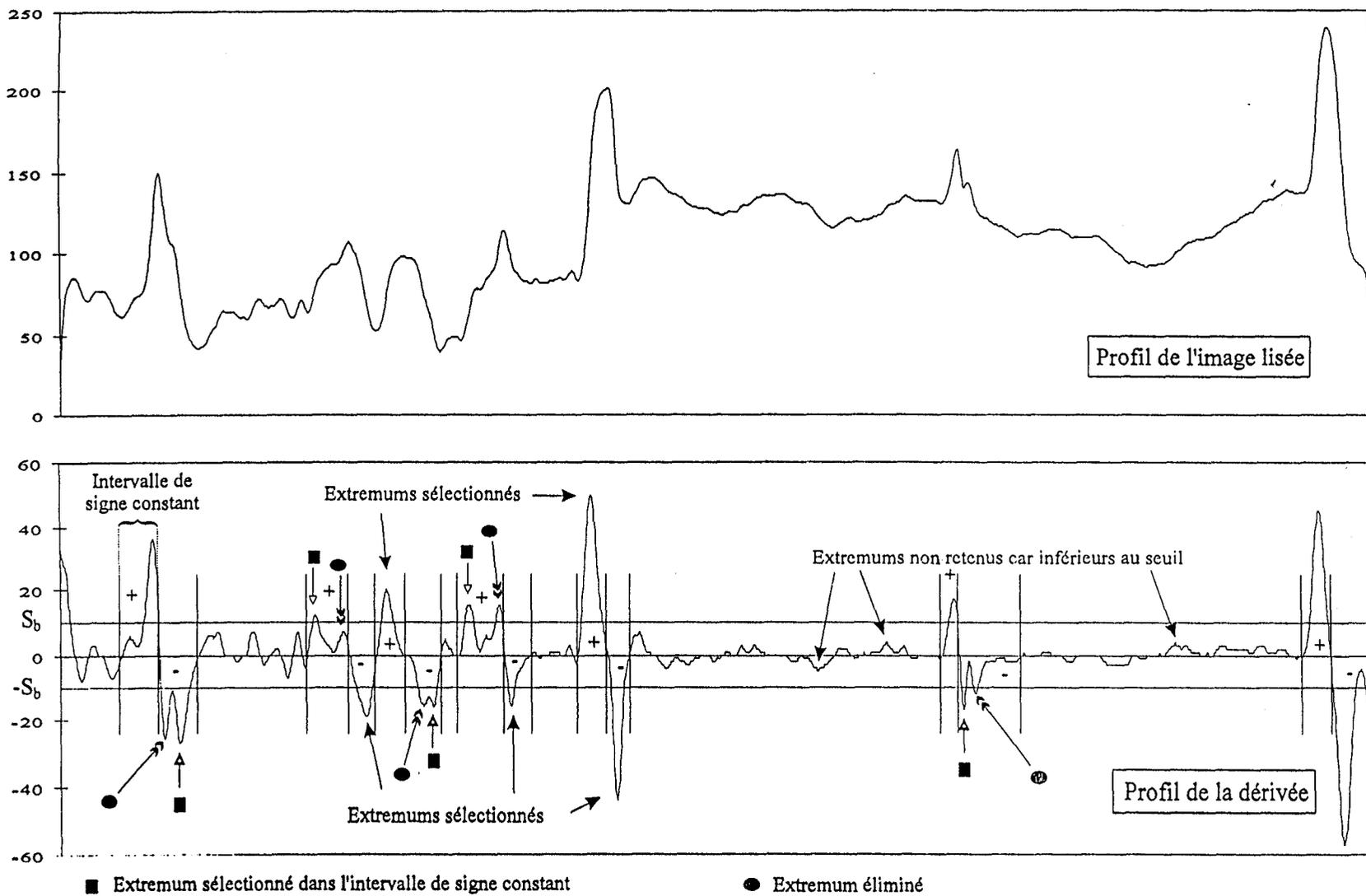
#### IV-5.2 Sélection des extremums significatifs par intervalle

Malgré la robustesse du détecteur choisi, on constate toujours, dans la pratique, qu'un contour peut donner naissance à plusieurs extrema locaux. En effet, les contours peuvent être plus ou moins larges, et la variation locale du niveau de gris peut être plus ou moins régulière de telle sorte que la dérivée peut présenter plusieurs extremums correspondant néanmoins à un seul contour.

La procédure de sélection consiste à diviser le signal dérivé en une suite d'intervalles adjacents où il conserve un signe constant. L'analyse est faite du début à la fin de l'image et un nouvel intervalle est créé chaque fois que la dérivée passe par zéro. Un seul extremum, quand il existe, est ensuite sélectionné dans chaque intervalle : il s'agit de l'extremum présentant l'amplitude du gradient la plus forte (Cf. Figure IV-10). L'image étant un signal discret, les extremums se traduisent souvent par l'apparition de plateaux. L'extremum  $g_n$  d'une image  $g$  est alors défini par :

$$\exists \text{ un entier positif } k \text{ tel que } \begin{cases} |g_n| = |g_{n+i}|, \forall i \in ]-k, k[ \\ |g_n| > |g_{n+k}| \\ |g_n| > |g_{n-k}| \end{cases} .$$

Dans ce cas, le point de contour est, par convention, localisé au centre du plateau.



### IV-5.3 Attributs des primitives

Les attributs des primitives seront utilisés, lors de la mise en correspondance, pour évaluer leur ressemblance, et pour déterminer si les contraintes d'appariement sont satisfaites ou non.

Dans notre cas, les primitives retenues sont les points de contour extraits des images linéaires. Lors de la phase d'extraction, nous conservons donc, pour chaque point de contour, les éléments suivants :

- Sa position dans l'image.
- La nature du point de contour, c'est à dire s'il s'agit d'un front montant ou d'un front descendant, ce qui revient à mémoriser le signe du gradient en ce point.
- La valeur de l'amplitude du gradient.

Les deux listes des points de contour, établies pour les images gauche et droite, seront transmises par la suite à la procédure de mise en correspondance.

### IV-5.4 Conclusion

La méthode d'extraction proposée par Burie est originale dans le sens où la sélection des points de contour significatifs ne s'effectue pas en fonction d'un seuil. Elle permet ainsi d'éviter les inconvénients des procédés de seuillage classique qui sont sensibles à la non-uniformité des objets, du fond et de l'éclairage. Le seuil utilisé, choisi délibérément petit, permet simplement d'éliminer les points de contour engendrés par le bruit et les faibles variations non significatives présentes dans l'image. L'efficacité de cette technique de sélection est nettement supérieure à celle résultant d'une technique de seuillage. En effet :

- Elle permet de supprimer les extrema non significatifs provenant des réponses multiples du détecteur.
- Elle conserve pour chaque objet présent dans l'image les points de contour les plus significatifs.

## IV-6 RESULTATS D'EXTRACTION DES CONTOURS

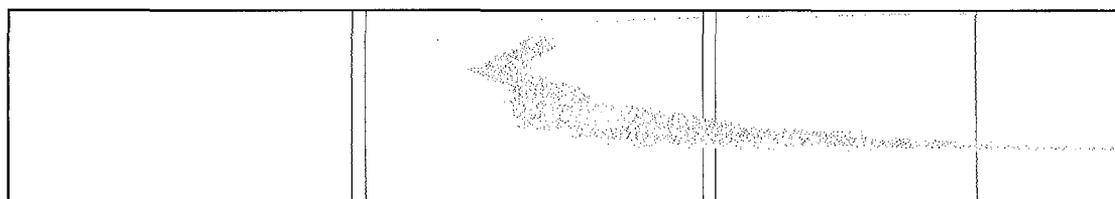
Les images linéaires sont traitées par le module d'extraction qui est organisé en trois étapes successives :

- Lissage de l'image brute (Cf. § IV-4.3.2).
- Différentiation, ou calcul du gradient, de l'image lissée (Cf. § IV-4.3.3).
- Extraction des points de contour (Cf. § IV-5).

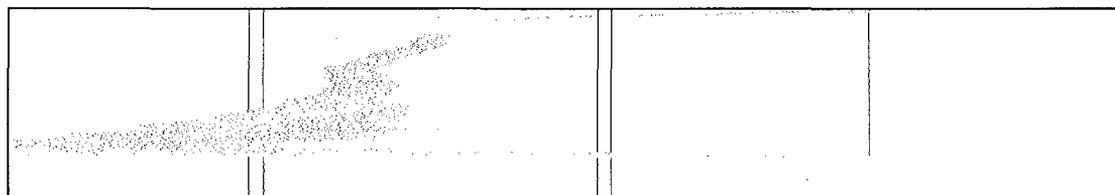
### IV-6.1 Piéton évoluant devant le stéréoscope

La figure IV-11 illustre les résultats obtenus sur la séquence stéréoscopique "piéton", dont les images brutes sont présentées au chapitre III (Cf. § III-5.2.1).

Pour obtenir une meilleure lisibilité, les points de contour extraits par la procédure décrite au § IV-5 sont représentés par des points noirs, et ceux quelque soit la valeur de leur amplitude du gradient.



*a) Extraction des contours de l'image gauche.*



*b) Extraction des contours de l'image droite.*

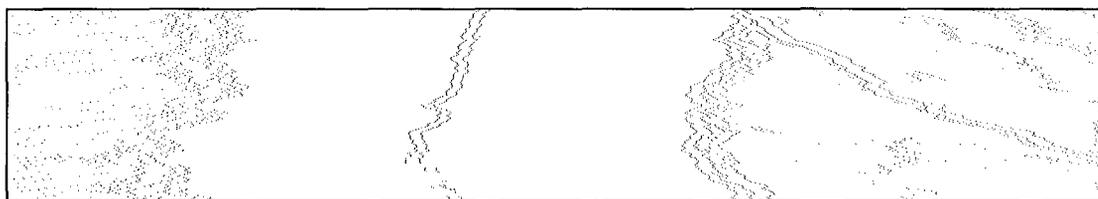
**Figure IV-11.** Résultat obtenu sur la séquence "piéton".

On constate une très bonne détection des points de contour. Les bords des lignes blanches ainsi que la frontière entre la route et la zone d'ombre sont parfaitement détectés. De plus, aucune fausse détection n'apparaît au niveau de la chaussée malgré la présence de certaines irrégularités.

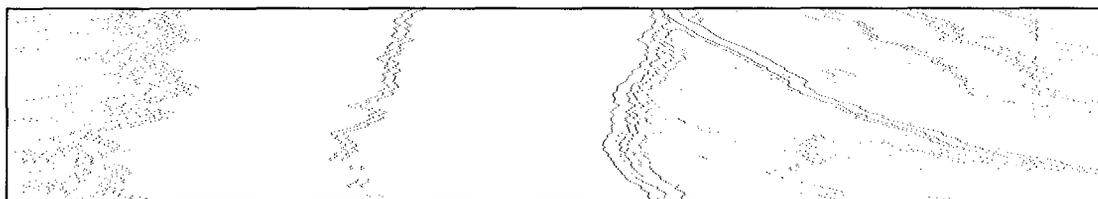
Il faut noter que le piéton engendre de nombreux points de contour. L'ensemble de ces points forme une structure cohérente au cours du temps. Mais les points de contour pris séparément ne respectent pas cette cohérence temporelle, car un point de contour d'une image de la séquence n'a pas obligatoirement de correspondant dans l'image suivante.

### IV-6.2 Route dégagée et passage d'une sortie d'autoroute

La figure IV-12 illustre les résultats obtenus sur la séquence stéréoscopique "passage d'une sortie d'autoroute", dont les images brutes sont présentées au chapitre III (Cf. § III-5.2.2).



a) Extraction des contours de l'image gauche.



a) Extraction des contours de l'image droite.

**Figure IV-12.** Résultat obtenu sur la séquence "passage d'une sortie d'autoroute".

Comme dans l'exemple précédent, les lignes blanches sont très bien détectées et aucun point de contour n'apparaît sur les deux voies principales. Quelques points de contour, dus aux taches sombres situées sur la chaussée, apparaissent néanmoins au niveau de la bretelle de sortie d'autoroute. On peut également remarquer que le bord gauche de la route engendre de nombreux points de contour qui ne correspondent pas à des obstacles. Ces points de contour sont dus aux herbes situées sur le bord de la route.

## IV-7 CONCLUSION

La première étape en stéréovision consiste à extraire, dans chaque image, les primitives à apparier. Le choix de ces dernières joue un grand rôle dans la phase de mise en correspondance, mais aussi dans la reconstruction de la scène. Dans le cas des images linéaires, le faible contenu informationnel réduit considérablement les possibilités de ce choix. Seulement deux types de primitive sont envisageables : les points de contour et les segments. Nous avons montré que ces deux types de primitive présentaient une dualité, c'est à dire qu'ils étaient dépendants l'un de l'autre. De plus, quelque soit le type de primitive choisi, il est nécessaire d'extraire les points de contour présents dans les images linéaires. Nous avons confié cette tâche à un module de

traitement utilisant le détecteur de Deriche et une méthode d'extraction développée par Burie [Burie 95]. Ce module est constitué de trois étapes distinctes et successives :

- Une étape de lissage destinée à réduire l'effet du bruit présent dans l'image.
- Une étape de différentiation, appliquée à l'image lissée, destinée à mettre en évidence les points de contour, c'est à dire les points correspondant aux extrema locaux de l'image gradient.
- Une étape d'extraction destinée à ne conserver que les extrema locaux significatifs correspondants aux points de contour pertinents de l'image.

A ce stade du traitement, nous disposons d'une liste de points de contour auxquels sont associés les éléments suivants :

- L'abscisse du point de contour dans l'image.
- Le signe du gradient en ce point.
- La valeur de l'amplitude du gradient.

Cette liste de points, établie pour chacune des images stéréoscopiques gauche et droite, est transmise au module de mise en correspondance. Avant de nous lancer dans la description de l'approche que nous avons développée pour apparier les images linéaires, nous présentons, dans le chapitre suivant, celle proposée par Burie [Burie 95].

## **Chapitre V**

# **Appariement stéréoscopique Séquentiel des images linéaires**

## V-1 INTRODUCTION

Nous exposons, dans ce chapitre, l'approche proposée par Burie [Burie 95] pour appairier les images linéaires en nous limitant aux parties essentielles. Les images sont issues du même stéréoscope que celui avec lequel nous travaillons, et les primitives d'appariement sont les mêmes que celles que nous utilisons. Ceci nous permettra, par la suite, de comparer ses résultats avec les nôtres.

## V-2 DESCRIPTION DE L'APPROCHE

### V-2.1 Principes de base

Il s'agit d'une méthode séquentielle de mise en correspondance. Elle est basée sur les deux principes suivants :

- L'appariement des primitives s'effectue *séquentiellement* en explorant les images linéaires du début à la fin.
- Lorsqu'un couple de primitives est apparié, il sert de base pour trouver l'appariement suivant.

### V-2.2 Définitions

Nous décrivons ici les différentes notations et termes que nous utiliserons au cours de l'exposé, ainsi que leurs significations.

L'extraction des primitives, détaillée dans le chapitre précédent, fournit deux listes de points de contour ordonnés, de la gauche vers la droite, selon leurs abscisses croissantes :

- Soit  $G(n)$  la liste des points de contour issus de l'image gauche, avec  $n = 0, 1, 2, \dots, N$ .
- Soit  $D(m)$  la liste des points de contour issus de l'image droite, avec  $m = 0, 1, 2, \dots, M$ .

Pour des raisons qui seront exposées ultérieurement, le premier et le dernier pixel de chaque image sont considérés comme des points de contour. Ceux-ci portent respectivement les indices  $n = 0$  et  $n = N$  dans l'image gauche, et  $m = 0$  et  $m = M$  dans l'image droite. De ce fait, la liste  $G(n)$  contient donc réellement  $N_g$  points de contour, avec  $N_g = N - 1$ . De même la liste  $D(n)$  en contient  $M_d$ , avec  $M_d = M - 1$ .

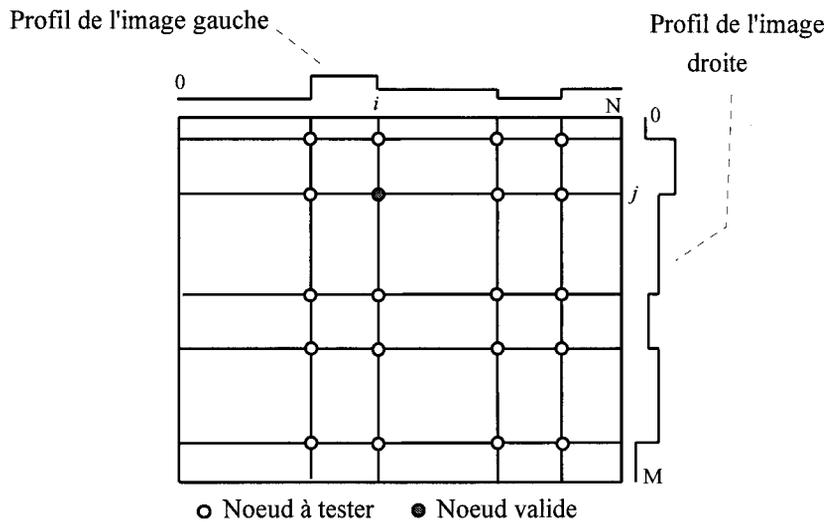
Lorsqu'on tente d'appairier le point de contour  $G(i)$  avec le point de contour  $D(j)$ , ils constituent une paire notée  $\{G(i), D(j)\}$ . Lorsqu'une paire  $\{G(i), D(j)\}$  représente un couple de points de contour appariés, elle est appelée "paire valide".

La dernière définition concerne la notion d'intervalle, défini comme l'ensemble des pixels d'une zone de l'image délimitée par deux points de contour consécutifs ou non. Un intervalle est noté  $[G(i);G(i+k)]$ , avec  $1 \leq k \leq N-i$  pour l'image gauche et  $[D(j);D(j+k')]$ , avec  $1 \leq k' \leq M-j$  pour l'image droite. Quand  $k = 1$  et  $k' = 1$ , les intervalles  $[G(i);G(i+1)]$  et  $[D(j);D(j+1)]$  représentent des zones délimitées par deux points de contour consécutifs. Nous retrouvons ici la dualité contours-régions que nous avons évoquée au chapitre IV (Cf. § IV-2).

Pour appairer les points de contour d'un couple d'images stéréoscopique, Burie était amené à comparer les niveaux de gris des pixels contenus dans les intervalles de l'image gauche et de l'image droite. Dans tous ses calculs, il s'est servi des images lissées par le filtre de Deriche, afin de limiter l'influence du bruit (Cf. Chapitre IV, § IV-4.3.2). Ainsi, les images brutes ne sont pas utilisées lors de la mise en correspondance.

### V-2.3 Représentation graphique

Pour permettre une meilleure compréhension, la méthode est représentée avec l'aide d'un graphique appelé "grille de mise en correspondance" (Cf. Figure V-1).



**Figure V-1.** Grille de mise en correspondance.

L'image gauche (resp. droite) est représentée par son profil reproduit horizontalement (resp. verticalement) en haut (resp. à droite) de la grille. Les lignes verticales (resp. horizontales) indiquent les positions des points de contour sélectionnés dans l'image gauche (resp. droite). Chaque intersection, appelée noeud, est définie par une paire  $\{G(i), D(j)\}$  et symbolise un emplacement où un test d'appariement doit être effectué. Un noeud devant être testé est représenté graphiquement par un disque blanc. Une paire valide  $\{G(i), D(j)\}$ , c'est à dire une paire

de points de contour appariés, est représentée graphiquement par un disque gris, appelé "noeud valide".

Si l'on dispose de  $N_g$  points de contour dans l'image gauche et de  $M_d$  points de contour dans l'image droite,  $N_g \times M_d$  noeuds doivent être testés par la procédure de mise en correspondance. Dans le cas de la figure V-1, où  $N_g = 4$  et  $M_d = 5$ , 20 noeuds sont donc examinés.

#### **V-2.4 Contraintes de mise en correspondance**

Dans sa thèse, Burie [Burie 95] a proposé d'utiliser un certain nombre de contraintes afin de faciliter le processus de mise en correspondance. Ces contraintes seront détaillées plus loin (Cf. Chapitre VII).

##### ***V-2.4.1 Contrainte d'unicité***

Rappelons que la contrainte d'unicité impose que tout point de contour de l'image gauche a au plus un correspondant dans l'image droite et réciproquement (Cf. Chapitre II, § II-7.2.5). Grâce à l'analyse séquentielle des images, l'utilisation de cette contrainte lors de la phase de mise en correspondance est implicite. En effet, aucune primitive ne peut être appariée deux fois puisqu'on se déplace toujours simultanément et dans le même sens dans les deux images.

##### ***V-2.4.2 Contrainte d'ordre***

Nous avons également vu que la contrainte d'ordre suppose que l'ordre des points de contour est le même dans chaque image linéaire (Cf. Chapitre II, § II-7.2.3). Il faut préciser que cette contrainte peut être mise en défaut dans certaines situations que nous avons évoquées au chapitre II. Celles-ci sont rares dans un contexte routier. Au cours du processus de mise en correspondance, cette contrainte est prise en compte implicitement du fait que les appariements sont réalisés séquentiellement.

##### ***V-2.4.3 Contrainte de position***

La contrainte de position est appliquée aux abscisses  $x_g$  et  $x_d$  des points de contour sur les capteurs gauche et droit. Lorsqu'on tente d'apparier un couple de points de contour, on impose que leurs abscisses vérifient la relation géométrique  $x_d < x_g$ . Celle-ci est déterminée à partir des relations liant la position d'un point objet dans la scène réelle et celles de ses images sur les capteurs du stéréoscope (Cf. Chapitre III, § III-4.3). Graphiquement, cette contrainte peut être représentée sous la forme d'une diagonale coupant la grille de mise en correspondance en deux (Cf. Figure V-2). Ainsi, l'ensemble des appariements possibles est représenté par les noeuds se trouvant dans la partie supérieure de la grille de mise en correspondance. Dans le cas de

l'exemple de la figure V-1, la contrainte de position permet de réduire le nombre de tests d'appariement de 20 à 12.

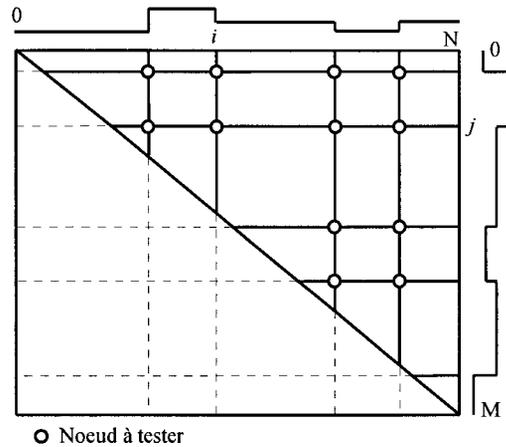


Figure V-2. Application de la contrainte de position.

#### V-2.4.4 Contrainte de signe du gradient

La contrainte de signe du gradient suppose que deux points de contour gauche et droit engendrés par le même point objet présentent localement des dérivées de même signe, c'est à dire des transitions de niveau de gris de même signe. Elle utilise pour cela l'attribut de signe du gradient associé à chaque point de contour lors de sa détection (Cf. Chapitre IV, § IV-5.3). Pour représenter cette contrainte graphiquement, il suffit de conserver les noeuds correspondant à des couples de points de contour associés à des fronts compatibles. La figure V-3 montre l'ensemble des appariements possibles après application de la contrainte de position et de la contrainte de signe du gradient. Ainsi, le nombre d'appariements possibles de l'exemple de la figure V-1 est réduit à 6.

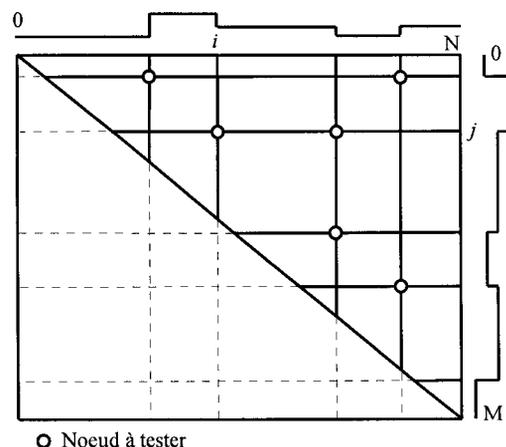


Figure V-3. Application des contraintes de position et de signe du gradient.

### V-3 ALGORITHME DE MISE EN CORRESPONDANCE

L'algorithme de mise en correspondance consiste à explorer les images séquentiellement du début à la fin. Il s'agit d'analyser les listes des points de contour  $G(i)$  et  $D(j)$  en faisant croître les indices  $i$  et  $j$ . Graphiquement, cela revient à aller d'une "extrémité" à l'autre de la grille (Cf. Figure V-4). La mise en correspondance peut donc être vue comme la recherche d'un chemin dont chaque étape aboutit à un noeud. Initialement, la procédure débute dans le "coin supérieur gauche" de la grille. L'objectif est d'atteindre le "coin inférieur droit".

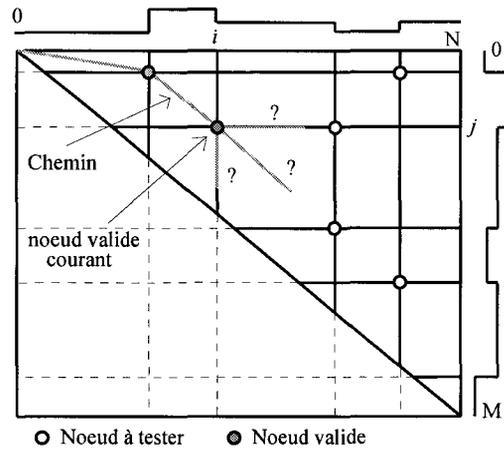


Figure V-4. Algorithme de mise en correspondance.

La recherche des appariements est basée sur le principe suivant : utiliser le dernier couple apparié qui forme la paire valide courante  $\{G(i), D(j)\}$  pour trouver la paire suivante. Graphiquement, il s'agit, en partant du noeud valide courant, de déterminer le chemin qui aboutit au prochain noeud valide.

#### V-3.1 Critère de ressemblance

Pour juger de la qualité des différents appariements possibles, la procédure de mise en correspondance utilise un critère de ressemblance photométrique. Supposons que  $\{G(i), D(j)\}$  soit la paire valide courante. Soit, par exemple, à évaluer la qualité de l'appariement du point de contour  $G(i+k)$  avec le point de contour  $D(j+k')$ , avec  $1 \leq k \leq N-i$  et  $1 \leq k' \leq M-j$ . On considère alors les intervalles définis par  $[G(i); G(i+k+\varepsilon)]$  et  $[D(j); D(j+k'+\varepsilon')]$ , avec  $(\varepsilon, \varepsilon') \in \{0, 1\}^2$  et  $\varepsilon + \varepsilon' \neq 0$ . Ces deux intervalles sont ensuite comparés en utilisant une mesure de similarité.

### V-3.1.1 Normalisation des intervalles

Avant de comparer les deux intervalles ainsi définis, deux opérations doivent être réalisées.

#### V-3.1.1.1 Recentrage des intervalles

Généralement, les deux caméras dont sont issues les images ne présentent pas exactement les mêmes gains de sortie. Cela se traduit par deux images dont les niveaux de gris moyens sont différents. Leur comparaison est par conséquent délicate à effectuer. Pour remédier à ce problème, les pixels de chaque intervalle sont ramenés à une valeur moyenne nulle de la façon suivante.

Soit un intervalle de longueur  $l$ . On note  $h(w)$  les niveaux de gris de l'image correspondant à cet intervalle, avec  $w = 0, 1, \dots, l-1$ . On calcule alors la moyenne  $m$  définie par :

$$m = \frac{1}{l} \sum_{z=0}^{z=l-1} h(z) .$$

Les niveaux de gris des pixels de l'intervalle sont alors modifiés pour avoir une valeur moyenne nulle. Les nouvelles valeurs,  $h_m(w)$ , des niveaux de gris des pixels de l'intervalle sont obtenues par la relation suivante :

$$h_m(w) = h(w) - m \quad \text{avec } w = 0, 1, \dots, l-1.$$

#### V-3.1.1.2 Rééchantillonnage spatial

Les intervalles à comparer ont peu de chance d'être exactement de la même longueur. Dans ce cas, une comparaison pixel à pixel ne peut être effectuée. Pour ramener les deux intervalles à une même longueur, on procède alors de la façon suivante.

Soit  $l$  la longueur de l'intervalle le plus petit et  $L$  celle du plus grand. On note  $h(w)$  les niveaux de gris de l'image correspondant à l'intervalle le plus petit, avec  $w = 0, 1, \dots, l-1$ . De la même manière on note  $H(W)$ , avec  $W = 0, 1, \dots, L-1$ , les niveaux de gris de l'autre image, correspondant à l'intervalle le plus grand.

Les pixels de l'intervalle le plus petit sont conservés dans leur totalité. L'opération consiste à prélever  $l$  échantillons, régulièrement espacés, dans l'intervalle le plus grand. On obtient alors l'intervalle  $H_e$  de longueur  $l$  défini de la façon suivante :

$$H_e(w) = H\left(\text{int}\left(\frac{w \cdot L}{l}\right)\right) \quad \text{avec } w = 0, 1, \dots, l-1.$$

L'opérateur  $\text{int}(x)$  fournit la partie entière de  $x$ .

### V-3.1.2 Mesure de similarité

A la suite de ces deux opérations de normalisation, on dispose de deux ensembles de pixels, correspondant aux intervalles  $[G(i); G(i+k+\varepsilon)]$  et  $[D(j); D(j+k'+\varepsilon')]$ , composés du même nombre de pixels  $Z$ , et avec une valeur moyenne du niveau de gris nulle. On note  $H_g(z)$  (resp.  $H_d(z)$ ) les niveaux de gris de l'image correspondant à l'intervalle gauche (resp. droit), avec  $z = 1, 2, \dots, Z$ .

Pour comparer les deux intervalles  $[G(i); G(i+k+\varepsilon)]$  et  $[D(j); D(j+k'+\varepsilon')]$ , qui correspondent au test de l'appariement des deux points de contour candidats  $G(i+k)$  et  $D(j+k')$ , Burie a défini une mesure de similarité comme étant la somme des écarts de niveaux de gris des pixels de ces deux intervalles :

$$S[G(i+k+\varepsilon), D(j+k'+\varepsilon')] = \frac{1}{Z} \sum_{z=1}^{z=Z} |H_g(z) - H_d(z)|$$

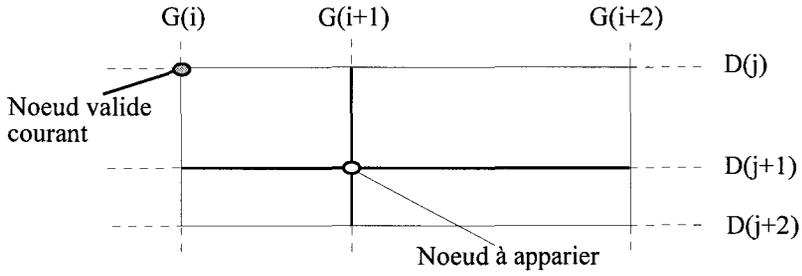
On constate que la ressemblance entre les deux intervalles est d'autant plus grande que la mesure  $S$  est proche de zéro. Comme il s'agit de comparer des mesures faites sur des groupes d'intervalles différents, cette mesure est normalisée par le coefficient  $Z$  qui correspond aux nombres de pixels des intervalles normalisés.

## V-3.2 Module de mise en correspondance élémentaire

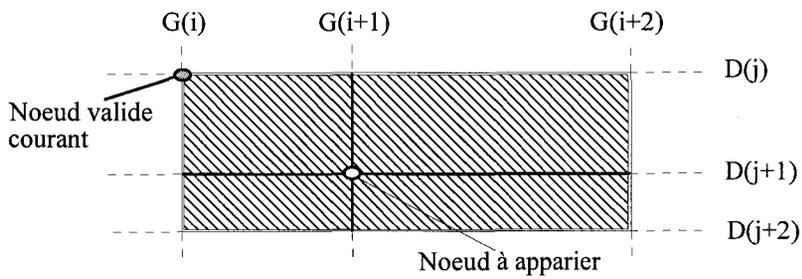
### V-3.2.1 Tests d'appariement élémentaire

En partant de la paire valide courante  $\{G(i), D(j)\}$ , on cherche à appairer le point de contour  $G(i+1)$  avec le point de contour  $D(j+1)$ . On suppose, dans un premier temps, que ces deux points de contour sont compatibles, c'est à dire qu'ils vérifient la contrainte de position et la contrainte de signe du gradient. On propose alors de considérer les intervalles définis par  $[G(i); G(i+1+\varepsilon)]$  et  $[D(j); D(j+1+\varepsilon')]$ , avec  $(\varepsilon, \varepsilon') \in \{0, 1\}^2$  et  $\varepsilon + \varepsilon' \neq 0$  et d'effectuer une mesure de similarité telle qu'elle est définie au § V-3.1.2. On est donc amené à calculer les trois mesures de similarité suivantes (Cf. Figure V-5) :

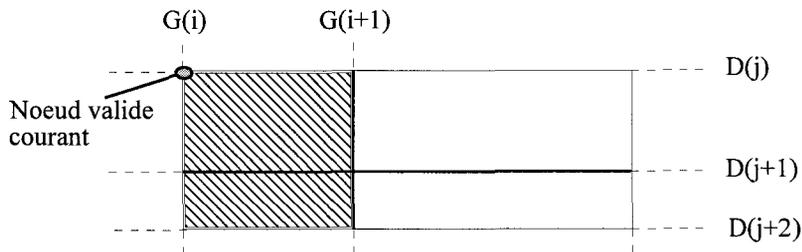
- $S_{22} = S[G(i + 2), D(j + 2)]$
- $S_{12} = S[G(i + 1), D(j + 2)]$
- $S_{21} = S[G(i + 2), D(j + 1)]$



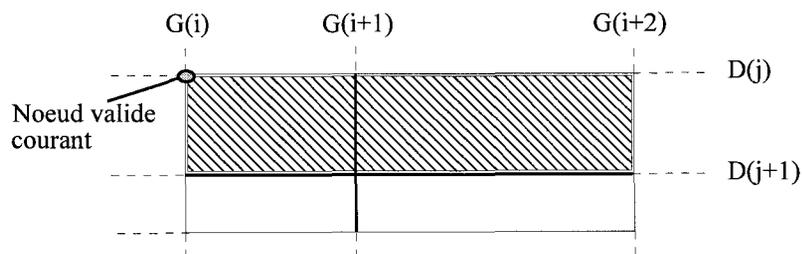
a) Partie de la grille de mise en correspondance.



b) Calcul de  $S_{22}$ .



c) Calcul de  $S_{12}$ .



d) Calcul de  $S_{21}$ .

Figure V-5. Tests de mise en correspondance.

### **Interprétation de la mesure $S_{22}$**

La mesure  $S_{22}$  sert à déterminer si l'appariement entre les points de contour  $G(i+1)$  et  $D(j+1)$  est correct. On évalue pour cela la mesure de similarité entre les intervalles normalisés définis par  $[G(i);G(i+2)]$  et  $[D(j);D(j+2)]$ . Ce choix est basé sur l'hypothèse suivante : *si, dans une image, un point de contour  $C$  délimite les régions  $R1$  et  $R2$ , alors, dans l'autre image, le point de contour  $C'$  correspondant délimite les deux régions  $R1'$  et  $R2'$  telles que la région  $R1'$  correspond à  $R1$  et  $R2'$  correspond à  $R2$ .*

Lors de la mise en correspondance des points de contour  $G(i+1)$  et  $D(j+1)$ , les points de contour  $G(i+2)$  et  $D(j+2)$  servent uniquement à définir la limite droite des intervalles. Leur compatibilité au sens des contraintes de position et de signe du gradient n'est pas prise en compte. Ceci pose un problème. En effet, dans le cas où les deux points de contour  $G(i+2)$  et  $D(j+2)$  sont incompatibles, ils ne seront pas appariés, et par suite les points de contour  $G(i+1)$  et  $D(j+1)$  délimitent deux régions qui ne se correspondent pas. On est alors en contradiction avec l'hypothèse faite au départ.

### **Interprétation de $S_{12}$ et $S_{21}$**

Les mesures  $S_{12}$  et  $S_{21}$  ne correspondent pas à une tentative d'appariement. Elles sont destinées à prendre en compte la présence de points de contour sans correspondant.

$S_{12}$  évalue la mesure de similarité entre les intervalles normalisés  $[G(i);G(i+1)]$  et  $[D(j);D(j+2)]$  afin de savoir si le point de contour  $D(j+1)$  n'a pas de correspondant.

De même,  $S_{21}$  mesure la similarité entre les intervalles normalisés  $[G(i);G(i+2)]$  et  $[D(j);D(j+1)]$  afin de savoir si le point de contour  $G(i+1)$  n'a pas de correspondant.

#### ***V-3.2.2 Règles de décision***

Les trois mesures  $S_{22}$ ,  $S_{12}$  et  $S_{21}$  sont comparées afin de déterminer la mesure la plus faible. En fonction des résultats obtenus, l'une des règles suivantes est appliquée :

- **$S_{22}$  est la mesure la plus faible** : le point de contour  $G(i+1)$  est apparié avec le point de contour  $D(j+1)$ . Le couple  $\{G(i+1), D(j+1)\}$  devient la paire valide courante. On considère alors le couple  $\{G(i+2), D(j+2)\}$  qui devient la paire à tester.
- **$S_{12}$  est la mesure la plus faible** : cela signifie que le point de contour  $D(j+1)$  n'a pas de correspondant dans l'image gauche. Il est donc ignoré et remplacé par  $D(j+2)$

dans la procédure d'appariement élémentaire. La paire  $\{G(i), D(j)\}$  reste la paire valide courante et la paire  $\{G(i+1), D(j+2)\}$  devient la paire à tester.

- $S_{2L}$  est la mesure la plus faible : cela signifie que le point de contour  $G(i+1)$  n'a pas de correspondant dans l'image droite. Il est donc ignoré et remplacé par  $G(i+2)$  dans la procédure d'appariement élémentaire. La paire  $\{G(i), D(j)\}$  reste la paire valide courante et la paire  $\{G(i+2), D(j+1)\}$  devient la paire à tester.

### V-3.3 Filtrage des paires incompatibles

Nous avons vu au § V-2.3 que seuls les points de contour compatibles, c'est à dire les points de contour respectant les contraintes de position et de signe du gradient, pouvaient être appariés. Ainsi, avant de lancer le module de mise en correspondance élémentaire pour valider l'appariement entre deux points de contour, une procédure de filtrage est utilisée pour vérifier leur compatibilité et éviter des traitements inutiles. La question qui se pose est la suivante : dans le cas où les deux points de contour ne sont pas compatibles, c'est à dire lorsque l'une des deux contraintes n'est pas respectée, comment relancer l'algorithme de mise en correspondance ? En d'autres termes, comment trouver une autre paire compatible pour permettre à l'algorithme de continuer sa progression ?

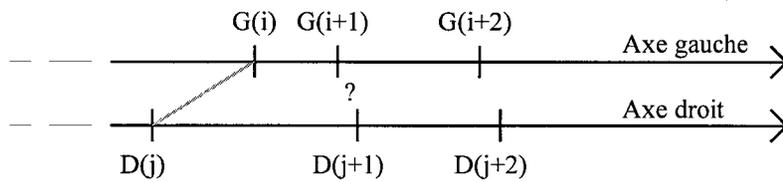
Burie considère alors, comme précédemment, la paire valide courante  $\{G(i), D(j)\}$ . Il s'agit de déterminer si les points de contour  $G(i+1)$  et  $D(j+1)$  sont compatibles. Ce test de compatibilité se décompose en deux étapes successives.

#### V-3.3.1 1<sup>ère</sup> étape : vérification de la contrainte de position

La contrainte de position (Cf. § V-2.4.3) impose que les abscisses des points de contour homologues dans leurs images respectives vérifient la relation  $x_d < x_g$ . Quand on compare les abscisses  $x_g$  de  $G(i+1)$  et  $x_d$  de  $D(j+1)$ , deux cas sont possibles :

1. La contrainte de position est respectée. Dans ce cas, on passe à la vérification de la contrainte de signe du gradient (Cf. § V-3.3.2).
2. La contrainte de position n'est pas respectée. Dans ce cas, les deux points de contour ne sont pas compatibles. Il faut donc trouver une autre paire compatible pour permettre à l'algorithme de poursuivre les appariements. En analysant la figure V-6, nous pouvons déduire la règle suivante : *si  $G(i+1)$  et  $D(j+1)$  ne vérifient pas la contrainte de position, toutes les paires  $\{G(i+1), D(j+k')\}$  avec  $1 \leq k' \leq M - j$  ne vérifient pas la contrainte de position. La*

solution consiste donc à ignorer le point de contour  $G(i+1)$  et chercher le premier point de contour  $G(i+k)$  avec  $2 \leq k \leq N - i$  vérifiant la contrainte avec le point de contour  $D(j+1)$ .



**Figure V-6.** Vérification de la contrainte de position.

### V-3.3.2 2<sup>ème</sup> étape : vérification de la contrainte de signe du gradient

L'application de la contrainte de signe du gradient (Cf. § V-2.4.4) se fait juste après celle de la contrainte de position. Il s'agit de vérifier que les points de contour  $G(i+1)$  et  $D(j+1)$  présentent localement dans chaque image des dérivées de même signe. Deux cas peuvent se présenter :

1. Les dérivées sont de même signe. Dans ce cas, la paire  $\{G(i+1), D(j+1)\}$  est compatible et donc le module de mise en correspondance élémentaire (Cf. § V-3.2) peut être lancé pour valider l'appariement de ce couple de points de contour.
2. Les dérivées sont de signes opposés. Dans ce cas, les deux points de contour ne sont pas compatibles. Il faut donc trouver une autre paire compatible pour relancer l'algorithme de mise en correspondance. Ce cas est plus complexe que pour la contrainte de position où une analyse géométrique est suffisante pour trouver la paire suivante. Pour trouver la paire suivante, la solution proposée par Burie fait appel au module de mise en correspondance élémentaire. Pour plus de détails, le lecteur pourra consulter la thèse de Burie [Burie 95].

### V-3.3.3 Conclusion sur de filtrage

Le processus de filtrage ne permet pas seulement d'éliminer les paires de points de contour incompatibles ne pouvant être appariés. Il met également en jeu des règles d'évolution qui permettent à l'algorithme de sélectionner une nouvelle paire compatible à tester, lorsque celui-ci se trouve dans une impasse. Il faut toutefois noter que le problème de la recherche d'une paire compatible, qui permet de relancer l'algorithme, est dû à la nature séquentielle de la méthode de mise en correspondance. Pour la même raison, cette recherche devient très lourde surtout dans le

cas où l'impasse est causée par la contrainte de signe du gradient, qui fait appel au module de mise en correspondance élémentaire.

### **V-3.4 Initialisation et arrêt de l'algorithme**

#### ***V-3.4.1 Condition initiale***

Nous avons vu, au § V-2.1, que Burie apparie les points de contour en fonction du dernier couple apparié. Cependant, l'algorithme a besoin d'un couple déjà apparié pour commencer l'exploration séquentielle des listes de points de contour gauche et droite. A cet effet, Burie choisit d'utiliser les premiers pixels de chaque image comme couple de base. Ces pixels, considérés comme des points de contour, sont repérés par l'indice zéro. L'algorithme utilise alors la paire valide initiale  $\{G(0), D(0)\}$  pour construire tous les appariements successifs.

Cette initialisation peut être, dans de nombreux cas, une source d'erreur. En effet, le couple initial, qui ne représente pas un appariement correct, est utilisé pour construire le couple suivant. Par conséquent, la procédure de mise en correspondance élémentaire, basée sur des mesures de similarité, peut fournir un appariement erroné. De plus, comme le dernier couple apparié sert de base pour trouver l'appariement suivant, cette erreur peut se propager sur l'ensemble des appariements.

#### ***V-3.4.2 Condition d'arrêt***

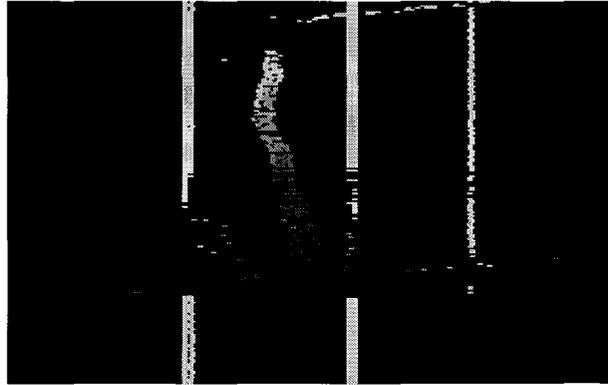
La procédure de mise en correspondance s'arrête lorsque l'un des points de contour  $G(N)$  ou  $D(M)$  fait partie de la nouvelle paire à appier. Ces points de contour correspondent en fait aux derniers pixels des images gauche et droite. Ils sont utilisés également lorsque l'algorithme tente d'apparier les derniers points de contour  $G(N-1)$  et  $D(M-1)$ . Dans ce cas, ils peuvent provoquer une erreur d'appariement car ils ne correspondent pas à des vrais points de contour.

### **V-3.5 Résultats de la mise en correspondance séquentielle**

#### ***V-3.5.1 Piéton évoluant devant le stéréoscope***

La figure V-7 montre le résultat de la mise en correspondance obtenu par Burie sur la séquence stéréoscopique "piéton", dont les images brutes sont présentées au chapitre III (Cf. § III-5.2.1). Chaque paire de points de contour appariés permet, grâce aux formules de triangulation (Cf. § V-5), de déterminer la distance et la position d'un point de la scène réelle. Les résultats représentés sur cette image s'interprètent de la manière suivante :

- L'axe horizontal représente l'axe des abscisses dans la scène. Un point se trouvant sur le bord gauche (resp. droit) de l'image est situé à 5 m à gauche (resp. droite) de l'axe optique.
- L'axe vertical représente l'axe du temps, celui-ci défilant du haut vers le bas de l'image.
- La distance est représentée par le niveau de gris. Celui-ci est d'autant plus clair que la distance par rapport au stéréoscope linéaire est grande.



*Figure V-7. Résultat obtenu sur la séquence "piéton".*

On constate sur cette image que les points de contour des bandes blanches sont correctement appariés et que leur détection est stable au cours du temps. En effet, les positions et les distances calculées sont constantes. En ce qui concerne le piéton, les niveaux de gris variant du clair au sombre signalent le rapprochement de celui-ci du stéréoscope. La transition entre la chaussée et la zone d'ombre à droite de l'image est également bien détectée.

On remarque cependant la présence de mauvais appariements lorsque des occlusions se produisent, c'est à dire lorsque le piéton cache l'une des bandes blanches à l'une des caméras. On peut constater aussi l'absence de plusieurs appariements. Sur certaines images-ligne, les points de contour droits de la bande gauche n'ont pas été appariés. La même remarque peut être faite sur les points de contour du piéton qui laissent apparaître dans la scène reconstituée des "trous" correspondant à des appariements manqués.

### ***V-3.5.2 Route dégagée et passage d'une sortie d'autoroute***

La figure V-8 montre le résultat de la mise en correspondance obtenu sur la séquence stéréoscopique "sortie d'autoroute", dont les images brutes sont présentées au chapitre III (Cf. § III-5.2.2).



Figure V-8. Résultat obtenu sur la séquence "sortie d'autoroute".

Sur cette figure, un point se trouvant sur le bord gauche de l'image est situé à 9 m à gauche de l'axe optique du stéréoscope. Un point se trouvant sur le bord droit de l'image est situé à 9 m à droite de ce même axe.

La scène reconstituée montre nettement les bandes blanches, qui sont les seuls éléments présents dans la séquence (Cf. Figure V-8). La bande centrale et celle de droite présentent une bonne stabilité dans le temps, par contre la bande de gauche n'apparaît pas nettement. Les nombreux points présents sur sa gauche sont dus aux herbes situées sur le bord de la chaussée.

Quelques faux appariements apparaissent sur la voie de gauche à la fin de la séquence ainsi que sur la voie de droite.

Comme l'exemple précédent, on peut observer de temps à autre des appariements manqués qui se manifestent sous la forme de discontinuités au niveau de la bande centrale et de la bande de droite.

## V-4 AMELIORATIONS DE L'ALGORITHME

### V-4.1 Mise en correspondance bidirectionnelle

#### V-4.1.1 Principe

Afin de limiter les risques d'erreurs, Burie a proposé d'appliquer la procédure de mise en correspondance séquentielle deux fois. La première passe se fait du début vers la fin des images, c'est ce qu'il appelle l'analyse progressive. La seconde passe est effectuée de la fin au début, il s'agit de l'analyse rétrograde. Une comparaison des résultats permet ensuite d'éliminer certaines erreurs d'appariements possibles.

Lors de l'analyse rétrograde, le principe de la procédure de mise en correspondance est identique à celui mis en oeuvre pour l'analyse progressive. Mais la différence essentielle vient du

fait que les listes des points de contour gauche et droite sont explorées en sens inverse, respectivement des indices  $N$  et  $M$  à l'indice zéro. Les derniers pixels de chaque image sont utilisés pour former la paire initiale  $\{G(N), D(M)\}$  qui constitue le point de départ pour construire les appariements successifs. L'analyse rétrograde s'arrête quand l'un des pixels  $G(0)$  ou  $D(0)$ , considérés comme des points de contour, fait partie de la nouvelle paire à appairer.

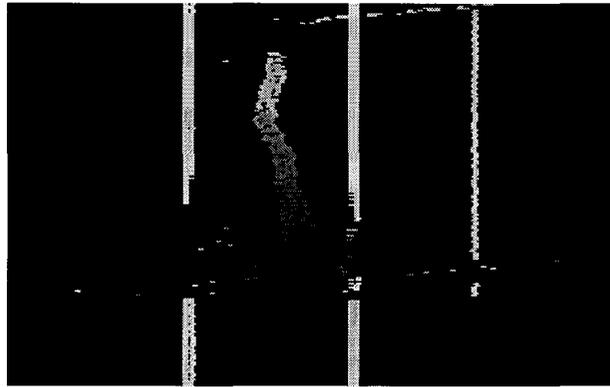
Chaque analyse fournit une liste de paires de points de contour appariés. Dans chaque liste, chaque paire est caractérisée par les abscisses des points de contour sur les deux capteurs et par la valeur de la mesure de similarité  $S_{22}$  obtenue lors de l'appariement. Pour construire la liste définitive des appariements, les deux listes sont comparées, pour ne conserver que les couples respectant les contraintes d'ordre et d'unicité. On rencontre trois types de possibilités à traiter :

1. Une paire  $\{G(i), D(j)\}$  apparaît dans les deux listes. Dans ce cas, la paire a été appariée deux fois. Elle est donc rangée dans la liste définitive.
2. Un point de contour d'une image est apparié à deux points de contour différents de l'autre image. Dans ce cas, la contrainte d'unicité impose de faire un choix parmi ces deux appariements. On conserve alors le couple qui respecte la contrainte d'ordre avec le couple précédemment stocké dans la liste définitive. Si les deux couples vérifient la contrainte d'ordre, on garde celui qui présente la mesure de similarité la plus faible.
3. Une paire  $\{G(i), D(j)\}$  n'est présente que dans une seule liste. Dans ce cas, la paire est conservée si elle vérifie la contrainte d'ordre avec le dernier couple stocké dans la liste définitive.

### **V-4.1.2 Résultats**

#### *V-4.1.2.1 Piéton évoluant devant le stéréoscope*

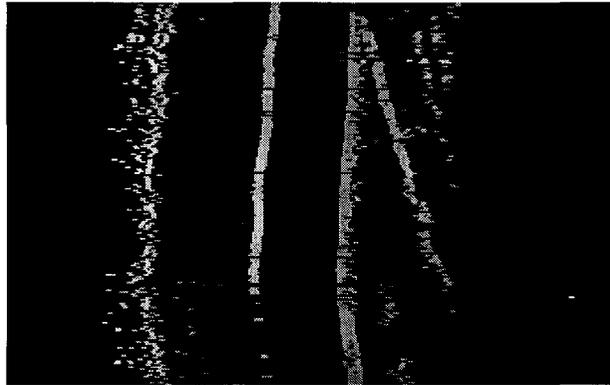
La figure V-9 montre le résultat de la mise en correspondance bidirectionnelle sur la séquence stéréoscopique "piéton". Celui-ci est meilleur que le résultat obtenu par l'algorithme séquentiel. En effet, l'analyse rétrograde a réussi là où l'analyse progressive a échoué. On constate que les faux appariements sont moins nombreux. De même, la bande droite et le piéton sont mieux détectés avec la méthode bidirectionnelle. Cependant, on constate que quelques "trous", qui correspondent à des appariements manqués, subsistent toujours dans la scène reconstruite.



*Figure V-9. Résultat obtenu sur la séquence "piéton".*

#### *V-4.1.2.2 Route dégagée et passage d'une sortie d'autoroute*

La figure V-10 montre le résultat de la mise en correspondance bidirectionnelle sur la séquence stéréoscopique "sortie d'autoroute".



*Figure V-10. Résultat obtenu sur la séquence "sortie d'autoroute".*

On ne remarque pas de véritable amélioration sur cette figure. En effet, cette séquence est très simple et l'analyse progressive suffit pour obtenir les appariements corrects. On constate cependant une diminution du nombre d'appariements manqués. Celle-ci se manifeste par la disparition de quelques "trous" dans la bande centrale et la bande de droite.

#### *V-4.1.3 Conclusion sur la méthode bidirectionnelle*

La méthode de mise en correspondance bidirectionnelle est intéressante car elle construit l'ensemble des appariements en combinant les résultats obtenus par chacune des deux analyses progressive et rétrograde. Elle offre ainsi la possibilité de confirmer certains appariements, mais également celle d'en éliminer d'autres plus ambigus. Cependant, cette méthode est lourde à mettre en oeuvre car elle nécessite d'effectuer deux passes ainsi qu'une comparaison des résultats fournis par les deux analyses. Il en résulte alors une augmentation du temps de traitement.

## V-4.2 Mise en correspondance hiérarchique

### V-4.2.1 Principe

Avec la recherche bidirectionnelle, Burie a jugé que cette méthode n'est pas assez robuste. Il a alors proposé une analyse hiérarchique du problème. L'idée principale est d'apparier hiérarchiquement les points de contour en utilisant la notion de niveau de recherche. Il s'agit de déterminer des couples de points de contour de référence à partir desquels sera effectuée la mise en correspondance des autres points de contour. Pour un niveau de recherche donné, il commence par sélectionner dans chaque image un certain nombre de points de contour très significatifs, en fonction de l'amplitude de leur dérivée (Cf. § V-4.2.2). Il applique ensuite la procédure de mise en correspondance bidirectionnelle pour apparier ces points de contour significatifs. Les couples de points de contour appariés servent alors de base pour rechercher les appariements des points de contour un peu moins significatifs.

La figure V-11 montre le niveau 1 de la recherche où les 4 points de contour les plus significatifs ont été sélectionnés dans chaque image. Ainsi, la grille à analyser devient plus simple. L'algorithme de mise en correspondance bidirectionnelle est ensuite appliqué dans la zone de recherche représentée graphiquement par un rectangle dont les limites sont dessinées avec un trait épais. On constate qu'au niveau 1, la zone de recherche est confondue avec la grille de mise en correspondance.

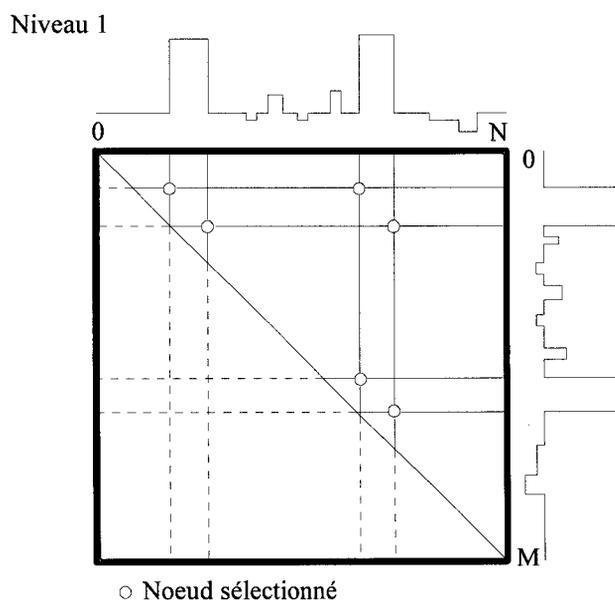
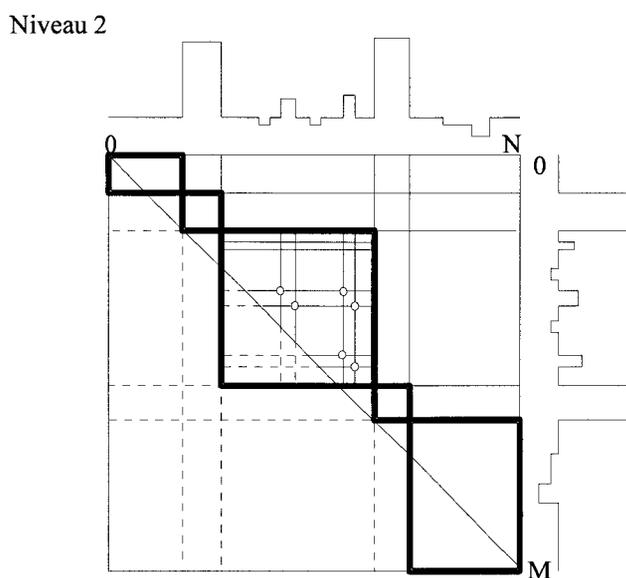


Figure V-11. Niveau 1 de la mise en correspondance hiérarchique.

Lorsque la mise en correspondance bidirectionnelle est terminée, elle fournit une liste de paires de points de contour appariés. Cette liste sert de base pour mettre en correspondance les points de contour sélectionnés au niveau suivant. La figure V-12 montre le niveau 2 de la recherche. Deux couples successifs appariés au niveau 1 servent à définir une nouvelle zone de recherche. Dans l'exemple de la figure V-12, 5 zones de recherche sont définies à partir des 4 noeuds valides.



**Figure V-12.** Niveau 2 de la mise en correspondance hiérarchique.

Au niveau 2, la technique consiste à sélectionner dans chaque zone de recherche des points de contour restants les plus significatifs pour les appairer. Dans le cas de la figure V-12, une seule zone de recherche contient des noeuds à traiter. Dans le cas général, la mise en correspondance bidirectionnelle est appliquée dans chaque zone de recherche. Les différents résultats obtenus permettent ensuite de définir de nouvelles zones de recherche plus petites, dans lesquelles la mise en correspondance bidirectionnelle est également appliquée. Afin de l'optimiser, cette procédure hiérarchique est implantée de façon récursive.

#### ***V-4.2.2 Sélection des points de contour de référence***

Soit le niveau  $n$  de recherche. Avant de lancer la procédure de mise en correspondance bidirectionnelle dans chaque zone de recherche, il faut sélectionner, dans chaque image, les points de contour significatifs à appairer. Le premier critère de sélection impose que les points de contour se trouvent dans la zone de recherche considérée. Seuls ceux dont les abscisses sont comprises entre les extrémités de la zone de recherche sont conservés. Le second critère de

sélection est basé sur l'amplitude du gradient. Soit  $d(x)$ , l'amplitude du gradient associée au point de contour d'abscisse  $x$ . L'algorithme de sélection ne conserve donc que les points de contour gauches vérifiant la relation :

$$\frac{Max_g}{2 \cdot n} \leq d(x) \leq Max_g \text{ ou } Min_g \leq d(x) \leq \frac{Min_g}{2 \cdot n}$$

où  $n$  est le niveau de recherche et  $Max_g$  (resp.  $Min_g$ ) l'amplitude maximale (resp. minimale) du gradient de l'image gauche.

Pour sélectionner les points de contour droits, on utilise une relation similaire :

$$\frac{Max_d}{2 \cdot n} \leq d(x) \leq Max_d \text{ ou } Min_d \leq d(x) \leq \frac{Min_d}{2 \cdot n}$$

où  $n$  est le niveau de recherche et  $Max_d$  (resp.  $Min_d$ ) l'amplitude maximale (resp. minimale) du gradient de l'image droite.

### V-4.2.3 Résultats

#### V-4.2.3.1 Piéton évoluant devant le stéréoscope

La figure V-13 montre les résultats de la mise en correspondance hiérarchique obtenus sur la séquence stéréoscopique "piéton".

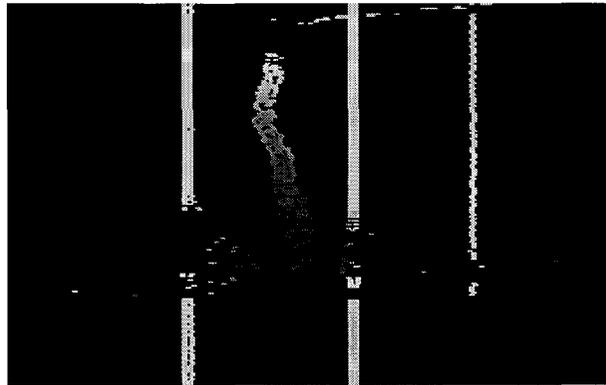


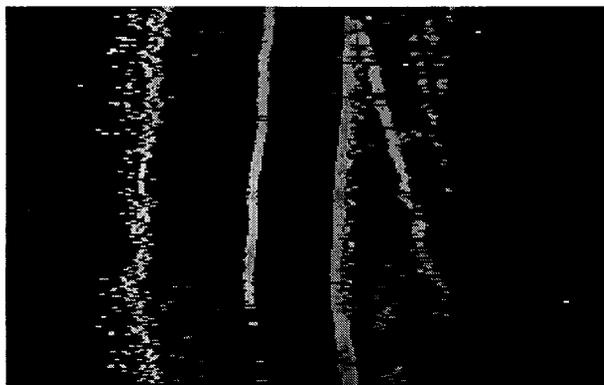
Figure V-13. Résultat obtenu sur la séquence "piéton".

On constate sur cette figure que les résultats sont meilleurs que ceux obtenus par la méthode bidirectionnelle. Les bandes blanches sont mieux détectées. On constate toujours la présence de faux appariements sur les images où se produisent des occlusions. Les points de contour du piéton sont bien mis en correspondance. Les variations de position et de distance sont

cohérentes d'une ligne à l'autre. Néanmoins, on constate toujours, sur certaines images, quelques "trous" correspondant à des appariements manqués.

#### *V-4.2.3.2 Route dégagée et passage d'une sortie d'autoroute*

La figure V-14 montre les résultats de la mise en correspondance hiérarchique obtenus sur la séquence stéréoscopique "sortie d'autoroute".



*Figure V-14. Résultat obtenu sur la séquence "sortie d'autoroute".*

Sur cette figure, on constate que les bandes blanches sont mieux détectées. Le nombre de faux appariements a diminué. De même, la plupart des "trous" ont disparu. Il faut néanmoins noter que la détection de la bande de gauche n'est toujours pas nette.

#### *V-4.2.4 Conclusion sur l'algorithme hiérarchique*

L'algorithme hiérarchique est plus fiable que les méthodes décrites précédemment car il décompose le problème en une succession de sous problèmes plus simples. Il permet ainsi de diminuer la complexité de la grille de mise en correspondance. En effet, avec cet algorithme, Burie traite 16 à 18 couples d'images stéréoscopiques à la seconde. De plus, cette technique présente l'avantage suivant : si une erreur intervient, elle ne se répercute pas sur l'ensemble des résultats comme cela pouvait se produire pour l'analyse séquentielle ou bidirectionnelle. L'erreur reste donc localisée dans la zone de recherche où elle s'est produite. Il faut toutefois noter que l'algorithme hiérarchique peut produire de faux appariements. En effet, la sélection des points de contour à appairer à un niveau de recherche est réalisée par un critère de seuillage. Ainsi, on peut se trouver dans des situations où un point de contour est sélectionné dans l'image gauche et son homologue n'est pas sélectionné dans l'image droite et inversement. La procédure de mise en correspondance peut donc appairer le point de contour sélectionné dans l'image gauche avec un point de contour de l'image droite qui n'est pas en réalité le bon point de contour homologue. De

plus, cette erreur d'appariement peut se propager dans l'ensemble des résultats car cet appariement erroné sera utilisé pour trouver les appariements aux niveaux de recherche inférieurs.

Il faut noter que cette méthode de mise en correspondance a été implantée dans le système de détection d'obstacles réalisé par le Centre d'Automatique de Lille dans le cadre du programme Prometheus [Burie 95]. Ce système de détection a été intégré au véhicule expérimental Prolab 2. Celui-ci a été présenté aux pouvoirs publics et à la presse au cours de la démonstration clôturant le programme Prometheus ainsi qu'à de nombreuses reprises où il a donné entière satisfaction. Les résultats obtenus par le système de détection, qui a été chargé à détecter les piétons, ont été excellents. La détection des piétons a atteint un taux de réussite approchant les 100 %.

## V-5 TRIANGULATION GEOMETRIQUE

Lorsque la mise en correspondance est terminée, la procédure de triangulation reçoit la liste des couples de points de contour appariés. Son rôle est de déterminer pour chaque couple, la distance et la position du point objet correspondant dans la scène tridimensionnelle. Cette procédure utilise :

- Les positions des deux points de contour sur les capteurs gauche et droit.
- Les paramètres de calibration déterminés lors de la phase de calibrage du stéréoscope linéaire (Cf. Chapitre III).

A partir des équations définies au chapitre III (Cf. § III-4.3), les formules de triangulation permettant de déterminer la distance  $Z_p$  et la position  $X_p$  d'un point  $P$  de la scène sont :

$$\begin{cases} Z_p = \frac{f \cdot E \cdot N}{l \cdot (x_g - x_d)} \\ X_p = \frac{l \cdot Z_p \cdot (x_g - o_g)}{f \cdot N} - \frac{E}{2} \end{cases}$$

où:

- $X_p$  est la position du point objet  $P$  le long de l'axe **OX**.
- $Z_p$  est la distance du point objet  $P$  le long de l'axe **OZ**.
- $x_g$  et  $x_d$  sont respectivement la position du point image sur le capteur gauche et sur le capteur droit.

- $o_g$  est la position du pixel central sur le capteur gauche, c'est à dire le pixel sur lequel se projette un point situé le long de l'axe optique.
- $f$  est la distance focale des objectifs.
- $E$  est l'entre axe du stéréoscope linéaire.
- $N$  est la résolution du capteur, c'est à dire le nombre de pixels dans l'image.
- $l$  est la longueur des capteurs.

Les résultats obtenus sont liés au repère du stéréoscope linéaire, c'est à dire au repère  $\{\mathbf{O}, \mathbf{X}, \mathbf{Z}\}$  (Cf. Chapitre III, Figure III-11).

On remarquera qu'on ne détermine pas la hauteur à laquelle le plan optique coupe les objets. Celle-ci est, en effet, inutile car elle n'apporte aucune information supplémentaire sur les objets détectés.

## V-6 CONCLUSION

Au cours de ce chapitre, nous avons présenté succinctement l'approche proposée par Burie pour résoudre le problème de la mise en correspondance stéréoscopique des images linéaires. Au début, il a proposé un algorithme séquentiel qui construit les appariements en analysant les images du début vers la fin. Afin d'améliorer les résultats de mise en correspondance, il a appliqué son algorithme séquentiel deux fois, du début vers la fin et de la fin au début des images linéaires. Une procédure de comparaison des résultats des deux analyses permet ensuite de construire une liste définitive d'appariements. Malgré cette première amélioration, les résultats obtenus présentaient toujours de faux appariements ainsi que des appariements manqués. Pour limiter ces problèmes, il a développé un algorithme hiérarchique basé sur la notion de niveau de recherche. Les résultats obtenus sont meilleurs que les précédents. Néanmoins, quelques problèmes subsistent toujours, surtout en ce qui concerne les appariements manqués.

Afin d'améliorer les résultats et de pouvoir traiter des scènes plus complexes, nous proposons une autre approche pour le problème de la mise en correspondance stéréoscopique des images linéaires. Il s'agit d'une approche utilisant un outil neuronal. Dans un premier temps, le problème de l'appariement est exprimé sous forme d'un problème de minimisation d'une fonction de coût. Celle-ci est construite à partir des différentes contraintes du problème. Un réseau de neurones de Hopfield est ensuite utilisé pour assurer le processus de minimisation. Avant de décrire notre approche, nous allons commencer par une présentation générale de l'utilisation d'un

réseau de neurones de Hopfield pour résoudre un problème de minimisation. Celle-ci fait l'objet du chapitre suivant.

## **Chapitre VI**

# **Modèle de Hopfield et optimisation combinatoire**

## VI-1 INTRODUCTION

Sous le terme de réseaux de neurones, on regroupe aujourd'hui un certain nombre de modèles connexionnistes dont l'intention est d'imiter quelques fonctions du cerveau humain en reproduisant certaines de ses structures de base. Parmi les plus connus de ces modèles on trouve le modèle de Hopfield. Celui-ci est basé sur le fait que, dans le cerveau humain, l'information circule jusqu'à ce que les décisions soient prises par étapes successives. Selon Hopfield, le système nerveux recherche des états stables, ou attracteurs, qui correspondent à des faits mémorisés. Les états voisins tendent à se rapprocher d'un état stable, ce qui autorise la correction des erreurs et la capacité à compléter des informations manquantes. Le modèle de Hopfield est donc fondamentalement une mémoire adressable par son contenu : lorsqu'une information inconnue ou bruitée est présentée au réseau, elle sera attirée vers l'élément mémorisé qui lui correspond le mieux.

Pour réaliser de telles propriétés, Hopfield a proposé un réseau où les neurones sont tous interconnectés. Chaque neurone effectue une somme pondérée de tous les signaux d'entrées qui lui parviennent et émet un signal de sortie à travers une fonction d'activation à seuil. Si cette somme dépasse le seuil, le neurone est actif, sinon il est inactif. A partir d'un état initial, le réseau effectue un certain nombre d'itérations avant de se stabiliser dans un état où tous les réponses des neurones ne peuvent plus changer.

Un autre apport essentiel de Hopfield au domaine des réseaux de neurones artificiels est l'analogie formelle établie entre son modèle et certains systèmes physiques tels que les réseaux de particules orientées en interaction : les verres de spins [Casaent 92][Hinton 84]. Il montre que durant l'évolution du réseau qui tend à se rapprocher de son état stable, une fonction d'énergie, analogue à celle des verres de spins d'Ising, décroît vers un minimum local. Cette analogie a permis de développer un formalisme riche et bien maîtrisé pour résoudre un très grand nombre de problèmes.

## IV.2 RESEAUX NEURONAUX

Un réseau neuronal, ou connexionniste, est un ensemble d'unités de calcul, appelées neurones artificiels, interconnectées selon une architecture spécifique. Les principales caractéristiques qui définissent de tels réseaux sont :

- Le modèle du neurone artificiel utilisé,
- La structure du réseau,
- La dynamique du réseau,
- L'apprentissage du réseau.

Avant d'aborder le sujet de ce chapitre, il nous a paru judicieux de rappeler, dans ce paragraphe, ces différentes caractéristiques qui sont à la base des modèles neuronaux. Nous nous contenterons de donner un aperçu général. Pour plus de détails, le lecteur intéressé pourra se référer à la littérature [Hérault 94][Davallo 89][Haykin 94].

### VI-2.1 Le neurone artificiel

Le neurone artificiel, ou neurone formel, est l'élément de base d'un réseau connexionniste. C'est un automate qui reçoit des potentiels d'action en entrée et est chargé de produire lui-même un potentiel d'action en sortie. Il a été introduit pour la première fois par MacCulloch et Pitts [MacCulloch 43] qui ont proposé, de façon très simplifiée, le premier modèle mathématique du neurone biologique. Dans leur approche, le neurone artificiel effectue une somme pondérée des potentiels d'action qui lui parviennent. Chacun de ces potentiels d'action est représenté par une valeur numérique qui décrit l'état du neurone qui l'a émis. Cette somme est ensuite comparée à un seuil : si celle-ci est supérieure au seuil alors le neurone est actif et fournit une réponse égale à  $I$ , sinon il est inactif et transmet une réponse égale à  $-I$ .

D'une façon générale, un neurone artificiel peut être défini par (Cf. Figure VI-1) :

- Des signaux d'entrées  $e_i$ ,  $i = 1, \dots, N$ , qui représentent les réponses provenant d'autres neurones et un signal d'entrée externe  $I$ .
- Des coefficients de pondération  $W_i$ ,  $i = 1, \dots, N$ , appelés aussi poids synaptiques, qui caractérisent les connexions qui parviennent au neurone. Chacune de ces pondérations correspond à l'importance attribuée à la réponse du neurone qui l'a émise.
- Une fonction d'entrée totale  $h(\cdot)$  qui élabore un "potentiel" d'entrée  $U$  défini par :

$$U = h(\{e_i\}) = \sum_{i=1}^{i=N} W_i \cdot e_i + I$$

- Une fonction d'activation  $g(\cdot)$ , appelée aussi fonction de transfert ou de sortie, qui calcule, à partir du potentiel d'entrée  $U$ , la réponse  $V$  du neurone artificiel qui servira ensuite à exciter les autres neurones qui lui sont connectés.

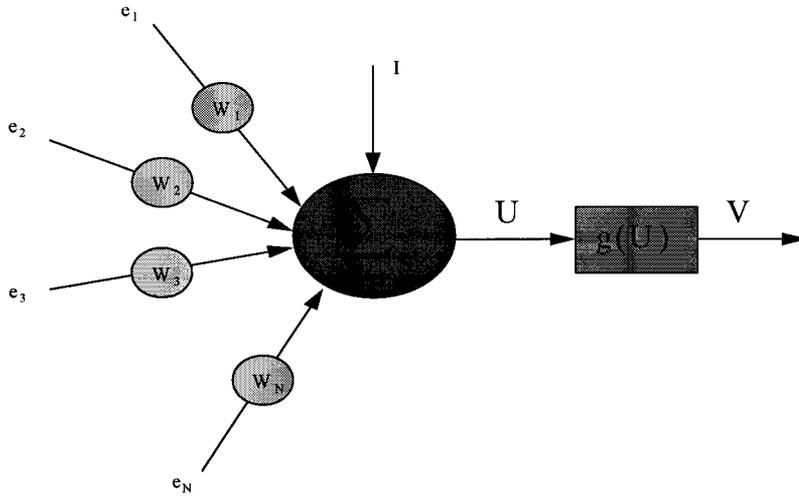


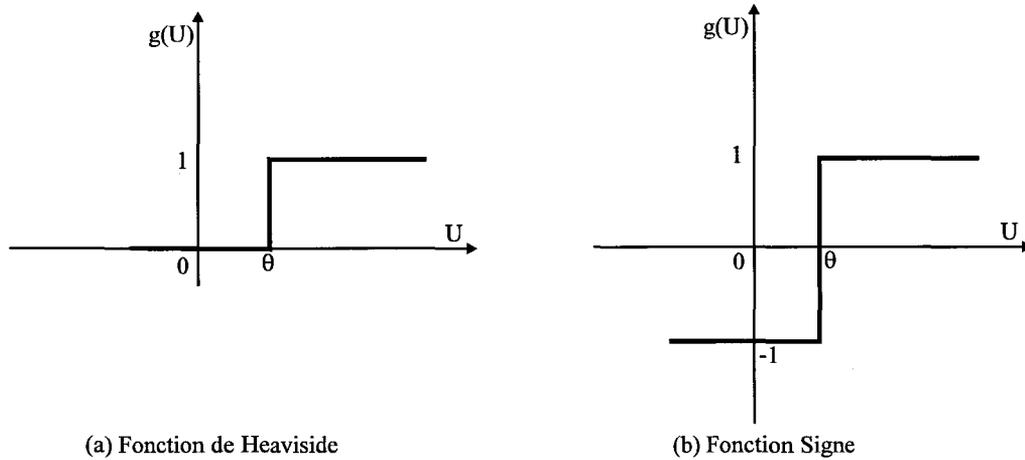
Figure VI-1. Le neurone artificiel.

Les différents choix de modèles de neurones artificiels se caractérisent généralement par le type de la fonction d'activation employée. Celle-ci peut être discrète, continue ou stochastique. On distingue ainsi trois grands modèles de neurones :

- **Le neurone artificiel binaire.** C'est un neurone artificiel dont la sortie est binaire. La fonction d'activation utilisée est une fonction binaire à seuil (Cf. Figure VI-2-b) :

$$\begin{cases} V = 1 & \text{si } U > \theta \\ V = -1 & \text{sinon} \end{cases}$$

où  $\theta$  est un seuil qui représente la limite à partir de laquelle le neurone s'active. Parfois on utilise la valeur  $0$  plutôt que  $-1$  pour représenter l'état inactif du neurone. La fonction signe est alors remplacée par la fonction de Heaviside (Cf. Figure VI-2-a).



**Figure VI-2.** Fonctions d'activations binaires.

- Le neurone artificiel lissé. Ce modèle fournit une sortie à valeur réelle. La fonction d'activation est une version lissée de la fonction binaire à seuil. Généralement, on utilise une fonction sigmoïde de la forme :

$$g(U) = \frac{\exp\left(\frac{U - \theta}{T}\right) - 1}{\exp\left(\frac{U - \theta}{T}\right) + 1} \quad (\text{Cf. Figure VI-3-a})$$

ou :

$$g(U) = \frac{\exp\left(\frac{U - \theta}{T}\right)}{\exp\left(\frac{U - \theta}{T}\right) + 1} \quad (\text{Cf. Figure VI-3-b})$$

$\theta$  est une constante positive et  $T$  est un facteur d'échelle rappelant le rôle d'une température dans les systèmes dynamiques complexes en physique statistique. Ce dernier contrôle la pente en  $U = \theta$ . Celle-ci augmente lorsque  $T$  tend vers 0. Le modèle tend alors vers le modèle binaire.

Il existe d'autres choix de fonctions d'activation pour ce modèle. Nous pouvons citer par exemple les fonctions linéaires à seuils (Cf. Figure VI-4).

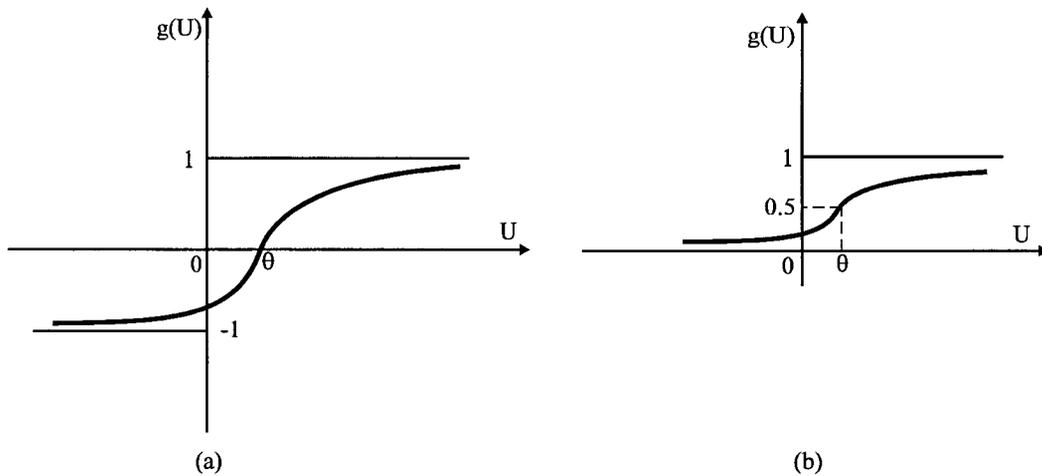


Figure VI-3. Fonctions sigmoïdes.

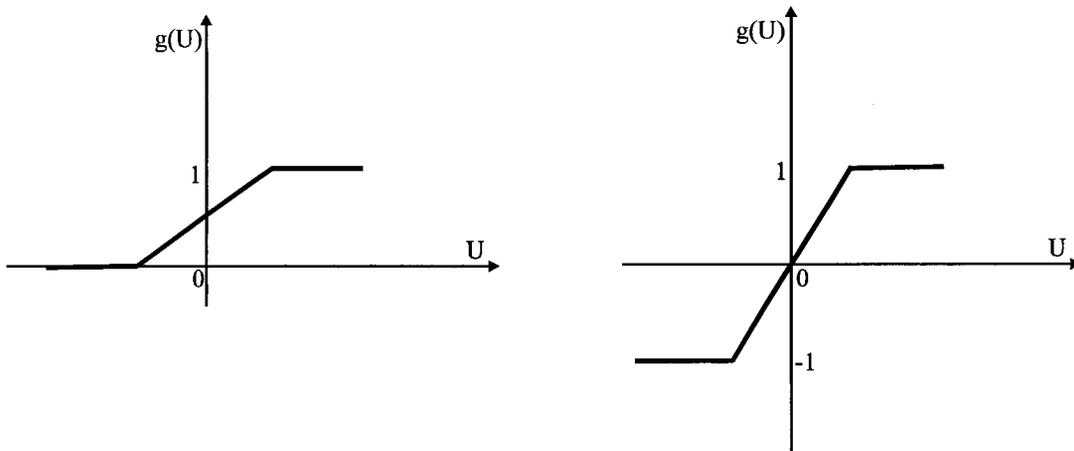


Figure VI-4. Fonctions linéaires à seuils.

- Le neurone artificiel probabiliste. Ce modèle utilise une fonction d'activation stochastique. La sortie du neurone artificiel est binaire et elle est déterminée en fonction d'une certaine probabilité :

$$P(V = 1) = \frac{1}{1 + \exp\left(-\frac{U - \theta}{T}\right)}$$

$$P(V = -1) = 1 - P(V = 1)$$

où  $\theta$  est une constante positive et  $T$  est un paramètre qui représente une température. Nous pouvons constater que lorsque  $T$  tend vers 0, le modèle stochastique tend vers le modèle binaire. Le neurone artificiel probabiliste est principalement utilisé dans les réseaux à recuit simulé comme les machines de Boltzmann [Davallo 89][Hinton 84] et de Cauchy [Szu 87].

## VI-2.2 Structure d'un réseau connexionniste

La structure d'un réseau connexionniste résulte de la façon dont ses neurones sont interconnectés. Elle définit ce qu'on appelle la topologie, ou l'architecture du réseau. Il existe deux grandes familles d'architectures de réseaux : les réseaux à couches et les réseaux entièrement connectés.

### VI-2.2.1 Les réseaux à couches

Ce type de réseaux (Cf. Figure VI-5) présente une couche d'entrée recevant les informations extérieures et une couche de sortie qui fournit les résultats du traitement. Entre ces deux couches extrêmes, se situent une ou plusieurs couches intermédiaires, appelées aussi couches cachées, qui effectuent le traitement. Les neurones de chaque couche reçoivent des signaux provenant des neurones de la couche précédente et en envoient aux neurones de la couche suivante, mais les neurones appartenant à une même couche ne communiquent pas entre eux. Les signaux ne peuvent alors se propager qu'en avant, de la couche d'entrée vers la couche de sortie. On parle de réseaux "feed-forward". Les connexions d'un neurone peuvent être totales ou partielles, c'est à dire qu'un neurone d'une couche peut être connecté soit à certains, soit à tous les neurones de la couche suivante.

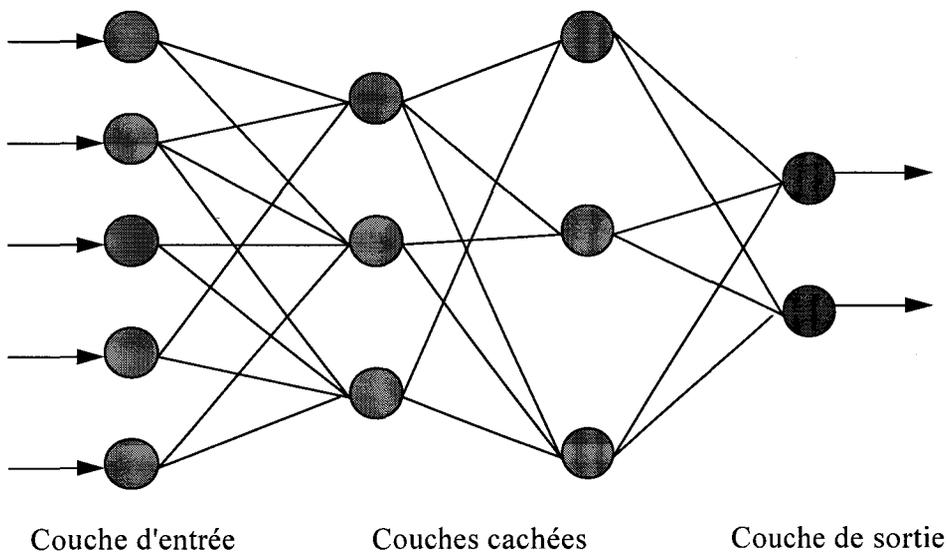
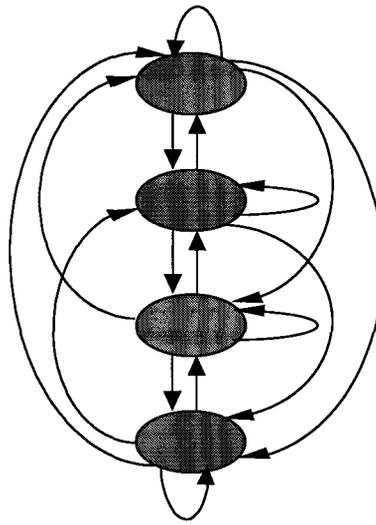


Figure VI-5. Structure d'un réseau à couches.

### VI-2.2.2 Les réseaux entièrement connectés

A l'opposé des réseaux à couches, un réseau entièrement connecté est composé d'une seule couche où chaque neurone communique avec tous les autres, et même éventuellement avec lui-même (Cf. Figure VI-6). Ces réseaux sont appelés aussi des réseaux "feed-back". On constate

que, dans ce type de réseau, une même couche joue à la fois le rôle de couche d'entrée, de couche de traitement et de couche de sortie.



*Figure VI-6. Structure d'un réseau entièrement connecté.*

### VI-2.3 Dynamique d'un réseau connexionniste

La dynamique d'un réseau connexionniste consiste à faire communiquer les neurones interconnectés. Elle s'intéresse à l'évolution des états des neurones et à leur propagation dans le réseau. Chaque neurone élabore son potentiel d'entrée à partir des signaux qui lui parviennent, puis calcule sa sortie qui servira de signal d'entrée pour les neurones qui lui sont connectés.

Pour une architecture multicouches, l'information à traiter est appliquée à la couche d'entrée du réseau, puis propagée, via les couches cachées, vers la couche de sortie. Pour changer d'état, un neurone d'une couche doit attendre les résultats des traitements des neurones de la couche précédente. En revanche, les neurones d'une même couche peuvent changer d'état simultanément dès que les résultats des traitements de la couche précédente sont disponibles.

Quand il s'agit d'un réseau entièrement connecté, l'information à traiter est appliquée à tous les neurones. Le réseau cherche ensuite à atteindre un état stable où les neurones ne peuvent plus évoluer. Pour changer d'état, un neurone utilise toutes les sorties issues des autres neurones. Il faut noter que le changement des états des neurones peut être effectué soit en mode synchrone (Cf. § VI-3.2.1), c'est dire que tous les neurones changent d'états en même temps, soit en mode asynchrone (Cf. § VI-3.2.2), c'est à dire qu'un seul neurone change d'état à la fois. Par ailleurs, l'évolution des états des neurones peut être continue ou non dans le temps.

### VI-2.4 Apprentissage d'un réseau connexionniste

Dans un système connexionniste, la connaissance est mémorisée dans les liaisons inter-neuronales. Chaque liaison est caractérisée par une valeur numérique, appelée poids de connexion ou poids synaptique, qui pondère le signal transmis d'un neurone à l'autre. L'apprentissage d'un réseau concerne l'évolution des poids synaptiques. Il consiste à déterminer les valeurs optimales des poids des connexions pour que le réseau accomplisse une certaine tâche.

### VI-2.5 Conclusion

Au cours de ce bref exposé, nous avons vu qu'un réseau connexionniste est un ensemble de neurones artificiels interconnectés selon une architecture spécifique. La propriété la plus intéressante de tels réseaux est le mécanisme d'apprentissage qui tente de copier le processus de mémorisation du cerveau humain. Les deux grandes familles d'architectures de réseaux, les réseaux à couches et les réseaux entièrement connectés, ont permis de développer un certain nombre de modèles qui sont, actuellement, utilisés pour résoudre différents types de problèmes. Dans le paragraphe suivant, nous allons présenter l'un des plus connus de ces modèles, à savoir le modèle de Hopfield.

## VI-3 LE MODELE DE HOPFIELD

Le modèle original de Hopfield [Hopfield 82], qui a fait par la suite l'objet de nombreuses améliorations, est basé sur l'utilisation des neurones artificiels de MacCulloch et Pitts, c'est à dire des neurones artificiels binaires (Cf. § VI-2.1). On parle alors du modèle discret de Hopfield. Dans ce paragraphe, nous utilisons les deux notations  $\{0,1\}$  ou  $\{-1,1\}$  pour représenter les sorties, ou les états, d'un neurone binaire artificiel. En fait, ces deux notations sont équivalentes puisqu'il est possible de passer du domaine  $\{0,1\}$  au domaine  $\{-1,1\}$  et vice-versa par un simple changement de variable. Ainsi, à chaque fois qu'il s'agira d'une simplification de calcul, on utilisera l'une ou l'autre de ces deux notations.

### VI-3.1 Structure du réseau

Le réseau de Hopfield est un réseau entièrement connecté. Chaque neurone  $n_i$  reçoit en entrées toutes les sorties des autres neurones (Cf. Figure VI-7). La connexion du neurone  $n_i$  vers le neurone  $n_j$  est caractérisée par un coefficient de couplage, ou poids synaptique, noté  $W_{ji}$ . Ce couplage correspond à l'importance accordée à l'action du neurone  $n_i$  sur le neurone  $n_j$ .  $V_i$  représente la sortie, ou l'état, du neurone  $n_i$ .

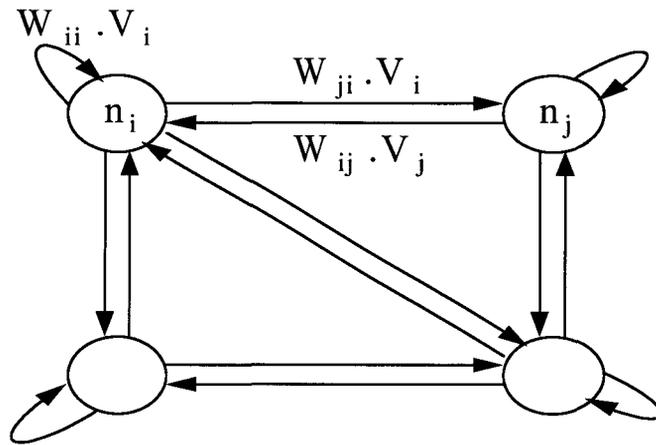


Figure VI-7. Connexion des neurones dans un réseau de Hopfield.

La figure VI-8 montre l'organisation d'un réseau de Hopfield possédant  $N$  neurones. Les poids des connexions sont matérialisés par des cercles blancs et les neurones par des cercles gris.

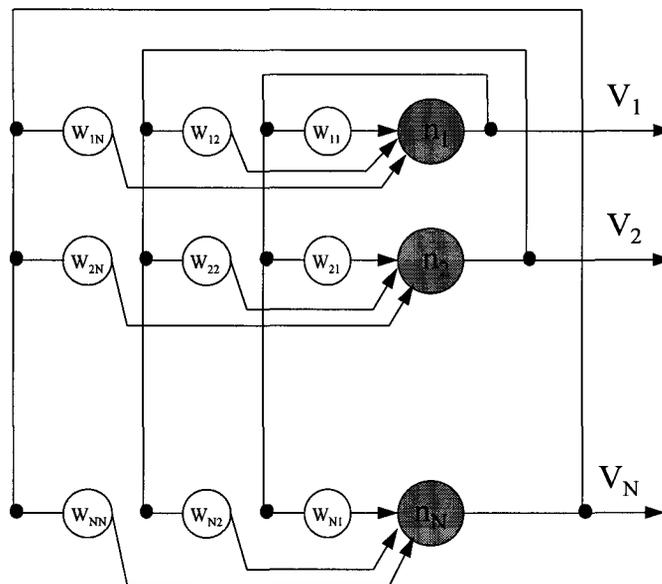


Figure VI-8. Architecture d'un réseau de Hopfield.

Un réseau de Hopfield est donc défini par un ensemble de  $N$  neurones en interaction totale et une matrice  $W$  constituée des poids des connexions  $W_{ij}$ ,  $i = 1, \dots, N$  et  $j = 1, \dots, N$ . En considérant les états élémentaires  $V_i$  de tous les neurones  $n_i$ ,  $i = 1, 2, \dots, N$ , le système peut être décrit par le vecteur  $\vec{V} = (V_1, V_2, \dots, V_N)^T \in \{-1, 1\}^N$  qui représente l'état du réseau.

### VI-3.2 Dynamique du réseau

Considérons un réseau de Hopfield à  $N$  neurones. L'évolution dynamique du réseau consiste à faire passer le réseau d'un état à un autre en modifiant les états élémentaires de ses

neurones. La mise à jour des éléments du vecteur d'état peut se faire en mode synchrone ou en mode asynchrone. Nous rappelons que le fonctionnement du réseau est à temps discret puisque le modèle de neurones utilisé est binaire.

### VI-3.2.1 Mise à jour en mode synchrone

En mode synchrone, toutes les composantes du vecteur d'état  $\vec{V}$  sont modifiées simultanément. Ainsi, l'état de chaque neurone à l'instant  $t+1$  est calculé à partir des états, à l'instant  $t$ , de tous les autres neurones :

$$\begin{cases} V_i(t+1) = 1 & \text{si } U_i(t+1) > 0 \\ V_i(t+1) = -1 & \text{si } U_i(t+1) < 0 \\ V_i(t+1) = V_i(t) & \text{si } U_i(t+1) = 0 \end{cases} \quad \text{Equation VI-1}$$

où  $U_i$  est le potentiel d'entrée du neurone  $n_i$  :

$$U_i(t+1) = \sum_{j=1}^N W_{ij} V_j(t) .$$

### VI-3.2.2 Mise à jour en mode asynchrone

C'est généralement le mode le plus utilisé. Dans ce cas, le vecteur d'état  $\vec{V}$  est mis à jour élément par élément en tenant compte des nouvelles composantes dès qu'elles sont disponibles. Dans cette optique, plusieurs variantes sont possibles, mais la plus importante est le mode asynchrone séquentiel où on calcule les nouvelles valeurs des états des neurones dans l'ordre où ils apparaissent dans le vecteur d'état. Ainsi, lorsqu'on arrive à l'élément  $V_i$ , on exploite le fait que les composantes  $V_1, V_2, \dots, V_{i-1}$  ont déjà été mises à jour, de sorte que la nouvelle valeur de  $V_i$  se calcule comme suit :

$$\begin{cases} V_i(t+1) = 1 & \text{si } U_i(t+1) > 0 \\ V_i(t+1) = -1 & \text{si } U_i(t+1) < 0 \\ V_i(t+1) = V_i(t) & \text{si } U_i(t+1) = 0 \end{cases} \quad \text{Equation VI-2}$$

où  $U_i$  est le potentiel d'entrée du neurone  $n_i$  :

$$U_i(t+1) = \sum_{j=1}^{i-1} W_{ij} V_j(t+1) + \sum_{j=i}^N W_{ij} V_j(t)$$

A partir du vecteur d'état  $\vec{V}(t) = (V_1(t), V_2(t), \dots, V_N(t))^T$ , qui représente l'état du réseau à l'instant  $t$ , une itération de mise à jour asynchrone séquentielle implique donc la mise à jour

successive de  $N$  composantes suivant l'équation VI-2 pour finalement aboutir au nouveau vecteur d'état  $\vec{V}(t+1) = (V_1(t+1), V_2(t+1), \dots, V_N(t+1))^T$  qui représente l'état du réseau à l'instant  $t+1$ . C'est ce type de mise à jour que nous utiliserons par la suite.

Concernant l'ordre de mise à jour des composantes du vecteur d'état, on peut envisager d'autres variantes que la stratégie séquentielle, par exemple en procédant de façon aléatoire, mais de telle manière que chaque composante soit mise à jour une fois au moins sur un intervalle de temps donné.

Hopfield [Hopfield 82] a montré que cette dynamique peut conduire le réseau vers un état stable sous les conditions suivantes (Cf. VI-4.2) :

- Les interconnexions entre les neurones sont symétriques, c'est à dire que la matrice  $W$  des poids des connexions est symétriques ( $W_{ij} = W_{ji}$ ).
- Les éléments diagonaux de cette matrice sont nuls ( $W_{ii} = 0$ ), c'est à dire qu'un neurone n'interagit pas avec lui-même.

La stabilité du réseau correspond à un vecteur d'état  $\vec{V}_s$  qui ne peut plus être modifié au cours de la procédure de mise à jour, c'est à dire tel que  $\vec{V}_s(t) = \vec{V}_s(t+1)$ . Il est important de noter qu'un même réseau peut admettre plusieurs états stables qui sont accessibles à partir de configurations initiales différentes.

### VI-3.3 Apprentissage du réseau

Le mécanisme d'apprentissage est la propriété la plus intéressante d'un réseau de neurones. Il consiste à entraîner le réseau en lui présentant successivement en entrée les éléments constitutifs d'une base d'apprentissage. Chaque élément de cette base est un vecteur qui représente une donnée à mémoriser. La dimension des vecteurs de la base d'apprentissage est égale au nombre de neurones de la couche d'entrée du réseau. Les composantes de ces vecteurs peuvent être réelles ou binaires selon le modèle de neurone utilisé dans le réseau.

L'apprentissage d'un réseau est essentiellement assuré par une dynamique de ses connexions : au cours de l'entraînement, le réseau adapte ses poids synaptiques à l'aide de règles, dites règles d'apprentissage.

Dans un réseau de Hopfield, l'apprentissage est assuré par une application quantitative de la règle de Hebb [Hebb 49]. Celle-ci consiste à augmenter le poids d'une connexion entre deux neurones chaque fois qu'ils sont actifs simultanément.

Supposons que l'on souhaite mémoriser dans un réseau de Hopfield à  $N$  neurones un ensemble  $E$  de  $M$  états :

$$E = \{E^m, m = 1, 2, \dots, M\} ,$$

où chaque état  $E^m$  est représenté par un vecteur  $\vec{E}^m$ , appelé vecteur d'état, constitué de  $N$  composantes binaires :

$$\vec{E}^m = (E_1^m, E_2^m, \dots, E_N^m)^T \in \{-1, 1\}^N .$$

Cela signifie que :

- Tous les états  $E^m, m = 1, \dots, M$ , sont des états stables du réseau, selon la dynamique décrite plus haut (Cf. § VI-3.2.2).
- Tous ces états sont, en outre, des états attracteurs, c'est à dire que lorsque le réseau part d'un état initial quelconque, il est attiré vers l'état stable le plus proche.

La procédure d'apprentissage peut se décomposer en trois phases, selon le schéma suivant :

1. Initialisation. On démarre avec un ensemble de poids de connexions tous nuls.
2. Présentation d'un état à mémoriser. On force le réseau dans un état à mémoriser, soit par exemple l'état  $E^{m_0}$ .
3. Modification des poids des connexions. On examine tous les couples  $(n_i, n_j)$  de neurones,  $i \neq j$ , et on augmente  $W_{ij}$  de la quantité  $\Delta W_{ij} = E_i^{m_0} \cdot E_j^{m_0}$  (Cf. Figure VI-9).

Ainsi, en présentant successivement les  $M$  états à mémoriser, on aboutit au résultat suivant :

$$W_{ij} = \sum_{m=1}^M E_i^m \cdot E_j^m \quad \text{Equation VI-3}$$

$E_i^{m_0}$	$E_j^{m_0}$	$\Delta W_{ij}$
1	1	1
1	-1	-1
-1	1	-1
-1	-1	1

Figure VI-9. Dynamique des connexions d'un réseau de Hopfield.

On constate que les poids des connexions sont symétriques, c'est à dire que  $W_{ij} = W_{ji}$ . Il faut noter aussi que l'interaction d'un neurone avec lui-même n'est pas prise en compte, c'est à dire que  $W_{ii} = 0$ . Ainsi, après la phase d'apprentissage, les deux conditions sur la matrice  $W$  sont vérifiées, ce qui permet d'assurer la convergence du réseau vers un état stable lors de son évolution (Cf. § VI-3.2.2).

En phase de fonctionnement, dite aussi de rappel, si l'on part d'un état initial suffisamment proche d'un état  $E^{m_0}$  mémorisé, la dynamique du réseau fait évoluer celui-ci vers l'état final désiré  $E^{m_0}$ . De cette manière, un réseau de Hopfield peut être assimilé à une mémoire associative adressable par son contenu. De façon similaire aux mécanismes de mémorisation du cerveau humain, une information mémorisée n'est pas localisée à des emplacements déterminés, mais elle est distribuée, ou codée, sur l'ensemble des neurones qui constituent ainsi la mémoire. L'intérêt de cette distribution de la mémoire sur plusieurs entités est ce qu'on appelle la résistance au bruit : la perte d'un élément de la mémoire n'entraîne pas la perte totale de l'information mémorisée.

Il est important de noter que l'apprentissage du réseau est statique, car il n'y a pas véritablement de dynamique des connexions. En effet, les poids des connexions peuvent être calculés directement, étant donné l'ensemble des états à mémoriser (Cf. Equation VI-3). Mais, en revanche, le fonctionnement du réseau est dynamique. A partir d'un état initial, le réseau effectue un certain nombre d'itérations afin de converger vers un état mémorisé. Au cours de chaque itération, le réseau passe d'un état à un autre en changeant les états des neurones à l'aide de leur règle d'évolution (Cf. Equation VI-2).

### VI-3.4 Stabilité des états mémorisés

Supposons qu'à l'instant  $t$  le réseau soit dans un état  $E^{m_0}$  mémorisé. Appliquons la règle de mise à jour, décrite au § VI-3.2.2 (Cf. Equation VI-2), aux neurones du réseau. Commençons par calculer l'état du neurone  $n_I$  à l'instant  $t+1$ . L'entrée totale du neurone  $n_I$  à l'instant  $t+1$  est donnée (Cf. § VI-3.2.2) :

$$U_I(t+1) = \sum_{j=1}^N W_{Ij} \cdot V_j(t) .$$

Comme  $V_j(t) = E_j^{m_0}$ , on a :

$$U_I(t+1) = \sum_{j=1}^N W_{Ij} \cdot E_j^{m_0} .$$

A partir de la définition de  $W_{ij}$  (Cf. Equation VI-3), on a :

$$\begin{aligned} U_I(t+1) &= \sum_{j=1}^N \left[ \sum_{m=1}^M E_I^m \cdot E_j^m \right] \cdot E_j^{m_0} \\ &= \sum_{m=1}^M E_I^m \cdot \left[ \sum_{j=1}^N E_j^m \cdot E_j^{m_0} \right] \end{aligned}$$

Supposons que les vecteurs d'état  $\vec{E}^m$ ,  $m = 1, 2, \dots, M$ , soient deux à deux orthogonaux. Dans ce cas on a :

$$\sum_{j=1}^N E_j^m \cdot E_j^{m_0} = N \cdot \delta_{mm_0}$$

où  $\delta_{mm_0}$  est le symbole de Kronecker, égal à 1 si  $m = m_0$  et à 0 sinon. L'entrée totale du neurone  $n_I$  devient alors :

$$U_I(t+1) = N \cdot E_I^{m_0} .$$

On constate que le signe de l'entrée totale  $U_I(t+1)$  du neurone  $n_I$  est le même que celui de son état  $E_I^{m_0}$ . Ainsi, d'après la dynamique du réseau (Cf. Equation VI-2), le neurone ne change pas d'état, c'est à dire que  $V_I(t) = V_I(t+1) = E_I^{m_0}$ .

Intéressons nous maintenant au neurone  $n_i$ . Ceci suppose que tous les neurones  $n_1, \dots, n_{i-1}$  ont été visités. L'entrée totale du neurone  $n_i$  à l'instant  $t+1$  est donnée par (Cf. § VI-3.2.2) :

$$U_i(t+1) = \sum_{j=1}^{i-1} W_{ij} \cdot V_j(t+1) + \sum_{j=i}^N W_{ij} \cdot V_j(t) .$$

Comme les neurones  $n_1, \dots, n_{i-1}$  n'ont pas changé d'état, on a :

$$U_i(t+1) = \sum_{j=1}^N W_{ij} \cdot E_j^{m_0} .$$

En suivant le même raisonnement que pour le neurone  $n_i$ , on obtient :

$$U_i(t+1) = N \cdot E_i^{m_0} .$$

Comme le signe de  $U_i(t+1)$  est le même que celui de  $E_i^{m_0}$ , l'état du neurone  $n_i$  ne change pas, c'est à dire que  $V_i(t) = V_i(t+1) = E_i^{m_0}$  (Cf. Equation VI-2). Ainsi, l'état du réseau ne change pas, c'est à dire qu'il est stable. Un état mémorisé correspond donc à un état stable du réseau.

### VI-3.5 Limites des réseaux de Hopfield

Particulièrement simple, un réseau de Hopfield est relativement puissant grâce à sa propriété de se comporter comme une mémoire associative. Toutefois les limites suivantes apparaissent :

- Etats poubelles. La définition des poids des connexions par la règle de Hebb dans les réseaux de Hopfield introduit des états stables qui ne correspondent pas à des états mémorisés et qui peuvent être fortement attracteurs. Ces états parasites correspondent en fait à des combinaisons linéaires de certains états mémorisés. Diverses solutions ont été proposées pour régler ce problème [Pöppel 87][Hopfield 83].
- Nécessité d'un prétraitement préalable. Avant de mémoriser les états dans le réseau, ceux-ci doivent être prétraités pour les orthogonaliser afin d'assurer leur stabilité (Cf. § VI-3.4). Afin d'éviter cette étape de prétraitement, Personnaz [Personnaz 86] propose une méthode de calcul des poids des connexions permettant d'assurer la stabilité des états à mémoriser, qu'ils soient orthogonaux deux à deux ou non.
- Oubli catastrophique. La capacité d'apprentissage de la règle de Hebb est très réduite. D'après Amit et al., en supposant le nombre  $N$  de neurones très grand, le nombre

maximal d'états correctement retrouvés est inférieur à  $0.138 \cdot N$  [Amit 85]. Si on dépasse cette limite, c'est à dire si on veut "surcharger" la mémoire, tous les états appris sont perdus.

### VI-3.6 Conclusion

Nous avons vu qu'un réseau de Hopfield, constitué de neurones entièrement connectés, est un système dynamique. L'apprentissage d'un tel réseau est réalisé par une application quantitative de la règle de Hebb. La principale propriété de ce type de réseau est son comportement similaire à des mémoires associatives, ce qui le rend très bien adapté pour résoudre des problèmes de reconnaissance. Il faut toutefois noter que, dans sa version la plus simple, un réseau de Hopfield présente un certain nombre d'insuffisances qui sont dues principalement à l'utilisation de la règle de Hebb dans la phase d'apprentissage du réseau.

## VI-4 RESEAU DE HOPFIELD ET OPTIMISATION COMBINATOIRE

On parle de réseaux connexionnistes dès que l'on trouve des neurones en interactions. Mais on connaît aussi d'autres types d'interactions, comme celles par exemple des particules en physique statistique telles que les verres de spins [Casaent 92][Hinton 84]. Il n'en fallait pas moins à quelques précurseurs comme Little et Hopfield pour constater ce parallélisme et l'exploiter [Little 74][Hopfield 82]. Ce parallélisme a permis d'introduire la notion de fonction d'énergie d'un ensemble de neurones tous interconnectés et de développer un formalisme riche et bien maîtrisé pour aborder un grand nombre de problèmes.

### VI-4.1 Structure générale du réseau

Nous avons vu que, dans un réseau de Hopfield, chaque neurone est soumis à l'action des autres neurones, y compris lui-même. Afin de rendre plus générale l'utilisation du réseau, les neurones sont soumis, en plus, à une action provenant de l'extérieure. Celle-ci se traduit, pour chaque neurone  $n_i$ , par une entrée externe notée  $I_i$  (Cf. Figure VI-10).

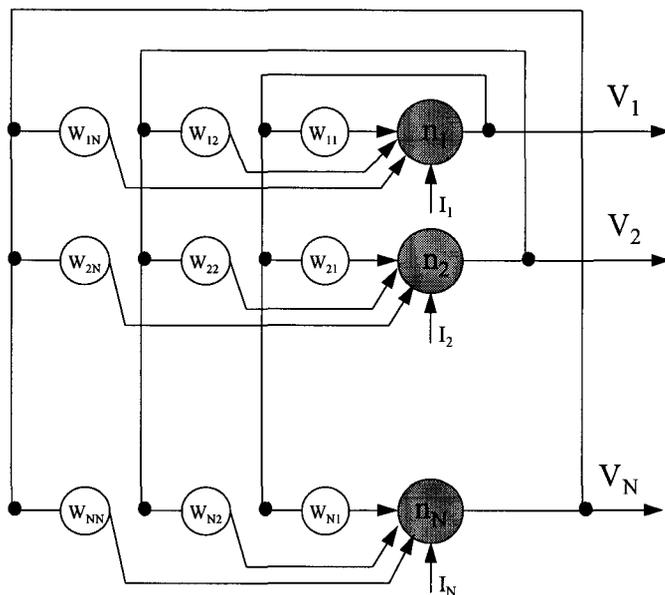


Figure VI-10. Structure générale d'un réseau de Hopfield.

Avec cette nouvelle structure, la dynamique du réseau ne change pas, mais il faut simplement tenir compte, lors du calcul de l'entrée totale d'un neurone \$n\_i\$, de l'entrée externe qui doit s'ajouter à la somme pondérée des entrées provenant des autres neurones du réseau (Cf. § VI-3.2.2) :

$$U_i(t+1) = \sum_{j=1}^{i-1} W_{ij} V_j(t+1) + \sum_{j=i}^N W_{ij} V_j(t) + I_i .$$

#### VI-4.2 Fonction d'énergie du réseau

Considérons un réseau de Hopfield à \$N\$ neurones. Soit \$W\$ la matrice des poids des connexions \$W\_{ij}\$, \$i = 1, \dots, N\$ et \$j = 1, \dots, N\$, et soit \$\vec{I}\$ le vecteur constitué par les \$N\$ entrées extérieures \$I\_i\$, \$i = 1, \dots, N\$. Hopfield définit une application, appelée fonction d'énergie du réseau, de \$\{-1, 1\}^N\$ dans \$R\$ qui à tout vecteur d'état \$\vec{V} = (V\_1, V\_2, \dots, V\_N)^T\$ associe la quantité :

$$H(\vec{V}) = -\frac{1}{2} \cdot \vec{V}^T \cdot W \cdot \vec{V} - \vec{I}^T \cdot \vec{V} - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N W_{ij} \cdot V_i \cdot V_j - \sum_{i=1}^N I_i \cdot V_i$$

Equation VI-4

Examinons l'évolution de cette fonction d'énergie au cours de l'évolution dynamique du réseau. Faisons évoluer l'état du neurone \$n\_i\$ selon la règle décrite au § VI-3.2.2. Pour la variation

$\Delta V_i = V_i(t+1) - V_i(t)$  de l'état  $V_i$  du neurone  $n_i$ , la variation correspondante de la fonction d'énergie vaut :

$$\begin{aligned} \Delta H &= -\frac{1}{2} \left[ \sum_{j=1}^{i-1} W_{ij} \cdot \Delta V_i \cdot V_j(t+1) + \sum_{j=i+1}^N W_{ij} \cdot \Delta V_i \cdot V_j(t) \right] \\ &\quad - \frac{1}{2} \left[ \sum_{j=1}^{i-1} W_{ji} \cdot \Delta V_i \cdot V_j(t+1) + \sum_{j=i+1}^N W_{ji} \cdot \Delta V_i \cdot V_j(t) \right] \\ &\quad - \frac{1}{2} W_{ii} \cdot \Delta V_i^2 - I_i \cdot \Delta V_i \\ &= -\frac{1}{2} \left[ \sum_{j=1}^{i-1} (W_{ij} + W_{ji}) \cdot \Delta V_i \cdot V_j(t+1) + \sum_{j=i+1}^N (W_{ij} + W_{ji}) \cdot \Delta V_i \cdot V_j(t) \right] \\ &\quad - \frac{1}{2} W_{ii} \cdot \Delta V_i^2 - I_i \cdot \Delta V_i \end{aligned}$$

Supposons que la matrice  $W$  des poids des connexions vérifie les conditions précédemment citées, c'est à dire qu'elle est symétrique avec des éléments diagonaux nuls. L'expression de la variation  $\Delta H$  de la fonction d'énergie s'écrit alors :

$$\begin{aligned} \Delta H &= -\Delta V_i \cdot \left[ \sum_{j=1}^{i-1} W_{ij} \cdot V_j(t+1) + \sum_{j=i+1}^N W_{ij} \cdot V_j(t) + I_i \right] \\ &= -\Delta V_i \cdot U_i(t+1) \end{aligned} \tag{Equation VI-5}$$

Appliquons au réseau la règle d'évolution décrite au § VI-3.2.2 (Cf. Equation VI-2). Le tableau représenté sur la figure VI-11 récapitule le sens de variation de la fonction d'énergie  $H$  en fonction de  $\Delta V_i$  et  $U_i$ .

$V_i(t)$	$U_i(t+1)$	$V_i(t+1)$	$\Delta V_i = V_i(t+1) - V_i(t)$	$\Delta H$
1	>0	1	0	0
1	<0	-1	-2	<0
-1	>0	1	2	<0
-1	<0	-1	0	0

Figure VI-11. Sens de variation de la fonction d'énergie du réseau.

On constate alors que la variation  $\Delta H$  de la fonction d'énergie engendrée par la variation de l'état d'un neurone est toujours négative. La fonction d'énergie  $H$  est donc monotone décroissante. Comme le nombre de neurones du réseau est fini, la fonction d'énergie  $H$  est bornée. Par conséquent, au bout d'un certain nombre d'itérations, une valeur minimale de cette fonction est atteinte et le système se stabilise. Ainsi, un état stable du réseau correspond à un minimum de sa fonction d'énergie  $H$ . Il faut cependant noter que le réseau peut converger vers l'un des minimums locaux de sa fonction d'énergie. Ces minimums locaux, qui peuvent correspondre à différentes valeurs de la fonction d'énergie, sont accessibles à partir de configurations initiales différentes.

Revenons au problème de la mémoire associative que nous avons évoqué au § VI-3.3. Nous avons vu qu'après apprentissage d'un réseau de Hopfield, celui-ci évolue, au cours de son fonctionnement, vers un état mémorisé, s'il a été stimulé par un état proche de celui-ci. Nous avons vu aussi que le réseau peut converger vers des états, appelés états poubelles, qui ne sont pas mémorisés (Cf. § VI-3.5). Ces états parasites correspondent en fait à des minimums locaux de la fonction d'énergie du réseau. Ils peuvent être atteints lorsque l'état initial du réseau est trop éloigné des états mémorisés. Ces états sont aussi liés à la capacité d'apprentissage du réseau [Peretto 84]. Un remède à ce problème consiste à introduire du bruit pour permettre au réseau de sortir de ces états parasites que constituent les minima locaux de sa fonction d'énergie. Cette technique, que nous analyserons au § VI-5, relève du principe du recuit simulé.

#### VI-4.3 Formulation duale

Nous avons vu qu'un réseau de Hopfield peut être décrit par une fonction d'énergie définie sous une forme quadratique à partir de la matrice des poids de ses connexions et du vecteur de ses entrées externes. Nous avons constaté que l'évolution dynamique du réseau l'amène à se stabiliser sur des minima locaux de cette fonction. En d'autres termes, les minima locaux de la fonction d'énergie du réseau agissent comme des attracteurs au cours de son évolution.

Cette propriété qu'a le réseau de "rechercher" un minimum de sa fonction d'énergie le rend apte à résoudre certains problèmes d'optimisation où l'objectif est de minimiser une fonction de coût représentée sous une forme quadratique. Cette dualité a été introduite par Hopfield lui-même, dans le but de donner une solution au problème du voyageur de commerce [Hopfield 85].

D'une manière générale, la méthodologie neuronale appliquée à un problème d'optimisation comprend trois étapes successives :

1. Codage du problème. Il s'agit de définir un système caractérisé par un ensemble de variables d'état binaires et une fonction de coût, sous forme quadratique, telle que, par construction, les solutions du problème correspondent à des minimums de cette fonction.
2. Construction du réseau. Il s'agit, ici, de considérer un réseau entièrement connecté, où chaque neurone représente une variable d'état binaire, et de déterminer ses paramètres, c'est à dire la matrice des poids des connexions et le vecteur des entrées externes, en identifiant la fonction de coût du problème et la fonction d'énergie du réseau définie au § VI-4.2.
3. Processus de minimisation. A partir d'une configuration initiale du réseau, les neurones évoluent selon la dynamique décrite au § VI-3.2.2 (Cf. Equation VI-2) jusqu'à ce qu'ils ne peuvent plus changer d'état. L'état stable du réseau correspond alors à un minimum local de sa fonction d'énergie qui n'est autre que la fonction de coût. Il faut rappeler que pour assurer la convergence du réseau, la matrice des poids des connexions doit vérifier les conditions énoncées par Hopfield, c'est à dire qu'elle doit être symétrique avec une diagonale nulle.

L'intérêt de cette méthode d'optimisation est qu'elle est simple à mettre en oeuvre grâce à l'utilisation d'une architecture neuronale. Cependant, on peut objecter que la classe des problèmes qui peuvent être abordés semble restreinte puisque les fonctions à minimiser doivent être des formes quadratiques. D'autre part, on doit être conscient du fait que le processus de minimisation induit n'est que locale.

#### **VI-4.4 Applications**

L'utilisation d'un réseau de Hopfield pour minimiser une fonction de coût a permis d'aborder plusieurs problèmes d'optimisation combinatoire. Le plus connu de ces problèmes est celui du voyageur de commerce où il s'agit, étant donné un certain nombre de villes, de trouver le plus court chemin passant une fois et une seule par chaque ville. Hopfield a montré que ce problème peut être résolu en utilisant la méthodologie neuronale que nous avons décrite plus

haut [Hopfield 85]. Pour cela, il a construit une fonction de coût, sous forme quadratique, telle que les solutions du problème correspondent à des minimums de cette fonction.

Cette approche a été utilisée également dans le domaine de la vision par ordinateur où plusieurs problèmes sont formalisés en termes de minimisation de fonctions de coût. Récemment, Yacoubi a développé une méthode basée sur un réseau de Hopfield pour estimer le mouvement dans une séquence d'images [Yacoubi 96]. Une méthode similaire a été proposée par Lelaurin dans le cadre de la même application [Lelaurin 96]. Le modèle connexionniste de Hopfield a été utilisé aussi pour résoudre le problème de la mise en correspondance stéréoscopique (Cf. Chapitre II, § II-8.7). En se basant sur ce même type d'approche, nous allons présenter, dans le chapitre VII, la méthode que nous avons développée pour le problème de l'appariement stéréoscopique des images linéaires.

#### **VI-4.5 Conclusion**

Nous avons vu qu'un réseau de Hopfield est caractérisé par une fonction d'énergie qui est définie, sous une forme quadratique, par la matrice des poids des connexions du réseau et le vecteur de ses entrées extérieures. Le fonctionnement du réseau a montré que, sous certaines conditions, cette fonction d'énergie décroît vers un minimum local. Cette propriété permet de proposer une utilisation duale des réseaux de Hopfield dans des problèmes d'optimisation où l'objectif est de minimiser une fonction de coût représentée sous une forme quadratique.

### **VI-5 RESEAUX DE HOPFIELD A RECUIT SIMULE**

Le principal inconvénient du modèle binaire de Hopfield est celui de la présence d'états stables indésirables qui correspondent à des minima locaux de la fonction d'énergie du réseau. Ces états "poubelles" sont fortement attracteurs car le processus de minimisation induit par la dynamique du réseau n'est que local. Nous allons montrer, dans ce paragraphe, comment ce problème peut être résolu en utilisant l'algorithme du recuit simulé. Grâce à sa nature stochastique, celui-ci permet, lors d'un processus de minimisation d'une fonction de coût, d'améliorer la probabilité de convergence vers un minimum global.

#### **VI-5.1 Le recuit simulé**

Introduit en 1983 par Kirkpatrick et al. [Kirkpatrick 83], l'algorithme du recuit simulé trouve ses origines dans la thermodynamique des systèmes physiques où l'on prend en compte les niveaux d'énergie.

Considérons un système constitué d'un grand nombre de particules. Un état de ce système peut être caractérisé par la donnée des états élémentaires de chacune des particules, par exemple la position ou le moment magnétique d'un atome. La probabilité de trouver ce système dans un état  $E$  est proportionnelle au facteur de Boltzmann  $\exp(-H(E)/T)$ , où  $H(E)$  représente l'énergie du système à cet état et  $T$  sa température. Ainsi, les probabilités d'occurrence de deux états  $E_1$  et  $E_2$  sont liées par la relation :

$$\frac{P(E_1)}{P(E_2)} = \exp\left(-\frac{H(E_1) - H(E_2)}{T}\right) \quad \text{Equation VI-6}$$

Dans [Métropolis 53], Métropolis a proposé un algorithme simple pour simuler un tel ensemble d'atomes en équilibre à une température  $T$  donnée. A chaque étape de la simulation, on donne à un atome un déplacement aléatoire, et on calcule la variation d'énergie  $\Delta H$  du système. Si  $\Delta H \leq 0$ , le déplacement est effectué. Le cas où  $\Delta H > 0$  est traité de façon probabiliste : la probabilité d'accepter le déplacement est  $p = \exp(-\Delta H/T)$ . La température  $T$  est un paramètre de contrôle, qui autorise, lorsqu'elle est élevée, des "sauts d'énergie" importants.

Par analogie, dans l'algorithme du recuit simulé, on considère un système caractérisé par un ensemble de variables d'état. On remplace l'énergie par une fonction de coût à minimiser et on utilise la méthode de Métropolis pour explorer les états du système. L'idée du recuit simulé est de "mettre" le système à une température  $T$  élevée, afin qu'il explore beaucoup d'états. Lorsque le système semble stabilisé, on abaisse cette température. Cette procédure est répétée jusqu'à ce que  $T$  atteigne la valeur 0.

On peut illustrer cette méthode de façon intuitive par la minimisation d'une fonction d'une variable réelle (Cf. Figure VI-12). On part d'un état initial  $E_0$  du système et on fixe la température  $T$  à une valeur élevée. A chaque étape, on examine la possibilité de faire varier  $E$  de  $\Delta E$  dans un sens quelconque. Cette variation est acceptée suivant la règle de Métropolis. Comme  $T$  est élevée, on va accepter des "sauts" importants qui vont amener  $E$  au voisinage du minimum global avec une bonne probabilité, alors qu'une méthode de descente de gradient, par exemple, aurait été rapidement coincée dans le premier minimum local rencontré. On diminue ensuite  $T$  progressivement pour "geler" la variable  $E$ .

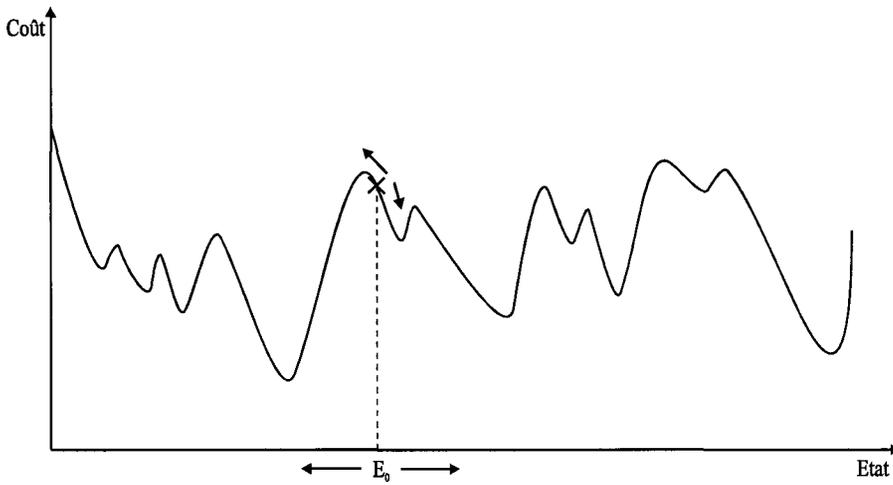


Figure VI-12. Principe du recuit simulé pour la minimisation d'une fonction d'une variable réelle.

Une autre image de cet algorithme est celle d'une bille lâchée au-dessus d'un paysage d'énergie. Au début, on "agite" beaucoup le paysage (valeur de  $T$  élevée) afin que la bille puisse passer d'une vallée à une autre. Puis, progressivement, on diminue l'agitation, c'est à dire  $T$ , de façon à piéger la bille dans une des vallées les plus profondes. A la fin, on arrête totalement l'agitation ( $T = 0$ ) pour que la bille tombe au fond de cette vallée qui correspond à un minimum absolu de l'énergie.

Les températures peuvent être également interprétées comme un moyen pour "aplanir le relief" (Cf. Figure VI-13). Au début, les fortes températures nivellent le paysage. L'abaissement progressif de la température permet ensuite de retrouver doucement tous les accidents du relief.

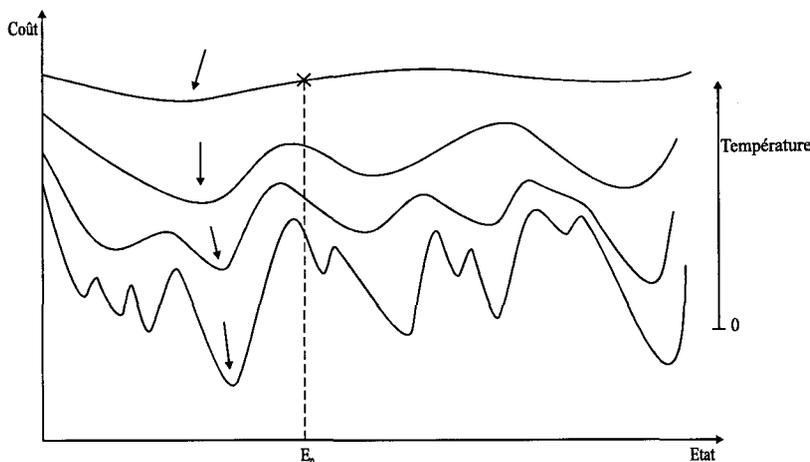


Figure VI-13. Paysage de la fonction de la figure VI-12 à différentes températures.

Face à un problème d'optimisation, la qualité des solutions produites par l'algorithme du recuit simulé est excellente, car il permet, lors du processus de minimisation d'une fonction de

coût, de converger, avec une bonne probabilité, vers un minimum global. Cependant, sa nature intrinsèquement stochastique rend sa convergence très lente. De plus, il faut exécuter le processus de stabilisation pour chaque température de la séquence du recuit, ce qui nécessite une grande quantité de calculs.

### VI-5.2 Utilisation dans les réseaux de Hopfield

Le principe du recuit simulé est à priori indépendant des réseaux de neurones. Nous allons montrer comment on peut le mettre en oeuvre sur un réseau de Hopfield.

Nous avons vu que, dans un réseau de Hopfield, un neurone change d'état dès que son entrée totale vérifie une condition de signe (Cf. § VI-3.2.2, Equation VI-2). Nous avons constaté, ensuite, que ce changement d'état va toujours dans le sens de la diminution de la fonction d'énergie associée au réseau (Cf. § VI-4.2). Pour avoir un réseau fonctionnant suivant le principe du recuit simulé, il suffit de permettre à un neurone des changements d'état augmentant sa fonction d'énergie.

Considérons un réseau de Hopfield constitué de neurones fonctionnant suivant une règle stochastique (Cf. § VI-2.1). Dans ce cas, on parle d'un modèle stochastique de Hopfield. Dans ce type de réseau, les neurones fonctionnent de façon aléatoire, c'est à dire qu'ils envisagent simultanément les deux états possibles  $\{-1, 1\}$  pour modifier leurs états. Le changement d'état d'un neurone du réseau est effectué par tirage au sort avec des probabilités proportionnelles au facteur de Boltzmann  $\exp(-\Delta H/T)$ . La variation de la fonction d'énergie  $\Delta H$  correspondant à un changement d'état d'un neurone  $n_i$  de  $V_i(t)$  à  $V_i(t+1)$  est, comme on l'a vu plus haut (Cf. § VI 4.2) :

$$\Delta H = -(V_i(t+1) - V_i(t)) \cdot U_i(t+1) \quad .$$

Sur le tableau de la figure VI-14, on représente le facteur de Boltzmann pour les quatre transitions possibles entre  $V_i(t)$  et  $V_i(t+1)$ . Pour des raisons de simplification, nous utilisons la valeur  $\theta$ , au lieu de  $-1$ , pour représenter l'état inactif d'un neurone.

$V_i(t) \backslash V_i(t+1)$	1	0
1	1	$\exp(-U_i(t+1)/T)$
0	$\exp(U_i(t+1)/T)$	1

Figure VI-14. Facteur de Boltzmann pour les quatre transitions possibles.

Cherchons maintenant les probabilités de transition d'un neurone  $n_i$ . Supposons qu'à l'instant  $t$  le neurone soit dans un état donné  $V_i(t)$ . En utilisant le tableau de la figure VI-14, le facteur de proportionnalité des probabilités de transition peut être exprimé de la manière suivante :

$$\frac{P(V_i(t+1)=1)}{P(V_i(t+1)=0)} = \exp(U_i(t+1)/T).$$

En normalisant :

$$P(V_i(t+1)=1) + P(V_i(t+1)=0) = 1,$$

on obtient la règle stochastique suivante :

$$P(V_i(t+1)=1) = \frac{1}{1 + \exp(-\frac{U_i(t+1)}{T})}$$

$$P(V_i(t+1)=0) = \frac{\exp(-\frac{U_i(t+1)}{T})}{1 + \exp(-\frac{U_i(t+1)}{T})} \quad \forall V_i(t) \quad \text{Equation VI-7}$$

Ainsi, le nouvel état d'un neurone stochastique est indépendant de son état antérieur. Nous pouvons constater que, lorsque  $T$  tend vers 0, le neurone stochastique tend vers le neurone binaire déterministe (Cf. § VI-2.1).

Pour compléter la mise en oeuvre de l'algorithme du recuit simulé sur un réseau de Hopfield, il reste à choisir une séquence de recuit, c'est à dire une suite décroissante de températures. Le principe de l'algorithme consiste alors à faire évoluer le réseau, pour chaque

température, vers un état stable. Au départ, à des températures très élevées, chaque neurone évolue de façon quasi aléatoire :

$$P(V_i(t+1) = 1) \approx P(V_i(t+1) = 0) \approx \frac{1}{2}.$$

Ensuite, lorsque la température s'abaisse, c'est l'entrée interne  $U_i$  qui détermine la sortie d'un neurone. Enfin, pour des températures proches de 0, le comportement aléatoire est supprimé et le réseau se comporte comme un réseau déterministe.

L'implantation de l'algorithme du recuit simulé sur un réseau de Hopfield prend ainsi la forme suivante :

1. Commencer avec une température  $T_0$  élevée.
2. Initialiser les états des neurones du réseau.
3. Faire évoluer le réseau, à cette température, vers un état stable. Le fonctionnement des neurones est régi par la règle stochastique précédemment décrite (Cf. Equation VI-7).
4. Diminuer la température.
5. Répéter les étapes (3) et (4) jusqu'à ce que le réseau ait atteint un minimum de sa fonction d'énergie, c'est à dire lorsqu'elle ne varie pratiquement plus.

### VI-5.3 Conclusion

Nous venons de montrer qu'il est possible de mettre en oeuvre le principe du recuit simulé sur un réseau de Hopfield. Il suffit pour cela d'utiliser des neurones fonctionnant avec une règle stochastique et de choisir une suite décroissante de températures pour chercher de manière itérative le minimum de la fonction d'énergie du réseau. Ce nouveau fonctionnement permet au réseau de converger, avec une bonne probabilité, vers un minimum global, mais au prix d'une grande quantité de calculs.

## VI-6 RESEAUX ANALOGIQUES DE HOPFIELD

Hopfield et Tank [Hopfield 85] ont remarqué que :

- Lorsque les états des neurones sont restreints à des valeurs binaires, par exemple 0 ou 1, le paysage énergétique, défini par la fonction d'énergie du réseau, possède beaucoup de minima locaux du fait de sa discrétisation.

- Le nombre de minima locaux est plus petit quand les états des neurones sont autorisés à prendre, d'une manière continue, n'importe quelle valeur de l'intervalle  $[0, 1]$ .

Ils ont alors proposé un autre modèle pour implanter la minimisation de la fonction d'énergie du réseau. Il s'agit du modèle analogique de Hopfield qui n'est autre qu'une extension du modèle binaire pour des états continus. La structure du réseau est celle que nous avons vue plus haut (Cf. Figure VI-10), mais les états des neurones, qui peuvent prendre n'importe quelle valeur de l'intervalle  $[0, 1]$ , sont des fonctions continues de la variable temporelle  $t$ , elle aussi continue.

On préfère généralement utiliser des réseaux analogiques plutôt que des réseaux binaires pour les deux raisons suivantes [Hérault 91] :

- Un réseau analogique tend à éviter les oscillations entre les états stables durant son évolution. De ce fait, la convergence des réseaux analogiques est plus rapide que celle des réseaux binaires.
- Les solutions proposées par un réseau analogique sont meilleures que celles produites par des réseaux binaires car les vallées du paysage énergétique sont généralement plus larges.

Néanmoins, le processus de minimisation est local, c'est à dire qu'un réseau analogique tend à proposer pour solution le premier minimum local rencontré.

### VI-6.1 Dynamique d'un réseau analogique

Considérons un réseau analogique à  $N$  neurones. Soient  $W$  la matrice des poids de ses connexions et  $\vec{I}$  le vecteur de ses entrées extérieures. Comme pour la cas binaire, le réseau est caractérisé par une fonction d'énergie qui à tout vecteur d'état  $\vec{V} = (V_1, V_2, \dots, V_N)^T \in [0, 1]^N$  associe la quantité :

$$H(\vec{V}) = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N W_{ij} \cdot V_i \cdot V_j - \sum_{i=1}^N I_i \cdot V_i \quad \text{Equation VI-8}$$

A partir de cette fonction d'énergie, Hopfield et Tank [Hopfield 85] ont introduit un terme de bruit pour construire une autre forme de fonction d'énergie, appelée fonction d'énergie perturbée :

$$\tilde{H}(\vec{V}, \lambda) = H(\vec{V}) + \sum_{i=1}^N B(V_i, \lambda) \quad \text{Equation VI-9}$$

où  $\lambda$  est un paramètre analogue à celui de la température pour un système physique. La fonction  $B(V_i, \lambda)$  est choisie de telle façon que lorsque  $\lambda$  tend vers 0, les minimums locaux des deux fonctions d'énergie  $\tilde{H}$  et  $H$  coïncident, c'est à dire que :

$$\lim_{\lambda \rightarrow 0} B(V_i, \lambda) = 0 \quad \text{Equation VI-10}$$

Le potentiel d'entrée  $U_i$  d'un neurone  $n_i$  du réseau est donné par une transformation non linéaire  $f(\cdot)$  de sa sortie  $V_i$ . Cette transformation est définie via la fonction  $B(\cdot)$  par :

$$\begin{aligned} U_i &= f(V_i, \lambda) \\ &= \tau \cdot \frac{dB(V_i, \lambda)}{dV_i} \end{aligned} \quad \text{Equation VI-11}$$

où  $\tau$  est un paramètre qui représente une constante de temps pour le neurone  $n_i$ . On suppose que cette constante est la même pour tous les neurones.

Le gradient de la fonction d'énergie perturbée s'écrit :

$$\frac{d\tilde{H}}{dV_i} = \frac{dH}{dV_i} + \frac{dB(V_i, \lambda)}{dV_i}$$

A partir des équations VI-8 et VI-11 on obtient :

$$\begin{aligned} \frac{d\tilde{H}}{dV_i} &= -\frac{1}{2} \left[ 2 \cdot W_{ii} \cdot V_i + \sum_{\substack{j=1 \\ j \neq i}}^N W_{ij} \cdot V_j + \sum_{\substack{j=1 \\ j \neq i}}^N W_{ij} \cdot V_j \right] - I_i + \frac{U_i}{\tau} \\ &= -\frac{1}{2} \left[ 2 \cdot W_{ii} \cdot V_i + \sum_{\substack{j=1 \\ j \neq i}}^N (W_{ij} + W_{ji}) \cdot V_j \right] - I_i + \frac{U_i}{\tau} \end{aligned} \quad \text{Equation VI-12}$$

Si on suppose que la matrice  $W$  des poids des connexions soit symétrique, on obtient alors :

$$\frac{d\tilde{H}}{dV_i} = -\sum_{j=1}^N W_{ij} \cdot V_j - I_i + \frac{U_i}{\tau} \quad \text{Equation VI-13}$$

A partir de l'équation VI-13, Hopfield et Tank ont défini l'équation qui régit l'évolution temporelle du neurone  $n_i$  :

$$\frac{dU_i}{dt} = -\frac{d\tilde{H}}{dV_i} = -\frac{U_i}{\tau} + \sum_{j=1}^N W_{ij} \cdot V_j + I_i \tag{Equation VI-14}$$

La sortie du neurone  $n_i$  est donnée par la transformation inverse  $g(\cdot)$  de celle de l'équation VI-11 :

$$V_i = g(U_i, \lambda) = f^{-1}(V_i, \lambda) \tag{Equation VI-15}$$

Nous rappelons que la procédure de mise à jour des états des neurones, au cours de l'évolution du réseau, est asynchrone séquentielle (Cf. § VI-3.2.2). Ainsi, la mise à jour de l'état  $V_i$  du neurone  $n_i$  est réalisée de la manière suivante (Cf. Figure VI-15) :

- A partir des sorties  $V_1(t+1), \dots, V_{i-1}(t+1), V_i(t), \dots, V_N(t)$  des neurones  $n_j, j = 1, \dots, N$ , on calcule le potentiel d'entrée  $U_i(t+1)$  du neurone  $n_i$  en utilisant l'équation VI-14.
- On calcule ensuite la nouvelle sortie  $V_i(t+1)$  du neurone  $n_i$  via la fonction  $g(\cdot)$  donnée par l'équation VI-15.

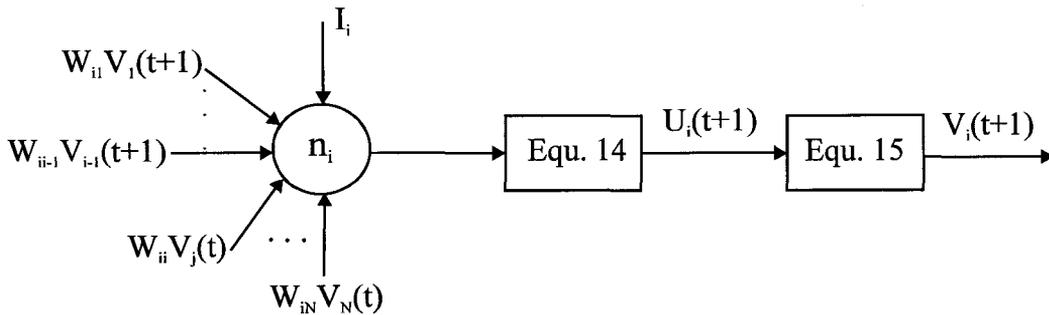


Figure VI-15. Fonctionnement d'un neurone du réseau.

Analysons maintenant l'évolution temporelle de la fonction d'énergie  $\tilde{H}$  le long de la trajectoire définie par l'équation VI-14. Dérivons la fonction d'énergie perturbée  $\tilde{H}$  par rapport à la variable temporelle  $t$  :

$$\frac{d\tilde{H}}{dt} = \sum_{i=1}^N \frac{d\tilde{H}}{dV_i} \cdot \frac{dV_i}{dt} \tag{Equation VI-16}$$

En tenant compte de l'équation VI-14, on obtient :

$$\frac{d\tilde{H}}{dt} = - \sum_{i=1}^N \frac{dU_i}{dt} \cdot \frac{dV_i}{dt} ,$$

ou encore, en utilisant l'équation VI-11 :

$$\frac{d\tilde{H}}{dt} = - \sum_{i=1}^N \left( \frac{df}{dV_i} \right) \cdot \left( \frac{dV_i}{dt} \right)^2 .$$

En conséquence, comme le terme  $\left( \frac{dV_i}{dt} \right)^2$  est toujours positif ou nul, on a :

$$\frac{d\tilde{H}}{dt} \leq 0 \text{ si } \frac{df}{dV_i} > 0 .$$

En d'autres termes, la fonction d'énergie perturbée  $\tilde{H}$ , fonction continue de la variable temporelle continue  $t$ , diminue au cours du temps vers un minimum local, si la fonction  $f(\cdot)$  est monotone croissante. Cette dernière est généralement choisie en utilisant une sigmoïde pour la fonction de sortie  $g(\cdot)$  (Cf. Figure VI-16) :

$$\begin{aligned} V_i = g(U_i, \lambda) &= \frac{I}{1 + \exp\left(-\frac{U_i}{\lambda}\right)} \\ &= \frac{I}{2} \left( 1 + \tanh\left(\frac{U_i}{2 \cdot \lambda}\right) \right) \end{aligned} \quad \text{Equation VI-17}$$

En inversant, on obtient :

$$U_i = f(V_i, \lambda) = \lambda \cdot \ln\left(\frac{V_i}{I - V_i}\right) \quad \text{Equation VI-18}$$

A partir de l'équation VI-11, l'expression de la fonction  $B(\cdot)$  s'écrit :

$$\begin{aligned} B(V_i, \lambda) &= \frac{I}{\tau} \cdot \int_{\frac{1}{2}}^{V_i} f(x, \lambda) \cdot dx \\ &= \frac{\lambda}{\tau} \cdot \int_{\frac{1}{2}}^{V_i} \ln\left(\frac{x}{I - x}\right) \cdot dx \end{aligned} \quad \text{Equation VI-19}$$

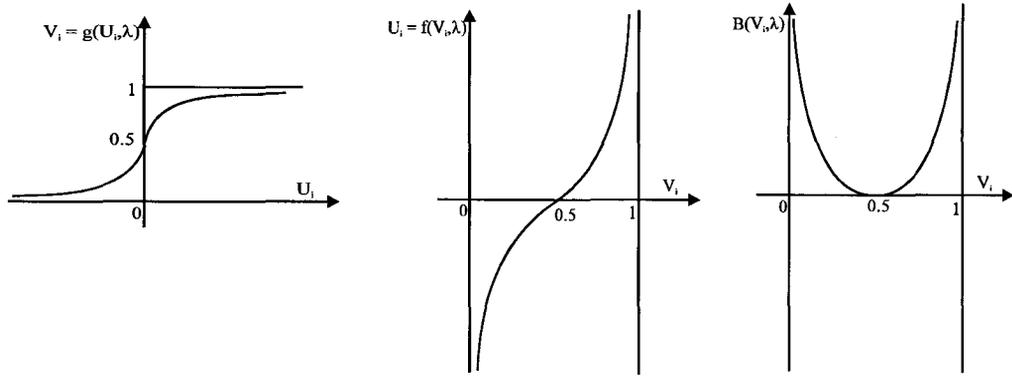


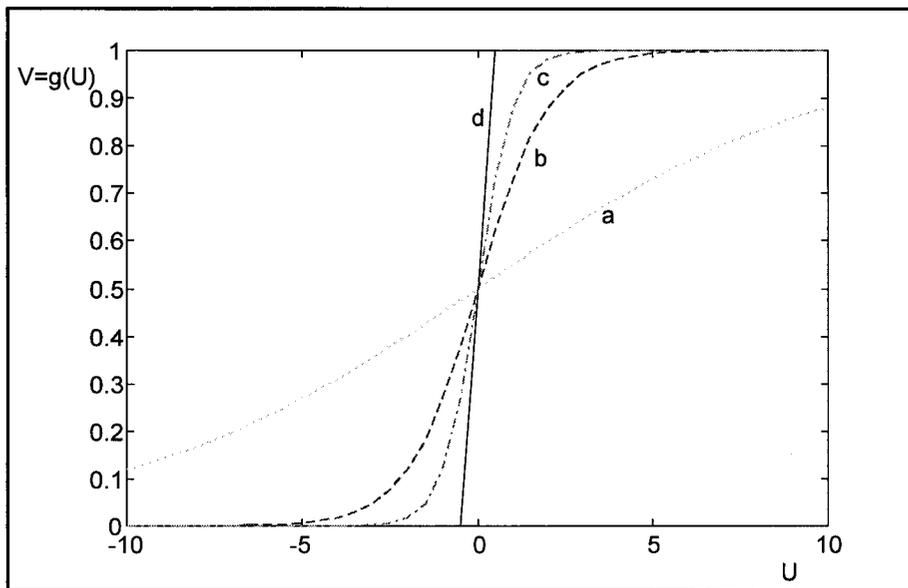
Figure VI-16. Fonctions d'entrée et de sortie d'un neurone.

Ainsi, la fonction d'énergie :

$$\tilde{H}(\vec{V}, \lambda) = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N W_{ij} \cdot V_i \cdot V_j - \sum_{i=1}^N I_i \cdot V_i + \frac{\lambda}{\tau} \sum_{i=1}^N \int_{1/2}^{V_i} \ln\left(\frac{x}{1-x}\right) \cdot dx$$

décroît au cours de l'évolution du réseau dont le comportement des neurones est régi par les équations VI-14 et VI-15. Il faut noter que le mode de fonctionnement de ces derniers est asynchrone, c'est à dire qu'un seul neurone change d'état à la fois (Cf. § VI-3.2.2). On notera qu'ici, contrairement aux réseaux binaires fonctionnant en temps discret, il n'est pas nécessaire que les termes diagonaux de la matrice  $W$  des poids des connexions soient nuls. Pour assurer la convergence du réseau, il suffit que les connexions entre les neurones soient symétriques, c'est à dire  $W_{ij} = W_{ji}$ . Cette condition est en fait utilisée pour passer de l'équation VI-12 à l'équation VI-13.

Il est important de rappeler que lorsque la température du système, représentée par le paramètre  $\lambda$ , est faible, les minima locaux des deux fonctions d'énergie  $H$  et  $\tilde{H}$  tendent à coïncider. Il faut noter aussi que lorsque  $\lambda$  tend vers 0, le modèle continue tend vers le modèle binaire (Cf. Figure VI-17). Par conséquent, les états stables du réseau tendent à se rapprocher des sommets de l'hypercube  $[0, 1]^N$ .



- (a) : Fonction de transfert pour  $\lambda = 5$   
 (b) : Fonction de transfert pour  $\lambda = 1$   
 (c) : Fonction de transfert pour  $\lambda = 0.5$   
 (d) : Fonction de transfert pour  $\lambda = 0.01$

*Figure VI-17. Fonction de transfert d'un neurone analogique en fonction de  $\lambda$ .*

### VI-6.2 Influence du terme de bruit sur l'évolution de la fonction d'énergie $H$

Nous avons vu que le fonctionnement des neurones, décrit par les équations VI-14 et VI-15, amenait le réseau, en partant d'une configuration initiale, à se stabiliser dans un minimum local de la fonction d'énergie perturbée  $\tilde{H}$  (Cf. Equation VI-9). Les minima locaux de la fonction d'énergie  $H$  (Cf. Equation VI-8) sont donc déplacés à cause du bruit introduit par la fonction  $B(\cdot)$  (Cf. Equation VI-19), et ce n'est que lorsque le paramètre  $\lambda$  tend vers 0 que les minima locaux des deux fonctions d'énergie  $H$  et  $\tilde{H}$  coïncident.

Il faut noter qu'au cours de la convergence du réseau, ce terme de bruit augmente la fonction d'énergie  $H$  (Cf. Equation VI-8) sous certaines conditions. En effet, dérivons celle-ci par rapport à la variable temporelle  $t$  :

$$\frac{dH}{dt} = \sum_{i=1}^N \frac{dV_i}{dt} \cdot \frac{dH}{dV_i} ,$$

soit en remplaçant  $\frac{dH}{dV_i}$  par  $\left( -\frac{U_i}{\tau} - \frac{dU_i}{dt} \right)$  (Cf. Equation VI-14) :

$$\begin{aligned}
\frac{dH}{dt} &= \sum_{i=1}^N \frac{dV_i}{dt} \cdot \left( -\frac{U_i}{\tau} - \frac{dU_i}{dt} \right) \\
&= -\sum_{i=1}^N \frac{dV_i}{dt} \cdot \frac{U_i}{\tau} - \sum_{i=1}^N \frac{dV_i}{dt} \cdot \frac{dU_i}{dt} \\
&= -\sum_{i=1}^N \frac{dV_i}{dt} \cdot \frac{U_i}{\tau} - \sum_{i=1}^N \left( \frac{dU_i}{dt} \cdot \frac{dV_i}{dU_i} \right) \cdot \frac{dU_i}{dt} \\
&= -\sum_{i=1}^N \frac{dV_i}{dt} \cdot \frac{U_i}{\tau} - \sum_{i=1}^N \frac{dV_i}{dU_i} \cdot \left( \frac{dU_i}{dt} \right)^2
\end{aligned}$$

Comme la fonction  $g(\cdot)$  est strictement monotone (Cf. Equation VI-17), le terme  $\frac{dV_i}{dU_i}$  est

strictement positif. Par conséquent le second terme  $-\sum_{i=1}^N \frac{dV_i}{dU_i} \cdot \left( \frac{dU_i}{dt} \right)^2$  est toujours négatif ou

nul. Par contre, le premier terme  $-\sum_{i=1}^N \frac{dV_i}{dt} \cdot \frac{U_i}{\tau}$  peut avoir n'importe quel signe. Ainsi, lorsque

les deux conditions suivantes sont vérifiées :

- $\left| \sum_{i=1}^N \frac{dV_i}{dt} \cdot \frac{U_i}{\tau} \right| > \left| \sum_{i=1}^N \frac{dV_i}{dU_i} \cdot \left( \frac{dU_i}{dt} \right)^2 \right|$
- $\frac{dV_i}{dt} \cdot \frac{U_i}{\tau} < 0 \quad \forall i = 1, \dots, N$

$\frac{dH}{dt}$  est positif, c'est à dire que la fonction d'énergie  $H$  augmente.

Afin d'éviter ce problème, certains auteurs ont suggéré d'éliminer le terme  $-\frac{U_i}{\tau}$  de l'équation VI-14 [Funabiki 93]. Ce terme correspond en fait au terme de bruit

$\frac{\lambda}{\tau} \sum_{i=1}^N \int_{1/2}^{V_i} \ln\left(\frac{x}{1-x}\right) \cdot dx$  qui tend vers zéro lorsque  $\lambda$  tend est très faible (Cf. Equations VI-9, VI-10

et VI-19). D'autre auteurs se sont contentés d'attribuer une grande valeur au paramètre  $\tau$  afin de réduire les effets indésirables de ce terme [Funabiki 93].

### VI-6.3 Conclusion

Nous avons vu comment implanter le processus de minimisation d'une forme quadratique sur un réseau analogique de Hopfield. L'évolution de celui-ci est continue dans le temps et le comportement des neurones est régi par une équation différentielle non linéaire. En comparaison avec un réseau binaire, un réseau analogique fournit des solutions meilleures car les sorties des neurones ne sont pas restreintes à des valeurs binaires durant la convergence du réseau. Pour la même raison, la convergence d'un réseau analogique est plus rapide que celle d'un réseau binaire. Néanmoins, comme les réseaux binaires, un réseau analogique tend à proposer pour solution le premier minimum local rencontré.

## VI-7 CONCLUSION

Nous avons vu, au cours de ce chapitre, que le modèle de Hopfield permet d'étudier les réseaux de neurones artificiels complètement connectés. Dans sa version la plus simple, ce modèle est basé sur l'utilisation des neurones formels de MacCulloch et Pitts. Utilisant un apprentissage basé sur la règle de Hebb, ce type de réseaux permet d'obtenir des résultats intéressants en se comportant comme une mémoire associative caractérisée par la propriété que l'on accède à son contenu et non à son adresse comme dans les calculateurs actuels. Toutefois, ce modèle présente un certain nombre d'insuffisances, en particulier la présence des états poubelles.

Nous avons vu aussi que le fonctionnement d'un réseau de Hopfield se prête, sous certaines conditions, à la minimisation automatique des formes quadratiques. Cette propriété intéressante permet de proposer une utilisation duale des réseaux de Hopfield pour résoudre des problèmes d'optimisation où l'objectif est de minimiser une fonction de coût représentée sous une forme quadratique. Néanmoins, cette méthode de minimisation n'est que locale. En effet, elle fournit une solution qui correspond au premier minimum local rencontré.

Nous avons montré ensuite comment mettre en oeuvre le principe du recuit simulé sur un réseau de Hopfield. Cette technique permet d'améliorer le processus de minimisation en augmentant la probabilité de convergence vers un minimum global. En revanche, cette approche nécessite une grande quantité de calculs et de tests, à cause de sa nature intrinsèquement stochastique.

Nous avons enfin présenté le modèle analogique de Hopfield qui est une extension du modèle binaire pour des états continus. Le fonctionnement de celui-ci est continu dans le temps

et le comportement des neurones est régi par des équations différentielles non linéaires. Par rapport à un réseau binaire, un réseau analogique fournit des solutions meilleures et converge rapidement. Néanmoins, il tend à proposer une solution qui est le premier minimum local rencontré.

Le travail que nous proposons, dans le chapitre suivant, porte sur l'utilisation du modèle analogique de Hopfield pour le problème de l'appariement stéréoscopique des images linéaires.

## **Chapitre VII**

# **Approche globale pour l'appariement stéréoscopique des images linéaires par réseau de neurones de Hopfield**

## VII-1 INTRODUCTION

La phase la plus difficile et la plus délicate en stéréovision est la mise en correspondance des indices visuels extraits des images stéréoscopiques. Nous avons passé en revue, au chapitre II, les différentes approches qui ont été proposées pour résoudre ce problème dans le cas des images stéréoscopiques matricielles. La plupart de ces méthodes donnent d'excellents résultats, mais ont un inconvénient majeur : le temps de traitement est trop long pour envisager une application temps réel, telle que la détection d'obstacles à l'avant des véhicules routiers. Cette limitation est due principalement à la grande quantité d'informations à traiter lors de l'analyse de couples d'images matricielles.

Afin de pallier cet inconvénient des systèmes stéréoscopiques classiques, nous avons opté pour un système stéréoscopique à base de caméras linéaires (Cf. Chapitre III). Dans le chapitre V, nous avons présenté la méthode développée par Burie pour appairer des images linéaires stéréoscopiques. Celle-ci donne de bons résultats, mais présente certains défauts qui peuvent, dans certains cas, conduire à des mises en correspondance erronées, sinon à ne pas trouver les appariements recherchés.

Dans le but d'améliorer les résultats de mise en correspondance, nous proposons, dans le présent chapitre, une approche basée sur l'utilisation des réseaux neuronaux en ramenant le problème de l'appariement à un problème d'optimisation. Après avoir codé le problème sous forme d'un système de variables d'état, une fonction de coût, représentant les contraintes du problème, est construite de telle sorte que ses minima correspondent à des appariements corrects. Le processus de minimisation est ensuite mis en oeuvre en utilisant un réseau de neurones de Hopfield (Cf. Chapitre VI).

## VII-2 CODAGE DU PROBLEME

Pour alléger la présentation de notre démarche, les notations définies au chapitre V, concernant les listes gauche et droite des points de contour, ne sont plus utilisées. Toutes les notations que nous utiliserons seront définies au fur et à mesure de leur apparition dans cet exposé.

Considérons un couple d'images stéréoscopiques. Soit  $G$  (resp.  $D$ ) la liste des points de contour extraits de l'image gauche (resp. droite). Soit  $N_G$  (resp.  $N_D$ ) le nombre de points de contour de l'image gauche (resp. droite).

Le principe de base de la méthodologie neuronale est, dans un premier temps, de coder le problème en termes d'états, qui sont des variables binaires. Chacun de ces états représente une possibilité de mise en correspondance entre un point de contour de l'image gauche et un point de contour de l'image droite. Nous désignons par  $E_{gd}$  la variable d'état qui représente la possibilité d'appariement d'un point de contour  $g$  de l'image gauche avec un point de contour  $d$  de l'image droite :  $E_{gd} = 1$  si  $g$  et  $d$  sont appariés,  $E_{gd} = 0$  dans le cas contraire. Ainsi, le codage du problème de la mise en correspondance pour un couple d'images stéréoscopiques nécessite un système de  $N_G \times N_D$  variables d'état. Pour permettre une meilleure compréhension, nous nous aidons d'un graphique qui représente ce que nous appelons le tableau de mise en correspondance (Cf. Figure VII-1). La liste des points de contour de l'image gauche (respectivement droite) est représentée à gauche (respectivement en bas) du tableau de mise en correspondance. Chaque case de ce dernier représente un appariement possible entre deux points de contour, l'un de l'image gauche et l'autre de l'image droite.

5	$E_{51}$	$E_{52}$	$E_{53}$	$E_{54}$	$E_{55}$	$E_{56}$	
4	$E_{41}$	$E_{42}$	$E_{43}$	$E_{44}$	$E_{45}$	$E_{46}$	
3	$E_{31}$	$E_{32}$	$E_{33}$	$E_{34}$	$E_{35}$	$E_{36}$	
2	$E_{21}$	$E_{22}$	$E_{23}$	$E_{24}$	$E_{25}$	$E_{26}$	
1	$E_{11}$	$E_{12}$	$E_{13}$	$E_{14}$	$E_{15}$	$E_{16}$	
G							
	D	1	2	3	4	5	6

Figure VII-1. Tableau de mise en correspondance.

Pour trouver une solution au problème de la mise en correspondance, certaines approches procèdent d'une manière locale, c'est à dire tentent d'apparier séquentiellement les points de contour couple par couple pour obtenir l'ensemble des appariements corrects (Cf. Chapitre V). Le principe général de ce type d'approche répond au schéma suivant :

1. Application d'un certain nombre de contraintes, dites locales, afin de réduire la combinatoire du problème. On ne garde, comme candidats à l'appariement, que les couples de primitives qui respectent ce type de contrainte.
2. Calcul d'un critère de mise en correspondance. On ne garde que les couples de primitives pour lesquels ce critère prend une valeur jugée satisfaisante. On

garde souvent plusieurs possibilités concurrentes, parmi lesquelles la suite des traitements permettra de ne conserver que la meilleure.

3. Vérification, grâce à des contraintes globales, de la compatibilité des appariements possibles avec les appariements déjà acquis. On élimine, ainsi, les hypothèses d'appariement incompatibles.

Le principale inconvénient de ce type d'approche est que la vérification de la compatibilité des appariements n'est pas globale. En effet, dès que l'on veut valider un couple de primitives, on vérifie sa compatibilité uniquement avec les couples de primitives déjà validés. De plus, si une erreur de mise en correspondance se produit, elle se propage tout au long du processus d'appariement.

Contrairement à ce genre de méthode, notre approche est basée sur une démarche de recherche globale. Le problème de la mise en correspondance est considéré comme un problème de satisfaction de contraintes. Une solution optimale au problème est celle dont l'ensemble des appariements respecte au mieux les contraintes appliquées. Pour quantifier cette optimalité, il suffit de construire une fonction de coût  $H$  qui est une application de  $\{0,1\}^{N_G \times N_D}$  dans  $R$ . Représentant les contraintes du problème, celle-ci associe à chaque possibilité d'appariement un nombre réel qui représente la qualité de la solution en question ou le degré de compatibilité des appariements vis-à-vis les contraintes imposées. Il s'agit alors de chercher la solution qui correspond à un coût minimal.

Ainsi, le problème de la mise en correspondance peut être ramené à un problème de minimisation d'une fonction de coût. Celle-ci doit être définie à partir d'un certain nombre de contraintes que doivent respecter les appariements. Dans le paragraphe suivant, nous détaillons les différentes contraintes que nous utilisons dans notre procédure de mise en correspondance.

## VII-3 LES CONTRAINTES DE MISE EN CORRESPONDANCE

### VII-3.1 Les contraintes locales

Ces contraintes sont exprimées en fonction des attributs des primitives gauches et droites, prises deux à deux. Elles sont dites locales car elles sont appliquées à chaque couple de primitives candidat à l'appariement, sans faire intervenir les autres couples. Pour notre procédure

---

\* L'ensemble des  $(N_G \times N_D)$ -uplets dont les éléments prennent les valeurs 0 ou 1.

de mise en correspondance, nous utilisons deux contraintes locales : la contrainte de position et la contrainte de signe du gradient.

### VII-3.1.1 La contrainte de position

Nous avons vu, au chapitre II, que la contrainte épipolaire impose que deux primitives homologues se trouvent nécessairement sur des droites épipolaires conjuguées. Cette contrainte, qui ne nécessite pas d'hypothèses sur la scène observée, dépend uniquement de la géométrie du capteur stéréoscopique. Dans notre cas, cette contrainte résulte de la géométrie du stéréoscope linéaire. Nous avons vu, au chapitre III, que la position d'un point objet dans la scène réelle pouvait s'exprimer en fonction de l'abscisse  $x_g$  (respectivement  $x_d$ ) de son image sur le capteur gauche (respectivement droit) (Cf. Equation III-5) :

$$\begin{cases} X_p = \frac{x_g \cdot D}{f} - \frac{E}{2} \\ X_p = \frac{x_d \cdot D}{f} + \frac{E}{2} \end{cases} \quad \text{Equation VII-1}$$

où les paramètres  $E$ ,  $f$  et  $D$  représentent respectivement l'entre axe des caméras, la distance focale des objectifs et la distance du point objet par rapport au stéréoscope linéaire.

En identifiant ces deux équations, on obtient alors la relation liant les abscisses  $x_g$  et  $x_d$  :

$$x_g - x_d = \frac{E \cdot f}{D} . \quad \text{Equation VII-2}$$

Les paramètres  $E$ ,  $f$  et  $D$  étant positifs, on obtient la contrainte suivante :

$$x_g > x_d .$$

Nous appliquons donc cette contrainte, que nous appelons contrainte de position, aux abscisses des primitives que l'on tente d'apparier. Ainsi, les primitives de tout couple ne respectant pas cette condition sont considérées comme incompatibles.

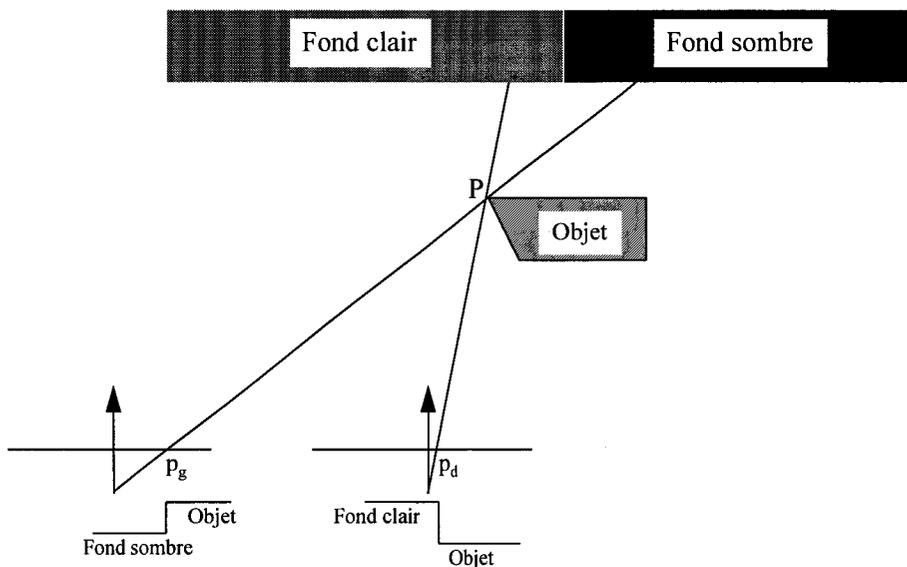
### VII-3.1.2 La contrainte de signe du gradient

Il s'agit d'une contrainte photométrique. Elle utilise l'attribut de signe du gradient que nous avons évoqué au chapitre IV (Cf. § IV-5.3) et postule que deux points de contour des images gauche et droite, engendrés par le même point objet, sont caractérisés localement par des dérivées de la fonction niveau de gris de même signe.

Dans une image linéaire, une dérivée positive correspond à un front montant, c'est à dire à une variation du niveau de gris évoluant du sombre vers le clair quand on explore l'image selon les abscisses croissantes. Par contre, une dérivée négative indique un front descendant, c'est à dire une variation du niveau de gris d'une zone claire vers une zone sombre.

La contrainte de signe du gradient stipule qu'un front montant ne peut pas être mis en correspondance avec un front descendant. Ainsi, les points de contour de tout couple ne respectant pas cette contrainte sont considérés comme incompatibles.

Il faut toutefois noter que cette affirmation peut être mise en défaut (Cf. Figure VII-2). Le point  $P$  placé à la surface de l'objet est vu par les deux caméras, mais avec des fonds différents : le point  $P$  est vu par la caméra gauche sur un fond sombre et est vu par la caméra droite sur un fond clair. De ce fait, les projections gauche et droite de ce point génèrent des gradients de signes différents. La contrainte n'est donc pas respectée, mais ce type de configuration est extrêmement rare dans un contexte routier car le fond est en général constituée par le revêtement de la chaussée qui est, en principe, homogène.



*Figure VII-2. Mise en défaut de la contrainte de signe du gradient.*

### VII-3.2 Les contraintes globales

Contrairement aux contraintes locales, les contraintes globales sont appliquées entre les couples de primitives candidates à l'appariement. Elles permettent de vérifier la compatibilité des appariements pour réduire les ambiguïtés de mise en correspondance. Notre procédure utilise

trois contraintes globales : la contrainte d'unicité, la contrainte d'ordre et la contrainte de continuité de la disparité.

### ***VII-3.2.1 La contrainte d'unicité***

La contrainte d'unicité repose sur l'un des aspects fondamentaux de la stéréovision : elle impose que toute primitive de l'image gauche a au plus un correspondant dans l'image droite, et réciproquement (Cf. Chapitre II, § II-4.3). Nous avons vu au chapitre II (Cf. § II-7.2.5) que certains alignements accidentels de primitives entraînaient une transgression de cette contrainte. En stéréovision linéaire, ce type de configuration, qui ne peut avoir lieu que de façon exceptionnelle, est évidemment possible, mais nous excluons cette possibilité pour des raisons identiques à celles mentionnées au § II-7.2.5 du chapitre II.

### ***VII-3.2.2 La contrainte d'ordre***

La contrainte d'ordre suppose que l'ordre des points de contour est le même dans chaque image linéaire. Soient deux couples  $(g_1, d_1)$  et  $(g_2, d_2)$  de points de contour homologues. La compatibilité d'appariement de ces deux couples, au sens de la contrainte d'ordre, est vérifiée si la position, dans l'image gauche, du point de contour  $g_1$  par rapport au point de contour  $g_2$  est la même que celle du point de contour  $d_1$  par rapport au point de contour  $d_2$ , dans l'image droite.

Il faut cependant noter que si l'on considère des objets ponctuels, nous pouvons nous trouver assez souvent dans la configuration représentée sur la figure VII-3, pour laquelle la contrainte d'ordre n'est pas vérifiée. Mais les objets ponctuels sont plutôt rares dans la nature. Les points de contour que nous utilisons en tant que primitives correspondent aux frontières d'objets présents dans des scènes réelles. Nous devons donc tenir compte de la taille de ces objets. La figure VII-4 illustre ce qui se produit dans une configuration identique à celle de la figure VII-3, mais en considérant la présence d'objets dont la taille ne permet pas de les réduire à des objets ponctuels. L'objet situé à l'avant plan occulte celui qui se trouve à l'arrière plan qui n'a donc plus d'image sur le capteur droit. On constate alors que le problème de l'ordre ne se pose plus. Il faut toutefois rappeler que, même en présence d'objets non ponctuels, on peut se retrouver devant des situations où la contrainte d'ordre n'est pas respectée (Cf. Figure VII-5). Dans le cadre de notre application, nous excluons ce type de situation qui ne peut avoir lieu que de façon exceptionnelle dans un contexte routier (Cf. Annexe 1).

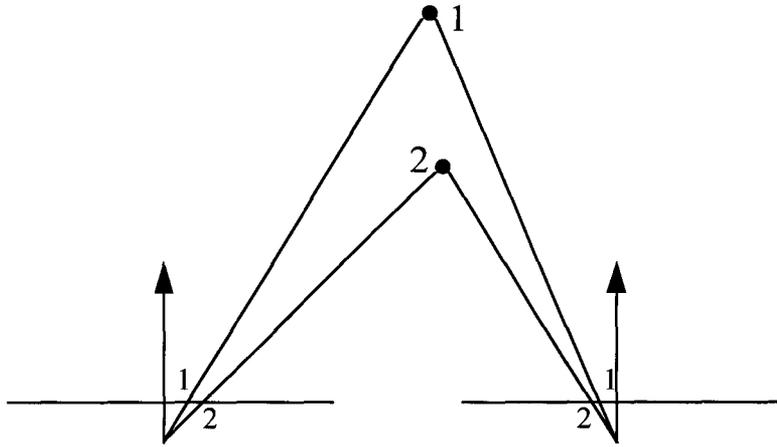


Figure VII-3. Cas d'objets ponctuels où la contrainte d'ordre n'est pas respectée.

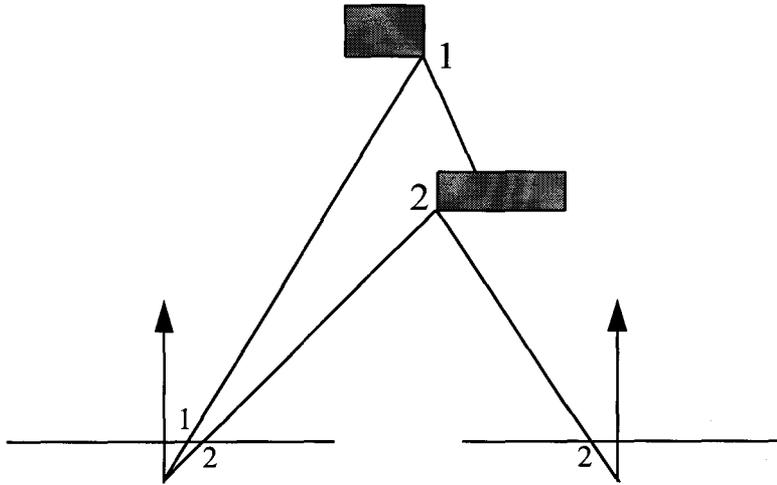


Figure VII-4. Le problème de l'ordre n'est plus posé.

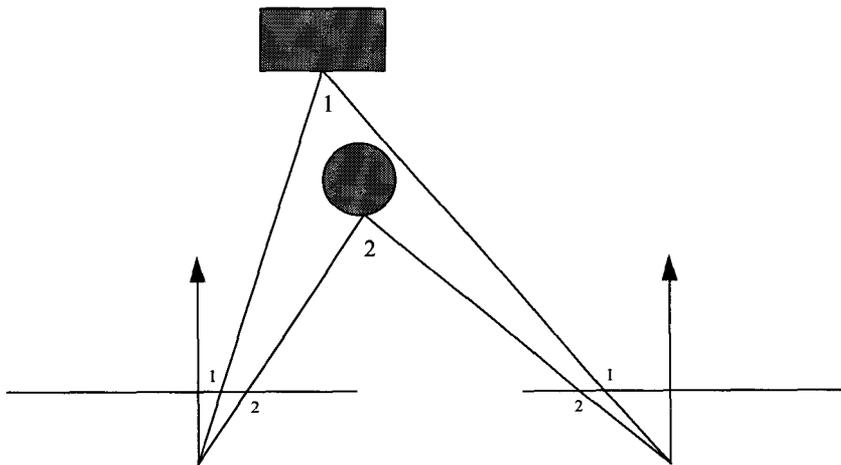


Figure VII-5. Transgression de la contrainte d'ordre en cas de présence d'objets.

### VII-3.2.3 La contrainte de continuité de la disparité

La contrainte de continuité de la disparité s'appuie sur l'hypothèse de continuité des surfaces par morceaux de telle sorte qu'un même objet ne peut présenter de variations brutales de profondeur par rapport au stéréoscope. La profondeur étant directement liée à la disparité (Cf. Chapitre III, § III-4.3), il en résulte que les disparités doivent varier continûment le long des surfaces constituant les objets. La contrainte de continuité de la disparité permet ainsi de vérifier la cohérence des appariements dans un contexte global. Dans le cadre de notre application, cette contrainte est très utile car les obstacles que nous sommes chargés de détecter présentent des formes 3D suffisamment régulières pour qu'il n'y ait pas, à leur surface, de fortes variations de profondeur.

### VII-3.3 Conclusion

Notre procédure de mise en correspondance utilise les cinq contraintes stéréoscopiques que nous venons de présenter :

1. La contrainte de position.
2. La contrainte de signe du gradient.
3. La contrainte d'unicité.
4. La contrainte d'ordre.
5. La contrainte de continuité de la disparité.

La prochaine étape consiste à construire une fonction de coût de telle façon que ses minima correspondent à des appariements corrects vis-à-vis les cinq contraintes imposées.

## VII-4 FONCTION DE COUT

Pour construire la fonction de coût du problème, chacune des contraintes présentées ci-dessus est exprimée en termes de coût.

### VII-4.1 Fonction de coût représentant la contrainte de position

Soit un couple de points de contour,  $g$  provenant de l'image gauche et  $d$  de l'image droite. Nous avons vu que l'appariement de ce couple n'est compatible que s'il satisfait la contrainte de position (Cf. § VII-3.1.1). Celle-ci impose que les abscisses des deux points de contour  $g$  et  $d$  vérifient la relation  $x_g > x_d$ , où  $x_g$  (respectivement  $x_d$ ) est la position de  $g$  (respectivement  $d$ ) dans l'image gauche (respectivement droite). L'illustration graphique de cette contrainte est représentée sur la figure VII-6.

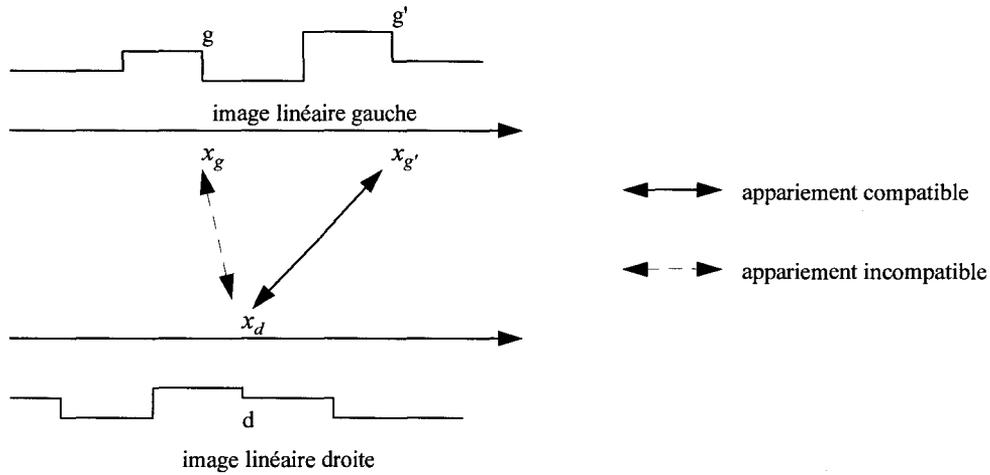


Figure VII-6. Contrainte de position.

La fonction de coût représentant cette contrainte doit donc être construite de telle façon que les appariements incompatibles soient défavorisés en termes de coût. En utilisant le codage défini au § VII-2, cette fonction peut s'exprimer sous la forme :

$$H_p = \sum_{g \in G} \sum_{d \in D} P_{gd} E_{gd} \tag{Equation VII-3}$$

où  $P_{gd}$  est une pénalité qui intervient au cas où la contrainte n'est pas respectée par le couple  $(g, d)$  :

$$P_{gd} = \begin{cases} 0 & \text{si } x_g > x_d \\ 1 & \text{sinon} \end{cases} .$$

**VII-4.2 Fonction de coût représentant la contrainte de signe du gradient**

La contrainte de signe du gradient impose que deux points de contour gauche et droit engendrés par le même point objet soient caractérisés par deux dérivées de la fonction niveau de gris de même signe. En d'autres termes, l'appariement de deux points de contour n'est possible que s'ils correspondent à des fronts de même nature (Cf. Figure VII-7).

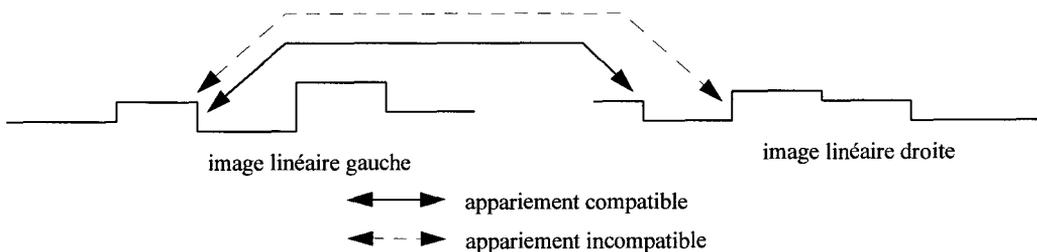


Figure VII-7. Contrainte de signe du gradient.

La formulation, en termes de coût, de cette contrainte est la même que celle de la contrainte de position. La fonction de coût doit être construite de telle sorte que les appariements ne respectant pas la contrainte de signe du gradient soient défavorisés en termes de coût. Celle-ci peut s'exprimer sous la forme :

$$H_g = \sum_{g \in G} \sum_{d \in D} S_{gd} E_{gd} \tag{Equation VII-4}$$

où  $S_{gd}$  est une pénalité qui intervient au cas où le couple  $(g, d)$  ne respecte pas la contrainte :

$$S_{gd} = \begin{cases} 0 & \text{si } g \text{ et } d \text{ correspondent à des fronts de même nature} \\ 1 & \text{sinon} \end{cases}$$

### VII-4.3 Fonction de coût représentant la contrainte d'unicité

Reprenons la figure VII-1 qui représente le tableau de mise en correspondance des points de contour de deux images stéréoscopiques. Nous avons vu que chaque variable  $E_{gd}$  code une possibilité d'appariement entre les points de contour  $g$  de l'image gauche et  $d$  de l'image droite. Si  $E_{gd} = 1$ , le couple  $(g, d)$  est apparié, sinon le couple n'est pas apparié. Une ligne (respectivement une colonne) du tableau de mise en correspondance représente l'ensemble des appariements possibles d'un point de contour de l'image gauche (respectivement droite). Pour satisfaire la contrainte d'unicité, il faut que chaque ligne mette en évidence un seul appariement, c'est à dire qu'il doit y avoir une seule case à 1 et les autres cases à 0 dans le tableau de la figure VII-1. Le même raisonnement s'applique pour les colonnes. La figure VII-8-a représente un exemple d'appariements respectant la contrainte d'unicité. Cette dernière n'est pas respectée pour l'exemple de la figure VII-8-b car la présence de deux 1 sur la ligne 4 du tableau de mise en correspondance indique que le point de contour 4 de l'image gauche est apparié aux deux points de contour 3 et 5 de l'image droite.

5	0	0	0	0	0	1
4	0	0	1	0	0	0
3	0	0	0	1	0	0
2	0	1	0	0	0	0
1	1	0	0	0	0	0
G	1	2	3	4	5	6
D	1	2	3	4	5	6

(a) La contrainte d'unicité est respectée.

5
4
3
2
1

0	0	0	0	0	1
0	0	1	0	1	0
0	0	0	1	0	0
0	1	0	0	0	0
1	0	0	0	0	0

1	2	3	4	5	6
---	---	---	---	---	---

(b) La contrainte d'unicité n'est pas respectée.

Figure VII-8. Codage de la contrainte d'unicité.

La fonction de coût représentant la contrainte d'unicité doit être construite de telle façon que les possibilités d'appariement ne respectant pas cette contrainte soient défavorisées en termes de coût. Cette fonction peut être exprimée sous la forme :

$$H_u = \sum_{g \in G} \left( 1 - \sum_{d \in D} E_{gd} \right)^2 + \sum_{d \in D} \left( 1 - \sum_{g \in G} E_{gd} \right)^2 \tag{Equation VII-5}$$

Le premier terme de cette fonction sert à défavoriser les situations où il y a plus d'un 1 sur chaque ligne du tableau de mise en correspondance. Il contribue à n'autoriser qu'un seul appariement pour chaque point de contour de l'image gauche. Le second terme sert à défavoriser les situations où il y a plus d'un 1 sur chaque colonne. Il contribue à n'autoriser qu'un seul appariement pour chaque point de contour de l'image droite.

**VII-4.4 Fonction de coût représentant la contrainte d'ordre**

Nous avons vu que la contrainte d'ordre est une contrainte globale. Elle permet de vérifier la compatibilité des appariements en supposant que l'ordre des points de contour est le même dans chaque image. Il faut toutefois rappeler que cette contrainte peut être transgressée dans certains cas que nous excluons dans le cadre de notre application (Cf. § VII-3.2.2).

Considérons deux couples  $(g,d)$  et  $(g',d')$  de points de contour homologues. La compatibilité des appariements de ces deux couples n'est vérifiée que si l'ordre des points de contour  $g$  et  $g'$  dans l'image gauche est le même que celui des points de contour  $d$  et  $d'$  dans l'image droite. En d'autres termes, les deux couples  $(g,d)$  et  $(g',d')$  vérifient la contrainte d'ordre si la position, dans l'image gauche, du point de contour  $g$  par rapport au point de contour  $g'$  est la

même que celle du point de contour  $d$  par rapport au point de contour  $d'$  dans l'image droite (Cf. Figure VII-9).

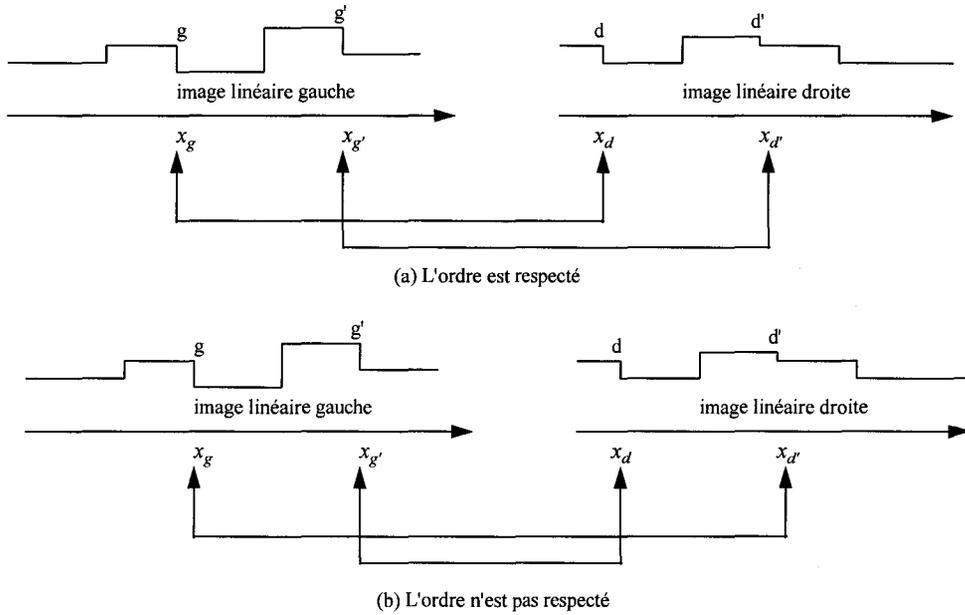


Figure VII-9. La contrainte d'ordre.

La figure VII-10-a montre un exemple de tableau de mise en correspondance où la contrainte d'ordre est respectée. En suivant l'ordre croissant des points de contour dans chaque image, les appariements sont disposés de telle sorte qu'ils constituent un chemin ascendant. Un exemple de possibilité d'appariement où la contrainte d'ordre est mise en défaut est illustré sur la figure VII-10-b. Les deux appariements  $(4,3)$  et  $(3,4)$  sont incompatibles.

5	G	0	0	0	0	0	1
4	D	0	0	0	0	1	0
3		0	0	0	1	0	0
2		0	1	0	0	0	0
1		1	0	0	0	0	0
		1	2	3	4	5	6

(a) La contrainte d'ordre est respectée.

5
4
3
2
1

0	0	0	0	0	1
0	0	1	0	0	0
0	0	0	1	0	0
0	1	0	0	0	0
1	0	0	0	0	0

1	2	3	4	5	6
---	---	---	---	---	---

(b) La contrainte d'ordre n'est pas respectée.

Figure VII-10. Codage de la contrainte d'ordre.

Pour construire une fonction de coût représentant la contrainte d'ordre, il faut que les possibilités de mise en correspondance mettant en défaut cette contrainte soient défavorisées en termes de coût. Cette fonction peut s'écrire sous la forme :

$$H_o = \sum_{g \in G} \sum_{d \in D} \sum_{\substack{g' \in G \\ g' \neq g}} \sum_{\substack{d' \in D \\ d' \neq d}} O_{gdg'd'} E_{gd} E_{g'd'} \tag{Equation VII-6}$$

Le coefficient  $O_{gdg'd'}$ , qui indique si la contrainte d'ordre est respectée pour les deux appariements  $(g,d)$  et  $(g',d')$ , est calculé comme suit :

$$O_{gdg'd'} = |s(x_g - x_{g'}) - s(x_d - x_{d'})|$$

$x_g, x_{g'}, x_d$  et  $x_{d'}$  sont respectivement les positions des points de contour  $g, g', d$  et  $d'$  dans les images linéaires stéréoscopiques.  $s(x)$  est la fonction signe telle que :

$$s(x) = \begin{cases} 1 & \text{si } x > 0 \\ 0 & \text{sinon} \end{cases}$$

**VII-4.5 Fonction de coût représentant la contrainte de continuité**

La contrainte de continuité stipule que les appariements de points de contour voisins doivent donner des disparités voisines. Le terme de coût représentant cette contrainte peut être exprimé sous la forme :

$$H_c = - \sum_{g \in G} \sum_{d \in D} \sum_{\substack{g' \in G \\ g' \neq g}} \sum_{\substack{d' \in D \\ d' \neq d}} C_{gdg'd'} E_{gd} E_{g'd'} \tag{Equation VII-7}$$

Le coefficient  $C_{gdg'd'}$  représente une mesure de compatibilité entre les deux paires de points de contour  $(g,d)$  et  $(g',d')$ . Il est exprimé en utilisant la fonction non linéaire suivante [Nasrabadi 92b] :

$$C_{gdg'd'} = C(X) = \frac{2}{1 + e^{(X-\theta)}} - 1 \quad \text{Equation VII-8}$$

où  $X$  est la valeur absolue de la différence des disparités entre les deux paires  $(g,d)$  et  $(g',d')$  :

$$X = |(x_g - x_d) - (x_{g'} - x_{d'})|.$$

Le paramètre  $\theta$ , qui contrôle la position où la fonction  $C(X)$  s'annule, permet une certaine tolérance afin de prendre en compte les effets du bruit et les distorsions (Cf. Figure VII-11). Il est ajusté de telle sorte qu'une compatibilité maximale, c'est à dire proche de 1, soit obtenue pour les bons appariements qui correspondent à des valeurs de  $X$  voisines de zéro, et qu'une compatibilité minimale, c'est à dire proche de -1, soit obtenue pour les mauvais appariements qui correspondent à des valeurs élevées de  $X$ .

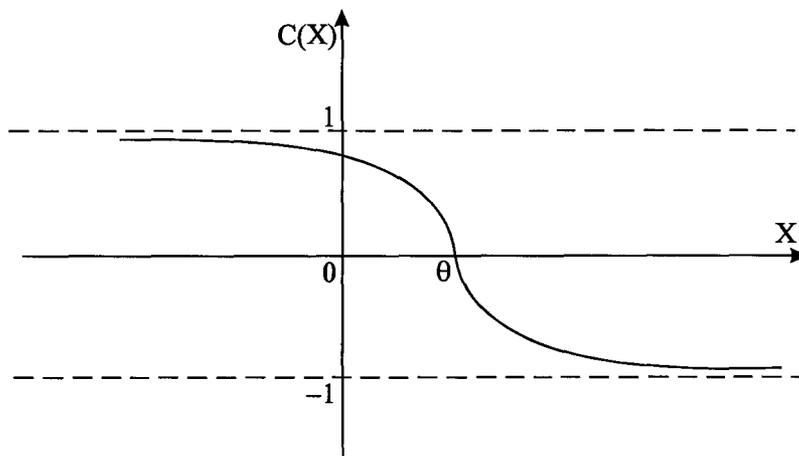


Figure VII-11. Graphe de la fonction non linéaire  $C(X)$ .

#### VII-4.6 Fonction de coût total

En combinant les cinq termes de coût (Cf. Equations VII-3 à VII-7), la fonction de coût total  $H_p$  utilisée pour résoudre le problème de la mise en correspondance prend la forme suivante :

$$\begin{aligned}
H_p = & K_{position} \sum_{g \in G} \sum_{d \in D} P_{gd} E_{gd} + K_{signe} \sum_{g \in G} \sum_{d \in D} S_{gd} E_{gd} \\
& + K_{unicité} \sum_{g \in G} \left( 1 - \sum_{d \in D} E_{gd} \right)^2 + K_{unicité} \sum_{d \in D} \left( 1 - \sum_{g \in G} E_{gd} \right)^2 \\
& + K_{ordre} \sum_{g \in G} \sum_{d \in D} \sum_{\substack{g' \in G \\ g' \neq g}} \sum_{\substack{d' \in D \\ d' \neq d}} O_{gdg'd'} E_{gd} E_{g'd'} \\
& - K_{continuité} \sum_{g \in G} \sum_{d \in D} \sum_{\substack{g' \in G \\ g' \neq g}} \sum_{\substack{d' \in D \\ d' \neq d}} C_{gdg'd'} E_{gd} E_{g'd'}
\end{aligned}$$

Equation VII-9

où  $K_{position}$ ,  $K_{signe}$ ,  $K_{unicité}$ ,  $K_{ordre}$  et  $K_{continuité}$  sont des coefficients positifs qui permettent de régler les pondérations des différents termes de la fonction de coût.

Le problème de la mise en correspondance peut être vu maintenant comme un problème de satisfaction de contraintes, caractérisé par la minimisation de la fonction de coût  $H_p$ . Dans le prochain paragraphe, nous présentons le processus neuronal que nous avons développé pour minimiser cette fonction de coût.

## VII-5 MINIMISATION DE LA FONCTION DE COUT

Nous avons montré, au paragraphe précédent, comment ramener le problème de la mise en correspondance à un problème d'optimisation. Une fonction de coût, représentant toutes les contraintes stéréoscopiques du problème, est construite de telle sorte que les bons appariements correspondent à un minimum de cette fonction.

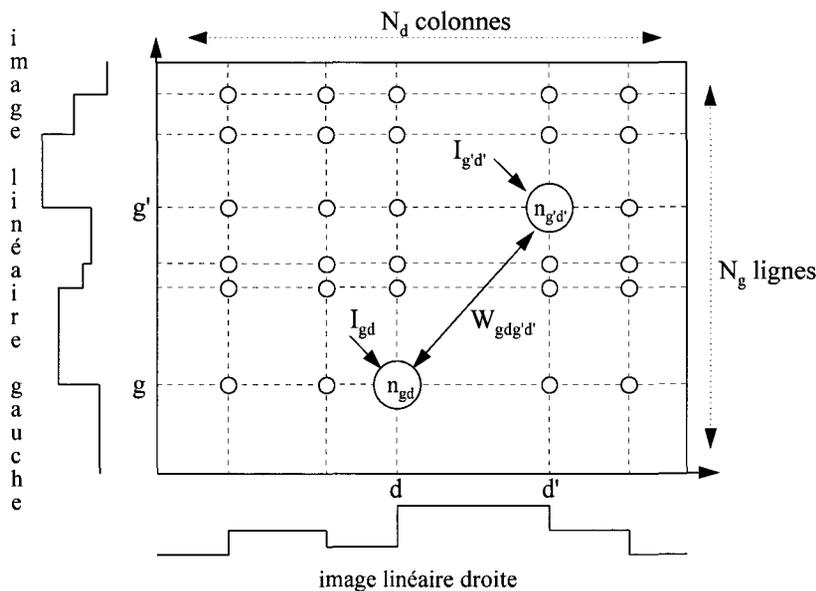
Pour minimiser la fonction de coût (Cf. Equation VII-9), nous avons fait appel aux réseaux de Hopfield qui sont très bien adaptés pour résoudre ce genre de problème d'optimisation. Comme on l'a déjà vu au chapitre VI, le principe général de cette approche neuronale peut se décomposer en deux étapes successives :

1. Construction du réseau. Il s'agit d'abord de donner une formulation neuronale au problème pour qu'il puisse être caractérisé par un réseau de Hopfield. Il faut ensuite déterminer les paramètres du réseau, c'est à dire les poids des connexions entre les neurones et leurs entrées externes, en identifiant la fonction de coût du problème à la fonction d'énergie du réseau.
2. Minimisation de la fonction d'énergie. Cette deuxième étape consiste à faire évoluer le réseau jusqu'à ce qu'il atteigne un état stable. Cet état, qui correspond

à un minimum de la fonction d'énergie du réseau, et donc de la fonction de coût du problème, est ensuite analysé afin d'obtenir le résultat de la mise en correspondance.

**VII-5.1 Construction du réseau**

En se basant sur la formulation exposée au paragraphe VII-2, la représentation neuronale du problème de mise en correspondance via un réseau de Hopfield est immédiate. Chaque neurone du réseau représente une possibilité d'appariement entre un point de contour de l'image gauche et un point de contour de l'image droite. Ainsi, pour caractériser le problème de mise en correspondance pour un couple d'images stéréoscopiques, il faut disposer d'un réseau de Hopfield organisé en une matrice de  $N_G \times N_D$  neurones, où  $N_G$  et  $N_D$  sont respectivement le nombre des points de contour de l'image gauche et de l'image droite (Cf. Figure VII-12).



**Figure VII-12.** Caractérisation du problème de la mise en correspondance via un réseau de Hopfield.

Plus explicitement, un neurone  $n_{gd}$  du réseau représente une possibilité de mise en correspondance entre le point de contour  $g$  de l'image gauche et le point de contour  $d$  de l'image droite. En faisant une analogie avec le codage du problème en termes de variables d'état (Cf. § VII-2), chaque neurone du réseau correspond à une case du tableau de mise en correspondance. De ce fait, la sortie, ou état, du neurone  $n_{gd}$  est représentée par la variable d'état  $E_{gd}$  qui indique si un appariement est possible ou non. Si  $E_{gd} = 1$  alors le couple  $(g,d)$  est apparié, si  $E_{gd} = 0$ , l'appariement est rejeté.

Pour compléter la construction du réseau, il faut calculer la matrice  $W$  des poids des connexions  $W_{gdg'd'}$ ,  $(g,d) \in G \times D$  et  $(g',d') \in G \times D$ , et le vecteur  $\vec{I}$  des entrées externes  $I_{gd}$ ,  $(g,d) \in G \times D$ . Ces paramètres sont complètement déterminés à partir de la fonction de coût du problème (Cf. Equation VII-9). Pour cela, il suffit de mettre cette dernière sous la forme de la fonction d'énergie du réseau (Cf. Annexe 2) :

$$H_r = -\frac{1}{2} \sum_{g \in G} \sum_{d \in D} \sum_{g' \in G} \sum_{d' \in D} W_{gdg'd'} E_{gd} E_{g'd'} - \sum_{g \in G} \sum_{d \in D} I_{gd} E_{gd} \quad \text{Equation VII-10}$$

En identifiant les deux expressions des équations VII-9 et VII-10, on obtient :

$$\begin{aligned} W_{gdg'd'} = & -K_{unicité} [\delta_{gg'} (1 - \delta_{dd'}) + \delta_{dd'} (1 - \delta_{gg'})] \\ & - K_{ordre} \cdot O_{gdg'd'} (1 - \delta_{gg'}) (1 - \delta_{dd'}) \\ & + K_{continuité} \cdot C_{gdg'd'} (1 - \delta_{gg'}) (1 - \delta_{dd'}) \end{aligned} \quad \text{Equation VII-11}$$

$$I_{gd} = 2K_{unicité} - \frac{K_{position} \cdot P_{gd} + K_{signe} \cdot S_{gd}}{2}$$

où  $\delta_{ij}$  est l'indice de Kroneker tel que :

$$\delta_{ij} = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{sinon} \end{cases}.$$

Notons que les poids des connexions du réseau sont symétriques. Cette propriété est très importante car elle assure la convergence du réseau (Cf. Chapitre VI, § VI-6.1). On note aussi que les éléments diagonaux de la matrice des poids des connexions sont nuls.

Le réseau de Hopfield ainsi défini caractérise complètement le problème de la mise en correspondance et permet de donner directement le résultat des appariements sous forme d'une matrice des états des neurones. Celle-ci est obtenue en faisant évoluer le réseau jusqu'à ce qu'il se stabilise dans un état qui correspond en fait à un minimum de sa fonction d'énergie (Cf. Chapitre VI). Une description détaillée du fonctionnement du réseau est présentée dans le prochain paragraphe.

## VII-5.2 Fonctionnement du réseau

Nous avons vu, au chapitre VI, qu'il existe trois modes de fonctionnement pour un réseau de Hopfield. Ceux-ci sont basés sur trois modèles distincts :

1. Le modèle binaire
2. Le modèle continu
3. Le modèle à recuit simulé

Nous avons vu aussi que chacun de ces modes de fonctionnement induit un processus de minimisation de la fonction d'énergie du réseau. En partant d'un état initial, le réseau évolue jusqu'à ce qu'il atteigne un état stable qui correspond à un minimum de sa fonction d'énergie.

Nous avons montré que la convergence des deux premiers modèles est locale, c'est à dire que le réseau se stabilise au premier minimum local rencontré. D'autre part, le processus de minimisation associé au modèle continu est plus rapide que celui induit par le modèle binaire. Par contre, le modèle à recuit simulé permet une convergence globale, mais plus lente, à cause de sa nature intrinsèquement stochastique.

Dans le cadre de notre application, le modèle à recuit simulé ne peut être utilisé car il implique une très grande quantité de calculs et donc un temps de traitement trop long. Nous avons donc opté pour le mode de fonctionnement basé sur le modèle continu. Nous rappelons que, dans ce modèle, les états des neurones sont autorisés à prendre n'importe quelle valeur dans l'intervalle  $[0, 1]$ .

Nous avons vu au § VII-5.1 qu'un neurone représente une possibilité de mise en correspondance entre deux points de contour et que son état binaire indique s'il y a appariement ou pas. Avec l'utilisation du modèle continu, l'état d'un neurone n'a plus la même interprétation. Celle-ci devient plus générale : l'état d'un neurone continu représente la probabilité de vraisemblance de l'appariement entre deux points de contour.

### VII-5.2.1 Mise à jour des états des neurones du réseau

Le fonctionnement du réseau consiste à faire évoluer les états des neurones jusqu'à ce qu'il atteigne un état stable. Le réseau est dit stable lorsqu'aucun de ses neurones ne peut changer d'état. L'évolution temporelle d'un neurone  $n_{gd}$  du réseau est régie par l'équation différentielle suivante (Cf. Chapitre VI, § VI-6.1) :

$$\frac{dU_{gd}}{dt} = l(U_{gd}) = -\frac{U_{gd}}{\tau} + \sum_{g' \in G} \sum_{d' \in D} W_{gdg'd'} E_{g'd'} + I_{gd} \quad \text{Equation VII-12}$$

où  $U_{gd}$  est le potentiel d'entrée du neurone  $n_{gd}$ . Le paramètre  $\tau$ , fixé à 1, représente une constante de temps qui est ici identique pour tous les neurones.

En partant d'un état initial, le réseau effectue un certain nombre d'itérations, ou cycles de mise à jour, pour atteindre un état stable. A chaque itération, tous les neurones sont consultés selon un schéma asynchrone séquentiel (Cf. Chapitre VI, § VI-3.2.2). Ainsi, dès que l'état d'un neurone est actualisé, il est pris en compte pour mettre à jour les états des autres neurones.

Supposons que la mise à jour, à l'instant  $t$ , des états des neurones soit effectuée. Nous allons décrire la procédure de mise à jour de l'état du neurone  $n_{gd}$  à l'instant  $t+1$ . Le même raisonnement est appliqué aux autres neurones du réseau. Cette procédure est réalisée en deux étapes successives (Cf. Figure VII-13). La première étape consiste à calculer le nouveau potentiel d'entrée  $U_{gd}(t+1)$  à partir de l'équation différentielle qui régit le fonctionnement du neurone  $n_{gd}$  (Cf. Equation VII-12). Pour cela, nous avons fait appel à la méthode numérique de Runge-Kutta qui permet de donner une solution approchée à partir d'une condition initiale [Piskounov 80]. Il s'agit, dans notre cas, de calculer le nouveau potentiel d'entrée  $U_{gd}(t+1)$  en partant de l'ancien potentiel d'entrée  $U_{gd}(t)$ . On obtient alors :

$$U_{gd}(t+1) = U_{gd}(t) + q \quad \text{Equation VII-13}$$

où  $q$  est un incrément associé à  $U_{gd}$  :

$$q = (q_1 + 2q_2 + 2q_3 + q_4) / 6$$

avec :

$$q_1 = h \cdot l(U_{gd}(t))$$

$$q_2 = h \cdot l(U_{gd}(t) + q_1/2)$$

$$q_3 = h \cdot l(U_{gd}(t) + q_2/2)$$

$$q_4 = h \cdot l(U_{gd}(t) + q_3)$$

où  $h$  représente le pas de discrétisation, associé à la variable temporelle  $t$ , utilisé pour résoudre l'équation différentielle VII-12.

La deuxième étape consiste à calculer le nouvel état  $E_{gd}(t+1)$  du neurone  $n_{gd}$  en utilisant une fonction de transfert de type sigmoïde (Cf. Chapitre VI, § VI-6.1) :

$$E_{gd}(t+1) = g(U_{gd}(t+1)) = \frac{1}{2} \left( 1 + \tanh \left( \frac{U_{gd}(t+1)}{\lambda} \right) \right) \quad \text{Equation VII-14}$$

où  $\lambda$  est un paramètre qui est analogue à la température pour un système physique (Cf. Chapitre VI, § VI-6.1).

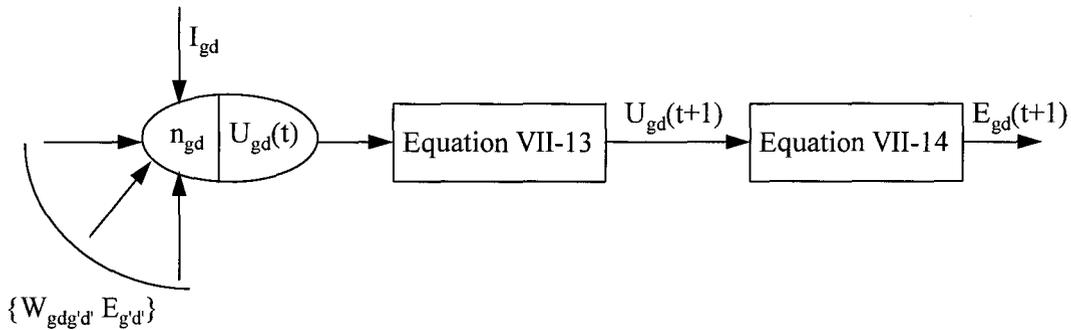


Figure VII-13. Procédure de mise à jour de l'état d'un neurone.

### VII-5.2.2 Initialisation du réseau

Il convient de noter que le processus de minimisation induit par le fonctionnement du réseau ne réalise qu'une descente dans un puits d'énergie, c'est à dire une recherche d'un minimum local, déterminé par une configuration initiale convenable. Ce processus ne comporte pas la caractéristique essentielle du recuit simulé qui est de visiter plusieurs puits d'énergie avant de se stabiliser dans l'un d'eux (Cf. Chapitre VI, § VI-5), mais présente l'avantage, si le système n'est pas trop loin de la solution lors de l'initialisation, de converger beaucoup plus rapidement vers une solution acceptable.

Dans le cas de notre réseau, nous avons suivi une stratégie d'initialisation très simple. Celle-ci consiste à affecter une probabilité de vraisemblance égale à 0.5 à tous les appariements. Ainsi, l'état initial du réseau est tel que  $E_{gd}(0) = 0.5$ , quelque soit le neurone  $n_{gd}$ . Pour calculer les potentiels d'entrées initiaux des neurones, nous appliquons la transformation inverse de la fonction sigmoïde  $g$  de l'équation VII-14 :

$$\begin{aligned} U_{gd}(0) &= \lambda \cdot \ln \left( \frac{E_{gd}(0)}{1 - E_{gd}(0)} \right) \\ &= 0 \end{aligned} \quad \text{Equation VII-15}$$

A partir de cet état initial du réseau, les états des neurones qui correspondent à des appariements corrects tendent vers 1 au cours de leur évolution. De même, les états des neurones correspondant à des faux appariements tendent vers 0 au cours de leur évolution. Les résultats de mise en correspondance obtenus avec cette stratégie d'initialisation sont très satisfaisants (Cf. § VII-6.2).

### VII-5.2.3 Stabilité du réseau

Nous rappelons que la stabilité du réseau est assurée du fait que sa matrice des poids des connexions est symétrique (Cf. Chapitre VI, § VI-6.1). Le réseau est dans un état stable lorsque les états de ses neurones n'évoluent plus, c'est à dire lorsque  $E_{gd}(t+1) = E_{gd}(t)$ , quelque soit le neurone  $n_{gd}$ .

### VII-5.3 Analyse de l'état final du réseau

Le résultat de la mise en correspondance est donné par l'état final du réseau qui correspond à un minimum de sa fonction d'énergie. Nous avons vu que l'état de chaque neurone du réseau représente une probabilité de vraisemblance de l'appariement entre deux points de contour gauche et droit. Selon cette logique, l'extraction des paires de points de contour appariés consiste donc à chercher les neurones qui présentent des réponses maximales dans chaque ligne et dans chaque colonne du réseau. Nous rappelons qu'une ligne du réseau est constituée de l'ensemble des neurones correspondants aux appariements possibles d'un point de contour de l'image gauche (Cf. Figure VII-12). De même, une colonne du réseau est constituée de l'ensemble des neurones correspondants aux appariements possibles d'un point de contour de l'image droite.

La recherche des neurones à réponse maximale est basée sur le principe suivant. Dans une première étape, on cherche, au sein du réseau, le neurone présentant la réponse maximale. La deuxième étape consiste à éliminer les neurones qui se situent dans la ligne et dans la colonne contenant le neurone sélectionné. On réitère ces deux étapes jusqu'à ce que tous les neurones du réseau aient été visités.

### VII-5.4 Procédure de mise en correspondance

A la lumière de tout ce qui a été dit, la procédure de mise en correspondance peut être décomposée en trois étapes successives :

- La première étape consiste à construire un réseau de Hopfield à partir de la fonction de coût du problème. Il s'agit de déterminer la matrice  $W$  des poids des connexions et le vecteur  $\vec{I}$  d'entrées externes (Cf. § VII-5.1).
- La deuxième étape consiste à faire évoluer le réseau, à partir d'un état initial, vers un état stable (Cf. § VII-5.2). Celui-ci est atteint lorsqu'aucun des neurones du réseau ne peut changer d'état.
- La dernière étape consiste à déterminer les paires de points de contour appariés en analysant l'état final du réseau (Cf. § VII-5.3).

Le schéma général de la procédure de mise en correspondance est illustré par le diagramme de la figure VII-14.

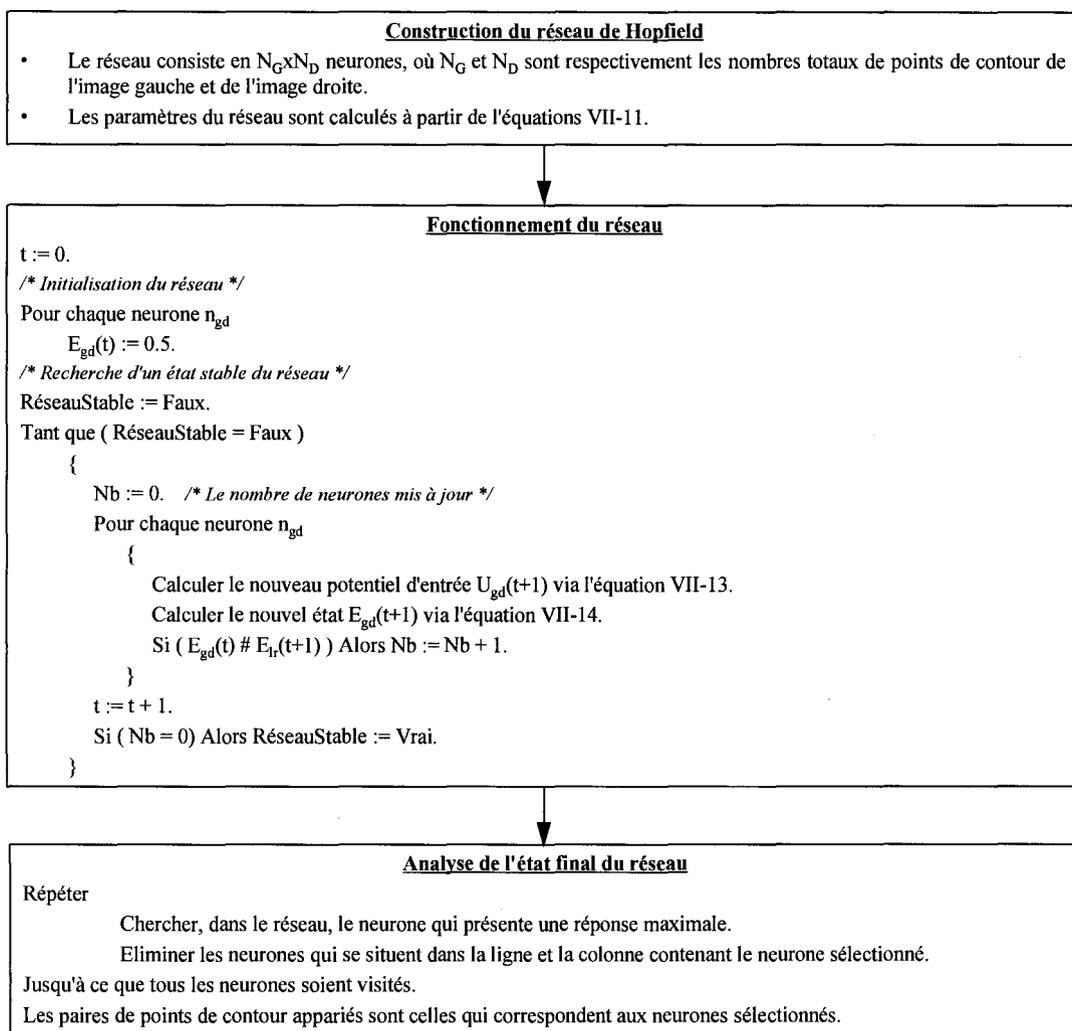


Figure VII-14. Algorithme de mise en correspondance.

### VII-5.5 Conclusion

Nous avons développé une procédure de mise en correspondance basée sur une approche connexioniste. Dans un premier temps, le problème est exprimé en termes de minimisation d'une fonction de coût. Un réseau de neurones de Hopfield est ensuite construit pour assurer le processus de minimisation. Le fonctionnement du réseau consiste à faire évoluer celui-ci, à partir d'un état initial, vers un état stable. Cet état final, qui correspond à un minimum de la fonction de coût, est enfin analysé pour en extraire les résultats de la mise en correspondance.

## VII-6 RESULTATS DE LA MISE EN CORRESPONDENCE

### VII-6.1 Choix des paramètres

Le choix des paramètres est un point important de notre méthode. Comme nous pouvons le constater, celle-ci fait intervenir 4 groupes de paramètres. Le premier groupe est constitué du seul paramètre  $\theta$  qui intervient dans l'équation VII-8. Comme nous l'avons déjà signalé, ce paramètre permet une certaine tolérance afin de prendre en compte les effets du bruit et les distorsions lors de l'évaluation de la contrainte de continuité de la disparité.

Le deuxième groupe de paramètres est constitué des coefficients de pondération des termes de la fonction de coût (Cf. Equation VII-9) :

- $K_{position}$  qui représente le coefficient de pondération du terme correspondant à la contrainte de position.
- $K_{signe}$  qui représente le coefficient de pondération du terme correspondant à la contrainte de signe du gradient.
- $K_{unicité}$  qui représente le coefficient de pondération des termes correspondant à la contrainte d'unicité.
- $K_{ordre}$  qui représente le coefficient de pondération du terme correspondant à la contrainte d'ordre.
- $K_{continuité}$  qui représente le coefficient de pondération du terme correspondant à la contrainte de continuité de la disparité.

Le troisième groupe est composé des deux paramètres  $\lambda$  et  $\tau$  qui sont associés au réseau. Comme nous l'avons mentionné au chapitre VI (Cf. § VI-6.1), le paramètre  $\lambda$  est analogue à celui de la température pour les systèmes physiques. Nous avons vu, au chapitre VI, comment ce

paramètre influe sur le comportement des neurones du réseau (Cf. Figure VI-17). Le deuxième paramètre  $\tau$  représente une constante de temps (Cf. Chapitre VI, § VI-6.1).

Le quatrième groupe de paramètres est constitué de l'incrément  $h$  associé à la variable temporelle  $t$  lors de la résolution numérique de l'équation différentielle qui régit l'évolution d'un neurone du réseau (Cf. Equation VII-12).

Il faut noter que les paramètres de chaque groupe sont ajustés indépendamment de ceux des autres groupes. Cette indépendance dans le réglage nous paraît logique puisque les phases où interviennent ces groupes de paramètres sont elles-mêmes indépendantes les unes des autres.

#### ***VII-6.1.1 Choix du paramètre $\theta$***

Nous rappelons que le paramètre  $\theta$  est utilisé dans l'équation VII-8 qui permet de calculer le coefficient de compatibilité de deux appariements. Il est ajusté de telle sorte qu'une compatibilité maximale, c'est à dire proche de 1, soit obtenue pour les bons appariements qui correspondent à des valeurs de  $X$  voisines de zéro, et qu'une compatibilité minimale, c'est à dire proche de -1, soit obtenue pour les mauvais appariements qui correspondent à des valeurs élevées de  $X$ . Les tests expérimentaux ont montré que  $\theta = 20$  est une valeur satisfaisante pour ce paramètre afin de prendre en compte les effets du bruit et les distorsions.

#### ***VII-6.1.2 Choix des coefficients intervenant dans la fonction de coût***

La stratégie que nous avons adoptée pour choisir les valeurs des coefficients de pondération des différents termes de la fonction de coût est basée sur l'importance que nous avons donnée à chacune des contraintes utilisées. Il est évident que les contributions de toutes les contraintes au problème de l'appariement n'ont pas obligatoirement la même importance.

Nous considérons que la contrainte d'unicité est celle qui doit avoir la plus forte contribution par rapport aux autres contraintes car sans cette contrainte, qui est d'ailleurs toujours imposée en stéréovision, la reconstruction 3D est impossible. En effet un point objet doit correspondre à deux points image homologues. De plus, cette contrainte est plus stable que les autres dans le sens où elle n'est pratiquement jamais transgressée, contrairement aux autres contraintes qui sont fondées sur des hypothèses sur la scène observée pour vérifier la compatibilité des appariements ou pour éliminer les couples de primitives incompatibles.

D'autre part, nous avons choisi de forcer la contribution des contraintes de position et de signe du gradient par rapport aux contraintes d'ordre et de continuité de la disparité. Ce choix

nous semble tout à fait normal car les contraintes de position et de signe du gradient sont locales alors que les contraintes d'ordre et de continuité de la disparité sont globales : une contrainte globale concerne l'ensemble des couples de primitives alors qu'une contrainte locale ne tient compte que du couple de primitives considéré. De plus, une contrainte globale n'est utilisée que pour assurer la compatibilité des appariements alors qu'une contrainte locale est appliquée pour éliminer les couples de primitives incompatibles.

En tenant compte de ces remarques, les résultats de notre procédure de mise en correspondance, que nous présentons dans le § VII-6.2, sont obtenus en utilisant pour les différents coefficients les valeurs suivantes :

- $K_{unicité} = 5$ ,
- $K_{position} = K_{signe} = 2$ ,
- $K_{ordre} = K_{continuité} = 1$ .

Il est important de noter que si le choix de ces coefficients mérite attention, il est essentiel aussi de regarder s'il est préférable de définir leur valeurs en fonction des images à traiter, ou bien de les fixer à des valeurs constantes quelque soit le couple d'images stéréoscopiques. Notre méthode ne prend pas en compte cette remarque, car les conditions d'application des contraintes utilisées sont inchangées d'un couple d'images à l'autre. Les mêmes coefficients sont donc utilisés pour traiter tous les couples d'images stéréoscopiques.

### VII-6.1.3 Choix des paramètres $\lambda$ et $\tau$

Nous avons vu que deux paramètres interviennent dans le fonctionnement du réseau. Comme nous l'avons montré au chapitre VI (Cf. § VI-6.1), la valeur du paramètre  $\lambda$  doit être choisie proche de 0 afin que les minima de la fonction de coût et ceux de la fonction d'énergie du réseau coïncident. Nous avons vu aussi, au chapitre VI, comment ce paramètre influe sur le comportement des neurones du réseau (Cf. Figure VI-17). Pour une valeur de  $\lambda$  éloignée de 0, les sorties des neurones n'atteignent pas les valeurs 0 et 1. Au cours de nos essais, nous avons constaté que, lorsque  $\lambda$  est loin de 0, le réseau converge vers des états tels que les sorties des neurones se situent aux alentours de 0.5, ce qui ne permet pas d'effectuer les mises en correspondance (Cf. § VII-5.3). Après un certain nombre de tests, nous avons retenu pour ce paramètre la valeur 0.01. Avec cette valeur, le réseau réussit toujours à converger vers des états tels que les réponses des neurones sont proches des valeurs 0 et 1.

Concernant le paramètre  $\tau$ , qui représente une constante de temps, nous nous sommes contenté du même choix que celui proposé par certains auteurs qui utilisent des constantes de temps égales à l'unité [Hopfield 85][Nichani 94][Hérault 91]. Il faut cependant noter que d'autres auteurs proposent d'attribuer une grande valeur à  $\tau$  afin de réduire les effets indésirables du terme

$$-\frac{U_{gd}}{\tau}$$

de l'équation VII-12 (Cf. Chapitre VI, § VI-6.2).

#### VII-6.1.4 Choix du paramètre $h$

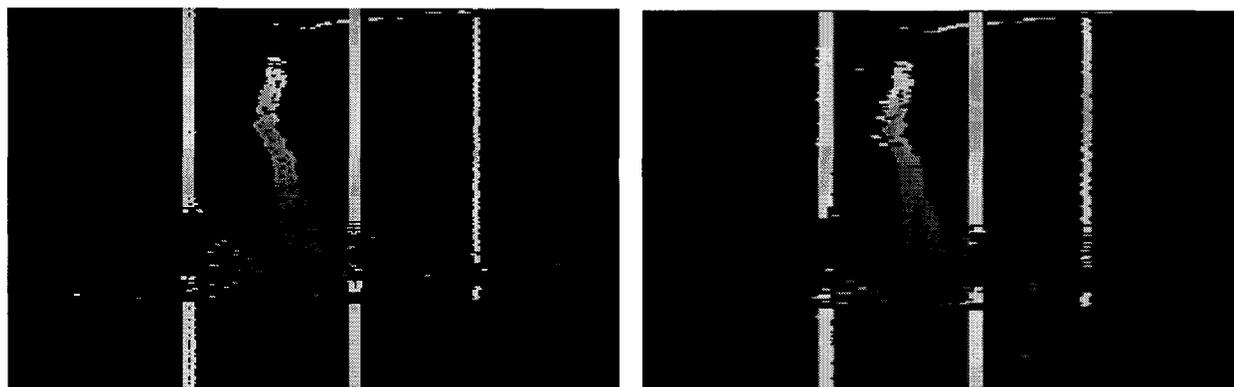
Le dernier paramètre  $h$  représente le pas de discrétisation utilisé pour résoudre l'équation différentielle qui régit l'évolution de chaque neurone (Cf. Equation VII-12). Nous avons constaté, au cours des tests, que lorsqu'une grande valeur est utilisée pour ce paramètre, on obtient de mauvais résultats. Ceci s'explique du fait que lorsque  $h$  est grand, la résolution de l'équation VII-12 est moins précise. Cependant, plus le paramètre  $h$  devient petit, meilleur est le résultat de la résolution de l'équation VII-12. Mais cette amélioration n'est obtenue qu'au prix d'une augmentation du temps de calcul. Après de nombreux tests, nous avons constaté que des résultats satisfaisants sont obtenus pour  $h = 0.0001$ .

### VII-6.2 Traitement des séquences stéréoscopiques

La procédure de mise en correspondance décrite ci-dessus est appliquée à des séquences d'images enregistrées grâce au stéréoscope linéaire. Chaque couple d'images stéréoscopiques est traité indépendamment des autres. A l'issue de chaque traitement, une liste des paires de points de contour mis en correspondance est constituée. Chaque paire de points de contour appariés permet, grâce aux formules de triangulation, de déterminer la distance et la position d'un point de la scène réelle (Cf. chapitre V, § V-5).

#### VII-6.2.1 Piéton évoluant devant le stéréoscope

La figure VII-15-a montre le résultat obtenu par la méthode proposée par Burie (Cf. Chapitre V, § V-4.2) sur la séquence stéréoscopique "*piéton*", dont les images brutes sont présentées au chapitre III (Cf. § III-5.2.1). Le résultat de traitement par notre procédure sur la même séquence est illustré sur la figure VII-15-b.



(a) Résultat obtenu par la méthode proposée par Burie.

(b) Résultat obtenu par notre méthode.

*Figure VII-15. Résultat obtenu sur la séquence "piéton".*

Les résultats représentés sur ces images s'interprètent de la manière suivante :

- L'axe horizontal représente l'abscisse dans la scène. Un point se trouvant sur le bord gauche de chaque image est situé à  $5\text{ m}$  à gauche de l'axe optique du stéréoscope linéaire. Un point se trouvant sur le bord droit de chaque image est situé à  $5\text{ m}$  à droite de ce même axe.
- L'axe vertical représente le temps. Celui-ci défilant du haut vers le bas de l'image.
- La distance est représentée par le niveau de gris. Celui-ci est d'autant plus clair que la distance par rapport au stéréoscope linéaire est importante.

Les points de contour des bandes blanches sont correctement appariés et leur détection est stable au cours du temps. En effet, les positions et distances calculées sont constantes. Les niveaux de gris clairs indiquent que les bandes sont éloignées du stéréoscope. La transition entre la chaussée et la zone d'ombre à droite de la figure est également bien détectée. En ce qui concerne le piéton, les niveaux de gris varient du clair au sombre, signalant bien que celui-ci se rapproche du stéréoscope.

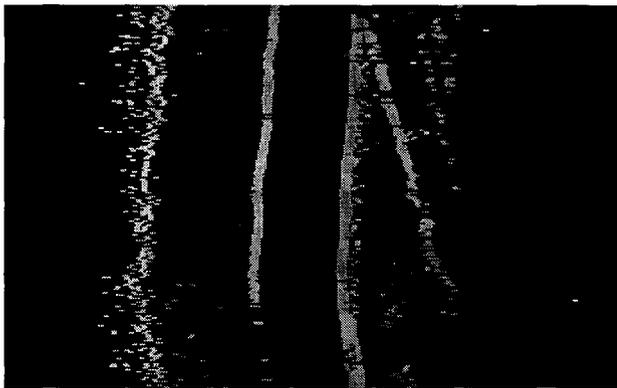
On remarque cependant, sur les deux résultats, la présence de mauvais appariements lorsque des occlusions se produisent, c'est à dire lorsque le piéton cache l'une des bandes blanches à l'une des caméras. Ces appariements erronés, qui sont peu nombreux, sont provoqués par la mise en correspondance des points de contour d'une des bandes blanches avec ceux du piéton.

Une simple comparaison visuelle de nos résultats avec ceux obtenus avec l'approche développée par Burie sur la même séquence stéréoscopique "piéton" (Cf. Figure VII-15) montre

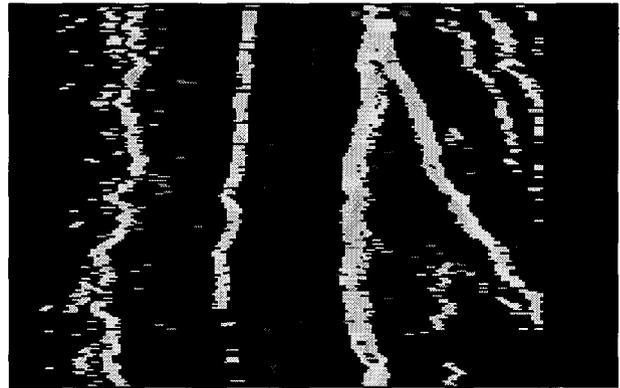
que notre approche est plus fiable et plus robuste. On constate, en effet, que notre procédure de mise en correspondance a réussi à appairer plus de points de contour du piéton que celle proposée par Burie. On peut ainsi voir clairement sur les résultats de la figure VII-15-a certains appariements manqués qui se manifestent par la présence de "trous" que l'on ne retrouve pas sur la figure VII-15-b.

### VII-6.2.2 Route dégagée et passage d'une sortie d'autoroute

Les figures VII-16-a montre le résultat obtenu par la procédure développée par Burie (Cf. § V-4.2) sur la séquence stéréoscopique "sortie d'autoroute", dont les images brutes sont présentées au chapitre III (Cf. § III-5.2.2). Le résultat obtenu par notre procédure est illustré sur la figure VII-16-b.



(a) Résultat obtenu par la méthode proposée par Burie.



(b) Résultat obtenu par notre méthode.

**Figure VII-16.** Résultat obtenu sur la séquence "sortie d'autoroute".

Sur ces figures, un point se trouvant sur le bord gauche de chaque image est situé à 9 m à gauche de l'axe optique du stéréoscope linéaire. Un point se trouvant sur le bord droit de chaque image est situé à 9 m à droite de ce même axe.

Les bandes blanches sont les seuls éléments présents dans cette séquence. Elles sont correctement détectées. La bande centrale et celle de droite présentent une bonne stabilité dans le temps. Par contre, la bande gauche n'apparaît pas nettement. Les nombreux points présents sur sa gauche sont dus aux herbes situées sur le bord de la route. Il faut cependant noter que, malgré quelques faux appariements, les herbes sont correctement détectées puisqu'elles se trouvent à la même distance que les bandes blanches.

La comparaison de nos résultats avec ceux de Burie (Cf. Figure VII-16) concernant cette séquence "sortie d'autoroute" montre, une fois de plus, que notre approche est la plus efficace du

fait qu'elle réussit toujours à appairer plus de points de contour. On peut constater qu'avec notre algorithme la bande blanche gauche ainsi que les bordures de la bretelle de sortie de l'autoroute sont mieux détectées, en ce sens qu'elles présentent moins de discontinuités.

### VII-6.2.3 Approche d'un véhicule

La figure VII-17 montre le résultat obtenu par notre procédure sur la séquence stéréoscopique "approche d'un véhicule", dont les images brutes sont présentées au chapitre III (Cf. § 5.2.3).

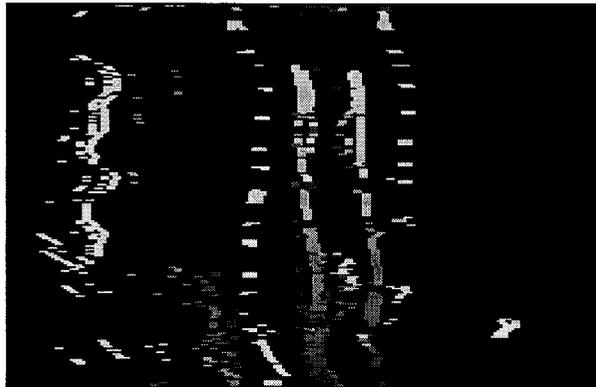


Figure VII-17. Résultat obtenu sur la séquence "approche d'un véhicule".

Sur cette figure, un point se trouvant sur le bord gauche de l'image est situé à 7 m à gauche de l'axe optique du stéréoscope linéaire. Un point se trouvant sur le bord droit de l'image est situé à 7 m à droite du même axe.

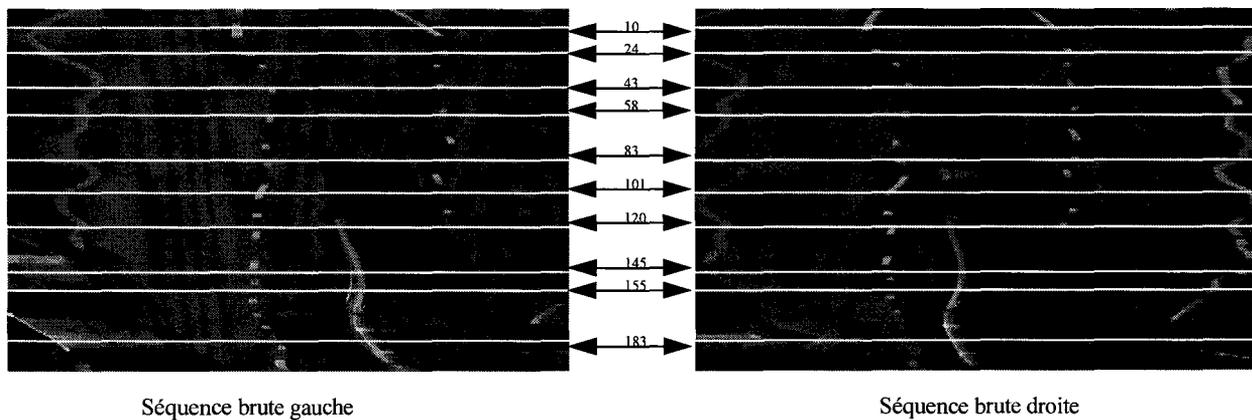
Sur la scène reconstituée (Cf. Figure VII-17), la plupart des points de contour objet de la *R4* sont reconstruits correctement. On note que, contrairement à ce qu'on constate sur les images brutes, la largeur apparente de la *R4* est ici la même pour toutes les lignes de la séquence reconstruite. Ceci est dû à la définition de l'axe horizontal, qui représente la largeur réelle des objets, et non pas celle de leur projection sur les capteurs. Le niveau de gris des points de contour de la *R4*, de plus en plus foncé, traduit son rapprochement au cours du temps.

On voit également, en pointillés, les deux lignes blanches qui séparent les voies de l'autoroute. Leur ondulation est due au mouvement de lacet du véhicule prototype par rapport à la chaussée. En effet, le repère de travail est lié à notre véhicule, et non à l'axe de la chaussée. La ligne blanche qui délimite le terre-plein central est également détectée, du moins lorsqu'elle est dans le champ des deux caméras.

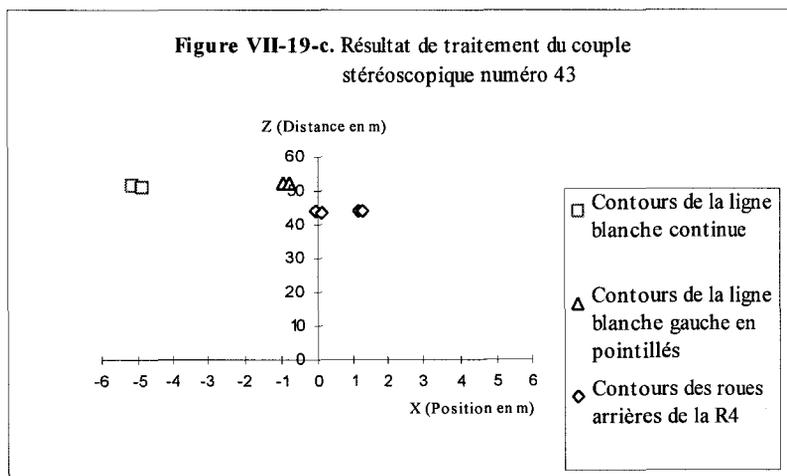
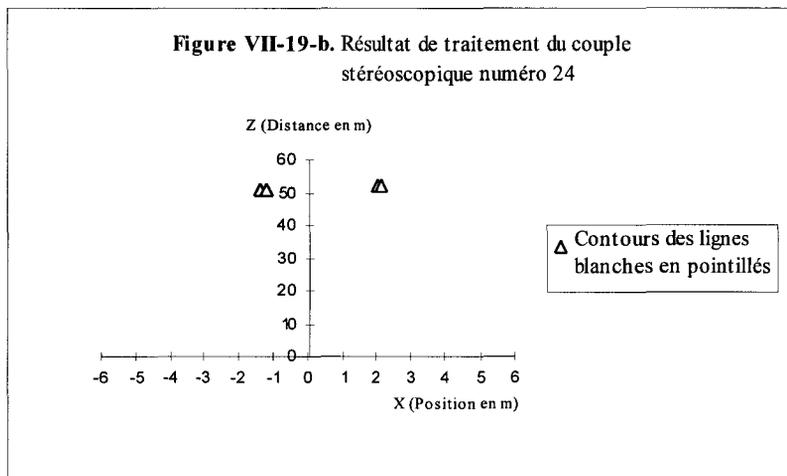
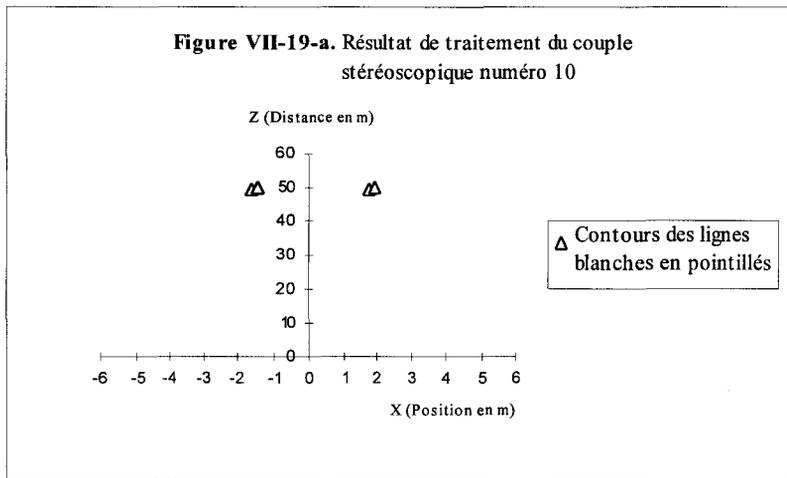
Le véhicule qui se rabat devant la *R4* est clairement visible en bas de l'image, sous la forme d'une ligne claire continue correspondant à son extrémité gauche. Son extrémité droite n'est pas visible en raison d'une occlusion par la *R4* qui la masque à la caméra droite. L'extrémité de sa longue ombre est visible en bas, à l'extrême droite de l'image.

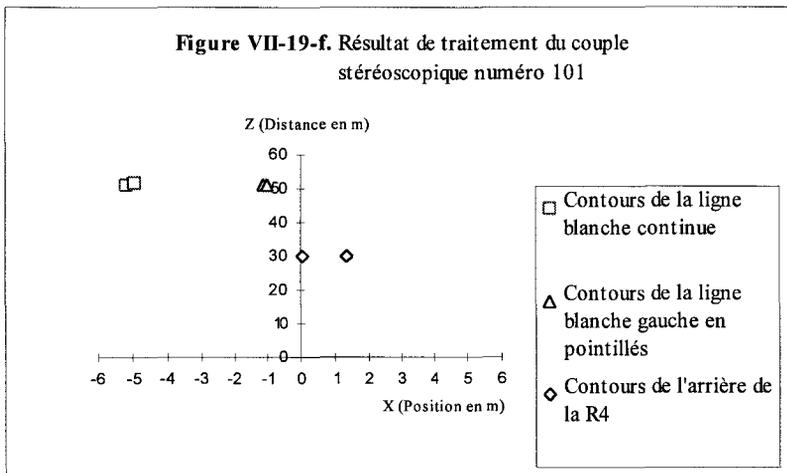
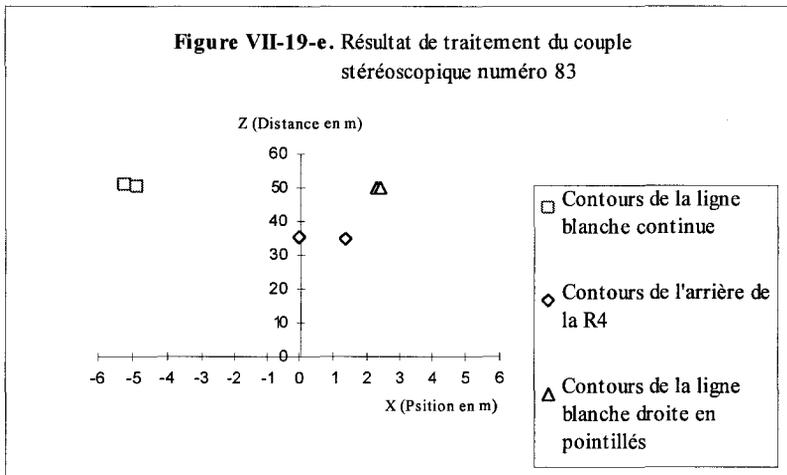
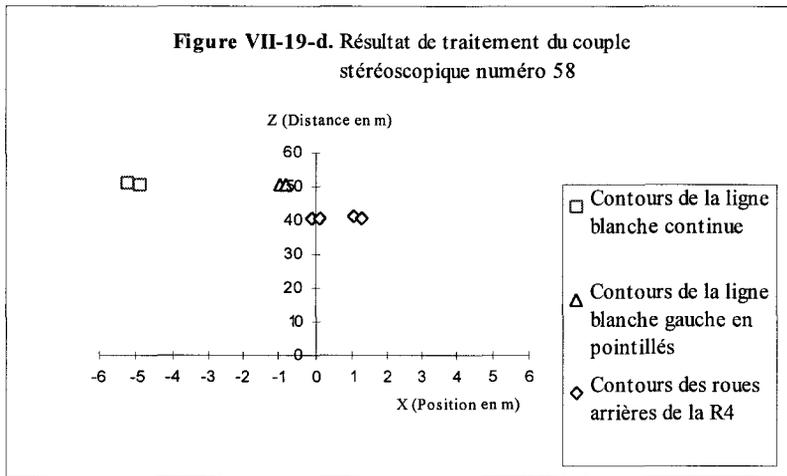
La voiture qui double le véhicule prototype est visible, en bas et à gauche de l'image, comme une ligne oblique très sombre en raison de sa proximité. Malgré son faible contraste par rapport à la chaussée, il y a peu de détections manquées au cours du temps, ce qui indique une bonne robustesse de la détection des points de contour ainsi que de la mise en correspondance. Les quelques erreurs d'appariement sont dues à l'apparition momentanée, sur l'image droite, de la ligne blanche qui délimite le terre-plein central : l'algorithme semble confondre cette ligne avec le bord droit de la voiture.

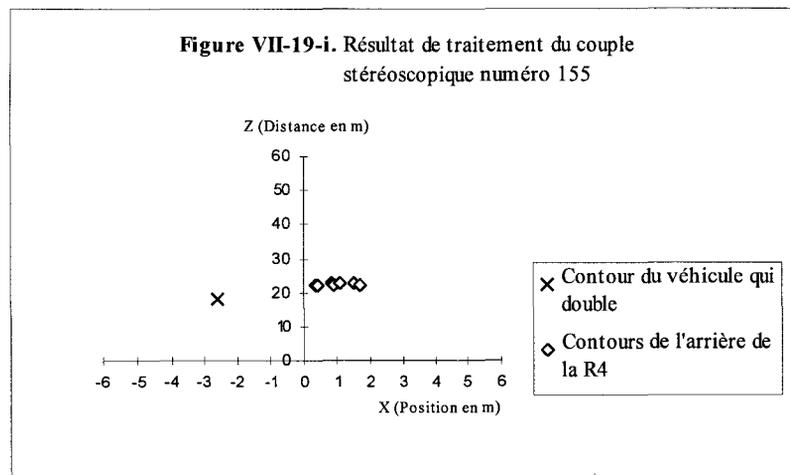
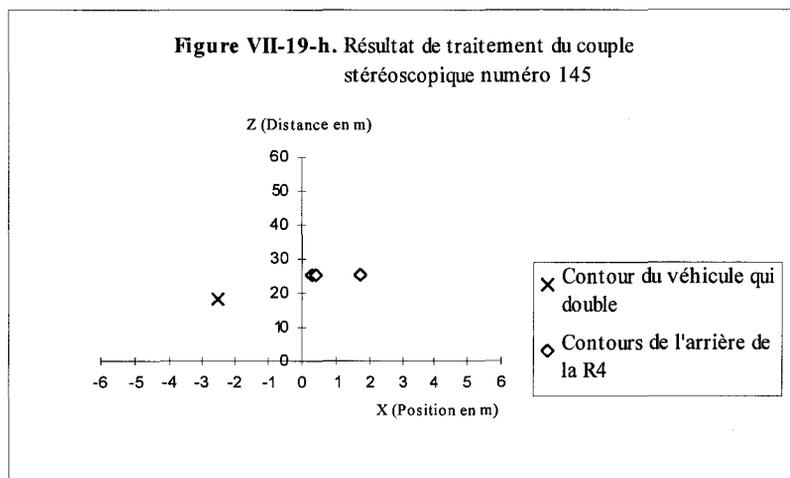
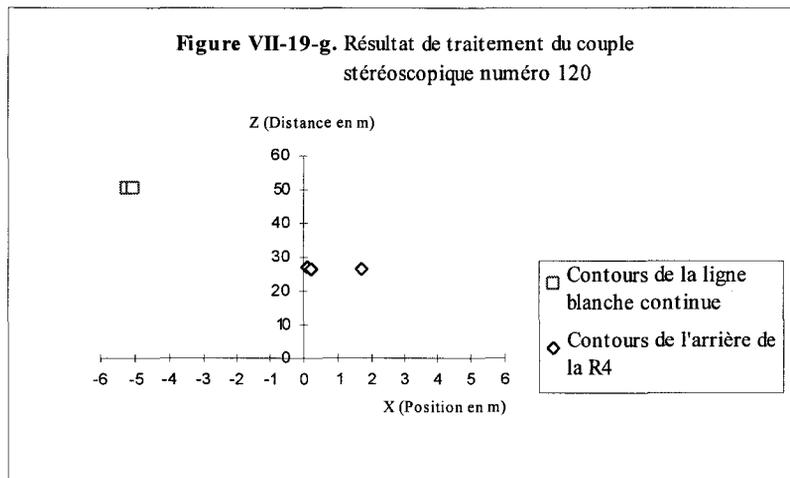
Nous avons retenu cette séquence complexe pour montrer comment est réalisée la reconstruction 3D. Sur la figure VII-18, nous avons sélectionné 10 couples d'images ligne représentatifs de différentes situations couramment rencontrées. Pour chacun de ces couples, la figure VII-19 indique, de manière quantifiée, les résultats de la reconstruction 3D.

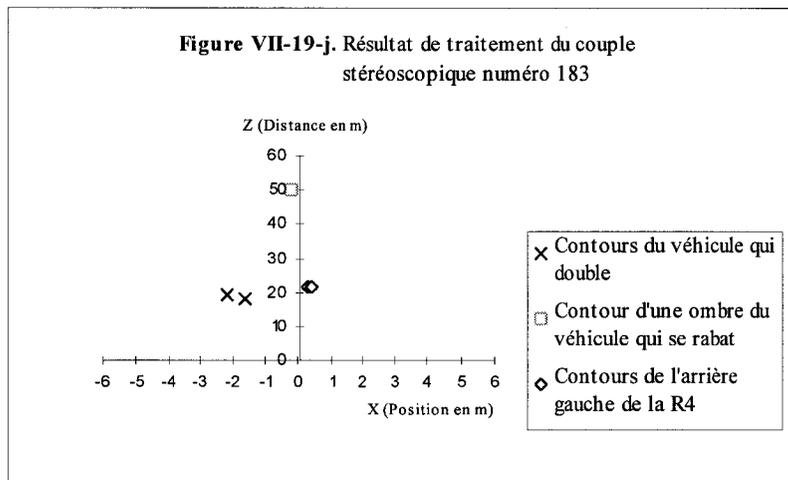


*Figure VII-18. Couples d'images stéréoscopiques sélectionnés.*









*Figure VII-19. Résultats de traitement des couples d'images stéréoscopiques sélectionnés.*

### ***Couples stéréoscopiques 10 et 24 :***

Au début de la séquence, le secteur de stéréovision coupe la chaussée à 50 m à l'avant du stéréoscope. Les points de contour des deux lignes blanches en pointillés sont mis en correspondance correctement. Les figures VII-19-a et VII-19-b indiquent la reconstruction des points de contour objet correspondants par rapport au repère lié au stéréoscope.

### ***Couples stéréoscopiques 43 et 58 :***

Comme le véhicule prototype se rapproche de plus en plus de la R4, le secteur de stéréovision coupe les deux roues arrières qui sont localisées à environ 40 m à l'avant du stéréoscope (Cf. Figures VII-19-c et VII-19-d). Une seule ligne blanche en pointillés est vue par les deux caméras. La ligne blanche continue de gauche, qui délimite le terre-plein central, est également vue par les deux caméras. Ses points de contour objet sont reconstruits à environ 50 m à l'avant du stéréoscope.

### ***Couples stéréoscopiques 83, 101 et 120 :***

Comme la distance entre les deux véhicules diminue, le secteur de stéréovision coupe l'arrière de la R4 à un niveau de plus en plus élevé. Les points de contour sont reconstruits correctement au cours de ce rapprochement du véhicule prototype (Cf. Figures VII-19-e, VII-19-f et VII-19-g). Pendant ce temps, les lignes blanches de la chaussée apparaissent et disparaissent. Cependant, quand elles sont vues par les deux caméras du stéréoscope, elles sont toujours reconstruites correctement.

**Couple stéréoscopique 145 :**

Les points de contour objet de la *R4* sont correctement détectés (Cf. Figure VII-19-h). Un point de contour objet situé sur le bord droit d'un autre véhicule qui double le véhicule prototype sur la voie de gauche est également reconstruit.

**Couple stéréoscopique 155 :**

Plusieurs points de contour objet de la *R4* sont vus par les deux caméras, au niveau où elle est coupée par le secteur de stéréovision. Tous ces points sont reconstruits à la même distance (Cf. Figure VII-19-i).

**Couple stéréoscopique 183 :**

Avec ce couple stéréoscopique, le véhicule qui double le véhicule prototype est détecté par l'intermédiaire de deux points de contour objet (Cf. Figure VII-19-j). Par contre, seuls deux couples de points de contour de la *R4* sont mis en correspondance, à cause de l'ombre produite, à gauche des images, par le véhicule qui se rabat devant la *R4*. On note aussi la détection d'un point de contour objet correspondant à une ombre entre le véhicule prototype et celui qui le double. Ce point est reconstruit correctement puisqu'il est détecté à *50 m* à l'avant du stéréoscope.

La figure VII-20 synthétise les résultats des traitements de ces *10* couples d'images stéréoscopiques en considérant l'évolution des distances des "objets" détectés. Les distances des points de contour objet correspondant aux lignes blanches oscillent autour de *50 m*. Ceci est dû au mouvement de lacet du véhicule prototype par rapport à la chaussée. L'évolution des distances des points de contour objet de la *R4* traduit son rapprochement par rapport au véhicule prototype au cours du temps. Les points de contour objet du véhicule qui double le véhicule prototype sont presque toujours à la même distance par rapport à ce dernier. Ceci est tout à fait normal car le secteur de stéréovision coupe le côté droit de ce véhicule qui est donc vu par les deux caméras à une distance constante au cours du dépassement (l'arrière du véhicule qui double n'est pas vu au cours de cette partie de la manoeuvre).

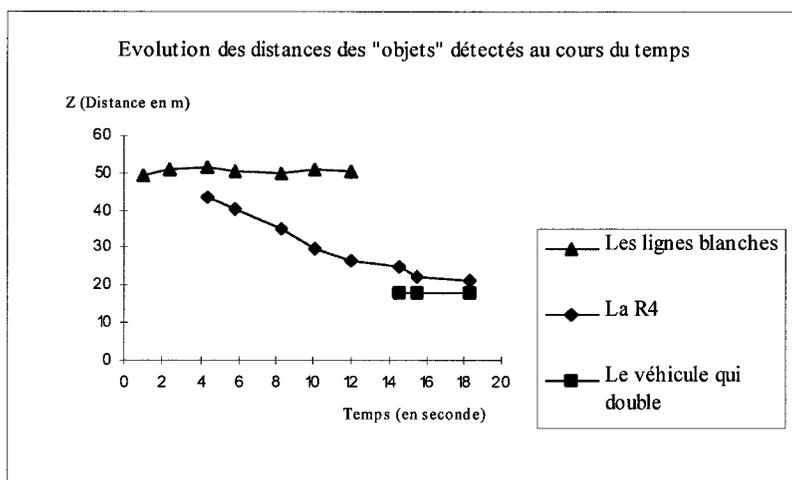


Figure VII-20. Evolution des distances calculées au cours du temps.

### VII-6.3 Traitement informatique des séquences stéréoscopiques

Pour permettre une implantation temps réel des algorithmes, l'ensemble des traitements est effectué par un ordinateur de type *PC-AT* équipé d'une carte *Alacron AL860-XP* qui comporte deux processeurs *RISC* cadencés à *50 MHz* (Cf. Figure VII-21). Dans le cadre du projet, le *PC-AT* choisi est équipé d'un processeur très simple : un *Intel 486SX* cadencé à *33 MHz*.

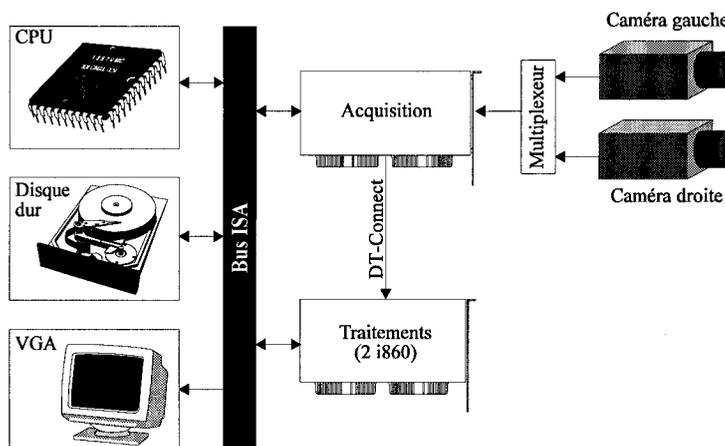


Figure VII-21. Support informatique des traitements.

La détection d'obstacles nécessite d'importants calculs qui sont confiés à la carte de traitement. La totalité des procédures exécutées sur cette carte est télé-chargée par le *486* du *PC-AT* à l'initialisation, via le bus *ISA*. Par la suite, les procédures de traitement sont accessibles depuis le *486* par un simple appel. A partir de l'instant où les images arrivent dans la mémoire de la carte, celle-ci va effectuer la totalité des traitements nécessaire à la détection, à savoir :

- L'extraction des points de contour.
- La mise en correspondance.

Pour plus de détails sur l'architecture informatique du système stéréoscopique, le lecteur pourra consulter la thèse de Burie [Burie 95].

L'implantation de notre procédure de mise en correspondance, ainsi que celle d'extraction des points de contour, sur l'un des deux processeurs de la carte, nous permet de traiter 6 à 8 paires d'images stéréoscopiques par seconde. Cette vitesse de traitement n'est pas suffisante pour une application temps réelle telle que la détection d'obstacles à l'avant des véhicules routiers. Cette limitation est due aux quantités de calculs importantes que nécessite la procédure de mise en correspondance. En effet, le réseau de neurones qui est à la base de cette procédure nécessite le calcul de ses paramètres, en particulier la matrice des poids de ses connexions. Celle-ci est de très grande taille lorsque le réseau contient un nombre important de neurones : si chaque image contient 10 points de contour, le réseau comporte 100 neurones et par la suite la matrice des poids de connexions est de taille  $100 \times 100$ . D'autre part, plus la taille du réseau est importante, moins le processus de stabilisation du réseau est rapide. Afin de pallier cet inconvénient, nous avons développé une autre procédure de mise en correspondance basée sur le même principe que la première, mais qui utilise un réseau de neurones de taille moins importante (Cf. § VII-7).

#### **VII-6.4 Conclusion**

Nous avons présenté quelques résultats de mise en correspondance obtenus sur des séquences stéréoscopiques correspondant à des scènes réelles. Ceux-ci sont jugés très satisfaisants et montrent que notre approche de mise en correspondance est fiable robuste. Une comparaison visuelle de nos résultats avec ceux obtenus par Burie a montré que notre procédure est la plus efficace. Cependant, la cadence avec laquelle les séquences sont traitées est insuffisante pour une application temps réel. Afin de pallier cette limitation, nous proposons une deuxième implantation de la procédure de mise en correspondance.

#### **VII-7 OPTIMISATION DU RESEAU**

La principale limitation de la procédure de mise en correspondance que nous avons développée provient d'un temps de traitement trop long. Ce problème est dû, sans aucun doute, à la taille importante du réseau qui est à la base du traitement. Impliquant un nombre important de neurones, ce réseau nécessite une grande quantité de calculs pour parvenir à la solution du problème.

Afin de diminuer le temps des traitements, nous proposons une nouvelle implantation basée sur la même approche que celle présentée précédemment. L'idée est de réduire la taille du réseau en éliminant les neurones dont on sait qu'ils correspondent, dès le départ, à des appariements incompatibles. Ceux-ci sont connus par application des contraintes de position et de signe du gradient (Cf. § VII-3.1). La démarche consiste à traiter séparément ces deux contraintes locales pour ne laisser, dans le réseau, que les contraintes globales d'unicité, d'ordre et de continuité de la disparité (Cf. § VII-3.2). Ainsi, les contraintes locales sont exploitées sous la forme d'une procédure de sélection préalable de telle sorte que le nouveau réseau n'implique que les neurones qui correspondent à des appariements respectant ces deux contraintes. Avec cette procédure de sélection, nous réduisons la taille du réseau de la figure VII-12 de 30 à 10 neurones (Cf. Figure VII-22). Les neurones impliqués dans le réseau sont ceux colorés en blanc. Les neurones colorés en gris sont ceux qui correspondent aux appariements incompatibles.

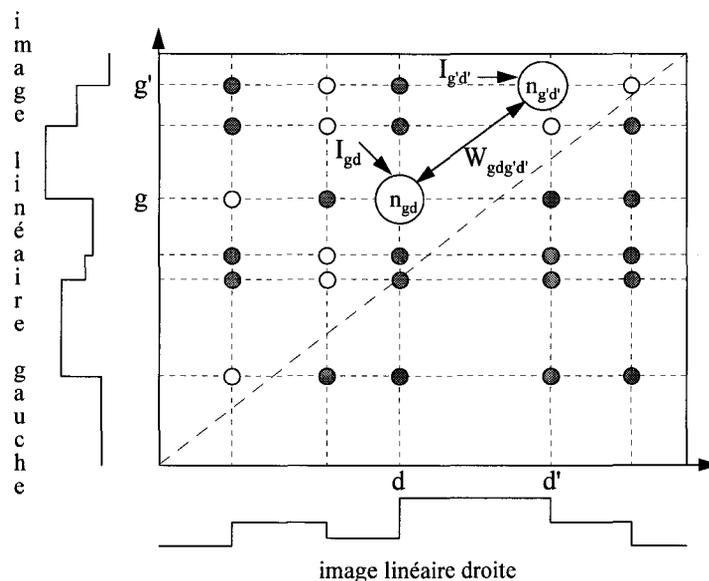


Figure VII-22. Nouvelle implantation de la mise en correspondance via le réseau de Hopfield.

### VII-7.1 Calcul des paramètres du réseau

Comme les contraintes locales de position et de signe du gradient sont utilisées pour éliminer les appariements incompatibles, elles ne sont plus représentées dans la nouvelle fonction de coût. Celle-ci ne représente alors que les contraintes d'unicité, d'ordre et de la continuité de la disparité, qui sont des contraintes globales (Cf. § VII-3.2). Elle s'exprime de la façon suivante :

$$\begin{aligned}
H_p = & K_{unicité} \left[ \sum_{g \in G} \left( 1 - \sum_{d \in D / (g,d) \in \Omega} E_{gd} \right)^2 + \sum_{d \in D} \left( 1 - \sum_{g \in G / (g,d) \in \Omega} E_{gd} \right)^2 \right] \\
& + K_{ordre} \sum_{(g,d) \in \Omega} \sum_{\substack{(g',d') \in \Omega \\ g' \neq g \text{ et } d' \neq d}} O_{gdg'd'} E_{gd} E_{g'd'} \\
& - K_{continuité} \sum_{(g,d) \in \Omega} \sum_{\substack{(g',d') \in \Omega \\ g' \neq g \text{ et } d' \neq d}} C_{gdg'd'} E_{gd} E_{g'd'}
\end{aligned} \tag{Equation VII-16}$$

où  $K_{unicité}$ ,  $K_{ordre}$  et  $K_{continuité}$  sont les coefficients de pondération des différents termes de coût.  $\Omega$  représente l'ensemble des appariements possibles après prise en compte des contraintes de position et de signe du gradient :

$$\Omega = \{(g,d) \in G \times D / g \text{ et } d \text{ vérifient les contraintes de position et de signe du gradient}\}.$$

Nous rappelons que le premier terme de la nouvelle fonction de coût représente la contrainte d'unicité. Le troisième et le quatrième termes représentent respectivement la contrainte d'ordre et la contrainte de continuité de la disparité.

De la même manière que précédemment (Cf. § VII-5.1), les poids des connexions entre les neurones et leurs entrées externes sont déterminés en mettant la fonction de coût du problème sous la forme de la fonction d'énergie du réseau :

$$H_r = -\frac{1}{2} \sum_{(g,d) \in \Omega} \sum_{(g',d') \in \Omega} W_{gdg'd'} E_{gd} E_{g'd'} - \sum_{(g,d) \in \Omega} I_{gd} E_{gd} \tag{Equation VII-17}$$

En identifiant les deux expressions des équations VII-16 et VII-17, on obtient :

$$\begin{aligned}
W_{gdg'd'} = & -K_{unicité} [\delta_{gg'} (1 - \delta_{dd'}) + \delta_{dd'} (1 - \delta_{gg'})] \\
& - K_{ordre} \cdot O_{gdg'd'} (1 - \delta_{gg'}) (1 - \delta_{dd'}) \\
& + K_{continuité} \cdot C_{gdg'd'} (1 - \delta_{gg'}) (1 - \delta_{dd'})
\end{aligned} \tag{Equation VII-18}$$

$$I_{gd} = 2K_{unicité}$$

où  $\delta_{ij}$  est l'indice de Kroneker tel que :

$$\delta_{ij} = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{sinon} \end{cases}.$$

### VII-7.2 Procédure de mise en correspondance

La procédure de mise en correspondance consiste à faire évoluer, à partir d'un état initial, le réseau ainsi défini vers un état stable. L'état initial du réseau est obtenu par la même procédure que celle présentée au § VII-5.2.2. Le principe de fonctionnement du réseau est le même que celui de la première implantation (Cf. § VII-5.2.1). Les points de contour appariés sont ensuite déterminés en analysant l'état final du réseau selon la procédure décrite au § VII-5.3.

### VII-7.3 Résultats de mise en correspondance

La nouvelle procédure de mise en correspondance est appliquée sur les séquences stéréoscopiques que nous avons traitées auparavant. Les résultats sont strictement identiques à ceux obtenus avec la première procédure (Cf. § VII-6.2), ce qui est tout à fait logique, puisque la seule différence entre les deux procédures est que les contraintes locales de position et de signe du gradient, qui permettent d'éliminer les appariements incompatibles, sont utilisées de deux manières différentes.

Concernant la vitesse à laquelle les séquences sont traitées, nous avons obtenu une bonne amélioration. En effet, avec cette nouvelle implantation, nous pouvons traiter 10 à 12 couples d'images stéréoscopiques à la seconde. Bien que cette vitesse soit acceptable, elle peut encore être améliorée. Dans ce but, nous proposons, par la suite, en se basant sur la même approche de départ, une troisième implantation de la procédure de mise en correspondance (Cf. § VII-8).

Notons que ces résultats sont obtenus avec des paramètres de valeurs identiques à celles utilisées précédemment (Cf. § VII-6.1).

### VII-7.4 Conclusion

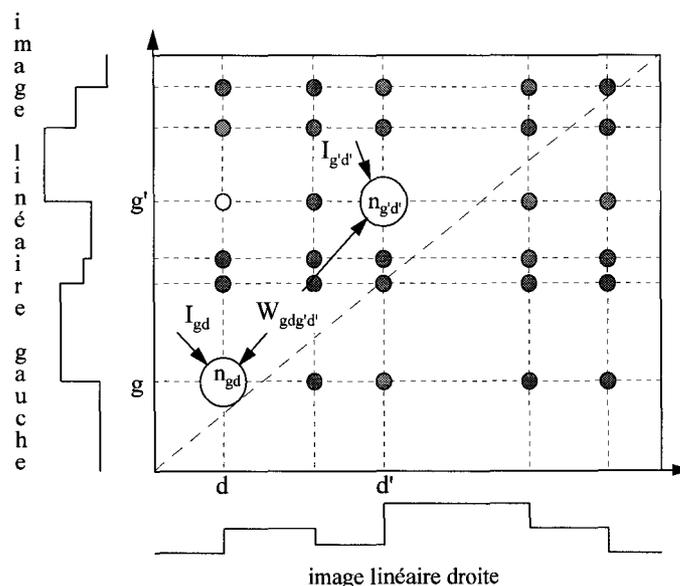
Nous avons présenté une nouvelle implantation pour la mise en correspondance des points de contour d'un couple d'images stéréoscopiques. Celle-ci est basée sur le même principe que la première, mais la nouvelle fonction de coût ne représente que les contraintes globales, c'est à dire les contraintes d'unicité, d'ordre et de continuité de la disparité. Pour minimiser cette fonction, nous avons utilisé un réseau de Hopfield composé seulement des neurones qui correspondent à des appariements compatibles vis-à-vis les contraintes locales de position et de signe du gradient. Les résultats de détection sur les séquences traitées sont identiques à ceux fournis par la première procédure. Cependant, nous avons obtenu une bonne amélioration concernant la vitesse de traitement des séquences stéréoscopiques. Le désir d'améliorer encore davantage cette cadence

nous a poussé à développer une troisième implantation basée sur la même approche que les précédentes. La présentation de celle-ci fait l'objet du paragraphe suivant.

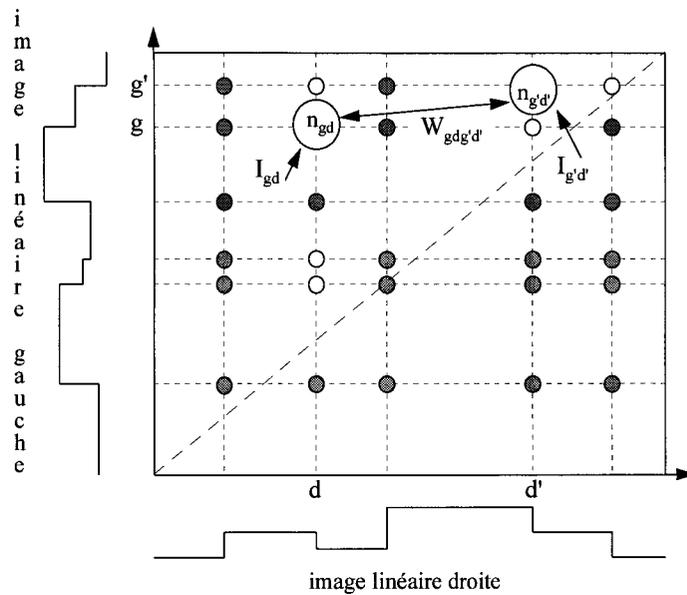
## VII-8 APPARIEMENT PAR SEPARATION DES PRIMITIVES

Afin d'améliorer d'avantage la vitesse des traitements, nous avons développé une troisième implantation pour la mise en correspondance en utilisant le même outil que pour les deux précédentes, c'est à dire les réseaux de Hopfield. L'idée principale est de séparer l'ensemble des points de contour de chaque image du couple stéréoscopique en deux sous-ensembles : l'ensemble des points de contour dont le signe du gradient est positif et l'ensemble des points de contour dont le signe du gradient est négatif. On décompose alors le problème en deux sous-problèmes identiques. Chacun de ces sous-ensembles consiste à mettre en correspondance les points de contour de même nature.

On doit construire alors un réseau de Hopfield pour chaque sous-problème (Cf. Figure VII-23). Comme pour l'implantation précédente, les neurones de chaque réseau ne doivent représenter que les appariements compatibles. A cet effet, une procédure de sélection, basée sur l'utilisation de la contrainte de position, est appliquée. Notons que la contrainte de signe du gradient ne sert plus à rien puisque les points de contour à mettre en correspondance ont le même signe du gradient. Les neurones impliqués dans chaque réseau sont ceux colorés en blanc. Les neurones colorés en gris sont ceux qui correspondent aux appariements incompatibles vis-à-vis la contrainte de position.



(a) Réseau construit à partir des points de contour de gradient positif.



(b) Réseau construit à partir des points de contour de gradient négatif.

Figure VII-23. Implantation via deux réseaux de Hopfield.

Par rapport aux deux précédentes implantations, cette stratégie permet de construire des réseaux de tailles moins importantes, ce qui rend leurs convergences plus rapides.

### VII-8.1 Calcul des paramètres des deux réseaux

Pour l'appariement de chaque type de points de contour, une fonction de coût est construite à partir des contraintes d'unicité, d'ordre et de continuité de la disparité (Cf. § VII-3.2). Pour les points de contour dont le signe du gradient est positif, cette fonction s'exprime sous la forme :

$$\begin{aligned}
 H_p = & K_{unicité} \left[ \sum_{g \in G^+} \left( 1 - \sum_{d \in D^+ / (g,d) \in \Gamma^+} E_{gd} \right)^2 + \sum_{d \in D^+} \left( 1 - \sum_{g \in G^+ / (g,d) \in \Gamma^+} E_{gd} \right)^2 \right] \\
 & + K_{ordre} \sum_{(g,d) \in \Gamma^+} \sum_{\substack{(g',d') \in \Gamma^+ \\ g' \neq g \text{ et } d' \neq d}} O_{gdg'd'} E_{gd} E_{g'd'} \\
 & - K_{continuité} \sum_{(g,d) \in \Gamma^+} \sum_{\substack{(g',d') \in \Gamma^+ \\ g' \neq g \text{ et } d' \neq d}} C_{gdg'd'} E_{gd} E_{g'd'}
 \end{aligned}
 \tag{Equation VII-19}$$

où  $K_{unicité}$ ,  $K_{ordre}$  et  $K_{continuité}$  sont les coefficients de pondération des différents termes de coût.  $G^+$  et  $D^+$  sont respectivement les sous-ensembles des points de contour de gradient positif de l'image gauche et de l'image droite.  $\Gamma^+$  représente l'ensemble des appariements possibles entre les deux sous-ensembles  $G^+$  et  $D^+$  après application de la contrainte de position :

$$\Gamma^+ = \{(g,d) \in G^+ \times D^+ / g \text{ et } d \text{ vérifient la contrainte de position}\}.$$

Nous rappelons que le premier terme de cette fonction de coût représente la contrainte d'unicité. Les contraintes d'ordre et de continuité de la disparité sont représentées respectivement par le troisième et le quatrième terme.

La fonction d'énergie du réseau associé au sous-problème en question est donnée par l'expression suivante :

$$H_r = -\frac{1}{2} \sum_{(g,d) \in \Gamma^+} \sum_{(g',d') \in \Gamma^+} W_{gdg'd'} E_{gd} E_{g'd'} - \sum_{(g,d) \in \Gamma^+} I_{gd} E_{gd} \quad \text{Equation VII-20}$$

La matrice des poids des connexions entre les neurones du réseau et son vecteur d'entrées externes sont déterminés en identifiant les expressions des équations VII-19 et VII-20. On obtient :

$$\begin{aligned} W_{gdg'd'} = & -K_{unicité} [\delta_{gg'}(1 - \delta_{dd'}) + \delta_{dd'}(1 - \delta_{gg'})] \\ & - K_{ordre} \cdot O_{gdg'd'}(1 - \delta_{gg'})(1 - \delta_{dd'}) \\ & + K_{continuité} \cdot C_{gdg'd'}(1 - \delta_{gg'})(1 - \delta_{dd'}) \end{aligned} \quad \text{Equation VII-21}$$

$$I_{gd} = 2K_{unicité}$$

où  $\delta_{ij}$  est l'indice de Kroneker tel que :

$$\delta_{ij} = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{sinon} \end{cases}.$$

Le même raisonnement s'applique pour calculer les paramètres du réseau correspondant à la mise en correspondance des points de contour de gradient négatif.

### VII-8.2 Procédure de mise en correspondance

Pour faire évoluer les deux réseaux vers des états stables, nous avons appliqué le même principe que celui utilisé pour les réseaux précédents (Cf. § VII-5.2.1). Comme pour les autres implantations, les deux réseaux sont initialisés par application de la procédure décrite au § VII-5.2.2. Les résultats de mise en correspondance fournis par chaque réseau sont ensuite regroupés pour reconstituer les appariements finaux des points de contour.

Par rapport aux procédures précédentes, celle-ci présente un avantage majeur : les deux réseaux peuvent être implantés parallèlement. Ce point est très important car il donne encore une possibilité de réduire le temps des traitements.

### VII-8.3 Résultats de mise en correspondance

Les mêmes séquences stéréoscopiques sont traitées par la procédure décrite ci-dessus. Les résultats ci-dessous sont obtenus avec les mêmes valeurs de paramètres que précédemment (Cf. § VII-6.1).

Les figures VII-24, VII-25 et VII-26 montrent respectivement les résultats de mise en correspondance sur les séquences "*piéton*", "*sortie d'autoroute*" et "*approche d'un véhicule*". Nous pouvons constater que les résultats de détection sont moins bons que ceux fournis par les deux procédures précédentes. En effet, quelques faux appariements supplémentaires sont générés. On signale aussi des appariements manqués qui se manifestent surtout sur le résultat obtenu sur la séquence "*approche d'un véhicule*" (Cf. Figure VII-26). Ces problèmes de mise en correspondance sont dus sans aucun doute au fait que les contraintes d'ordre et de continuité de la disparité, qui sont des contraintes globales, ne prennent pas en compte tous les points de contour des images. En effet, ces deux contraintes sont appliquées globalement, mais pour n'apparier que les points de contour de même signe. Ainsi, lors de l'appariement d'un groupe de points de contour, elles ne prennent pas en compte les points de contour de l'autre groupe.

Nos résultats sont comparables à ceux obtenus par Burie (Cf. Figures VII-24 et VII-25). En effet, nous pouvons constater que notre procédure réussit à apparier plus de points de contour. Sur les résultats de la figure VII-24, le piéton est mieux détecté avec notre algorithme. Sur les résultats de la figure VII-25, on voit clairement que notre procédure détecte mieux toutes les bandes blanches de la chaussée. En revanche, notre procédure de mise en correspondance fournit quelques faux appariements supplémentaires par rapport à celle proposée par Burie. Mais, un filtrage spatio-temporel permettrait de les supprimer sans difficulté [Burie 95].

Le traitement des séquences stéréoscopiques avec cette procédure est effectué avec une vitesse allant de 14 à 17 paires d'images stéréoscopiques par seconde. Cette cadence est très proche de celle obtenue par la méthode de Burie qui traite 16 à 18 paires d'images stéréoscopiques à la seconde. Ce résultat est très encourageant, surtout qu'une amélioration pourrait être encore obtenue grâce à une implantation parallèle des deux réseaux. Ceci est tout à

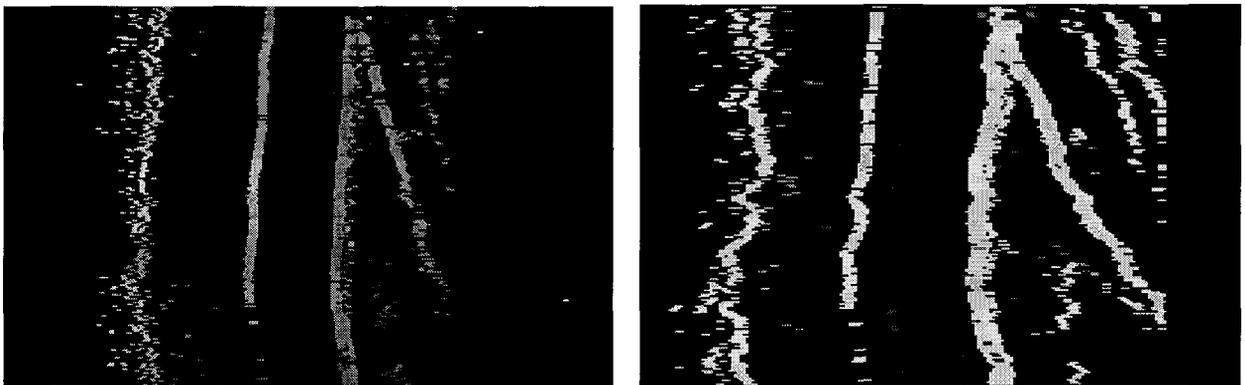
fait possible en considérant les deux processeurs de la carte de traitement que nous avons utilisée (Cf. § VII-6.3). Ainsi, chaque réseau pourrait être implanté sur un processeur *i860*.



(a) Résultat obtenu par la méthode proposée par Burie.

(b) Résultat obtenu par notre troisième implantation.

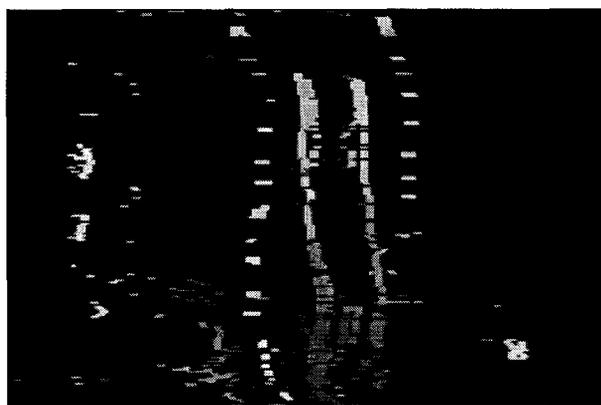
*Figure VII-24. Résultats obtenus sur la séquence "piéton".*



(a) Résultat obtenu par la méthode proposée par Burie.

(b) Résultat obtenu par notre troisième implantation.

*Figure VII-25. Résultats obtenus sur la séquence "sortie d'autoroute".*



*Figure VII-26. Résultat obtenu sur la séquence "approche d'un véhicule".*

#### VII-8.4 Conclusion

Nous avons proposé une troisième implantation pour la mise en correspondance des points de contour d'un couple d'images stéréoscopiques. Celle-ci est basée sur la séparation des points

de contour de chaque image en deux groupes suivant leur signe du gradient. Le problème de la mise en correspondance est alors décomposé en deux sous-problèmes identiques qui consistent à apparier les points de contour de même nature. A cet effet, une fonction de coût est construite pour chaque groupe de points de contour. Deux réseaux de Hopfield sont ensuite mis au point pour réaliser la mise en correspondance des points de contour de chaque groupe. Les résultats obtenus sont moins bons que ceux fournis par les deux implantations précédentes. Cependant, ils restent toujours comparables avec les résultats obtenus par la méthode de Burie. En ce qui concerne la vitesse des traitements, nous avons obtenu une amélioration substantielle par rapport aux deux implantations précédentes. En effet, nous avons atteint une cadence allant de 14 à 17 paires stéréoscopique à la seconde. Cette vitesse est presque la même que celle obtenue par Burie. De plus, la stratégie utilisée permet d'envisager une parallélisation de l'implantation des deux réseaux, ce qui augmenterait davantage la cadence des traitements.

## VII-9 CONCLUSION

Nous avons présenté, au cours de ce chapitre, une approche neuronale pour le problème de la mise en correspondance en stéréovision linéaire. Elle consiste, dans un premier temps, à exprimer le problème en termes de minimisation d'une fonction de coût construite à partir d'un certain nombre de contraintes stéréoscopiques. Un réseau de neurones de Hopfield est ensuite utilisé pour implanter le processus de minimisation de cette fonction de coût.

Trois implantations sont proposées. La première consiste à construire un réseau qui prend en compte toutes les contraintes stéréoscopiques. Cette implantation, qui donne de bons résultats, demande beaucoup de temps de calcul, ce qui est peu compatible avec une application temps réel telle que la détection d'obstacles à l'avant des véhicules routiers.

Afin d'améliorer les temps de traitement des séquences stéréoscopiques, nous avons proposé une autre implantation de la mise en correspondance des images linéaires. Basée sur le même principe que la première, celle-ci utilise un réseau de taille moins importante en ne gardant que les neurones correspondant aux appariements compatibles vis-à-vis les contraintes locales de position et de signe du gradient. Dans ce cas, ces deux contraintes ne sont pas représentées dans la fonction de coût. Pour les mêmes résultats de détection que ceux obtenus par la première procédure, celle-ci apporte une amélioration substantielle au niveau de la vitesse avec laquelle les séquences stéréoscopiques sont traitées.

Cette vitesse est cependant jugée insuffisante pour notre application. Nous avons donc développé une troisième implantation dont le principe est de décomposer le problème en deux sous-problèmes identiques. Chacun de ces sous-problèmes consiste à mettre en correspondance les points de contour de même nature, c'est à dire de même signe du gradient. Deux réseaux sont ensuite construits pour minimiser les deux fonctions de coût correspondant aux deux sous-problèmes. En suivant cette stratégie, nous avons réduit considérablement le temps de traitement. De plus, les deux réseaux peuvent être implantés parallèlement, ce qui augmenterait davantage la cadence de traitement. Ce point est très important dans le cadre de notre application qui doit être réalisée en temps réel. Cependant, les résultats de mise en correspondance sont moins bons que ceux obtenus par les deux premières implantations, mais ils sont jugés suffisants en les comparant avec ceux obtenus par Burie.

## **Chapitre VIII**

# **Conclusion générale**

Depuis une dizaine d'année, de nombreux chercheurs et constructeurs automobiles se sont penchés sur le problème de l'amélioration de la sécurité routière. Les travaux qui ont été entrepris ont donné naissance à plusieurs programmes de recherche dans le monde entier. En Europe, le programme le plus ambitieux dans ce domaine est le projet Promethus, dans lequel nous sommes impliqués. L'objectif de ces programmes de recherche est de concevoir des systèmes intelligents embarqués dont le rôle est d'aider, de conseiller, voire à long terme de remplacer l'Homme dans certaines manoeuvres. La conception de systèmes de détection d'obstacles fait partie des projets les plus ambitieux entrepris dans le cadre de ces programmes. Les recherches effectuées pour la réalisation de tels systèmes ont permis de développer deux grands types d'approches : l'approche passive et l'approche active.

Le travail confié au Centre d'Automatique de Lille, dans le cadre du programme Promethus, était destiné à la réalisation d'un système de détection d'obstacles sur route, basé sur une approche par stéréovision passive.

La première partie de ce travail, réalisée par J.-L. Bruyelle dans le cadre de sa thèse, était axée sur la conception d'un capteur stéréoscopique, sa mise en oeuvre et sa validation [Bruyelle 94]. A partir d'une réflexion sur la nécessité de détecter les obstacles à l'avant d'un véhicule routier et après avoir étudié les différentes méthodes de détection existantes, il a conçu un nouveau capteur stéréoscopique composé de deux caméras linéaires, baptisé stéréoscope linéaire. L'utilisation de ce type de caméras à la place des caméras matricielles traditionnelles permet de diminuer très sensiblement la quantité de données à traiter, et d'apporter ainsi un gain de temps très substantiel au niveau de l'analyse des couples d'images stéréoscopiques tout en offrant une résolution horizontale nettement supérieure à celle des caméras matricielles. Afin de rendre ce capteur stéréoscopique opérationnel, il a été soumis à une procédure de calibrage spécifique. Celle-ci permet de lui donner les caractéristiques géométriques choisies à l'avance en fonction de l'application envisagée et, surtout, d'assurer la fusion des plans de visée des deux caméras linéaires qui le composent.

La deuxième partie des recherches était consacrée au traitement et analyse des images fournies par le stéréoscope linéaire. Le problème clef était celui de la mise en correspondance des couples d'images stéréoscopiques fournies par les caméras gauche et droite du capteur stéréoscopique. Pour résoudre ce problème, J.-C Burie a proposé une méthode basée sur une analyse photométrique des images à traiter [Burie 95]. Il s'agit d'une méthode locale qui

recherche les appariements couple par couple en explorant séquentiellement les images linéaires. Dans sa version la plus sophistiquée, cette méthode fournit de bons résultats grâce à une approche multi-résolution. Elle présente toutefois quelques limitations qui peuvent parfois mener à des erreurs de mise en correspondance et à des appariements manqués, ce qui est grave, surtout lorsqu'il s'agit de détecter les obstacles devant un véhicule routier.

Afin d'améliorer le processus de mise en correspondance des images linéaires, nous avons proposé une méthode basée sur l'emploi de réseaux neuronaux.

La première étape de l'analyse des couples d'images stéréoscopiques est l'extraction des primitives, ou indices visuels, représentant les objets présents dans la scène observée. Contrairement à l'utilisation d'images matricielles, l'emploi des images linéaires ne permet pas un choix de primitives. Seuls les points de type contour sont envisageables. Ces primitives sont bien adaptées aux images linéaires car elles correspondent aux contours des objets présents dans les images. Pour mettre en évidence ces indices visuels, nous avons fait appel à l'opérateur proposé par Deriche. Nous avons ensuite appliqué une méthode d'extraction afin de ne conserver que les contours significatifs. A l'issue du traitement de chaque couple d'images stéréoscopiques, deux listes de points de contour sont établies et transmises au module de mise en correspondance.

La seconde phase est destinée à appairer les primitives extraites des images gauche et droite. La méthode que nous avons développée est basée sur l'utilisation des réseaux neuronaux. Dans un premier temps, le problème de l'appariement est exprimé en termes de minimisation d'une fonction de coût construite à partir d'un certain nombre de contraintes stéréoscopiques. Un réseau de neurones de Hopfield est ensuite mis au point pour assurer le processus de minimisation. Le principal avantage de cette méthode est que la recherche des appariements est effectuée d'une manière globale. De plus, elle n'utilise aucune mesure de ressemblance photométrique dont l'application est mal adaptée lorsqu'il s'agit d'appairer des images où se produisent des occlusions.

Nous avons proposé trois algorithmes de mise en correspondance. Les résultats fournis par les deux premiers sont meilleurs que ceux obtenus avec l'algorithme proposé par Burie. Par ailleurs, notre méthode de mise en correspondance s'est montrée également fiable et robuste lorsqu'il s'agit d'analyser des images de scènes complexes. Cependant, la cadence de traitement obtenue ne permet pas d'envisager une application temps réel telle que la détection d'obstacles à

l'avant des véhicules routiers, à moins de faire appel à une architecture dédiée comme il en existe pour la majorité des réseaux neuronaux [Choi 91][Satanarayama 92][Sheu 92].

Afin d'améliorer le temps des traitements, nous avons proposé un troisième algorithme. Celui-ci donne des résultats un peu dégradés par rapport aux deux premiers algorithmes, mais comparables par rapport à ceux obtenus par la méthode de Burie. Avec cet algorithme, nous avons atteint pratiquement la même cadence de traitement que celle obtenue par Burie.

Nous travaillons actuellement sur le suivi temporel des objets détectés afin d'éliminer les fausses détections. Ce suivi temporel, que nous abordons également avec une approche neuromimétique, permettra aussi de déterminer les vitesses de déplacement des objets par rapport au stéréoscope.

Le système de détection que le Centre d'Automatique de Lille a développé a été intégré au véhicule expérimental Prolab 2. Ce véhicule, réalisé par le groupe ProArt France dans le cadre du programme Prometheus, a bénéficié du soutien des constructeurs automobiles P.S.A.

Le stéréoscope linéaire n'est évidemment pas limité à la seule détection d'obstacles sur route. Il peut être aussi utilisé pour la détection d'obstacles à l'avant des véhicules de transports guidés, tels que les trains ou les métros. Il peut être également utilisé en robotique mobile, pour le guidage et/ou la détection d'obstacles.

Actuellement, les recherches sont orientées vers la fusion multisensorielle, ou multicateurs, afin d'améliorer les systèmes de détection d'obstacles. Cette approche consiste à faire coopérer plusieurs capteurs passifs et actifs afin de surveiller une zone plus importante et/ou vérifier la validité des résultats obtenus par chaque capteur. J. Pontois, chercheur au Centre d'Automatique de Lille, travaille actuellement sur la fusion des données provenant du stéréoscope linéaire avec celles issues d'un radar hyperfréquence développé par L'I.E.M.N.<sup>1</sup> en collaboration avec le L.E.O.S.T.<sup>2</sup> de L'I.N.R.E.T.S.<sup>3</sup> et avec le soutien de la Région Nord/Pas de Calais.

La méthode neuronale de minimisation de fonctions de coût que nous avons développée a été aussi appliquée au problème de l'appariement stéréoscopique d'images matricielles en utilisant des contours actifs comme primitives [Ruichek 96]. Ce travail a été effectué au Centre

---

<sup>1</sup> Institut d'Electronique et Micro-électronique du Nord.

<sup>2</sup> Laboratoire d'Electronique, d'Ondes et des Signaux pour les Transports.

<sup>3</sup> Institut National de Recherche sur les Transports et leur Sécurité.

d'Automatique de Lille en collaboration avec C. Vieren. Comme dans le cas de l'appariement des images linéaires, une fonction de coût est construite et ensuite minimisée par un réseau de neurones de Hopfield. Cette approche a été également utilisée pour l'estimation du mouvement dans une séquence d'images. Le problème était celui de la mise en correspondance des images successives de la séquence [Vannoorenberghe 96]. Ce travail a été effectué en collaboration avec P. Vannoorenberghe, chercheur au L.I.S.I.R.<sup>1</sup>.

---

<sup>1</sup> Laboratoire d'Instrumentation du Signal Image et des Réseaux.

# **Annexe 1**

Nous allons montrer dans cette annexe que la contrainte d'ordre est presque toujours vérifiée dans le cadre de notre application.

Nous rappelons que la contrainte d'ordre stipule que l'ordre des primitives est le même dans chaque image du couple stéréoscopique (Cf. Figure 1).

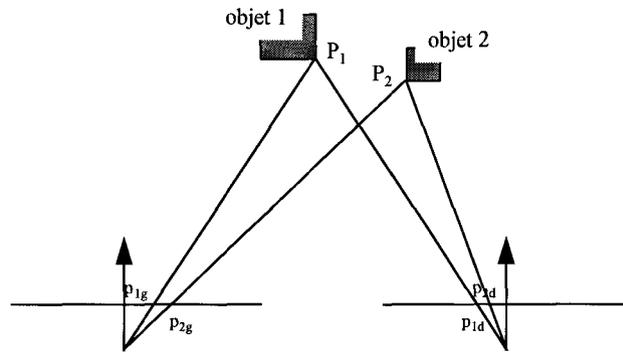


Figure 1. La contrainte d'ordre est respectée.

Comme nous pouvons le constater sur la figure 2, cette contrainte est transgressée dans le cas où l'objet 1 (resp. l'objet 2) est situé dans le champ défini par le triangle  $P_2p_{g2}P_{d2}$  (resp.  $P_1p_{g1}P_{d1}$ ).

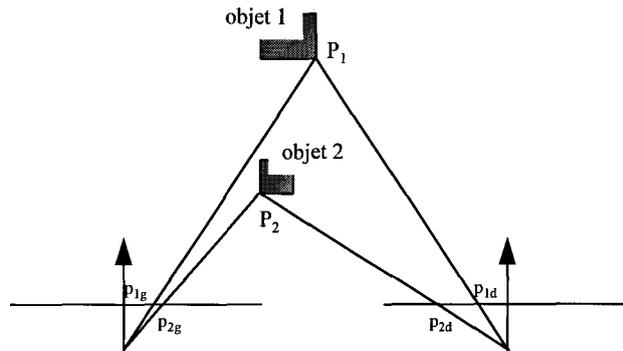


Figure 2. La contrainte d'ordre est transgressée.

Dans le cadre de notre application, la plupart des objets sont détectés entre 20 m et 50 m à l'avant du véhicule (Cf. Figure 3). Etant donné les triangles semblables  $P_1AB$  et  $P_1F_dF_g$ , on a :

$$\frac{AB}{z - 20} = \frac{E}{z},$$

ou encore :

$$AB = \frac{E \cdot (z - 20)}{z},$$

où  $F_g$  et  $F_d$  sont respectivement les centres optiques des caméras gauche et droite.  $E$  est l'entre-axe entre les deux caméras.

De la même manière pour les triangles  $P_2CD$  et  $P_2F_dF_g$ , on obtient :

$$CD = \frac{E \cdot (z - 20)}{z}$$

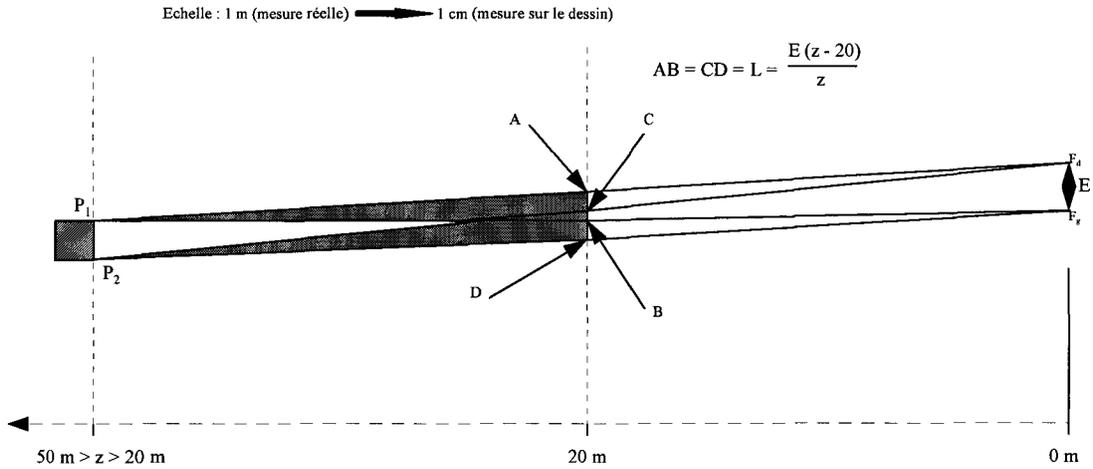


Figure 3. Conditions d'application de la contrainte d'ordre.

Comme  $E = 1 m$ , la longueur  $L = AB = CD$  des bases des triangles  $P_1AB$  et  $P_2CD$ , qui délimitent les champs où la contrainte d'ordre est transgressée, est comprise entre  $0 cm$  et  $60 cm$ , lorsque la distance  $z$  de l'objet "carré" varie entre  $20 m$  et  $50 m$  (Cf. Figure 3). Or, les véhicules, même les piétons, ont généralement une largeur supérieure à  $60 cm$ . Ainsi, on va se trouver dans des situations où les objets vont s'occulter entre eux (Cf. Figure 4). Dans ce cas, le problème de l'ordre des primitives ne se pose plus.

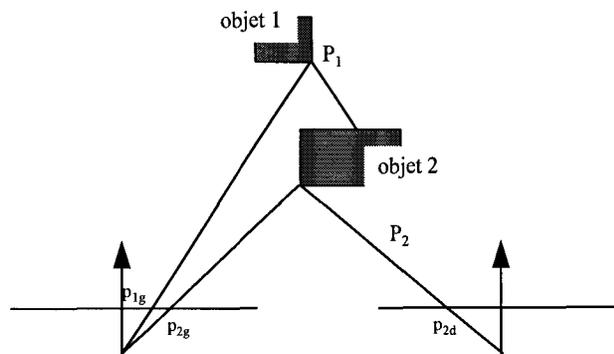


Figure 4. Dans une situation d'occlusion, le problème de l'ordre ne se pose plus.

## **Annexe 2**

L'objectif est de prouver que la fonction de coût  $H_p$  :

$$\begin{aligned}
H_p = & K_{position} \sum_{g \in G} \sum_{d \in D} P_{gd} E_{gd} + K_{signe} \sum_{g \in G} \sum_{d \in D} G_{gd} E_{gd} \\
& + K_{unicité} \sum_{g \in G} \left( 1 - \sum_{d \in D} E_{gd} \right)^2 + K_{unicité} \sum_{d \in D} \left( 1 - \sum_{g \in G} E_{gd} \right)^2 \\
& + K_{ordre} \sum_{g \in G} \sum_{d \in D} \sum_{\substack{g' \in G \\ g' \neq g}} \sum_{\substack{d' \in D \\ d' \neq d}} O_{gdg'd'} E_{gd} E_{g'd'} \\
& - K_{continuité} \sum_{g \in G} \sum_{d \in D} \sum_{\substack{g' \in G \\ g' \neq g}} \sum_{\substack{d' \in D \\ d' \neq d}} C_{gdg'd'} E_{gd} E_{g'd'}
\end{aligned} \tag{Equation 1}$$

est équivalente à la fonction d'énergie du réseau  $H_r$  :

$$H_r = -\frac{1}{2} \sum_{g \in G} \sum_{d \in D} \sum_{g' \in G} \sum_{d' \in D} W_{gdg'd'} E_{gd} E_{g'd'} - \sum_{g \in G} \sum_{d \in D} I_{gd} E_{gd} \tag{Equation 2}$$

Simplifions le troisième et le quatrième termes de la fonction de coût  $H_p$  (Cf. Equation 1) :

$$\begin{aligned}
\sum_{g \in G} \left( 1 - \sum_{d \in D} E_{gd} \right)^2 &= \sum_{g \in G} (1) + \sum_{g \in G} \left( \sum_{d \in D} E_{gd} \right)^2 - 2 \sum_{g \in G} \sum_{d \in D} E_{gd} \\
&= \sum_{g \in G} (1) + \sum_{g \in G} \left( \sum_{d \in D} \sum_{d' \in D} E_{gd} E_{gd'} \right) - 2 \sum_{g \in G} \sum_{d \in D} E_{gd} \\
&= N_G + \sum_{g \in G} \sum_{d \in D} \sum_{d' \in D} E_{gd} E_{gd'} - 2 \sum_{g \in G} \sum_{d \in D} E_{gd}
\end{aligned} \tag{Equation 3}$$

Or  $E_{gd} E_{gd'} = E_{gd} E_{g'd'} \delta_{gg'}$ . Ainsi, en substituant dans (Equation 3), nous obtenons :

$$\sum_{g \in G} \left( 1 - \sum_{d \in D} E_{gd} \right)^2 = N_G + \sum_{g \in G} \sum_{d \in D} \sum_{g' \in G} \sum_{d' \in D} E_{gd} E_{g'd'} \delta_{gg'} - 2 \sum_{g \in G} \sum_{d \in D} E_{gd}$$

De la même façon, le quatrième terme peut être écrit comme suit :

$$\sum_{d \in D} \left( 1 - \sum_{g \in G} E_{gd} \right)^2 = N_D + \sum_{g \in G} \sum_{d \in D} \sum_{g' \in G} \sum_{d' \in D} E_{gd} E_{g'd'} \delta_{dd'} - 2 \sum_{g \in G} \sum_{d \in D} E_{gd}$$

La fonction de coût  $H_p$  devient alors :

$$\begin{aligned}
H_p &= K_{unicité}(N_D + N_G) \\
&+ \sum_{g \in G} \sum_{d \in D} \sum_{g' \in G} \sum_{d' \in D} [K_{unicité}\delta_{gg'} + K_{unicité}\delta_{dd'} \\
&\quad + K_{ordre}(1 - \delta_{gg'})(1 - \delta_{dd'})O_{gdg'd'} \\
&\quad - K_{continuité}(1 - \delta_{gg'})(1 - \delta_{dd'})C_{gdg'd'}] E_{gd} E_{g'd'} \\
&+ \sum_{g \in G} \sum_{d \in D} (K_{position}P_{gd} + K_{signe}G_{gd} - 4K_{unicité})E_{gd}
\end{aligned}$$

où encore :

$$\begin{aligned}
H'_p &= \frac{H_p - K_{unicité}(N_G + N_D)}{2} = \\
&- \frac{1}{2} \sum_{g \in G} \sum_{d \in D} \sum_{g' \in G} \sum_{d' \in D} [-K_{unicité}\delta_{gg'} - K_{unicité}\delta_{dd'} \\
&\quad - K_{ordre}(1 - \delta_{gg'})(1 - \delta_{dd'})O_{gdg'd'} \\
&\quad + K_{continuité}(1 - \delta_{gg'})(1 - \delta_{dd'})C_{gdg'd'}] E_{gd} E_{g'd'} \\
&- \sum_{g \in G} \sum_{d \in D} \frac{-K_{position}P_{gd} - K_{signe}G_{gd} + 4K_{unicité}}{2} E_{gd}
\end{aligned}$$

Le terme  $-K_{unicité} \frac{N_G + N_D}{2}$  étant une constante, le problème revient à minimiser la

fonction de coût  $H'_p$  qui est équivalente à la fonction d'énergie du réseau (Cf. Equation 2). Ainsi, pour déterminer les poids des connexions entre les neurones et leurs entrées externes, il suffit d'identifier les deux expressions des fonctions  $H_r$  et  $H'_p$ .

Pour une raison de simplification, nous considérons que le poids de connexion entre un neurone et lui-même est nul. En fait, dans le cas de notre problème, les contraintes globales sont appliquées entre les paires de points de contour distinctes. Il faut donc annuler le terme  $-K_{unicité}\delta_{gg'} - K_{unicité}\delta_{dd'}$  de la fonction  $H'_p$  lorsque  $g = g'$  et  $d = d'$ . Ainsi, les poids des connexions entre les neurones du réseau et leurs entrées externes s'expriment sous la forme :

$$\begin{aligned}W_{gdg'd'} = & -K_{unicité} [\delta_{gg'}(1-\delta_{dd'}) - D\delta_{dd'}(1-\delta_{gg'})] \\ & - K_{ordre} O_{gdg'd'}(1-\delta_{gg'})(1-\delta_{dd'}) \\ & + K_{continuité} C_{gdg'd'}(1-\delta_{gg'})(1-\delta_{dd'})\end{aligned}$$

$$I_{gd} = 2K_{unicité} - \frac{K_{position}P_{gd} + K_{gradient}G_{gd}}{2}$$

## **Publications dans le cadre de la thèse**

Y. Ruichek et J. -G. Postaire "*Extraction of Depth Information from Stereo Pairs of Linear images Using a Neural Matching Algorithm*". Proc. of Int. Conf. on **Engineering Applications of Neural Networks**, Helsinki, Finlande, p. 25-32, 21-23 Août 1995.

Y. Ruichek et J. -G. Postaire "*Une Approche Neuronale pour la Stéréovision Linéaire*". Colloque **GRETSI 95**, Juan les Pins, France, Tome 2, p. 889-892, 18-22 Septembre 1995.

Y. Ruichek et J. -G. Postaire "*A Neural Implementation for High Speed Processing in Linear Stereo Vision*". Proc. of Int. Conf. on **Systems, Man and Cybernetics**, Vancouver, Canada, Vol. 5, p. 3902-3907, 22-25 Octobre 1995.

Y. Ruichek et J. -G. Postaire "*Real-Time Neural Vision for Obstacle Detection Using Linear Cameras*". Proc. of Int. Symp. **Intelligent Vehicles '95**, Detroit, U.S.A, p. 524-528, 25-26 Septembre 1995.

Y. Ruichek et J. -G. Postaire "*A Connexionist Approach for the Correspondence Problem in Linear Stereo Vision*". Proc. of **ASIAN Conference on Computer Vision**, Singapour, Vol. 3, p. 427-431, 5-8 Décembre 1995.

Y. Ruichek et J. -G. Postaire "*3-D Environment Reconstruction in front of a Car with a Stereo Vision System using a Hopfield Neural Network*". Proc. of Int. Conf. on **Road Vehicle Automation**, Bolton, Grande Bretagne, 11-13 Septembre 1995.

Y. Ruichek, J. -G. Postaire, L. Macaire et J. -C. Burie "*Implémentation Neuronale pour la Mise en Correspondance de Primitives en Stéréovision Linéaire*". Proc. of Int. Conf. on **Communication Signals and Systems**, Rabat, Maroc, Vol. 2, p. 574-582, 09-11 Octobre 1995.

Y. Ruichek et J.-G. Postaire "*A Neural Matching Algorithm for 3-D Reconstruction from Stereo Pairs of Linear Images*". **Pattern Recognition Letters** 4(17), p. 387-398, 1996.

Y. Ruichek, J.-G. Postaire et J.-L. Bruyelle "*A Neural Approach for Obstacle Detection with a Linear Stereoscopic Sensor*". **Mathematical and Computer Modelling** (sous presse).

Y. Ruichek et J.-G. Postaire "*A New Neural Real-Time Implementation for Obstacle Detection using Linear Stereo Vision*". Article en cours de révision pour publication dans le journal **Real Time Imaging**.

# **Bibliographie**

- [Abdi 94] Abdi H. "*Les réseaux de neurones*", Ed. Presses Universitaires de Grenoble, 1994.
- [Abe 95] Abe S. et Gee A. "*Global convergence of the Hopfield neural network with nonzero diagonal elements*", IEEE Trans. on Circuits and Systems-II : Analog and Digital Signal Processing, Vol. 42, n° 1, p. 39-45; Janvier 1995.
- [Adjouadi 94] Adjouadi M. et Candocia F. "*A stereo matching paradigm based on the Walsh transformation*", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. PAMI-16, n° 12, p. 1212-1218, Décembre 1994.
- [Aggarwal 75] Aggarwal J.K. "*Computer analysis of moving polygonal images*", IEEE Computer, Vol. C-24, n° 10, p. 966-976, Octobre 1975.
- [Aggarwal 81] Aggarwal J.K. "*Correspondence processes in dynamic scene analysis*", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol PAMI-69, n° 5, Mai 1981.
- [Aggarwal 85] Aggarwal J.K. et Duda R.O "*Computer analysis of moving polygonal images*", Digital Image Processing and Analysis, Vol. 2, p. 445-455, 1985.
- [Aiyer 90] Aiyer S.V.B., Niranjan M. et Fallside F. "*A theoretical investigation into performance of the Hopfield model*", IEEE Trans. on Neural Networks, Vol. 1, n° 2, p. 204-215, Juin 1990.
- [Alizon 90] Alizon J., Gallice J., Trassoudaine L. et Treuillet S. "*Multi-sensory data fusion for obstacle detection and tracking on motorway*", Proc. ProArt workshop on Vision, Sophia Antipolis, p. 179-187, 19-20 Avril 1990.
- [Amari 71] Amari S. "*Characteristics of randomly connected threshold-element networks and network system*", Proceedings IEEE, Vol. 59, p. 35-47, Janvier 1971.
- [Amari 72] Amari S. "*Learning patterns and pattern sequences by self-organizing nets of threshold elements*", IEEE Trans. on Computer, Vol. C-21, p. 1197-1206, Novembre 1972.
- [Ambellouis 95] Ambellouis S. et Cabestaing F. "*Analyse du mouvement par réseau de neurones à temps de propagation*", Proceedings Intern. AMSE Confer. on Communications, Signals and Systems, Rabat, Maroc, Vol. 2, p. 554-559, 09-11 Octobre 1995.
- [Amit 85] Amit D.J., Gutfreund H. et Sompolinsky H. "*Spin-glass models of neural networks*", Physical Review A, Vol. 32, n° 2, p. 1007-1018, 1985.
- [Amit 89] Amit D.J. "*Modeling brain function, the world of attractor dynamics*", Cambridge University Press, 1989.

- [Anderson 88] Anderson J.A. et Rosenfeld. "**Neurocomputing: Foundations of research**", MIT Press, Cambridge MA, 1988.
- [Ardizzone 90] Ardizzone E., Chella A. et Sorbello F. "**Application of the novelty filter to the motion analysis**", Int. Neural Network Conference, Paris, France, p. 46-49, 09-13 Juillet 1990.
- [Asada 86] Asada H. et Brady M. "**The curvative primal sketch**", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. PAMI-8, n° 1, p. 2-14, Janvier 1986.
- [Ayache 85] Ayache N. et Faferjon B. "**Un algorithme rapide de stéréoscopie passive utilisant la prédiction et la vérification récursive d'hypothèses**", 5<sup>ème</sup> Congrès AFCET de Reconnaissance des Formes et Intelligence Artificielle, Grenoble, p. 1217-1232, Novembre 1985.
- [Ayache 89] Ayache N. "**Vision stéréoscopique et perception multisensorielle - Application à la robotique mobile**", Ed. InterEditions, Paris, 1989.
- [Bailey 88] Bailey D.L., Thompson D.M. et Feinstein J.L. "**Options tradings using neural networks**", NeuroNîmes, Novembre 1988.
- [Baker 81] Baker H.H. et Binford T.O. "**Depth from edge and intensity based stereo**", Proc. 7<sup>th</sup> Int. Joint Conference on Artificial Intelligence, Vancouver, Canada, p. 631-636, Août 1981.
- [Ballard 82] Ballard D.H. et Brown C.M. "**Computer vision**", Prentice Hall Inc., Englewood Cliffs, New Jersey, 1982.
- [Ballard 83] Ballard D.H. et Kimball O.A. "**Rigid body motion from depth and optical flow**", Computer Vision Graphics and Image Processing, Vol. 22, n° 1, p. 95-115, Avril 1983.
- [Barto 83] Barto A.G., Sutton R.S. et Anderson C.W. "**Neuronlike adaptive elements that can solve difficult learning control problems**", IEEE Trans. on Systems Man and Cybernetics, Vol. SMC-13, n° 5, Septembre-Octobre 1983.
- [Basso 92] Basso A. et Kunt M. "**Autoassociative neural networks for image compression**", European Transactions on Telecommunications and Related Technologies, Vol. 3, n° 6, p. 593-598, 1992.
- [Behringer 92] Behringer R., Holt V.V. et Dickmanns D. "**Road and relative ego-state recognition**", IEEE Intelligent Vehicles Symposium '92, Detroit, Etats-Unis, p. 385-390, 29 Juin-01 Juillet 1992.
- [Ben-Hanan 92] Ben-Hanan U., Peleg K. et Gutman P. "**Classification of fruits by a Boltzmann Perceptron neural network**", Automatica, Vol. 28, n° 5, p. 961-968, 1992.

- [Benamrane 94] Benamrane N. "*Contribution à la vision stéréoscopique par mise en correspondance de régions*", Thèse de Doctorat, Université de Valenciennes et du Hainaut Cambresis, France, 24 Janvier 1994.
- [Benard 84] Benard M. "*Automatic stereophotogrammetry : a method based on feature detection and dynamic programming*", Photogrammetria, n° 39, p. 169-181, 1984.
- [Bers 80] Bers K.H. "*Object detection in image sequences*", Proc. IEEE 5<sup>th</sup> Int. Conf. on Pattern Recognition, 1980.
- [Bishop 95] Bishop M. "*Neural networks for pattern recognition*", CLARENDON PRESS, OXFORD, 1995.
- [Blosseville 87] Blosseville J.M. et Coll. "*TITAN: Système de mesure de trafic par analyse d'image*", Rapport interne INRETS, Arcueil, 1987.
- [Bolles 82] Bolles R.C. et Cain R.A. "*Recognizing and locating partially visible objects*", SRI Technical note 262, Juin 1982.
- [Bouthemy 86] Bouthemy P. "*Un nouveau schéma d'estimation du champ de vitesses sur les contours dans une séquence d'images*", 2<sup>ème</sup> Colloque Image CESTA, Nice, p. 237-245, Avril 1986.
- [Brice 70] Brice C. et Fenema C. "*Scene analysis using regions*", Artificial Intelligence, n° 1, p. 205-226, 1970.
- [Briot 79] Briot M., Talou J.-C. et Bauzil G. "*Le système de perception du robot mobile autonome Hilare*", Proc. 2<sup>ème</sup> congrès AFCET/INRIA de Reconnaissance des Formes et Intelligence Artificielle, Toulouse, 12-14 Septembre 1979.
- [Broughton 91] Broughton J., Colk H., Dryselius B., Fontaine H., Gastaldi G., Hydén C., Klöckner J.-H., Neumann L., Risser R. et Velding I. "*The Pro-Gen traffic safety checklist*", Rapport PROMETHEUS, Stuttgart, 1991.
- [Bruss 83] Bruss A.R. et Horn B.K.P. "*Passive navigation*", Computer Vision Graphics and Image Processing, Vol. 21, n° 1, p. 3-20, 1983.
- [Bruyelle 94] Bruyelle J.-L. "*Conception et réalisation d'un dispositif de prise de vue stéréoscopique linéaire - Application à la détection d'obstacles à l'avant des véhicules routiers*", Thèse de Doctorat, Université des Sciences et Technologies de Lille, France, 2 Décembre 1994.
- [Bulsari 95] Bulsari A.B. et Kallio S. "*Engineering applications of artificial neural networks*", Proceedings of EANN'95, Otaniemi, Finlande, 21-23 Août 1995.

- [Bulsari 96] Bulsari A.B., Kallio S. et Tsaptsinos D. "***Solving engineering problems with neural networks***", Proceedings of the Intl. Conf. on Engineering Applications of Neural Networks, London, England, 17-19 June 1996.
- [Burie 95] Burie J.-C. "***Mise en correspondance d'images linéaires stéréoscopiques - Application à la détection d'obstacles à l'avant des véhicules routiers***", Thèse de Doctorat, Université des Sciences et Technologies de Lille, France, 28 Septembre 1995.
- [Burnod 88] Burnod Y. "***An adaptive neural network: the cerebral cortex***", Collection biologie théorique, Ed. Masson, 1988.
- [Bye 93] Bye S.J. et Adams A. "***Neural network paradigm for visual pattern classification***", 3<sup>th</sup> Int. Conf. on Artificial Neural Networks, Brighton Conference Centre, UK, 25-27 Mai 1993.
- [Cabestaing 92] Cabestaing F. "***Détection de contours en mouvement dans une séquence d'images: conception et réalisation d'un processeur câblé temps-réel***", Thèse de Doctorat, Université des Sciences et Technologies de Lille, France, Janvier 1992.
- [Cafforio 76] Cafforio C. et Rocca F. "***Methods for measuring small displacements of television images***", IEEE Trans. on Inf. Theory, Vol. IT-22, p. 573-579, 1976.
- [Cai 92] Cai D.-F. et Zhou M. "***Adaptative image compression based on backpropagation neural networks***", Neural and Stochastic Methods in Image and Signal Processing, San Diego, USA, 20-23 Juillet 1992.
- [Canny 83] Canny J.-F. "***Finding edges and lines in images***", Tech. Rep. n° 720, Artificial Intelligence Laboratory, MIT, Cambridge, MA, USA, Juin 1983.
- [Canny 86] Canny J.-F. "***A computational approach to edge detection***", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 8, n° 6, p. 679-698, Novembre 1986.
- [Carlsson 90] Carlsson S. et Eklundh J.-O. "***Object detection using model based prediction and motion parallax***", Proc. Pro-Art Workshop on Vision, Sophia Antipolis, p. 139-145, 19-20 Avril 1990.
- [Casaent 92] Casaent D. et Telfer B. "***High capacity pattern recognition associative processors***", Neural Networks, Vol. 5, p. 687-698, 1992.
- [Castan 84] Castan S. et Shen J. "***A stereo algorithm taking into account the perspective distortions***", Proc. 7<sup>th</sup> ICPR, Montréal, Canada, p. 441-443, 1984.

- [Chebaro 91] Chebaro B., Massipailhes L. et Castan S. "**Description polygonale basée sur l'extraction d'ilots de confiance**", 8<sup>ème</sup> Congrès sur la Reconnaissance des formes et Intelligence Artificielle, Lyon, France, Vol. 2, p. 841-846, 25-29 Novembre 1991.
- [Cheng 80] Cheng N., Birk J.R. et Kelley R. "**Estimation workpiece pose using the feature points method**", IEEE, Vol. AC-25, n° 6, Décembre 1980.
- [Choi 91] Choi J., Sheu B.J. et Gowda S.M. "**Analog VLSI neural network implementations of hardware annealing and winner-take-all functions**", 34<sup>th</sup> Midwest Symp. Circuits and Systems, Monterey, CA, Mai 1991.
- [Chou 87] Chou T.C. et Kanatani K. "**Recovering 3D rigid motions without correspondence**", Proc. 1<sup>st</sup> Int. Conf. on Computer Vision, p. 534-538, 1987.
- [Chou 93] Chou J.-S., Chen C.-T. et Lin W.-C. "**Segmentation of dual-echo MR images using neural networks**", Proceedings of SPIE - The International Society for Optical Engineering, Society of Photo-Optical Instrumentation Engineers, Bellingham, WA, USA, 1898, p. 220-227, 1993.
- [Chung 94] Chung P.-C., Tsai C.-T., Chen E.-L. et Sun Y.-N. "**Polygonal approximation using a competitive Hopfield neural network**", Pattern Recognition, Vol. 27, n° 11, p. 1505-1512, 1994.
- [Claxton 87] Claxton P.R. et Kwok K.-Y. "**The use colour to segment and label images**", Proc. Alvey Vision Conf. p. 295-302, 1987.
- [Cochet 88] Cochet Y. et Paget G. "**Neural networks for images databases**", International Neural Networks Society, First Annual Meeting, Boston MA, 06-10 Septembre 1988.
- [Cocquerez 87a] Cocquerez J.P. et Gagalowicz A. "**Mise en correspondance de régions dans une paire d'images stéréo**", Proc. Colloque Image du MARI-CESTA, Paris, France, p. 165-173, Mai 1987.
- [Cocquerez 87b] Cocquerez J.P. et Monga O. "**Matching regions in stereovision**", Proc. 5<sup>th</sup> Scandinavian Conference on Image Analysis, Stockholm, p. 295-308, 1987.
- [Colle 90] Colle O. "**Vision stéréoscopique à l'aide de deux caméras linéaires : Application à la robotique mobile**", Thèse de Doctorat, Institut National des Sciences Appliquées de Lyon, France, 6 Novembre 1990.
- [Cruz 95] Cruz J.M., Pajares G. et Aranda J. "**A neural network model in stereovision matching**", Neural Networks, Vol. 8, n° 5, p. 805-813, 1995.

- [Davalò 89] Davalo E. et Naïm P. "*Des réseaux de neurones*", Edition Eyrolles, Paris, 1989.
- [Davis 79] Davis L.S., Wu Z. et Sun H. "*Contour based motion estimation*", Computer Graphics and Image Processing, Vol. 9, 1979.
- [Davis 81] Davis L.S. et Rosenfeld A. "*Cooperating processes for low-level vision : A survey*", Artificial Intelligence, Vol. 17, n° 1-3, p. 245-263, Août 1981.
- [Deloof 90] Deloof P., Haese N. et Branowski S. "*Recherche bibliographique sur les radars de prévention des collisions*", Recherche Transports Sécurité, n°26, p. 15-22, Juin 1990.
- [Deloof 94] Deloof P., Rolland P.-A. et Haese N. "*La prévention des collisions routières par radar anticollision micro-onde*", Synthèse INRETS, n° 25, Juillet 1994.
- [Deriche 87a] Deriche R. et Cocquerez J.P. "*Extraction de composants connexes basée sur une détection optimale des contours*", CESTA, Paris, 1987.
- [Deriche 87b] Deriche R. "*Using Canny's criteria to derive a recursively implemented optimal edge detector*", Int. Journal on Computer Vision, Vol. 1, n° 2, p. 167-187, Mai 1987.
- [Deriche 90] Deriche R. et Faugeras O. "*2-D curve matching using high curvature points : Application to stéréovision*", ISSN : CH2898-5/90/0000/0240\$01.00 © 1990 IEEE, p. 240-242, 1990.
- [Deriche 90b] Deriche R. "*Fast algorithms for low-level vision*", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 12, n° 1, p. 78-87, Janvier 1990.
- [Di Zenso 86] Di Zenso S. "*A note on the gradient of multi-image*", Computer Vision, Graphics and Image Processing, n° 33, p. 116-125, 1986.
- [Drive 89] Rapport Drive V1002. "*Study of the feasibility of a low cost radar front end*", FGM-CRESTA-USTL, 1989.
- [du Buf 90] du Buf J.M.H., Kardan M. et Spann M. "*Texture feature performance for image segmentation*", Pattern Recognition, Vol. 23, n° 3-4, p. 291-309, 1990.
- [Duvieubourg 93] Duvieubourg L., Carpentier T. et Postaire J.-G. "*Monitoring changes in length with a linear CCD camera in mechanics of deformable solids*", Proc. 5<sup>th</sup> Int. Congres on Condition Monitoring and Diagnostic Engineering Management, Bristol, Juillet 1993.

- [Elhor 96] ElHor N., Hamad D. et Postaire J.-G. "**Classification supervisée par réseau de neurones modulaire**", JNAEETSA 96, Marrakech, Maroc, p. 392-397, 24-25 Octobre 1996.
- [Enkelmann 90] Enkelmann W. "**Obstacle detection by evaluation of optical flow fields**", Proc. ProArt workshop on Vision, Sophia Antipolis, p. 146-155, 19-20 Avril 1990.
- [Fang 92] FANG W.-C., Sheu B.J., Chen O.T.-C. et Choi J. "**A VLSI neural processor for image data compression using self-organization networks**", IEEE Trans. on Neural Networks, Vol. 3, n° 3, p. 506-518, 1992.
- [Faugeras 88] Faugeras O.D. "**Quelques pas vers la vision artificielle en trois dimensions**", Technique et Science Informatiques, Vol. 7, n° 6, p. 547-590, 1988.
- [Faugeras 93] Faugeras O.D. et al. "**Real time correlation-based stereo : algorithm, implementations and applications**", Rapport de recherche n° 2013, INRIA Sophia-Antipolis, Août 1993.
- [Fennema 79] Fennema C.L. et Thompson W.B. "**Velocity determination in scenes containing several objects**", Computer Graphics and Image Processing, Vol. 9, p. 301-315, Avril 1979.
- [Flocchini 92] Flocchini P., Gardin F., Mauri G., Pensini M.P. et Stofella P. "**Combining image processing operators and neural networks in a face recognition system**", Int. Journal of Pattern Recognition and Artificial Intelligence, Vol. 6, n° 2-3, p. 447-467, 1992.
- [Freeman 79] Freeman M. "**Curved surface representation utilizing data extracted from multiple photographic image**", Workshop on the representation of 3D objects, Mai 1979.
- [Freeman 92] Freeman J.A. et Skapura D.M. "**Neural networks: Algorithms, applications and programming techniques**", Addison-Wesley, 1992.
- [Freixinho 95] Freixinho A.P., Salazar F.A., Vellasco M. et Pacheco M.A. "**Handwritten digits classification using neural networks**", Proceedings of the Intl. Conf. EANN'95, Otaniemi, Finland, p. 71-74, 21-23 August 1995.
- [Fukushima 75a] Fukushima K. "**Cognitron : a self-organizing multilayered neural network**", Biological Cybernetics, 20, p. 121-136, 1975.
- [Fukushima 75b] Fukushima F., Miyake S. et Ito T. "**Neocognitron : A neural network model for a mechanism of visual pattern recognition**", IEEE Trans. on Systems Man and Cybernetics, Vol. SMC-13, p. 826-834, 1975.

- [Fukushima 84] Fukushima K. "*A hierarchical neural network model for associative memory*", Biological Cybernetics, 50, p. 105-113, 1984.
- [Funabiki 93] Funabiki N., Takefuji Y. et Lee K.C. "*Comparisons of seven neural network models on traffic control problems in multistage interconnection networks*", IEEE Trans. on Computers, Vol. 42, n° 4, p. 497-501, Avril 1993.
- [Gallice 91] de Saint Blancard M., Gallice J. et Alizon J. "*Procédé de détection d'obstacles présents devant un véhicule automobile et dispositif pour la mise en oeuvre d'un tel procédé*", Dépôt français n° 9105628 du 7 Mai 1991, Dépôt européen n° 92400971 du 7 Avril 1992, Dépôt USA n° 07/879281 du 7 Mai 1992.
- [Gallice 94a] Trassoudaine L., Checchin P., Gallice J., Collange F. et Alizon J. "*A multisensory system for road obstacle detection and tracking*", Proc. IMACS Int. Symposium on Signal Processing Robotics and Neural Networks, Lille, France, p. 7-10, Avril 1994.
- [Gallice 94b] Martinet P., Gallice J., Montagne E. et Alizon J. "*Real time 3D location of a car from three characteristic points observed in a video image*", International Federation of Automatic Control, Proc. 7<sup>th</sup> Symposium on Transportation Systems: Theory and Application of Advanced technology, Tianjin, Chine, p. 385-390, 24-26 Août 1994.
- [Gaudiano 94] Gaudiano P., Zalama E. et Coronado J. "*A real-time unsupervised neural network for the control of a mobile robot*", Proceedings of IEEE Int. Conf. on Neural Networks, Vol. 5, p. 2848-2853, 28 Juin-02 Juillet 1994.
- [Gennery 80] Gennery D.B. "*Modelling the environment of an exploring vehicle by means of stereo vision*", PhD thesis, Stanford University, Juin 1980.
- [Gish 90] Gish S.L. et Blanz W.E. "*Comparing the performance of connectionist and statistical classifiers on an image segmentation problem*", Advances in Neural Information Processing Systems 2, Morgan Kaufmann Publishers, Lippmann R.P., Moody J.E. et Touretzky D.S. (eds.), San Mateo, California, 1990.
- [Glover 93] Glover F. et al. "*Modern heuristic techniques for combinatorial problems*", Black-well Scientific Publications, Oxford, U.K., 1993.
- [Graefe 90] Graefe V., Solder U. et Regensburger U. "*Recognition of objects on highways*", Proc. ProArt workshop on Vision, Sophia Antipolis, p. 133-138, 19-20 Avril 1990.
- [Graf 92] Graf H.P., Nohl C.R. et Ben J. "*Image segmentation with networks of variable scales*", Advances in Neural Network Information Processing Systems 4, Moody J.E., Hanson S.J. et Lippmann R.P. (eds.), Morgan Kaufmann Publishers, San Mateo, California, 1992.

- [Grimson 81] Grimson W.E.L. "*From images to surfaces*", MIT Press, Cambridge, Etats-Unis, 1981.
- [Grossberg 88] Grossberg S. "*Neural network and natural intelligence*", MIT Press, 1988.
- [Hancock 90] Hancock E.R. et Kittler J. "*Discrete relaxation*", Pattern Recognition, Vol. 23, n° 7, p. 711-733, 1990.
- [Hannah 84] Hannah M.J. "*Description of SRI's baseline stereo system*", Technical Report 342, SRI International, Octobre 1984.
- [Hannah 89] Hannah M.J. "*A system for digital stereo image matching*", Photogrammetric Engineering and Remote Sensing, Vol. 55, n° 12, p. 1765-1770, Decembre 1989.
- [Hanssens 92] Hanssens E., Verleysen M. et Legat J.-D. "*A dedicated neural network for visual motion detection*", 5<sup>th</sup> Int. Conf. on Neural Networks and their Applications, Nîme, France, 02-06 Novembre 1992.
- [Haskell 74] Haskell B.G. "*Frame to frame coding of television pictures using two-dimensional Fourier transform*", IEEE Trans. on Information Theory, IT-20, p. 119-120, 1974.
- [Hattori 92] Hattori A., Hosaka A., Taniguchi M. et Nakano E. "*Driving control system for autonomous vehicle using multiple observed point information*", IEEE Intelligent Vehicles Symposium '92, Detroit, Etats-Unis, p. 207-212, 29 Juin-01 Juillet 1992.
- [Haykin 94] Haykin S. "*Neural networks : a comprehensive foundation*", Macmillan College Publishing Company, Inc., 1994.
- [Hebb 49] Hebb D.O. "*The organization of the behaviour*", Wiley, New York, 1949.
- [Hecht-Nielsen 90] Hecht-Nielsen R. "*Neurocomputing*", HNC, Inc. and University of California, San Diego, Addison-Wesley Publishing Company, 1990.
- [Heikkonen 95] Heikkonen J. et Mäntynen M. "*Digit recognition on pulp bales*", Proceedings of the Intl. Conf. EANN'95, Otaniemi, Finland, p. 75-78, 21-23 August 1995.
- [Hérault 91] Hérault L. "*Réseaux de neurones récurrents pour l'optimisation combinatoire : Application à la théorie des graphes et à la vision par ordinateur*", Thèse de Doctorat, Institut National Polytechnique de Grenoble, 26 Février 1991.
- [Hérault 94] Hérault J. et Jutten C. "*Réseaux neuronaux et traitement du signal*", Edition Hermès, Paris, 1994.

- [Hinton 81] Hinton G.E. et Anderson J.A. (eds.). "*Parallel models of associative memory*", Hillsdale, Erlbaum, 1981.
- [Hinton 84] Hinton G., Sejnowski T. et Ackley D.H. "*Boltzmann machines : constraint satisfaction networks the learn*", Technical report, Carnegie Mellon University, 1984.
- [Hongo 96] Hongo S., Sonehara N. et Yoroizawa I. "*Edge-based binocular stereopsis algorithm - A matching mechanism with probabilistic feedback*", Neural Networks, Vol. 9, n° 3, p. 379-395, 1996.
- [Hopfield 82] Hopfield J.J. "*Neural networks and physical systems with emergent collective computational abilities*", Proc. of the National Academy Sciences, Vol. 79, USA, p. 2554-2558, 1982.
- [Hopfield 83] Hopfield J.J., Feinstein D.I. et Palmer R.G. "*Unlearning has a stabilizing effect in collective memories*", Nature, 304, p. 158-159, 1983.
- [Hopfield 85] Hopfield J.J. et Tank D.W. "*Neural computation of decisions in optimization problems*", Biological Cybernetics, 52, p. 141-152, 1985.
- [Horaud 88] Horaud R. et Skordas T. "*Structural matching for stereo vision*", Proc. 9<sup>th</sup> Int. Conf. on Pattern Recognition, Rome, 14-17 Novembre 1988.
- [Horaud 89] Horaud R. et Skordas T. "*Stereo matching through feature grouping and maximal cliques*", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. PAMI-11, n° 11, p. 1168-1180, Novembre 1989.
- [Horaud 93] Horaud R. et Monga O. "*Vision par ordinateur - outils fondamentaux*", Ed. Hermès, Paris, 1993.
- [Horn 81] Horn B.K.P. et Schunk B.G. "*Determining the optical flow*", Artificial Intelligence, Vol. 17, p. 185-203, 1981.
- [Horn 86] Horn B.K.P. "*Robot vision*", Cambridge, MA, MIT Press, 1986.
- [Horowitz 76] Horowitz S.L. et Pavlidis T. "*Picture segmentation by a tree transversal algorithm*", Assoc. Comput. Machine, n° 23, p. 368-388, 1976.
- [Hrycej 92] Hrycej T. "*Modular learning in neural networks*", John Wiley & Sons, Inc., 1992.
- [Hu 93a] Hu J.-E. et Siy P. "*An ordering-oriented Hopfield network and its application in stereo vision*", Proc. SPIE on Application of Artificial Neural Networks IV, Orlando, Vol. 1965, p. 556-567, 12-16 Avril 1993.
- [Hu 93b] Hu J.-E. et Siy P. "*Stereo correspondence through multiple constraint neural networks*", Proc. IEEE Int. Conf. on Neural Networks, San Francisco, CA, p. 126-131, 28 Mars - 01 Avril 1993.

- [Huang 81] Huang T.S. "*Image sequence analysis: what can we learn from applications?*", Image Sequence Analysis, Ed. T.S. Huang, p. 104-116, 1981.
- [Huang 92] Huang C.-N., Lim C.-C. et Liu M.C. "*Comparison of image processing algorithms and neural networks in machine vision inspection*", Computers & Industrial Engineering, Vol. 23, n° 1-4, p. 105-108, 1992.
- [Hubel 79] Hubel D. "*The brain*", Scientific American, Vol. 241, n° 3, p. 39-47, Septembre 1979.
- [Huntsberger 87] Huntsberger T.L. et Jayaramamurthy S.N. "*Determination of the optic flow field using the spatio temporal deformation of region properties*", Pattern Recognition Letters, Vol. 6, p. 169-177, Août 1987.
- [Hwang 93] Hwang J.-N. et Chen E.T.Y. "*Textured image segmentation via neural network probabilities modelling*", IEEE Conf. on Neural Networks, San Francisco, USA, p. 1702-1707, 1993.
- [Ikeuchi 83] Ikeuchi K. "*Constructing a depth map from images*", Technical rep., AI Memo n° 744, Massachusetts, Août 1983.
- [Inigo 85] Inigo R.M., Tkacik T. et McVey E.S. "*The application of linear image arrays to mobile robot guidance and navigation*", Proc. 3<sup>rd</sup> Int. Conf. on Automated Guided Vehicle Systems, Stockholm, Suède, p. 157-168, Octobre 1985.
- [Inigo 87] Inigo R.M. et Tkacik T. "*Mobile robot operation in real-time with linear image array based vision*", Proc. IEEE Int. Symposium Intelligent Control, Philadelphie, Etats-Unis, p. 228-233, Janvier 1987.
- [Ito 86] Ito M. et Ishii A. "*Three-view stereo analysis*", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. PAMI-8, p. 524-531, Juillet 1986.
- [Jain 79] Jain R. et Nagel H.H. "*On the analysis of accumulative difference pictures from image sequences of real world scenes*", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. PAMI-1, n° 2, p. 206-214, 1979.
- [Jain 81] Jain R, Miltzer D. et Nagel H.H. "*Separating non-stationary scene componentes in a sequence of real world TV images*", IJCAI-81, p. 612-618, 1981.
- [Jain 85] Jain R. "*Extraction of motion information from peripheral processes*", Digital Image Processing and Analysis, Vol. 2, p. 491-505, 1985.
- [Jodouin 94] Jodouin J.-F. "*Les réseaux de neurones : principes et définitions*", Collection Hermès, Paris, 1994.

- [Jorgsen 87] Jorgsen C.C. "*Neural network representation of sensor graphs in autonomous robot path planning*", Proceedings IEEE on Neural Networks, 1987.
- [Kachi 96] Kachi D. "*Une approche géométrique pour l'analyse d'une séquence d'images monoculaires*", Thèse de Doctorat, Université de Technologie de Compiègne, France, 1996.
- [Kamp 90] Kamp Y. et Hasler M. "*Réseaux de neurones récurssifs pour mémoires associatives*", Ed. Presses Polytechniques et Universitaires Romandes, Collection électricité, 1990.
- [Kanade 94] Kanade T. et Okutomi M. "*A stereo matching algorithm with an adaptative window : theory and experiement*", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. PAMI-16, n° 9, Septembre 1994.
- [Kass 87] Kass M., Witkin A. et Terzopoulos D. "*Snakes : Active contour models*", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 3, p. 259-267, 1987.
- [Kendall 92] Kendall G.D. et Hall T.J. "*Performing fundamental image processing operations using quantised neural networks*", 4<sup>th</sup> int. Conf. on Image Processing and its Applications.
- [Khotanzad 90] Khtanzad A. et Lee Y.W. "*Depth perception by a neural network*", IEEE Midcon/90 Conf. Record, Dallas, Texas, p. 424-427, 11-13 Septembre 1990.
- [Khotanzad 91] Khotanzad A. et Lee Y.W. "*Stereopsis by a neural network which learns the constraints*", Advances in Neural Information Processing Systems, Morgan Kaufmann Publishers, Vol. 3, p. 327-334, Inc. 1991.
- [Khotanzad 93] Khotanzad A., Bokil A. et Lee Y.W. "*Stereopsis by constraint learning feed-forward neural networks*", IEEE Trans. on Neural Networks, Vol. 4, n° 2, p. 332-342, Mars 1993.
- [Kirkpatrick 83] Kirkpatrick S., Gelatt C.D. et Vecchi M.P. "*Optimization by simulated annealing*", Science, Vol. 220, n° 4598, p. 671, Mai 1983.
- [Kirkpatrick 83] Kirkpatrick S., Gelatt C.D.Jr. et Vecchi M.P. "*Optimization by simulated annealing*", Science, 220, p. 671-680, 1983.
- [Kirkpatrick 84] Kirkpatrick S. "*Optimization by simulated annealing : quantitative studies*", Journ. Statist. Phys., Vol. 34, p. 974, 1984.
- [Kitchen 82] Kitchen L. et Rosenfeld A. "*Grey-level corner detection*", Pattern Recognition Letters, Vol. 1, p. 95-102, Décembre 1982.

- [Knerr 92] Knerr S. "*Réseaux de neurones pour la classification automatique : Application à la reconnaissance de chiffres manuscrits*", Thèse de Doctorat, Université Paris VI, 01 Avril 1992.
- [Koch 86] Koch C., Marroquin J. et Yuille A. "*Analog 'neuronal' networks in early vision*", Proc. Natl. Acad. Sci., USA, Vol. 83, p. 4263-4267, Juin 1986.
- [Koh 93] Koh J., Suk M. et Bhandarkar S.M. "*Multi-layer Kohonen's self-organizing feature map for range image segmentation*", IEEE Int. Conf. on Neural Networks, San Francisco, California, USA, p. 1270-1276, 1993.
- [Kohonen 82] Kohonen T. "*Analysis of a simple self-organizing process*", Biological Cybernetics, 44, p. 135-140, 1982.
- [Kohonen 84] Kohonen T. "*Self-organization and associative memory*", Springer Verlag, Whshington DC, 2<sup>nd</sup> edition, 1984.
- [Kohonen 88a] Kohonen T., Barna G. et Chrisley R. "*Statistical pattern recognition with neural networks : benchmarking studies*", IEEE International Conference on Neural Networks, San Diego (USA, CA), 1, p. 61-68, 1988.
- [Kohonen 88b] Kohonen T. "*Self organization and associative memory*", Springer Verlag, 1988.
- [Kohonen 90] Kohonen T., Kangas J.A. et Laaksonen J.T. "*Variants of self-organizing maps*", IEEE Trans. on Neural Networks, 1, p. 93-99, 1990.
- [Korn 80] Korn A. "*Motion analysis in natural scenes picked up by a moving optical sensor*", Proc. 5<sup>th</sup> Int. Conf. on Pattern Recognition, Miami, Etats-Unis, 1980.
- [Kriegman 89] Kriegman D.J, Triendl E. et Binford T.O. "*Stereo vision and navigation in buildings for mobile robots*", IEEE Trans. on Robotics and automation, Vol. 5, n° 6, 1989.
- [Krotkov 90] Krotkov E., Henriksen K. et Kories R. "*Stereo ranging with verging cameras*", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. PAMI-12, n° 12, p. 1200-1205, Décembre 1990.
- [Krüger 95] Krüger W., Enkelmann W. et Rössle S. "*Real-time estimation and tracking of optical flow vectors for obstacle detection*", IEEE Intelligent Vehicles Symposium '95, Detroit, Etats-Unis, p. 304-309, 25-26 Septembre 1995.
- [Lasserre 96] Lasserre P. "*Vision pour la robotique mobile en environnement naturel*", Thèse de Doctorat, L.A.A.S., Université Paul Sabatier, Toulouse, France, Septembre 1996.

- [Lawton 83] Lawton D.T. "*Processing translational motion sequences*", Computer Vision Graphics and Image Processing, p. 116-144, Avril 1983.
- [LeCun 87] Le Cun Y. "*Modèles connexionnistes de l'apprentissage*", Thèse de Doctorat, Université de Paris VI, 1987.
- [Lee 91] Lee H.-C. et Cok D.R. "*Detecting boundaries in a vector field*", IEEE Trans. on Signal Processing, Vol. 39, n° 5, p. 1181-1194, Mai 1991.
- [Lee 94a] Lee S.-H. et Leou J.-J. "*A dynamic programming approach to line segment matching in stereo vision*", Pattern Recognition, Vol. 27, n° 8, p. 961-986, Août 1994.
- [Lee 94b] Lee J.J., Shim J.C. et Ha Y.H. "*Stereo correspondence using the Hopfield neural network of a new energy function*", Pattern Recognition, Vol. 27, n° 11, p. 1513-1522, 1994.
- [Leese 71] Leese J.A. "*Analysis automatic technique for obtaining cloud motion from geosynchronous satellite data using cross-correlation*", J. Appl. Meteorol., Vol. 10, p. 118-132, Février 1971.
- [Lelaurin 96] Lelaurin L. "*Contribution à la définition d'un modèle générique de réseaux de neurones pour la minimisation de fonctionnelles - Application à l'estimation de mouvement dans des séquences d'images*", Thèse de Doctorat, Université des Sciences et Technologies de Lille, France, 27 Septembre 1996.
- [Leroi 89] Leroi M. "*Contribution à la vision 3D du Robot Marcheur EMA4 dans un Univers Polyédrique*", Thèse de Doctorat, Université de Valenciennes et du HainautCambresis, France, Février 1989.
- [Levin 93] Levin A.U. et Narendra K.S. "*Control of nonlinear dynamical systems using neural networks : Controllability and stabilization*", IEEE Trans. on Neural Networks, Vol. 4, n° 2, p. 192-206, Mars 1993.
- [Levine 73] Levine M., Handly D. et Yagi G. "*Computer determination of depth maps*", Computer Graphics and Image Processing, Vol. 2, 1973.
- [Levy 87] Levy B.C. et Adams M.B. "*Global optimization with stochastic neural networks*", IEEE Intl. Conf. on Neural Networks, San Diego, p. III-681, 1987.
- [Limb 75] Limb J.O. et Murphy J.A. "*Measuring the speed of moving objects from television images*", IEEE Trans. on Comm., V-Com. 23, p. 474-478, 1975.
- [Lippmann 87] Lippmann R.P. "*An introduction to computing with neural nets*", IEEE ASSP Magazine, p. 4-23, Avril 1987.

- [Lippmann 89] Lippmann R.P. "**Pattern classification using neural networks**", IEEE Communications Magazine, Vol. 27, n° 11, p. 47-64, 1989.
- [Little 74] Little W.A. "**The existence of persistent states in the brain**", Math. Biosci., 19, p. 101-120, 1974.
- [Loncelle 89] Loncelle J. "**Détection de contour par rétropropagation du gradient**", Concept logiciel expert, NeuroNîme, p. 373, 1989.
- [Loncelle 90] Loncelle J. "**Contribution des réseaux connexionnistes au traitement d'images bas-niveau**", Thèse de Doctorat, Université Paris XI, 1990.
- [Long 86] Long-Limozin P. "**Vision stéréoscopique appliquée à la robotique**", Thèse de Doctorat, Université de Nice, France, Octobre 1986.
- [Lumia 83] Lumia R., Haralick R.M., Zuniga O., Shapiro L., Pong T.-C. et Wang F.-P. "**Texture analysis of aerial photographs**", Pattern Recognition, Vol. 16, n° 1, p. 39-46, 1983.
- [Lustman 87] Lustman F. "**Vision stéréoscopique et perception du mouvement en vision artificielle**", Thèse de Doctorat, Université de Paris-Sud, Décembre 1987.
- [Ma 87] Ma J., Lu X. et Wu C. "**A motion constraint equation under space varying or time varying illumination**", Pattern Recognition Letters, Vol. 5, p. 301-315, 1987.
- [Maitre 84] Maitre H. et Lopez K.J. "**Transformation de Hough: théorie et application à la détection de symétries circulaires**", 4<sup>ème</sup> congrès de Reconnaissance des Formes, Paris, Janvier 1984.
- [Malaterre 92] Malaterra G. et Fontaine H. "**Les aides à la conduite: quels enjeux pour la sécurité?**", Recherche Transports Sécurité, n° 35, p. 43-54, Septembre 1992.
- [Mammone 94] Mammone R.J. "**Artificial neural networks for speech and vision**", First Edition, Chapman & Hall, London, 1994.
- [Maren 90] Maren A.J., Harston C.T. et Pap R.M. "**Handbook of neural computing applications**", Academic Press.
- [Marr 76] Marr D. et Poggio T. "**Cooperative computation of stereo disparity**", Science, n° 194, p. 283-287, 1976.
- [Marr 77a] Marr D., et Poggio T. "**A theory of human stereo vision**", A. I. Memo n° 451, Massachusetts Institute of Technology, Artificial Intelligence Laboratory, Novembre 1977.

- [Marr 77b] Marr D., et Poggio T. "*Analysis of a cooperative stereo algorithm*", A. I. Memo n° 446, Massachusetts Institute of Technology, Artificial Intelligence Laboratory, Octobre 1977.
- [Martinez 93] Martinez F., Thiesse B. et Marthon P. "*Reconstruction de scènes 3D en vision binoculaire animée*", 4<sup>ème</sup> Journées ORASIS, GDR-PRC Communication Homme-Machine, Mulhouse, p. 13-14, 11-14 Octobre 1993.
- [McCulloch 43] McCulloch W.S. et Pitts W. "*A logical calculus of the ideas immanent in nervous activity*", Bulletin of Mathematical Biophysics, Vol. 5, p. 115-133, 1943.
- [Mead 89] Mead C. "*Analog VLSI and neural systems*", Adisson-Wesley, 1989.
- [Medioni 84] Medioni G. et Nevatia R.N. "*Matching images using linear features*", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. PAMI-6, n° 6, p. 675-685, Novembre 1984.
- [Medioni 85] Medioni G. et Nevatia R. "*Segment-besed stereo matching*", Computer Vision, Graphics and Image Processing, Vol. 31, p. 2-18, 1985.
- [Métropolis 53] Métropolis N., Rosenbluth A.W., Rosenbluth M.N, Teller A.H. et Teller E. "*Equation of state calculations for fast computing machines*", Journal of Chemical Physics, 6, p. 1087, 1953.
- [Minsky 69] Minsky M. et Papert S. "*Perceptrons*", MIT Press, Cambridge, 1969.
- [Missler 95] Missler J.M. et Kamangar F.A. "*A neural network for pursuit tracking inspired by the fly visual system*", Neural Networks, Vol. 8, n° 3, p. 463-480, 1995.
- [Mitchell 78] Mitchell O.R. et Carlton S.G. "*Image segmentation using a local extrema texture measure*", Proc. IEEE Comp. Soc. Conf. on Pattern Recognition and Image Processing, p. 386, aussi dans Pattern Recognition, Vol. 10, p. 205-210, 1978.
- [Mitiche 86] Mitiche A. "*On kineopsis and computation of structure and motion*", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol PAMI-8, n° 1, Janvier 1986.
- [Mitiche 87] Mitiche A. "*Experiments in computing optical flow with the gradient-based multiconstraint method*", Pattern Recognition, Vol. 20, n° 2, p. 173-179, 1987.
- [Monchaud 86] Monchaud S. "*Contribution à la télémétrie pour robots de troisième génération*", Thèse de Doctorat d'Etat Es-Sciences, Université de Rennes 1 et INSA de Rennes, France, 2 Juillet 1986.

- [Moravec 77] Moravec H.P. "*Toward automatic visual obstacle avoidance*", 5<sup>th</sup> International Joint Conference on Artificial Intelligence, Cambridge, MA, USA, p. 584, Août 1977.
- [Moravec 80] Moravec H.P. "*Obstacle avoidance and navigation in the real world by a seeing robot rover*", PhD Thesis, Stanford Artificial Intelligence Laboratory, 1980.
- [Moravec 81] Moravec H.P. "*Rover visual obstacle avoidance*", Proc. 7<sup>th</sup> International Joint Conference on Artificial Intelligence, Vancouver, Canada, p. 785-790, Août 1981.
- [Moura 91] Moura L. et Martins F.C.M. "*Edge détection through cooperation and competition*", IEEE Int. Joint Conf. on Neural Networks, Singapore, p. 2588-2593, 18-21 Novembre 1991.
- [Mousavi 90] Mousavi M.S. et Schalkoff R.J. "*A neural network approach for stereo vision*", Proc. of IEEE SouthEastConf90, p. 808-812, 1990.
- [Nagao 83] Nagao M. et Matsuyama M. "*Structural image analysis*", 1983.
- [Nasrabadi 92a] Nasrabadi N.M. "*A stereo vision technique using curve-segments and relaxation matching*", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. PAMI-14, n° 5, p. 566-572, Mai 1992.
- [Nasrabadi 92b] Nasrabadi N.M. et Choo C.Y. "*Hopfield network for stereo vision correspondence*", IEEE Trans. on Neural Networks, Vol. 3, n° 1, p. 5-13, Janvier 1992.
- [Nichani 94] Nichani S. "*Solving the correspondence problem using a Hopfield network*", IEEE International Conference on Neural Networks, Orlando, USA, Vol. VI, p. 4107-4112, 28 Juin - 02 Juillet 1994.
- [Noble 87] Noble A. "*Finding corners*", Proc. Alvey Conference, Cambridge, Angleterre, 1987.
- [O'Toole 89] O'Toole A.J. "*Structure from stereo by associative learning of the constraints*", Perception, Vol. 18, p. 767-782, 1989.
- [Ögmen 90] Ögmen H. et Gagné S. "*Neural network architectures for motion perception and elementary motion detection in the fly visual system*", Neural Networks, Vol. 3, p. 487-505, 1990.
- [Ohta 85] Ohta Y. et Kanade T. "*Stereo by intra- and inter-scanline search using dynamic programming*", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. PAMI-7, n° 2, p. 139-154, Mars 1985.

- [Oksenhendler 94] Oksenhendler V., Bensrhair A., Miché P. et Debrie R. "*Un nouveau processus auto-adaptatif de stéréovision passive utilisant une nouvelle primitive : La déclivité*", Colloque AGI'94, Futuroscope, Poitiers, p. 333-336, 2-3 Juin 1994.
- [Olhander 78] Olhander R., Price K. et Reddy D.R. "*Picture segmentation using a recursive region splitting method*", Computer Graphics and Image Processing, n° 8, p. 313-333, 1978.
- [Oshima 79] Oshima M. et Shirai Y. "*A scene description method using three dimensional information*", Pattern Recognition, Vol. 11, p. 9-17, 1979.
- [Pang 90] Pang V. et Palaniswami M. "*Pattern classification using a self-organizing neural network*", IEEE Region 10 Conf. on Computer and Communication Systems, p. 562-566, 1990.
- [Pao 90] Pao Y. "*Adaptive pattern recognition and neural networks*", Case Western Reserve University Addison-Wesley Publishing Company, Inc., 1990.
- [Parker 85] Parker D.B. "*Learning logic*", Report AT-47, MIT Center for Computational Research in Economics and Management Science, 1985.
- [Pekergin 92] Pekergin M.F. "*Optimisation combinatoire par le calcul neuronal et parallélisme optimal*", Thèse de Doctorat, Université Paris V, 17 Mars 1992.
- [Peretto 84] Peretto P. "*Collective Properties of neural networks - A statistical physic approach*", Biological Cybernetics, 50, p. 51-62, 1984.
- [Personnaz 86] Personnaz L., Guyon I. et Dreyfus G. "*Collective computational properties of neural networks : new learning mechanisms*", Physical Review A, Vol. 34, n° 3, 1986.
- [Peterson 89] Peterson C. et Söderberg B. "*A new method for mapping optimization problems onto neural networks*", Int. Journal of Neural Systems, Vol. 1, n° 1, p. 3-22, 1989.
- [Pham 92] Pham D.T. et Bayro-Corrochano E.J. "*Neural networks for low-level image processing*", Artificial Neural Networks 2, Aleksander I. and Taylor J. (eds.), Proceedings of the 1992 Int. Conf. on Artificial Neural Networks, Brighton, UK, 4-7 Septembre 1992.
- [Pietikainen 86] Pietikainen M. et Harwood D. "*Depth from three-camera stereo*", Proc. IEEE Int. Conference on Computer Vision and Pattern Recognition, Miami Beach, Floride, Etats-Unis, p. 2-8, 1986.
- [Piskounov 80] Piskounov N. "*Calcul différentiel et intégral*", 9<sup>ème</sup> édition, Tome II.

- [Pollard 85] Pollard S.B., Mayhew J.E.W. et Frisby J.P. "***PMF : A stereo correspondence algorithm using a disparity gradient limit***", Perception, n° 14, p. 449-470, 1985.
- [Pollard 86] Pollard S.B., Porril J., Meyhew J.E.W. et Frisby J.P. "***Disparity gradient, lipschitz continuity and computing binocular correspondence***", Proc. 3<sup>th</sup> International Symposium on Robotics Research, France, 1986.
- [Pöppel 87] Pöppel G. et Krey U. "***Dynamical learning process for recognition of correlated patterns in symmetric spin glass models***", Europhysics Letters, Vol. 4, n° 9, p. 979-985, 1987.
- [Prewitt 70] Prewitt J.M.S. "***Object enhancement and extraction***", Picture Processing and Psychopictorics, 1970.
- [ProArt 95] Laboratoires impliquées dans le groupe ProArt. "***Rapport final des activités du groupe ProArt France***".
- [Protzel 93] Protzel P.W., Palumbo D.L. et Arras M.K. "***Performance and Fault-Tolerance of neural networks for optimization***", IEEE Trans. on Neural Networks, Vol. 4, n° 4, p. 600-614, Juillet 1993.
- [Qian 88] Qian N. et Sejnowski T.J. "***Learning to solve random-dot stereograms of dense and transparent surfaces with recurrent backpropagation***", Proc. 1988 Connectionist Models, Summer school, Carnegie Mellon University, Pittsburgh, PA, p. 435-443, 17-26 Juin 1988.
- [Rajala 83] Rajala S.A. "***Application of the one-dimensional Fourier transform for tracking moving objects in noisy environment***", Computer Vision Graphics and Image Processing, Vol. 21, 1983.
- [Rives 86] Rives G. "***Planar partially occluded objects scene analysis***", Proc. 8<sup>ème</sup> Conf. on Pattern Recognition, 1986.
- [Robert 86] Robert de Saint-Vincent. "***Perception et modélisation de l'environnement d'un robot mobile : une approche par stéréovision***", Thèse de l'Université Paul Sabatier, Toulouse, 1986.
- [Roberts 65] Roberts L.G. "***Machine perception of three-dimensional solids***", Optical and Electro-optical Information Processing, p. 159-197, 1965.
- [Rombaut 93] Rombaut M. "***ProLab2: A driving assistance system***", Proc. IEEE Tsukuba International Workshop on Advanced Robotics '93, Tsukuba, Japon, 8-9 Novembre 1993.
- [Rombaut 95] Rombaut M., Burie J.-C. et Postaire J.-G. "***ProLab2: Un système d'assistance à la conduite***", Proc. Conf. Applica '95, Lille, France, 24 Mars 1995.

- [Rosenblatt 62] Rosenblatt F. "***Principle of neurodynamics : Perceptrons and the theory of brain mechanisms***", Spartan Books, Whashington, 1962.
- [Rosenblatt 65] Rosenblatt F. "***The perceptron : a probabilistic model for information storage and organization in the brain***", Psychological Review, 65, p. 386-408, 1965.
- [Rosenfeld 71] Rosenfeld A. et Thurston M. "***Edge and curve detection for visual scene analysis***", IEEE Trans. on Computers, Vol. C-20, n° 5, p. 562-569, Mai 1971.
- [Rosenfeld 72] Rosenfeld A., Thurston M. et Lee Y. "***Edge and curve detection : further experiments***", IEEE Trans. on Computers, Vol C-21, n° 7, p. 677-691, Juillet 1972.
- [Ruichek 96] Ruichek Y., C. Vieren et J.-G. Postaire. "***Localising moving objects using active contour models and a neural stereo matching procedure***", Proceedings of the Int. Conf. on Engineering Applications of Neural Networks, Londres, Grande Bretagne, p. 355-358, 17-19 Juin 1996.
- [Rumelhart 86] Rumelhart D.E., Hinton G.E. et Williams R.J. "***Parallel distributed Processing***", MIT Press, 1986.
- [Satanarayama 92] Satanarayama S., Tsividis Y.P. et Graf H.P. "***A reconfigurable VLSI neural network***", IEEE Journal of Solid-State Circuits, Vol. 27, p. 67-81, Janvier 1992.
- [Sejnowski 87] Sejnowski T. et Rosenberg C. "***Parallel networks that learn to pronounce English text***", Complex Systems, Vol. 1, p. 145-168, 1987.
- [Selsis 96] Selsis M. "***Application des modèles de contours actifs au suivi et à la localisation 3D d'objets en mouvement***", Thèse de Doctorat, Université des Sciences et Technologies de Lille, France, 11 Janvier 1996.
- [Shadpey 94] Shadpey F., Khorasani K. et Patel R.V. "***An analog-Hopfield neural network for recognition of partially visible 2-D objects***", Proceedings of IEEE Int. Conf. on Neural Networks, Vol. 5, p. 2956-2961, 28 Juin-02 Juillet 1994.
- [Shariat 90] Shariat H. "***Motion estimation with more than two frames***", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. PAMI-12, n° 5, p. 417-434, Mai 1990.
- [Shen 86] Shen J. "***Filtarge rapide en traitement d'image et vision tridimensionnelle par ordinateur***", Thèse de Doctorat d'Etat, Université Paul Sabatier, Toulouse, Mai 1986.
- [Shen 87a] Shen J., Castan S. et Zhao J. "***Stereo vision by pyramidal BLI graph matching***", Proc. SPIE'87, Cannes, France, 1987.

- [Shen 87b] Shen J. et Castan S. "*A new strategy for multi-camera stereo vision*", Proc. 5<sup>th</sup> Scandinavian Conference on Image Analysis, Stockholm, Juin 1987.
- [Shen 89] Shen J. et Zhao J. "*A new trinocular stereo vision method*", Proc. 6<sup>th</sup> Scandinavian Conference on Image Analysis, Oulu, Finlande, Juin 1989.
- [Sheu 91] Sheu B.J. et Fang W.-C. "*Real-time high-ratio image compression using adaptative VLSI neuroprocessors*", Proceedings of the 1991 Int. Conf. on Acoustics Speech and Signal Processing - ICASSP 91, Toronto, Canada, 14-17 Mai 1991.
- [Sheu 92] Sheu B.J., Choi J. et Chang C.-F. "*An analog neural network processor for self-organizing mapping*", Tech. Dig. IEEE Int. Solid-State Circuits Conf., San Francisco, CA, p. 136-137, Février 1992.
- [Si-ahmed 86] Si-ahmed N. "*Contribution à l'integration de la vision dynamique au système de perception 3D du Robot Hilare*", Thèse de Docteur Ingénieur, CNRS, Décembre 1986.
- [Skordas 88a] Skordas T. et Horaud R. "*Mise en correspondance structurelle pour la vision stéréoscopique*", Technique et Science Informatiques, Vol. 7, n° 6, p. 591-608, 1988.
- [Skordas 88b] Skordas T. "*Mise en correspondance et reconstruction stéréo utilisant une description structurelle des images*", Thèse de Doctorat, Institut National Polytechnique de Grenoble, France, Octobre 1988.
- [Snyder 83] Snyder W.E et Rajala S.A. "*Track acquisition of sub-pixel targets*", Image Sequence Analysis, Ed. T.S. Huang, p. 503, 1983.
- [Sudharsanan 91] Sudharsanan S.I. et Sundareshan M.K. "*Exponential stability and a systematic synthesis of a neural network for quadratic minimization*", Neural Networks, Vol. 4, p. 599-613, 1991.
- [Sun 87] Sun G.Z., Chen H.H. et Lee Y.C. "*Learning stereopsis with neural networks*", Proc. IEEE First Int. Conf. on Neural Networks, San Diego, CA, p. 345-355, Juin 1987.
- [Sun 93] Sun K.T. et Fu H.C. "*A hybrid neural network model for solving optimisation problems*", IEEE Trans. on Computers, Vol. 42, n° 2, p. 218-227, Février 1993.
- [Szu 87] Szu H. "*Nonconvex optimization by fast simulated annealing*", Proc. of the IEEE, Vol. 75, n° 11, p. 1538-1540, 1987.
- [Tagare 86] Tagare H.D. et de Figueiredo R.J.P. "*On localization performance measure and optimal edge detection*", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. PAMI-12, p. 1186-1190, 1986.

- [Thompson 81] Thompson W.B. "**Lower level estimation and interpretation of visual motion**", Computer, Vol. 14, n° 8, p. 20-28, 1981.
- [Thorpe 90] Thorpe C.E. "**Vision and navigation - The carnegie mellon NavLab**", Kluwer Academic Publishers, Boston, 1990.
- [Tikhonov 77] Tikhonov A.N. et Arsenin V.Y. "**Solution to Ill-posed Problems**", Winston and Sons, Washington D.C, 1977.
- [Tsai 86] Tsai R. "**Multiframe image point matching and 3D surface reconstruction**", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. PAMI-5, n° 2, p. 159-174, 1986.
- [Ullman 79] Ullman S. "**The interpretation of visual motion**", MIT Press, Cambridge, Massachusetts, 1979.
- [Ulmer 92] Ulmer B. "**VITA - An Autonomous Road Vehicle (ARV) for collision avoidance in traffic**", IEEE Intelligent Vehicle Symposium '92, Detroit, Etats-Unis, p. 36-41, 29 Juin-01 Juillet 1992.
- [Ultré 96] Ultré V. "**Contribution à la segmentation d'images de mosaïque en couleur**", Thèse de Doctorat, Université des Sciences et Technologies de Lille, France, 25 Janvier 1996.
- [Unnikrishnan 91] Unnikrishnan K.P., Hopfield J.J. et Tank D.W. "**Connected digit speaker-dependant speech recognition using neural network with time delayed connections**", IEEE Trans. on Signal Processing, Vol. 39, n° 3, March 1991.
- [Vannoorenberghe 96] Vannoorenberghe P. Ruichek Y. et J.-G. Postaire. "**A neural matching procedure for tracking pedestrians in urban environment**", Proceedings of the Int. Conf. on Engineering Applications of Neural Networks, Londres, Grande Bretagne, p. 347-350, 17-19 Juin 1996.
- [Vinet 91] Vinet L. et Gagalowicz A. "**Mise en correspondance de régions utilisant la cohérence de leurs voisinages**", 8<sup>ème</sup> Congrès de Reconnaissance des Formes et Intelligence Artificielle, Lyon, France, Vol. 2, p. 821-826, 25-29 Novembre 1991.
- [Walker 94] Walker N.P., Eglén S.J. et Lawrence B.A. "**Image compression using neural networks**", GEC Journal of Research, Vol. 11, n° 2, p. 66-75, 1994.
- [Waltz 75] Waltz D. "**Understanding line drawings of scene with shadows**", Psychology of computer vision, Winston Ed., Mc Graw Hill, 1975.
- [Wane 93] Wane S. "**Contribution au pilotage des robots mobiles**", Thèse de Doctorat, Université de Technologie de Compiègne, France, 17 Mars 1993.

- [Wasserman 89] Wasserman P.D. "*Neural computing theory and practise*", V.N.R., New York, 1989.
- [Watanabe 74] Watanabe S. et Group C. "*An automated apparatus for cancer prescreening : cybest*", Computer Graphics and Image Processing, n° 3, p. 350-358, 1974.
- [Waxman 86] Waxman A.M. et al. "*A visual navigation system*", Proc. IEEE Conf. on Robotics and Automation, San Francisco, Etats-Unis, Avril 1986.
- [Webb 81] Webb J.A. et Aggarwal J.K. "*Structure from motion and jointed bodies*", Proc. 7<sup>th</sup> Int. Joint Conf. on Artificial Intelligence, Août 1981.
- [Webb 82] Webb J.A. et Aggarwal J.K. "*Structure from motion of rigid and jointed objects*", Artificial Intelligence, Vol. 19, p. 107-130, Septembre 1982.
- [Weszka 78] Weszka J.S. "*A survey of threshold selection techniques*", Computer Graphics and Image Processing, n° 7, p. 259-265, 1978.
- [Widrow 59] Widrow B. "*Adaptive sampled-data system : a statistical theory of adaptation*", Institute of Radio Engineers, Western Electronic Show and Convention, Convention Record, Vol. Part 4, p. 74-85, 1959.
- [Widrow 60] Widrow B. "*An adaptative 'Adaline' neuron using chemical 'memistor'*", Technical report number 1553-2, Standford Electronics Laboratories, p. 406-408, 17 Octobre 1960.
- [Williams 80] Williams T.D. "*Depth from camera motion in a real world scene*", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. PAMI-2, n° 6, Novembre 1980.
- [Wohn 83] Wohn K. "*Motion estimation based on multiple local constraints and nonlinear smoothing*", Pattern Recognition, Vol. 16, n° 6, 1983.
- [Wrobel 88] Worobel-Dautcourt B. "*Perception de la distance par mise en correspondance de régions entre des images stéréoscopiques*", Thèse de Doctorat, Institut Polytechnique de Lorraine, Mars 1988.
- [Yacoubi 96] Yacoubi A. "*Contribution de l'approche neuromimétique à l'analyse du mouvement dans une séquence d'images*", Thèse de Doctorat, Université des Sciences et Technologies de Lille, France, 05 Avril 1996.
- [Yalamachili 82] Yalamachili S., Martin W.N et Aggarwal J.K. "*Extraction of moving object descriptions via differencing*", Computer Graphics and Image Processing, Vol. 18, p. 188-201, 1982.
- [Yashida 81] Yashida M., Asada M. et Tsuji S. "*Automatic analysis of moving images*", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. PAMI-3, n° 1, p. 12-19, Janvier 1981.

- [Yashida 86] Yashida M., Kitamura Y. et Kimachi M. "***Trinocular vision : new approach for correspondence problem***", Proc. IEEE Int. Conf. on Pattern Recognition, Paris, p. 1041-1044, Octobre 1986.
- [Zagrouba 92] \* Zagrouba E., Krey C. et Hamrouni Z. "***Region matching by adjacency propagation in stereo vision***", 2<sup>nd</sup> Int. Conference on Automation, Robotics and Computer Vision, ICARCV'92, Singapour, Vol. 1, p. CV8.5.1-CV8.5.5, 16-18 Septembre 92.
- [Zeng 92] Zeng Z., Nakajima M. et Agui T. "***Study on image data compression by using neural networks***", Visual Communications and Image Processing '92, Boston, USA, Proceedings of the SPIE - The International Society for Optical Engineering, 1818(3), p. 1425-1433, 1992.
- [Zhao 89] Zhao J. "***Extraction d'information tridimensionnelle par stéréo vision***", Thèse de Doctorat, Université Paul Sabatier, Toulouse, France, Juillet 1989.
- [Zhou 88a] Zhou Y.T. et Chellappa R. "***Stereo matching using a neural network***", Proc. IEEE International Conf. Acoustics, Speech and Signal Processing, Vol. ICASSP-88, New York, p. 940-943, 11-14 Avril 1988.
- [Zhou 88b] Zhou Y.-T., Chellappa R., Vaid A. et Jenkins B.K. "***Image restoration using a neural network***", IEEE Trans. on Acoustics Speech and Signal Processing, Vol. 36, n° 7, Juillet 1988.
- [Zhou 88c] Zhou Y.T. et Chellappa R. "***Computation of optical flow using a neural network***", IEEE Proc. Intl. Conf. on Neural Networks, San Diego, CA, Vol. 2, p. 71-78, 1988.
- [Zhou 90] Zhou Y.T. et Chellappa R. "***A network for motion perception***", IEEE Intl. Proc. Intl. Joint Conf. on Neural Networks, San Diego, CA, Vol. 2, p. 875-884, June 1990.
- [Zhou 92] Zhou Y.T. et Chellappa R. "***Artificial neural network for computer vision***", Springer-Verlag, Berlin, 1992.
- [Ziou 91] Ziou D. "***La détection de contours dans les images à niveau de gris : mise en oeuvre et sélection de détecteurs***", Thèse de Doctorat, Institut National Polytechnique de Lorraine, France, 11 Octobre 1991.
- [Zuniga 83] Zuniga O. et Haralick R.M. "***Corner detection using the facet model***", International Conference on Pattern Recognition and Image Processing, p. 30-37, 1983.
- [Zwingelstein 95] Zwingelstein G. "***Diagnostic des défaillances***", Edition Hermès, Paris, 1995.