THÈSE

présentée à

L'UNIVERSITÉ DES SCIENCES ET TECHNOLOGIES DE LILLE

pour obtenir le titre de

DOCTEUR D'UNIVERSITE

Spécialité : Productique, Automatique et Informatique Industrielle

par

Muriel SELSIS

APPLICATION DES MODELES DE CONTOURS ACTIFS AU SUIVI ET A LA LOCALISATION 3D D'OBJETS EN MOUVEMENT.

Soutenue le 11 janvier 1996

Composition du jury :

P. VIDAL	Président
P. BAYLOU	Rapporteurs
M. BRIOT	
J. P. DEPARIS	Examinateur
J. G. POSTAIRE	Directeurs de thèse
C.VIEREN	

Travail préparé au Centre d'Automatique de l'Université des Sciences et Technologies de Lille

Remerciements

Je tiens à remercier Monsieur Pierre Vidal, professeur à l'U.S.T.L et directeur du Centre d'Automatique de Lille, pour m'avoir accueillie au sein de son laboratoire, et pour l'honneur qu'il me fait en présidant le jury de ma thèse.

Je remercie également Monsieur Jack-Gérard Postaire, professeur à l'U.S.T.L et responsable de l'équipe Image et Décision, codirecteur de mes travaux de recherche, pour l'aide qu'il m'a apportée.

Je remercie Christophe Vieren, maître de conférence à l'U.S.T.L et codirecteur de mes travaux de recherche, pour le temps qu'il m'a consacré et pour son aide précieuse, aussi bien en recherche que pour la rédaction de ce manuscrit.

Je remercie également Messieurs Pierre Baylou et Maurice Briot, respectivement professeurs à l'E.N.S.E.R.B de Bordeaux et au L.A.A.S de Toulouse, pour avoir accepté d'être rapporteurs de mon manuscrit.

Je remercie Monsieur Jean Pierre Deparis, pour avoir accepté d'être examinateur de mon travail de thèse.

Je remercie également tous les membres de l'équipe Image et Décision, et tout particulièrement François Cabestaing, pour les conversations fructueuses que nous avons eues et pour l'aide qu'ils m'ont apportée. Je remercie aussi particulièrement Luc Duvieubourg, pour m'avoir permis d'acquérir une certaine expérience dans l'enseignement.

Table des matières

Table des figures

Introduction générale

Le travail de recherche présenté dans ce mémoire a été effectué au sein de l'équipe « Image et Décision » du Centre d'Automatique de Lille. Il consiste en la modélisation, le suivi et la localisation des objets en mouvement dans des séquences d'images stéréoscopiques.

Il fait suite à différents travaux de recherche menés en collaboration par le Centre d'Automatique de Lille et l'INRETS (Institut National de REcherche sur les Transports et leur Sécurité) sur la surveillance des transports en vue d'améliorer leur sécurité, et notamment à la thèse de C. Vieren intitulée « Segmentation de scènes dynamiques en temps réel. Application au traitement de séquences d'images pour la surveillance de carrefours routiers ».

La surveillance de trafic basée sur le traitement d'images consiste essentiellement à suivre les véhicules en mouvement. Les algorithmes utilisés doivent être à la fois rapides, économiques et robustes.

Nous avons choisi d'adapter les contours actifs au suivi d'objets en mouvement, car nous pensons qu'ils répondent à ces trois critères. Les contours actifs permettent notamment de disposer d'un modèle facilitant grandement les procédures de suivi temporel et d'appariement stéréoscopique, d'où un important gain en temps de calcul. D'autre part, ils peuvent apporter des informations statistiques intéressantes pour la surveillance, comme par exemple la vitesse des véhicules, leur nombre pendant une période donnée, etc.

Enfin, dans le cadre des applications envisagées pour la surveillance de trafic, les objets en mouvement pénètrent dans la scène par sa périphérie. Cette hypothèse de travail nous permet de proposer une initialisation automatique des contours actifs à la périphérie de l'image, et ainsi d'automatiser le suivi.

Pour justifier notre choix, nous allons tout d'abord présenter les méthodes de segmentation classiques, afin de les comparer avec l'approche de la segmentation par les contours actifs.

En traitement d'images, la plupart des méthodes de segmentation utilisées pour détecter les objets présents dans les images agissent de manière locale. Des portions de contour (points ou segments) ou des portions de surface sont détectées et supposées appartenir à un même objet reconstruit à partir de ces différents éléments. Les diverses techniques de

segmentation utilisées pour des images statiques sont décrites dans le premier chapitre. Les techniques de segmentation dynamique sont décrites dans le second chapitre.

Une fois la segmentation statique ou dynamique effectuée, on peut soit localiser les objets dans la scène en utilisant une paire d'images stéréoscopiques, soit suivre les objets dans la séquence d'images afin d'analyser leur mouvement. Pour cela, il existe différentes méthodes d'appariement, basées sur l'analyse des caractéristiques des primitives obtenues par les méthodes de segmentation. Celles-ci sont présentées dans le troisième chapitre. Ces problèmes d'appariement sont complexes à résoudre en raison de l'approche locale de la segmentation.

Des méthodes plus récentes ont adopté un point de vue plus global. Une modélisation du contour des objets a été introduite à partir de la notion d'énergie : il s'agit des contours actifs. Les avantages de ces nouvelles méthodes découlent de la modélisation : les contours obtenus correspondent à une succession de points tous liés les uns aux autres par une formulation mathématique. Les modèles de contours actifs et leurs différentes applications en traitement d'images sont présentés dans le quatrième chapitre.

Ces modèles, qui offrent d'excellents résultats sur les images statiques, nous ont paru également bien adaptés pour l'analyse de séquences d'images. En effet, le caractère actif de cette méthode en fait un outil rapide et efficace pour traiter successivement une série d'images où les objets occupent des places relativement voisines d'une image à l'autre. L'objectif principal de notre travail a été l'adaptation des modèles de contours actifs à ce type de problème et plus précisément à la modélisation et au suivi des objets en mouvement dans les séquences d'images.

Les contours actifs sont des courbes dont l'évolution est régie par la minimisation de l'énergie qui leur est associée. Dans le modèle initial, cette énergie est définie de façon à ce que les contours actifs soient attirés par les contours des objets situés dans leur voisinage. Pour que les contours actifs puissent suivre les objets en mouvement, il faut extraire les contours de ces objets du reste de l'image. On effectue donc un pré-traitement des séquences avec un opérateur développé au sein de notre équipe et décrit dans le deuxième chapitre. Des modèles de contours actifs adaptés sont ensuite appliqués sur les images ainsi obtenues. Les adaptations concernent d'une part la formulation mathématique du modèle, et d'autre part, le processus de convergence, qui a été choisi pour satisfaire la contrainte essentielle de rapidité. Ces modifications sont décrites dans le cinquième chapitre.

Disposant alors d'un modèle des objets en mouvement, il devient possible de simplifier plusieurs problèmes cruciaux de l'analyse d'images. Ainsi, nous nous sommes attachés à utiliser cette modélisation pour simplifier les procédures d'appariement stéréoscopique. Nous montrons, dans le sixième chapitre, que les modèles de contours utilisés pour le suivi permettent d'apparier facilement, et de façon fiable, les objets présents dans chacune des images du couple stéréoscopique. Les modèles de contours actifs nous permettent ainsi de localiser les objets en mouvement dans l'espace tridimensionnel.

Enfin, le septième et dernier chapitre est consacré à la présentation des résultats que nous avons obtenus sur des séquences d'images de synthèse et sur des images réelles.

Chapitre I

La segmentation statique

A. Introduction

On peut distinguer trois grands domaines en analyse d'images :

- **la segmentation** qui consiste à extraire différents éléments plus ou moins complexes présents dans les images, et de natures très diverses suivant les images traitées : cellules biologiques, nuages, cristaux, etc. Cette étape est toujours nécessaire, car elle permet d'extraire l'information utile pour l'application envisagée [ASA-81], [HAR-85], [HOR-93].
- **la reconstruction** qui permet, à partir de deux images d'une même scène prises par deux caméras dont les points de vue sont différents, de reconstruire l'environnement 3D observé, même sans information *a priori* sur la scène analysée [HOF-89], [GRI-86]. Cette reconstruction utilisant deux images d'une même scène fait partie du processus de stéréovision. On peut également reconstruire une scène à partir d'images prises par une seule caméra, mais dans ce cas, il est nécessaire soit de disposer d'informations *a priori* sur la scène étudiée [BRO-86], soit de déplacer la caméra pour disposer de plusieurs points de vue (Depth from motion) [ADI-85], [JER-90].
- **la reconnaissance** qui consiste à comparer des modèles d'objets avec les objets présents dans l'image, afin d'identifier ces derniers. Une application bien connue est la reconnaissance de caractères.

Ce chapitre est uniquement consacré à la segmentation statique, étape préalable à toute analyse [ROS-71], [MAR-80]. Nous allons donner un aperçu des procédures classiques de segmentation d'images statiques, puis nous traiterons dans le chapitre suivant la segmentation dynamique, qui consiste à extraire les objets en mouvement d'une séquence d'images. On distingue en segmentation deux grandes classes de procédures complémentaires :

- les procédures basées sur la recherche de zones homogènes ou **régions**.
- les procédures basées sur la **détection des contours** [GAU-93].

B. La segmentation en régions

Le but de ce type de segmentation est de trouver des zones homogènes dans l'image (homogènes par leur niveau de gris, leur couleur ou leur texture), chacune de ces zones devant correspondre à un objet ou à une partie d'objet. Les problèmes rencontrés lors de l'élaboration d'algorithmes de ce type sont les suivants :

- la recherche de critères utilisant les valeurs de la fonction intensité et les relations géométriques entre pixels afin de caractériser les régions
- l'exploitation et la mise en oeuvre de ces critères.

On distingue deux grands types de méthodes de segmentation en régions :

- la segmentation par classification
- la segmentation par séparation/fusion de régions.

1. La segmentation par classification

Pour classer les pixels, on dispose de deux types d'informations : la luminance des pixels, c'est à dire la valeur de la fonction intensité, ou encore niveau de gris, et les coordonnées des pixels. Lors de la segmentation par classification, ces deux informations sont traitées successivement.

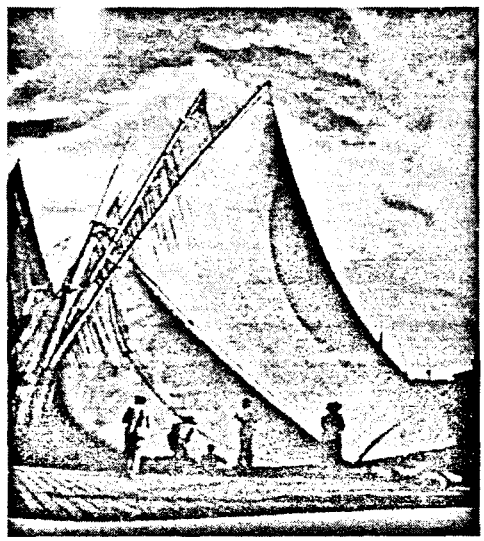
On détermine tout d'abord une classification des pixels à l'aide de l'histogramme des niveaux de gris de l'image. Quand l'histogramme est caractérisé par la présence de modes et de vallées, on regroupe dans une même classe l'ensemble des pixels compris entre deux vallées, car l'intensité des pixels correspondant à l'image d'un objet est supposée homogène. Ainsi, dans le cas d'images simples, un objet est représenté par un mode de l'histogramme [PRE-66].

Malheureusement, les images dont l'histogramme est simple à interpréter sont des cas particuliers : elles bénéficient d'un éclairage homogène, comportent peu d'objets et le fond est uniforme. Il est donc nécessaire d'utiliser d'autres critères que la valeur de l'intensité pour distinguer les différents objets.

On peut utiliser les relations spatiales entre pixels pour les classer en régions [OLH-78], [WES-74] : une région est alors définie comme un ensemble maximal de pixels connexes appartenant à une même classe. On peut également utiliser des vecteurs de propriétés associés à chaque pixel. Ces vecteurs peuvent contenir le niveau du pixel considéré ainsi que celui de

ses voisins. On réalise ensuite la segmentation par agrégation des points dont les vecteurs propriété sont suffisamment ressemblants au sens d'un certain critère [ASA-81], [HAR-85].

Ce type de méthode est difficilement utilisable si les images contiennent beaucoup d'objets ou sont très bruitées, car l'histogramme est alors peu exploitable. C'est le cas de la figure I-1 :



a) Image.

b) Histogramme.

Figure I-1 : Image et son histogramme

Les modes et les vallées de cet histogramme sont très nombreux et se distinguent difficilement. Ceci provient du fait que l'image contient beaucoup d'éléments différents, et qu'au sein d'un même objet, les niveaux de gris ne sont pas homogènes.

2. La segmentation par fusion et séparation de régions

Contrairement aux méthodes précédentes, ces méthodes utilisent simultanément les diverses informations concernant les pixels, à savoir leur luminance et leurs coordonnées :

- **fusion** : les algorithmes de fusion, également nommés algorithmes de croissance de régions, visent à fusionner les régions connexes de l'image qui présentent le même critère d'homogénéité, appelé prédicat d'homogénéité.

  A chaque itération, on traite des couples de régions connexes. On ne fusionne que les couples pour lesquels la réunion vérifie encore le prédicat d'homogénéité, et pour lesquels la qualité de la fusion est optimale [MUE-68], [BRI-70], [PON-80]. Il se pose donc le problème du choix du ou des prédicats d'homogénéité, ainsi que de la mesure de la qualité de la fusion. Voici à titre indicatif deux exemples de prédicats d'homogénéité :

  ◊ Soit P(R) un prédicat d'homogénéité, R étant une région de l'image, et A un point de la région R, de coordonnées (x,y) et d'intensité I. On définit P(R) comme suit :

  $$P(R) = (\max_{A \in R} I(A) - \min_{A \in R} I(A))$$

  Une région R vérifiant le prédicat P(R) équivaut à :

  l'inégalité [P(R)<seuil] est vraie.

  Avec ce prédicat, seules sont considérées comme régions les zones de l'image où les variations d'intensité sont faibles.

  ◊ On peut également choisir comme prédicat d'homogénéité :

  $$P(R) = \text{variance}(R)$$

  Le prédicat conditionne donc la qualité de la segmentation, un mauvais prédicat pouvant conduire par exemple au découpage de l'image en une multitude de petites régions non significatives. D'autre part, la segmentation en régions n'est pas adaptée au traitement de toutes les images. La segmentation d'images contenant des dégradés de lumière, par exemple, donne lieu à la création de régions qui ne correspondent à aucun objet réel de la scène, mais simplement aux différents niveaux d'éclairage des objets (cas de la voile de la figure I-1).

- **séparation** : les algorithmes de séparation de régions procèdent de façon opposée aux algorithmes de fusion : on cherche à chaque itération à diviser en régions plus petites des régions de l'image considérées comme non homogènes en fonction

d'un certain critère. Les problèmes avec ce type d'algorithmes sont identiques à ceux rencontrés avec les méthodes de fusion.

Les algorithmes de séparation/fusion de régions, quant à eux, combinent les deux techniques précédentes [HOR-76].

Il est avantageux d'utiliser les méthodes de segmentation en régions plutôt que des méthodes de segmentation par recherche de contours lorsque l'on dispose d'images comportant des contours bruités, et lorsque les objets sur lesquels on travaille sont plus facilement caractérisés par leur texture ou leur couleur que par leurs contours.

C. La détection de contours

La détection de contours s'effectue toujours en deux temps :

- détection des points de contours
- chaînage des points de contours.

La détection des points de contours met en évidence les frontières dans l'image, mais ces frontières ne correspondent pas forcément à des objets réels. Le vocable contour sera désormais utilisé pour désigner uniquement ces frontières de l'image.

Le chaînage des points de contours a pour but de regrouper les points de contours sensés constituer la frontière d'objets réels. Elle permet notamment d'éliminer les détections dues au bruit contenu dans les images.

1. La détection des points de contours

Les points de contours sont caractérisés par des variations locales importantes de la fonction intensité de l'image I(x,y). Dans les méthodes classiques, ces discontinuités de la fonction intensité sont mises en évidence grâce à l'utilisation d'opérateurs différentiels, comme le gradient ou le laplacien.

Dans le cas d'une image idéale non bruitée, les points de contours sont caractérisés par des maxima locaux de la norme du gradient de l'intensité I(x,y). L'expression du gradient est donnée par :

$$\nabla I(x_0, y_0) = \begin{pmatrix} \frac{\partial I(x,y)}{\partial x} \\ \frac{\partial I(x,y)}{\partial y} \end{pmatrix}_{(x_0,y_0)}$$

ou par des passages par zéro du laplacien de l'intensité donné par :

$$\Delta I(x_0,y_0)=\left(\frac{\partial^2 I(x,y)}{\partial x^2}\right)_{(x_0,y_0)}+\left(\frac{\partial^2 I(x,y)}{\partial y^2}\right)_{(x_0,y_0)}$$

Ces expressions ne sont valables que pour des fonctions continues.

On se trouve donc confronté en pratique à deux problèmes :

- la discrétisation de l'intensité image
- le bruit contenu dans les images.

Il faut donc utiliser des méthodes de calcul des dérivées de l'intensité qui éliminent ces problèmes, du moins en partie. Parmi les méthodes disponibles, mentionnons l'utilisation de masques de convolution, la méthode de Haralick d'interpolation de la fonction intensité, le détecteur optimal de Canny-Deriche, et enfin le détecteur optimal de Wan.

a) Les masques de convolution

Les masques de convolution permettent de calculer une approximation du gradient ou du laplacien. Pour détecter les contours, on cherche soit à obtenir une approximation de la valeur du gradient en tout point, c'est à dire des quantités $\frac{\partial I}{\partial x}(x, y)$ et $\frac{\partial I}{\partial y}(x,y)$, puis à localiser les maxima locaux de la norme du gradient [ROS-82], soit à établir une approximation du laplacien afin de rechercher ses passages par zéro ou, en raison de l'aspect discret, ses changements de signe [TOR-86].

Pour le calcul du gradient, on peut utiliser par exemple les masques de convolution de Prewitt [PRE-70] :

$$f_x=\begin{bmatrix}1 & 0 & -1\\ 1 & 0 & -1\\ 1 & 0 & -1\end{bmatrix} \text{ et } f_y=\begin{bmatrix}1 & 1 & 1\\ 0 & 0 & 0\\ -1 & -1 & -1\end{bmatrix}.$$

Le masque fx fournit une approximation de $\frac{\partial I}{\partial x}(x, y)$ et permet de détecter les contours verticaux alors que le masque fy fournit une approximation de $\frac{\partial I}{\partial y}(x, y)$ et permet de détecter les contours horizontaux. Ce masque est couramment utilisé pour son effet de lissage et permet de conserver une bonne localisation du contour.

Pour le calcul du laplacien, on utilise par exemple les masques :

$$f=\begin{bmatrix}0 & 1 & 0\\ 1 & -4 & 1\\ 0 & 1 & 0\end{bmatrix} \text{ ou } f=\begin{bmatrix}-1 & -1 & -1\\ -1 & 8 & -1\\ -1 & -1 & -1\end{bmatrix}.$$

L'extraction des points de contours peut être suivie ou précédée d'un lissage destiné à éliminer le bruit contenu dans les images. Le lissage le plus simple consiste à convoluer l'image avec un masque moyenneur qui affecte au pixel central la valeur moyenne de l'intensité sur la fenêtre. L'expression d'un masque de moyennage de taille 3x3 est la suivante :

$$f_m = \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Un autre lissage couramment utilisé consiste à convoluer la fonction intensité de l'image I(x, y) avec une fonction Gaussienne g(x, y, σ), dont l'expression est :

$$g(x,y,\sigma) = \frac{1}{2\pi\sigma^2} e^{-\left(\frac{x^2+y^2}{2\sigma^2}\right)}$$, où σ est l'écart type.

Pour ces deux masques, une augmentation de la taille du masque de moyennage ou de l'écart type σ du filtre gaussien accentue l'effet de lissage, au détriment de la netteté des frontières. La localisation des contours sera par conséquent d'autant moins précise que l'on aura mieux éliminé le bruit.

b) La méthode de Haralick

Pour éviter les problèmes d'instabilité numérique de la dérivation discrète, différents auteurs ont proposé d'interpoler la fonction intensité par une fonction dérivable [HUE-73], [HAR-84].

Haralick a proposé d'utiliser comme fonction d'interpolation un polynôme de Tchebicheff d'ordre 3, en approchant I(x, y) par une fonction de la forme :

$$I(x,y) = k_1 + k_2 x + k_3 y + k_4 x^2 + k_5 xy + k_6 y^2 + k_7 x^3 + k_8 x^2 y + k_9 xy^2 + k_{10} y^3$$

Chaque coefficient du polynôme est calculé par convolution de l'image avec un masque particulier. A titre d'exemple, le masque $\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$ équivalent au filtre moyenneur, permet de calculer le premier coefficient.

Le masque $\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$, qui est un détecteur de contours verticaux, permet de calculer le deuxième coefficient.

Cette méthode permet d'éliminer en partie l'instabilité due à la dérivation discrète, mais tout comme les méthodes précédentes, elle a pour inconvénient d'être sensible au bruit. En effet, en l'absence de pré-traitement de l'image, on calcule une approximation polynomiale d'une fonction intensité bruitée.

c) Le détecteur optimal de Canny-Deriche

Canny a cherché à obtenir un détecteur de contours qui vérifie les critères suivants [CAN-86] :

- la probabilité de détecter un vrai point de contour doit être maximale, alors que la probabilité de détecter un faux point de contour (détection parasite due au bruit) doit être minimale. Ceci revient à maximiser le rapport signal sur bruit, en considérant comme signal la réponse du détecteur.
- la localisation du point de contour doit être aussi précise que possible.

Le filtre de Canny est un filtre séparable en x et en y. On utilise successivement le filtre en x et en y. L'expression, selon x, du filtre optimal qui maximise ces critères est donnée par : $f(x) = a\, e^{\alpha|x|} \sin(\omega x), \alpha < 0, \omega > 0$

Le paramètre α permet de régler la largeur du filtre, c'est à dire le rapport détection/localisation : une valeur élevée de α permet une meilleure localisation alors qu'une valeur faible permet une meilleure détection.

Deriche a donné une solution exacte de l'équation de Canny. Le filtre de Deriche est le suivant : $s(x) = k(\alpha|x| + 1)e^{-\alpha|x|}$

Le paramètre k est un facteur de normalisation. Son expression est : $k = \dfrac{\left(1 - e^{-\alpha}\right)^2}{1 + 2ae^{-\alpha} - e^{-2\alpha}}$

Ce filtre, de réponse impulsionnelle infinie, est implémenté récursivement [DER-87].

d) Le détecteur hyperbolique de Wan

Un des projets du programme PROMETHEUS, « **Pro**gramme for **E**uropean **T**raffic with **H**ighest **E**fficiency and **U**nprecedented **S**afety », consiste à détecter des obstacles devant les véhicules par traitement des images fournies par des caméras placées à l'avant du véhicule. Y. F. Wan a développé dans le cadre de ce programme un nouveau filtre récursif dont les performances sont meilleures que celles du filtre de Deriche [WAN-95]. L'expression selon x de ce nouveau filtre est donnée par : $f(x) = e^{-\alpha|x|} \cdot \sinh(\beta \cdot x)$

e) Élimination des fausses détections dues au bruit

Le bruit ne pouvant être complètement éliminé de l'image par l'opération de lissage, certains points détectés par les divers opérateurs différentiels ne sont pas nécessairement des points de contours réels. Il est alors nécessaire d'éliminer les fausses détections dues au bruit, et de rassembler les points qui appartiennent au même contour :

- **par seuillage :** les faux points de contours dus au bruit sont pour la plupart caractérisés par de faibles valeurs du module du gradient. Un simple seuillage permet par conséquent de les éliminer [CAN-86]. On se trouve alors confronté au problème désormais courant du choix d'un seuil :

  $\Rightarrow$ un seuil trop bas n'élimine pas tous les faux points de contours,

  $\Rightarrow$ un seuil trop haut élimine par contre de vrais points de contours.

- **par relaxation :** en faisant l'hypothèse que les points de contours réels des objets ne sont pas isolés, on accorde une certaine confiance à un point détecté comme point de contour en fonction de ses voisins : si ses voisins sont des points de contour, la confiance en ce point augmente, dans le cas contraire, elle diminue. La confiance est une mesure de la probabilité qu'a un point d'être un vrai point de contour. Elle est comprise entre 0 et 1, 0 correspondant à un point dont la probabilité d'appartenir à un contour est nulle et 1 correspondant à un point ayant la probabilité maximale d'appartenir à un contour [PRA-80], [ROS-76]. La relaxation est une procédure itérative, dont la convergence n'est pas garantie pour des images complexes.

2. Modélisation des contours

Il est nécessaire de regrouper les points de contours obtenus à l'étape précédente en chaîne de contours. En effet, les points de contours représentent une grande quantité d'information qui n'est pas structurée et donc inutilisable directement. Une description plus intéressante est constituée par l'ensemble des segments de contours représentant une partie du contour des objets.

Pour obtenir cette description, deux étapes sont nécessaires :

- chaînage des points de contour,
- modélisation des chaînes par approximation, généralement polygonale.

A ces deux étapes, on peut ajouter l'élaboration d'un graphe d'adjacence contenant les relations de voisinage entre ces segments.

Chaînage des points de contours

Le chaînage des points de contours, appelé également suivi de contours, consiste à regrouper les points de contours connexes dans des chaînes de contour. Il permet de diminuer la quantité d'information puisque l'on passe d'une représentation matricielle des contours à une représentation sous forme de listes chaînées.

Les méthodes de chaînage utilisent la définition suivante d'un contour : les points appartenant au contour d'un même objet sont des points connexes dont la valeur du gradient de l'intensité est élevée. La méthode la plus simple consiste donc en une binarisation de l'image, suivie d'une procédure d'amincissement du contour par l'utilisation de fonctions morphologiques binaires spécifiques. Cependant, le choix du seuil est très souvent délicat, en particulier lorsque l'éclairage de l'image n'est pas uniforme.

Des méthodes plus robustes ont donc été proposées, travaillant directement sur l'image gradient en niveaux de gris [SAG-81], [TEH-89]. Une méthode de chaînage consiste par exemple à sélectionner les points de l'image gradient correspondant à un maximum local. A partir de ces points sélectionnés, notés P_i, on cherche si leur voisinage comporte des points de contour. Cette recherche s'effectue dans la direction orthogonale à l'orientation du gradient au point considéré. Il se peut que dans cette direction, plusieurs pixels voisins, notés P_j, puissent être considérés comme des points de contour. Il faut donc les discriminer. Pour cela, il est en général nécessaire d'introduire une fonction de coût. Cette fonction de coût doit prendre en compte la valeur du gradient du point P_j, l'écart entre les valeurs des gradients des points P_i et P_j, ainsi que l'écart entre les directions des gradients aux points P_i et P_j, afin de pénaliser les changements de direction dans le contour.

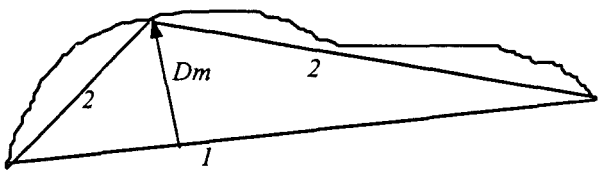
Si la fonction de coût ne permet pas de discriminer les points P_j, il faut les utiliser tous pour poursuivre la recherche, et à partir du résultat obtenu, revenir au choix du point P_j. L'algorithme de suivi s'arrête lorsque l'on rencontre un autre point du même contour, c'est à dire lorsque l'on a obtenu une chaîne fermée.

Si il n'y a pas de points de contours dans le voisinage d'un point Pi, il est alors nécessaire d'élargir la taille du voisinage pour la recherche des points Pj. L'utilisation d'un voisinage étendu rend le problème encore plus complexe.

Une autre méthode de chaînage consiste à utiliser le graphe d'adjacence des points de contour de l'image. Un contour est alors défini comme un chemin de coût minimum dans ce graphe. Une des difficultés de ce type de méthode consiste en le choix de la fonction de coût utilisée pour trouver le chemin optimal. La minimisation de la fonction de coût peut ensuite s'effectuer par programmation dynamique comme dans [TAN-91].

Modélisation des contours

Afin de diminuer encore la quantité d'information, les chaînes de contours sont généralement modélisées en segments par un algorithme d'approximation polygonale. L'algorithme d'approximation polygonale le plus connu est celui de Pavlidis [PAV-77]. En considérant le segment de droite reliant le premier et le dernier point d'une chaîne, on calcule la distance maximale *Dm* séparant le point le plus éloigné de la chaîne du segment. Si cette distance est supérieure à un certain seuil, on découpe la chaîne initiale en deux au point où la distance *Dm* est atteinte (*cf.* figure I-2). On réitère cet algorithme sur chacune des nouvelles chaînes jusqu'à obtenir des distances *Dm* inférieures au seuil fixé.



1 Segment initial
2 Segments après une itération
Figure I-2 : Algorithme d'approximation polygonale

Une autre méthode d'approximation polygonale consiste à rechercher le polygone circonscrit aux points chaînés, de périmètre minimal. Ce polygone est également appelé enveloppe convexe. Le coût en calcul de ce type d'algorithmes est en n.log(n), où n est le nombre de sommets du polygone [TOU-82], [ORL-83].

D'autres types de méthodes consistent à modéliser les chaînes par des arcs de cercle.

On voit donc que pour disposer d'un modèle compact des objets, trois étapes sont nécessaires : la détection des points de contour, le chaînage de ces points et enfin une modélisation de ces chaînes.

D. La combinaison des deux approches

Les méthodes de segmentation de type « régions » et « contours » étant complémentaires, certains auteurs ont voulu les combiner afin d'améliorer la qualité de la segmentation. Cette approche consiste à tenir compte des contours afin de ne pas fusionner des régions dont la frontière comporte trop de points de contours, comparativement à un seuil déterminé.

Ceci permet par exemple d'éviter la création de régions non significatives lors de la segmentation d'images contenant des dégradés de lumière. En effet, les limites artificielles issues de la segmentation en régions ne contiennent pas de points de contours, ce qui fait que les régions correspondantes peuvent être fusionnées [WRO-87].

E. Conclusion

Les méthodes de segmentation en régions impliquent en général des temps de calcul élevés, peu adaptés à des traitements de séquences d'images orientés temps réel. Elles permettent difficilement le suivi des objets au cours d'une séquence, car la segmentation en régions est peu stable : on n'obtient pas nécessairement le même nombre de régions d'une image à l'autre, et leurs caractéristiques varient également d'une image à l'autre.

De plus, comme les régions ne représentent pas forcément des objets, elles ne permettent pas non plus une reconstruction ultérieure de la scène segmentée, sauf si l'on dispose de connaissances *a priori* sur les objets. Toutes ces raisons font que nous avons préféré privilégier la détection des contours plutôt celle des régions.

Cependant, tout comme les méthodes de segmentation en régions, les méthodes classiques de détection des contours ont pour inconvénient majeur de travailler à un niveau local. Elles ne prennent en compte la propriété majeure des contours d'être des courbes continues. que dans la phase de chaînage. En effet, l'étape de chaînage s'effectue souvent séquentiellement à partir d'un point reconnu comme point de contour vers un voisinage très proche, et conserve donc un caractère local. C'est notamment pour éviter cet inconvénient qu'a été proposé le modèle des contours actifs que nous présentons dans le chapitre IV.

Chapitre II

La segmentation dynamique

A. Introduction

La segmentation dynamique a pour but l'extraction des objets en mouvement dans une séquence d'images, afin de pouvoir analyser leur mouvement et notamment de connaître leur vitesse et la direction de leur déplacement [AGG-75].

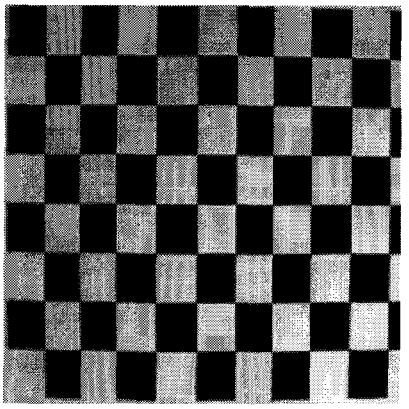
Les méthodes de segmentation dynamique les plus simples à mettre en oeuvre restent de loin celles basées sur la différence d'images [YAL-82]. De plus, elles satisfont l'aspect temps réel qui conditionne fortement notre étude.

Le principe de base de ces méthodes consiste à détecter les zones affectées par le mouvement des objets entre deux images, en calculant la différence pixel à pixel des deux images. Pour que les différences entre images proviennent uniquement du mouvement des objets, une hypothèse commune à toutes ces méthodes est que **la caméra de prise de vue soit fixe.**

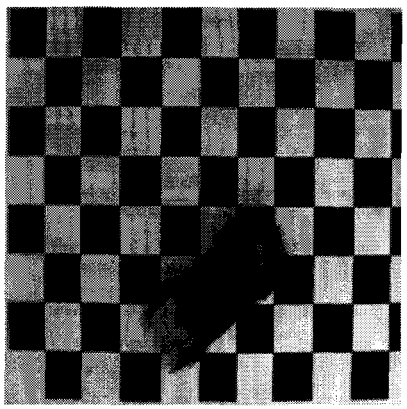
La diversité des méthodes réside dans le choix des images de la séquence utilisées pour effectuer l'opération différence. On peut mettre en évidence le mouvement en faisant, par exemple, la différence entre l'image courante et une image de référence. Ainsi, tout élément de l'image courante qui n'était pas présent dans l'image de référence est considéré comme un objet mobile [NAG-83].

La difficulté réside dans l'acquisition de l'image de référence, qui doit être prise à un moment où la scène ne comporte aucun objet mobile, ce qui n'est pas toujours possible. D'autre part, l'éclairage peut varier au cours de la séquence de telle sorte que la différence entre l'image courante et l'image de référence met en évidence non seulement les parties mobiles, mais également les variations d'éclairage. Quoique de nombreuses solutions aient été proposées pour pallier ces deux principaux problèmes afin de disposer d'une image de référence idéale, aucune n'est totalement efficace ni fiable sans hypothèses spécifiques.

De plus, si le fond n'est pas uniforme, la différence entre une image de la séquence et l'image de référence fait apparaître ce fond, comme l'illustre la figure II-1.



a) Image de référence



b) Image courante

c) Différence de ces deux images

Figure II-1 : Différence d'images avec image de référence.

Une autre méthode consiste à calculer la différence entre l'image courante et ses voisines immédiates dans la séquence. Ce type de méthode est moins sensible aux variations temporelles de l'éclairage, qui sont généralement faibles entre deux instants successifs de prise de vue [YAL-82].

B. Segmentation par la différence de deux images successives

La méthode la plus simple et la plus rapide consiste à calculer les différences entre toutes les paires d'images successives, et à les binariser. Cependant, chaque image différence entre l'image courante, notée C, et celle qui la précède, notée P, contient des régions de types différents qu'il est nécessaire de caractériser pour pouvoir retrouver les objets en mouvement. Diverses méthodes ont été proposées pour reconstruire ces mobiles [JAI-79a], [HAY-83]. Nous illustrons le problème de cette caractérisation par l'algorithme proposé par Jain et *al.* dans [JAI-79b]. Cet algorithme peut être décomposé en quatre phases :

- binarisation de l'image différence D(P,C) : B[D(P,C)]
- analyse et classification en trois types des régions d'intensité non nulle de l'image binaire B[D(P,C)].
- reconstruction de l'objet en mouvement selon le type de régions identifié dans l'image différence. L'objet en mouvement est reconstitué soit par agrandissement, soit par réduction de ces régions.
- affinement et vérification des résultats. Plusieurs règles heuristiques peuvent être appliquées à l'objet identifié afin d'améliorer la précision et de vérifier si sa forme est compatible avec l'ensemble des informations disponibles dans toute la séquence d'images.

L'efficacité de cette méthode repose sur la pertinence de la classification issue de la première phase. Cet algorithme a été baptisé Frog par référence au système de vision des batraciens, qui n'est sensible qu'au mouvement des objets présents dans le champ de vision.

1. Classification des régions

La classification des régions de l'image différence est nécessaire car ces régions ont des origines différentes, comme l'illustre l'exemple simple de la figure II-2. La région L résulte du recouvrement du fond par l'objet au cours de son déplacement, alors que la région T résulte du découvrement du fond par l'objet lors de son déplacement entre les images P et C.

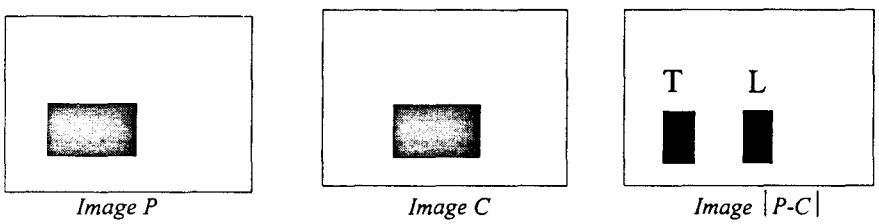


Figure II-2 :Obtention de région T et L par recouvrement et découvrement du fond.

Les auteurs ont proposé un algorithme qui permet de déterminer si une région est du type T ou L dans de nombreux cas. Pour cela, ils définissent les points extrêmes d'une région comme étant les pixels situés sur le bord de la région.

Pour chaque région de D(P,C), on calcule CC (respectivement CP) le nombre de points extrêmes qui sont aussi des points extrêmes de la région correspondante de C (respectivement P). Puis on définit le rapport :

$$CURPRE=CC/CP$$

Ce rapport permet de distinguer 3 types de régions:

- les régions de type L correspondent à CURPRE>>1 et permettent de reconstruire l'objet en mouvement dans la position qu'il occupe dans l'image C;
- les régions de type T correspondent à CURPRE<<1 et permettent de reconstruire l'objet en mouvement dans la position qu'il occupait dans l'image P;
- les régions de type X correspondent à une valeur de CURPRE proche de 1 et résultent souvent de l'association de plusieurs zones de types T et L, comme le montre la figure II 3.

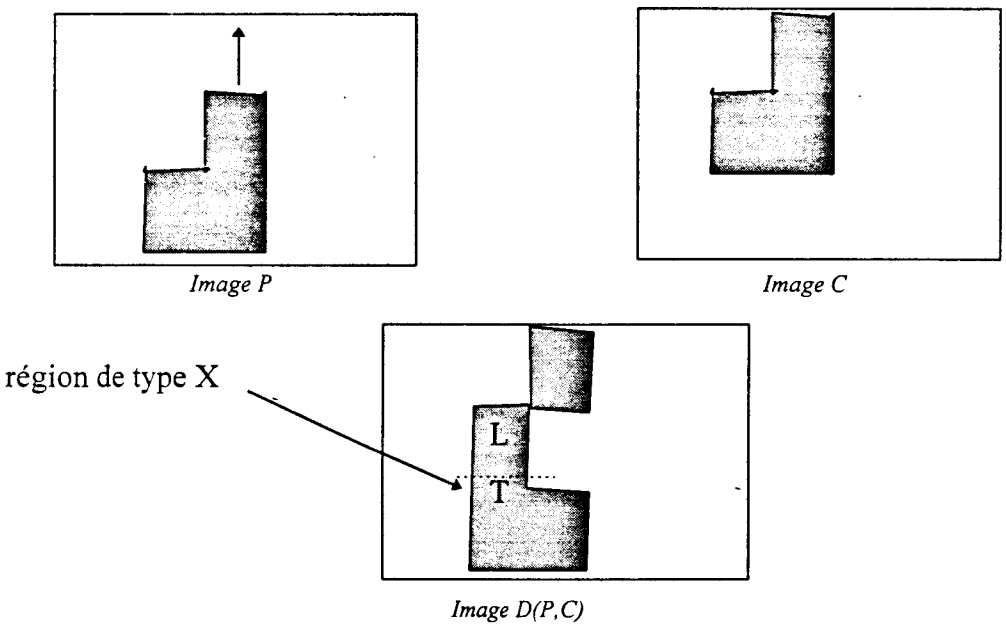


Figure II-3: Obtention de régions de types T, L et X par la différence de deux images successives.

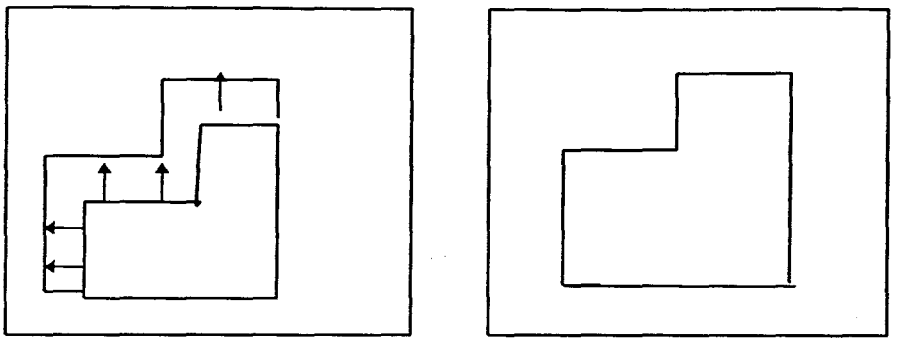
2. Reconstruction des objets en mouvement

Selon le type des régions T, L ou X, deux méthodes de reconstruction ont été proposées :

- une méthode de reconstruction par agrandissement,
- une méthode de reconstruction par réduction.

Une méthode classique d'accroissement de région permet de reconstruire l'objet en mouvement dans sa position dans l'image courante (région L), ou dans l'image précédente (région T). Un exemple de croissance d'une région L est donné figure II-4.

Pour les régions de type X, le problème est fondamentalement différent car par hypothèse ce sont des combinaisons de régions de types L et T. Le contour de l'objet dans l'image précédente ou courante est déterminé à partir des coordonnées des pixels de la région X par un algorithme similaire mais opérant par réduction.



Croissance d'une région L *Position de l'objet dans l'image C*

Figure II-4 : Accroissement de région.

3. Vérification

Des post-traitements peuvent être appliqués aux images des objets ainsi obtenues afin d'affiner leur forme et de vérifier si elle est effectivement compatible avec l'ensemble de la séquence dynamique. La reconstitution de l'objet peut notamment être menée conjointement dans les images P et C afin de confronter les résultats obtenus.

4. Conclusion

On voit bien à travers cette méthode la complexité du problème de reconstruction de la scène dynamique découlant d'un traitement bas-niveau tel que la soustraction pixel à pixel de deux images.

L'avantage résultant de la rapidité de l'opérateur de détection des zones en mouvement est perdu en raison des traitements ultérieurs nécessaires à la reconstruction des objets. Pour cette raison des opérateurs plus élaborés ont été développés,qui conservent cet avantage.

C. Segmentation par des opérateurs spécifiques

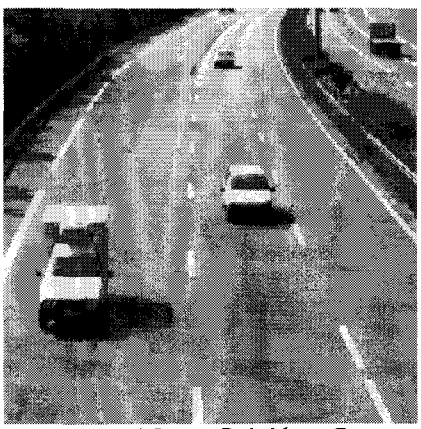
1. L'opérateur de Haynes et Jain

Cet algorithme, illustré figure II-5, utilise deux images successives. Ces images opérandes sont l'image courante notée C et l'image précédente notée P. Ces deux images successives contiennent les objets en mouvement dans leurs positions courante et précédente. On notera la valeur absolue de leur différence pixel à pixel D(P,C). Considérons les images C et P. L'image différence D(P,C) est donnée figure II-5c. Si l'on calcule le gradient de l'image courante G(C) avec l'opérateur de Sobel, on dispose des contours de tous les objets présents dans l'image courante, qu'ils soient statiques ou mobiles (*cf.* figure II-5d).

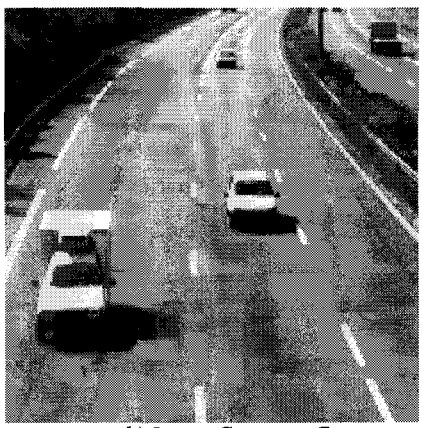
En multipliant l'image D(P,C) par l'image G(C), on obtient une image résultat qui ne contient plus, théoriquement, que les points de contours des objets en mouvement dans leur position courante [HAY-83]. L'opérateur CM(C) de Haynes et Jain de détection des **C**ontours en **M**ouvement dans l'image **C**ourante s'écrit ainsi :

$$CM(C) = D(P,C) \times G(C)$$

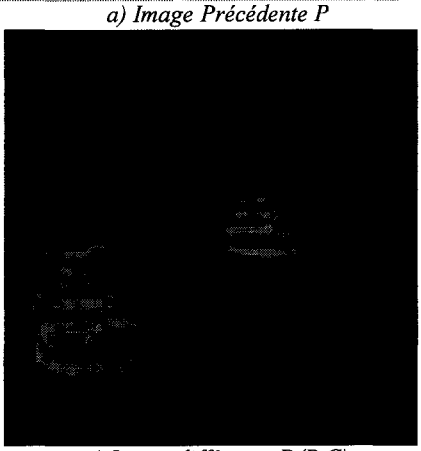
Son application aux images de la figure II-5a et II-5b donne l'image de la figure II-5e.



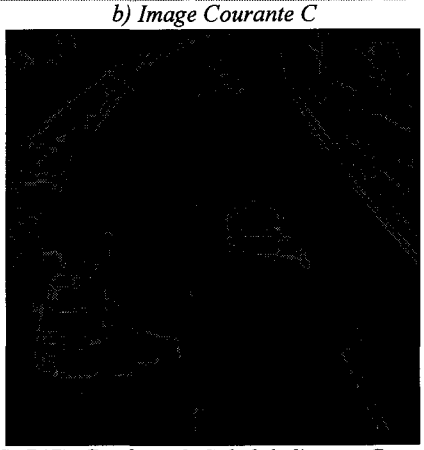
a) Image Précédente P



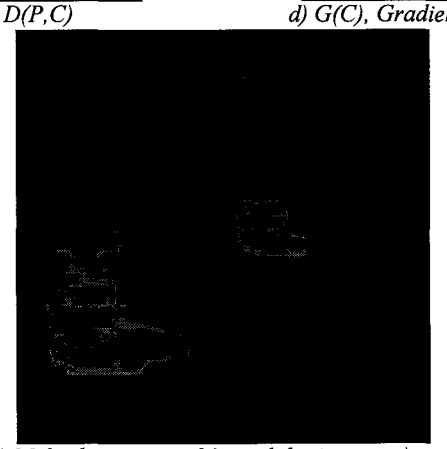
b) Image Courante C



c) Image différence D(P,C)



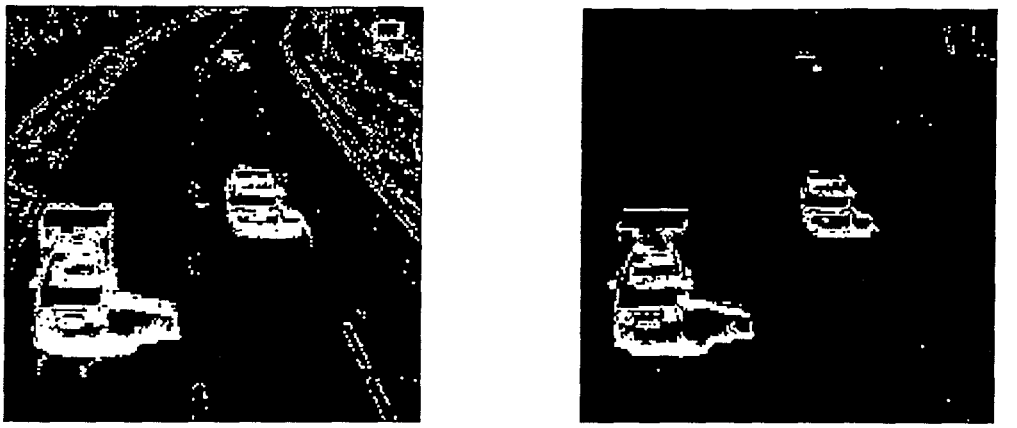
d) G(C), Gradient de Sobel de l'image C.



e) Multiplication pixel à pixel des images c) et d)

Figure II-5 : Opérateur CM(C) de Haynes et Jain

En binarisant cette image CM(C) afin de faire apparaître la plupart des points détectés par cet opérateur, on obtient l'image de la figure II-6a.



a) seuil de 5 sur 256 niveaux de gris *b) seuil de 25*

Figure II-6 : Binarisation de l'image CM(C)

On constate que cette image binaire contient de nombreux points de contours statiques parasites. Ces contours proviennent essentiellement de la différence d'images D(P,C), opération sensible au bruit. En effet, si l'on binarise l'image D(P,C) de la figure II-5c avec un seuil de 5, on obtient la figure II-7.

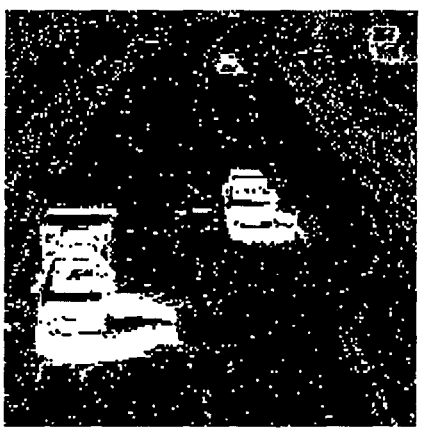


Figure II-7 : Binarisation de D(P,C) avec un seuil de 5

L'intensité de certains de ces points parasites dus au bruit est ensuite multipliée par l'intensité des points correspondants de l'image gradient, et ces points se retrouvent donc dans l'image résultat.

On pourrait binariser l'image résultat CM(C) de façon à éliminer les points de contours statiques qui ont un niveau de gris globalement plus faible que les points de contours en mouvement. Cependant, en utilisant un seuil de binarisation plus élevé, on élimine une partie des points de contours en mouvement. Le résultat de cette binarisation avec un seuil de 25 est donné figure II-6b.

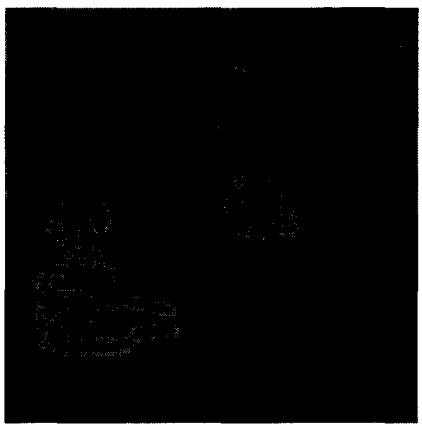
2. L'opérateur de Stelmaszyck

A la place de l'image différence de l'opérateur précédent de l'opérateur précédent, Stelmaszyck utilise le gradient de l'image différence D(P,C), combinant ainsi deux images de contours [STE-85].

Comme dans la méthode précédente, Stelmaszyck multiplie cette image par le gradient de l'image courante :

$$CM(C) = G(D(P,C)) \times G(C)$$

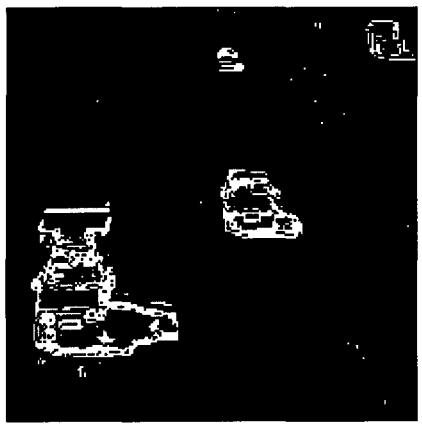
L'image résultant de cette multiplication est donnée sur la figure II-8b.



a) Gradient de Sobel de D(P,C)

b) Résultat de la multiplication des images G(D(P,C)) et G(C)

c) Binarisation de l'image b) avec un seuil de 5



d) Binarisation de l'image b) avec un seuil de 25.

Figure II-8 : Opérateur de Stelmaszyck

Si l'on binarise l'image de la figure II-8b avec le même seuil que pour l'opérateur de Haynes et Jain (seuil=5), afin de faire apparaître la plupart des points détectés, on obtient l'image binaire de la figure II-8c. Le seuil de binarisation nécessaire pour éliminer le bruit contenu dans l'image résultat est ici de 35. L'image binaire obtenue est donnée figure II-8d.

Tout comme avec l'opérateur précédent, on retient de nombreux contours statiques dus au bruit. Ces contours parasites proviennent du gradient de l'image différence, et sont ensuite multipliés par des points du gradient de l'image courante. Ce phénomène peut être mis en évidence sur nos images en binarisant le gradient de l'image différence avec un seuil bas (seuil=5), de façon à faire apparaître ces points parasites. On obtient ainsi l'image de la figure II-9.

Cet opérateur, tout comme avec le précédent, pose un problème quand un objet mobile laisse apparaître, en se déplaçant, un fond non uniforme. Les contours de cette partie du fond qui est cachée dans l'image précédente mais est présente dans l'image courante, seront présents dans la différence d'images, et donc dans l'image résultat. Ils seront donc considérés comme des contours mobiles. L'utilisation de trois images successives permet de pallier cet inconvénient. C'est ce que propose l'opérateur suivant.

Figure II-9 : Binarisation de D(C, P)) avec un seuil de 5

3. L'opérateur de Vieren

Vieren [VIE-88] a développé un opérateur qui permet d'extraire les contours extérieurs d'objets en mouvement devant un fond uniforme ou non. Celui-ci est illustré par la figure II-10. Pour cela, trois images successives de la scène dynamique sont nécessaires :

- l'image précédente, notée P,
- l'image courante, notée C,
- l'image suivante, notée S.

Deux images différences sont calculées à partir de ces trois images : l'image différence, notée D(P,C), entre l'image courante et l'image précédente et l'image différence entre l'image suivante et l'image courante, notée D(C,S). Ces différences d'images permettent d'extraire les zones de l'image affectées par le mouvement.

Les contours des objets en mouvement sont ensuite détectés grâce à un opérateur de type gradient. Les deux images résultantes, notées G(D(P,C)) et G(D(C,S)), contiennent plusieurs types de contours :

- les contours extérieurs des objets en mouvement, dans les positions qu'ils occupent dans l'image courante, se retrouvent dans les images G(D(P,C)) et G(D(C,S)).
- les contours extérieurs des objets en mouvement dans les positions qu'ils occupent dans l'image précédente se retrouvent dans l'image G(D(P,C)), et les contours dans les positions qu'ils occupent dans l'image suivante se retrouvent dans G(D(C,S)).
- les contours du fond dans des zones de l'image qui ont été recouvertes puis découvertes par les objets en mouvement peuvent être présents dans plusieurs des trois images successives.

On effectue la multiplication point à point des deux images G(D(P, C)) et G(D(C, S)) afin de ne garder parmi ces trois types de contours que ceux correspondant à la position des objets en mouvement dans l'image courante. En effet, seuls ceux-ci sont **communs à ces deux images**. Les contours du fond étant éliminés lors de la multiplication, la méthode permet donc de détecter des objets en mouvement devant un **fond non uniforme**. L'expression de CM(C), l'opérateur de détection des **C**ontours en **M**ouvement dans l'image **C**ourante, est ainsi donnée par :

$$CM(C) = G(D(P, C)) \times G(D(C, S))$$

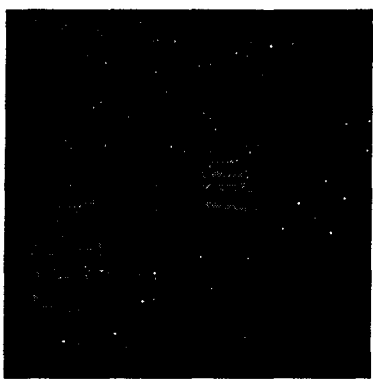
L'application de cet opérateur donne l'image de la figure II-10h.

La binarisation de l'image de la figure II-10h avec un seuil de 5 donne l'image de la figure II-11.

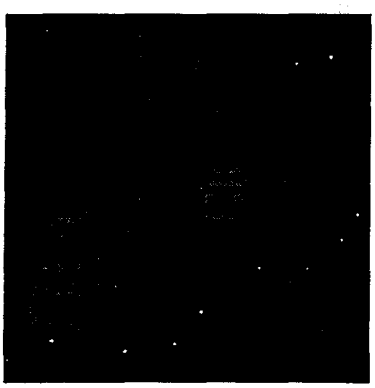
On constate que, contrairement aux opérateurs précédents, cet opérateur ne détecte pratiquement aucun point erroné, malgré un seuil de binarisation très faible. Cet opérateur fournit donc des résultats bien meilleurs que les précédents. Ceci est dû au fait que la multiplication des images gradient atténue l'amplitude des points de bruit contenus dans chacune de ces deux images, car il y a peu de chances que les points parasites apparaissent aux mêmes endroits.

Par contre, l'opérateur conserve les contours statiques situés à l'intérieur des contours extérieurs des objets en mouvement. Cet inconvénient n'est pas gênant pour notre application car nous n'exploitons que les contours **extérieurs** des objets pour leur modélisation à l'aide de contours actifs.

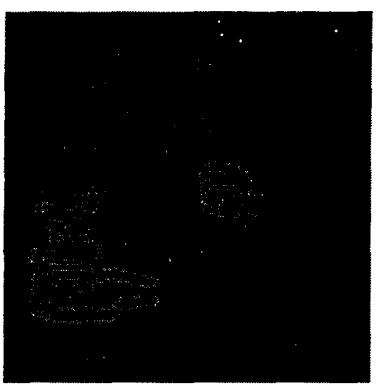
a) Image Précédente *b) Image Courante* *c) Image Suivante*



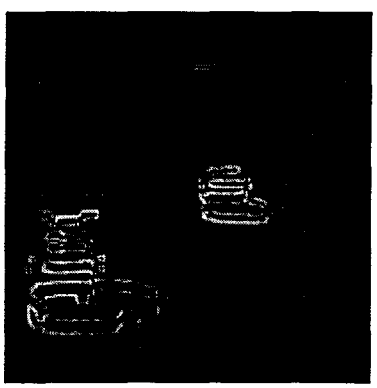
d) D(P,C)



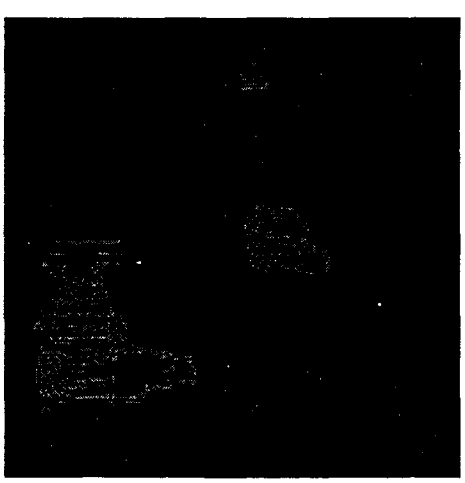
e) D(C,S)



f) G(D(P, C))



g) G(D(C, S))



h) CM(C)

Figure II-10 :Opérateur de Vieren

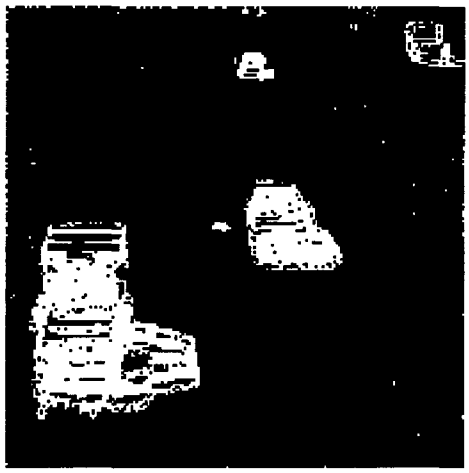


Figure II-11 : Binarisation de CM(C)

Le second inconvénient de l'opérateur est le retard d'une image introduit dans le traitement. En effet, il est nécessaire de disposer de l'image suivante pour extraire de l'image courante les contours des objets en mouvement.

Au niveau coût de calcul, l'opérateur de Vieren présente un avantage sur l'opérateur de Stelmaszyck puisque l'image G(D(C,S)) peut être réutilisée pour le calcul de l'opération de détection appliquée à l'image suivante, G(D(C,S)) devenant G(D(P,C)).

Notons que l'opérateur a d'ailleurs été implanté sous la forme d'un processeur câblé nommé S.T.R.E.A.M fournissant le résultat CM(C) en temps réel [CAB-92].

4. L'opérateur d'Orkisz

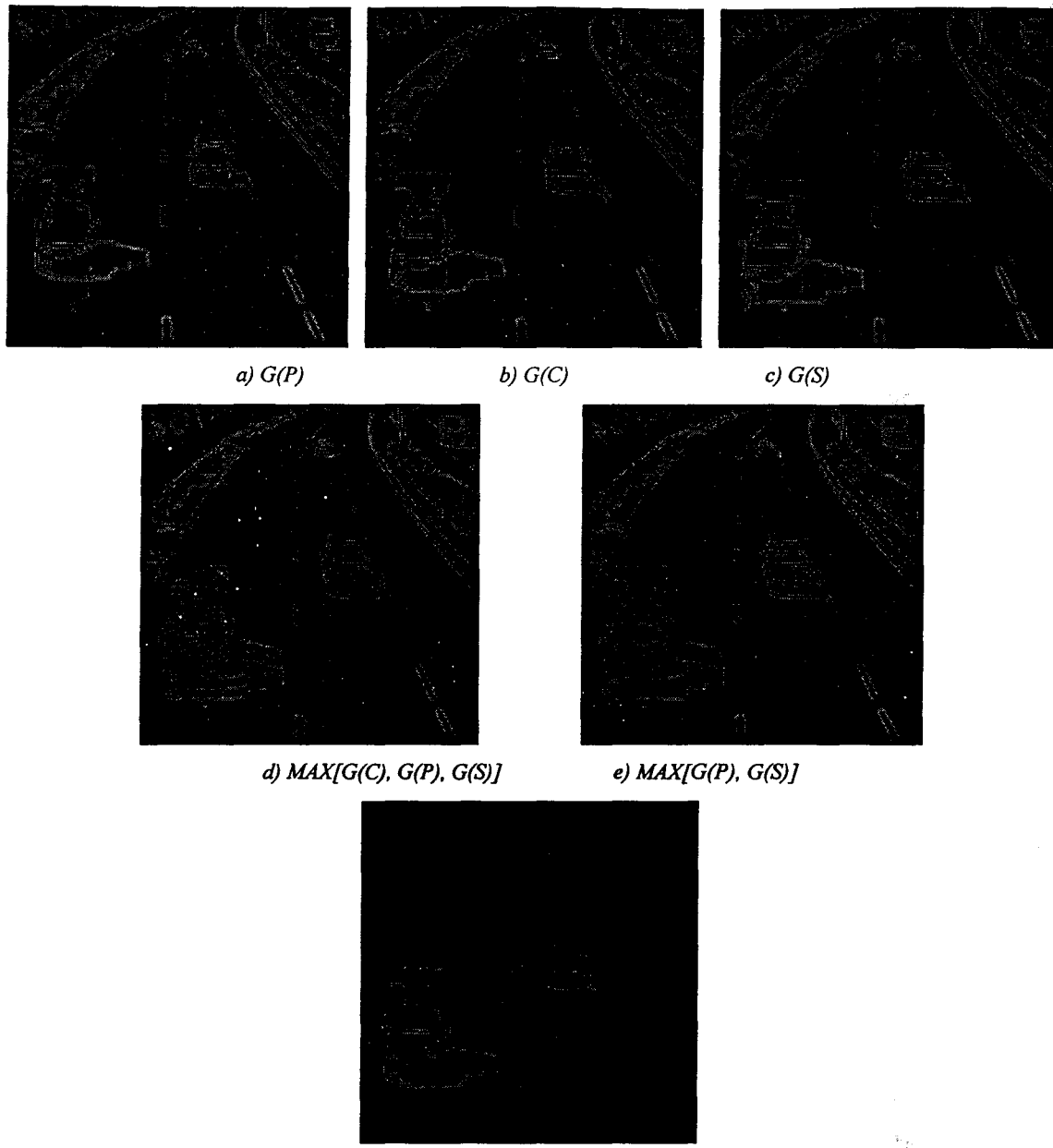
La multiplication introduisant un changement de dynamique dans les images résultats, Orkisz a préféré travailler avec un opérateur qui ne modifie pas cette dynamique [ORK-92]. C'est pourquoi il a choisi l'opérateur Max comme opérateur de coïncidence entre deux images. L'opérateur Max appliqué à deux images consiste à remplacer le niveau de gris de chaque pixel par le niveau de gris Maximal de ce pixel dans les deux images.

Orkisz s'est également attaché à construire un opérateur qui ne détecte pas les contours statiques présents à l'intérieur du contour extérieur des objets mobiles, contours provenant des éléments du fond qui sont découverts au cours du mouvement. Son opérateur est le suivant :

$$CM(C) = MAX[G(C), G(P), G(S)] - MAX[G(P), G(S)]$$

Cet opérateur utilise également trois images successives de la séquence, afin de s'affranchir des non-uniformités du fond.

Appliquons cet opérateur à nos images (*cf.* figure II-12)



a) G(P) *b) G(C)* *c) G(S)*

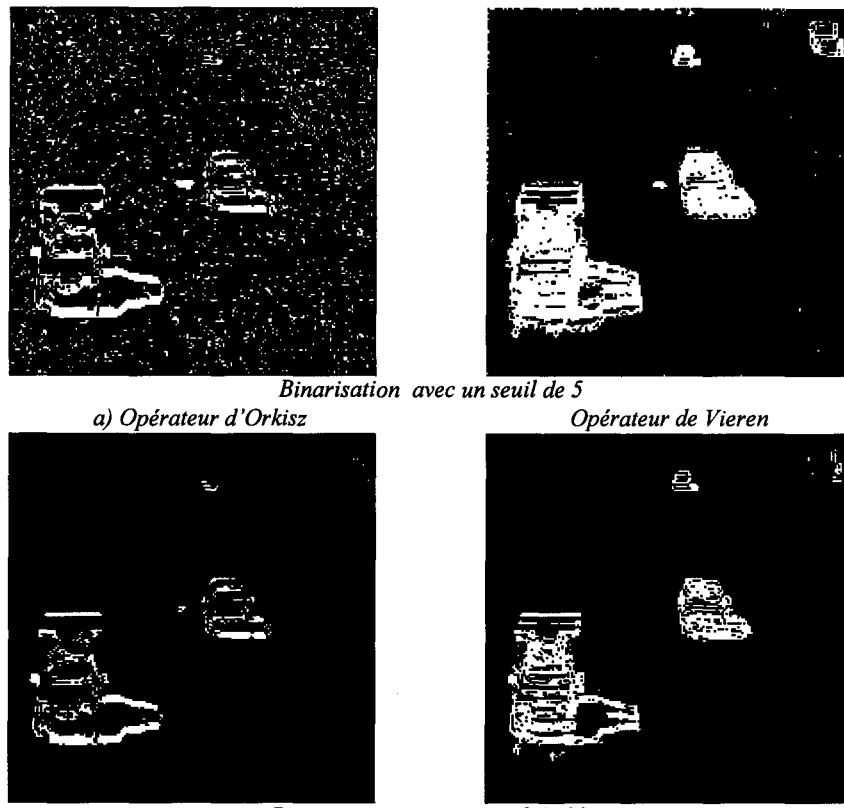
d) MAX[G(C), G(P), G(S)] *e) MAX[G(P), G(S)]*

f) CM(G)

Figure II-12 : Opérateur d'Orkisz.

Les contours des objets en mouvement sont plus visibles sur l'image binaire de la figure II-13c. La binarisation avec un seuil de 5 de l'image de la figure II-12f, destinée à mettre en évidence les détections parasites, fournit comme résultat l'image de la figure II-13a. On constate que le résultat est très bruité. Ceci est dû au fait que les opérateurs Max utilisés pour la coïncidence amplifient le bruit, et que la différence des deux images MAX[G(C), G(P), G(S)] et MAX[G(P), G(S)] ne permet pas d'éliminer ce bruit. On peut cependant binariser cette image de façon à éliminer le bruit en utilisant un seuil de 30 (*cf.* figure II-13c).

Notons qu'avec un tel seuil, l'opérateur de Vieren donne le résultat de la figure II-13d.



Binarisation avec un seuil de 5

a) Opérateur d'Orkisz *Opérateur de Vieren*

Binarisation avec un seuil de 30

c) Opérateur d'Orkisz *d) Opérateur de Vieren*

Figure II-13 : Comparaison des images CM(C) obtenues avec les opérateurs d'Orkisz et de Vieren

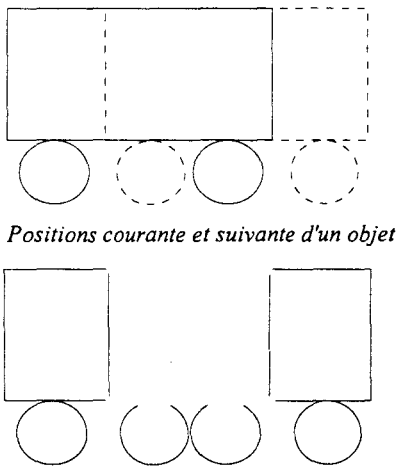
Les résultats de la binarisation des images de contours en mouvement issues des opérateurs de Vieren et d'Orkisz semblent montrer que l'opérateur de Vieren possède un meilleur rapport signal sur bruit que l'opérateur d'Orkisz. En effet, même avec un seuil très bas de 5, il y a beaucoup moins de fausses détections dues au bruit dans l'image binarisée de l'opérateur de Vieren que dans celle de l'opérateur d'Orkisz. De plus, le seuil de 30 nécessaire pour éliminer le bruit présent dans l'image des contours en mouvement obtenue par l'opérateur d'Orkisz élimine également une partie des contours des objets. Avec ce même seuil, il n'y a plus de détections parasites avec l'opérateur de Vieren, et les contours des objets en mouvement sont de meilleure qualité.

Remarquons que, tout comme l'opérateur de Vieren, l'opérateur d'Orkisz introduit un retard d'une image dans le traitement de la séquence, puisqu'il utilise aussi l'image suivante.

D. Conclusion

Les performances des différents opérateurs ont été étudiées par [CAB-92] et par [ORK-92]. Elles sont récapitulées de manière qualitative dans le tableau II-1. Au vu de ces résultats, les opérateurs de Vieren et d'Orkisz sont les plus performants, car ils éliminent en partie (Vieren) ou totalement (Orkisz) les contours statiques provenant d'un fond non uniforme. La présence de contours statiques à l'intérieur des contours extérieurs des objets en mouvement ne constitue pas pour notre application un inconvénient. Nous retiendrons donc par la suite l'opérateur de Vieren qui semble présenter une meilleure robustesse par rapport au bruit que celui d'Orkisz.

Signalons toutefois l'inconvénient que présentent tous les opérateurs précédemment cités, qui est la détection imparfaite des objets présentant des bords rectilignes, parallèles à la direction de leur mouvement, comme expliqué grâce à la figure II-14.



a) Positions courante et suivante d'un objet

Figure II-14 : Différence des images courante et suivante

La position courante de l'objet est représentée en traits pleins, et sa position dans l'image suivante est représentée en pointillés. Lors de l'opération différence, les contours dont les positions se chevauchent dans les deux images sont éliminés. Il ne reste plus que les pixels dont l'intensité a varié entre les deux images, c'est à dire les pixels des extrémités des segments rectilignes.

Les objets présentant des bords rectilignes parallèles à la direction du mouvement présentent donc des discontinuités dans les images différences, et dans le résultat final de la détection. Ce problème se rencontre notamment avec des séquences d'images comportant des voitures, type de séquences que nous avons à traiter. La méthode utilisée pour le suivi des objets en mouvement devra donc prendre en compte ces discontinuités éventuelles des contours.

La segmentation dynamique permet de mettre en évidence les objets en mouvement dans les séquences d'images. Il s'agit maintenant de modéliser les objets ainsi extraits, afin de les suivre tout au long de la séquence et de les localiser dans l'espace 3D. Le suivi des objets est réalisé grâce à des méthodes d'appariement temporel dans des séquences dynamiques. La localisation 3D est réalisée grâce à l'appariement stéréoscopique, c'est à dire l'appariement entre chaque paire d'images. Dans le chapitre suivant, nous présentons les différentes méthodes d'appariement temporel et stéréoscopique.

Opérateur	Temps de calcul	Précision de localisation des contours	Rapport signal sur bruit dans les images résultats	Problème du fond non uniforme
Haynes et Jain	tous les calculs sont à refaire à chaque image	mauvaise	faible	non résolu
Stelmaszyck	idem	bonne	faible	non résolu
Vieren	rapide car la moitié des calculs peut être réutilisée pour l'image suivante	bonne	élevé	résolution partielle (contours statiques à l'intérieur des contours en mouvement)
Orkisz	idem	bonne	élevé	résolu

Tableau II-1 : Performances des différents opérateurs de détection

Chapitre III

Les méthodes classiques d'appariement de primitives.

A. Introduction

La segmentation permet d'extraire des éléments caractéristiques des images tels que points de contour, segments, coins des objets, jonctions triples (point d'intersection de trois droites). A partir de ces éléments issus de différentes images, deux types de mise en correspondance, ou appariement, peuvent être recherchées, c'est pourquoi nous précisons ici la terminologie employée par la suite :

- dynamique ou temporel dans le cas de séquences monoculaires. Dans une séquence d'images contenant des objets en mouvement, ces éléments caractéristiques se déplacent, tout comme les objets dont ils constituent une représentation partielle. Afin de connaître et d'analyser le mouvement de ces objets, on peut chercher à mettre en correspondance ces éléments, appelés primitives, dans toutes les images de la séquence. En calculant les caractéristiques du mouvement de ces primitives, on peut alors connaître le mouvement des objets, en les reconstruisant à partir des primitives dont le mouvement est identique : même direction, même vitesse, etc. Ce type d'appariement est utilisé pour réaliser le suivi.
- statique ou stéréoscopique, dans le cas des images stéréoscopiques. Dans le cas de la vision stéréoscopique, apparier deux à deux les éléments caractéristiques issus de chacune des deux images constitue l'appariement stéréoscopique. Il permet de retrouver la position des objets par rapport aux caméras.

Le schéma de la figure III-1 illustre ces deux types d'appariements, dynamique pour suivre les objets en mouvement, statique pour localiser les objets dans l'espace 3D.

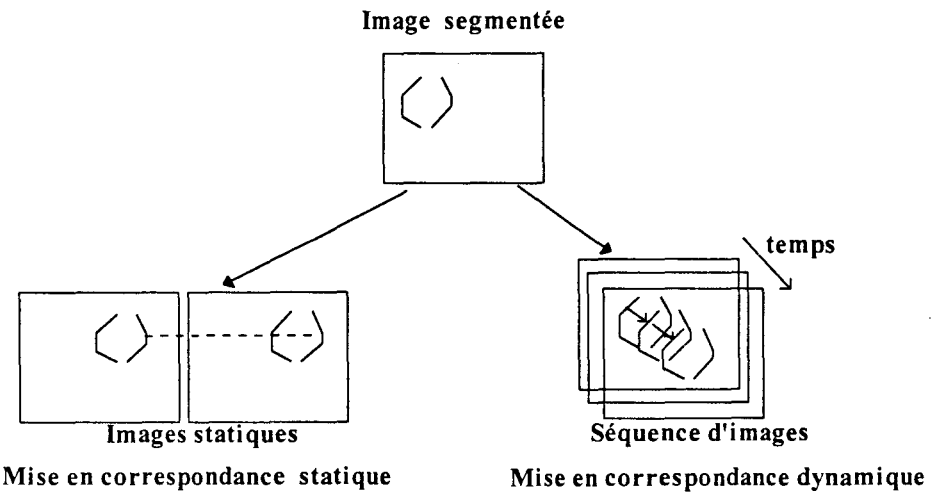


Figure III-1 : Appariement temporel et appariement stéréoscopique.

Nous allons, dans un premier temps, exposer diverses méthodes d'appariement temporel. Nous exposerons ensuite les méthodes d'appariement stéréoscopique.

B. L'appariement temporel

Dans le cas de la segmentation dynamique, la séparation en deux étapes d'extraction des primitives puis d'appariement temporel est parfois difficile à établir, car ces deux étapes peuvent interagir entre elles.

Nous scinderons la présentation des différentes approches en trois parties :

- l'approche globale;
- l'approche locale;
- l'approche synthétique.

1. L'approche globale

L'approche globale permet de déterminer la vitesse et la trajectoire d'un certain nombre d'éléments caractéristiques des objets en mouvement, comme les coins [MOR-79], ou les angles [DRE-81]. Les éléments caractéristiques des images sont généralement extraits grâce à un opérateur de bas niveau. Chacun des éléments caractéristiques est ensuite associé à un descripteur constitué d'un ensemble d'attributs géométriques ou topologiques, afin de faciliter leur discrimination et donc la phase ultérieure de mise en correspondance. Les éléments caractéristiques associés à leur descripteur sont appelés " tokens " en anglais (*cf.* figure III-2)

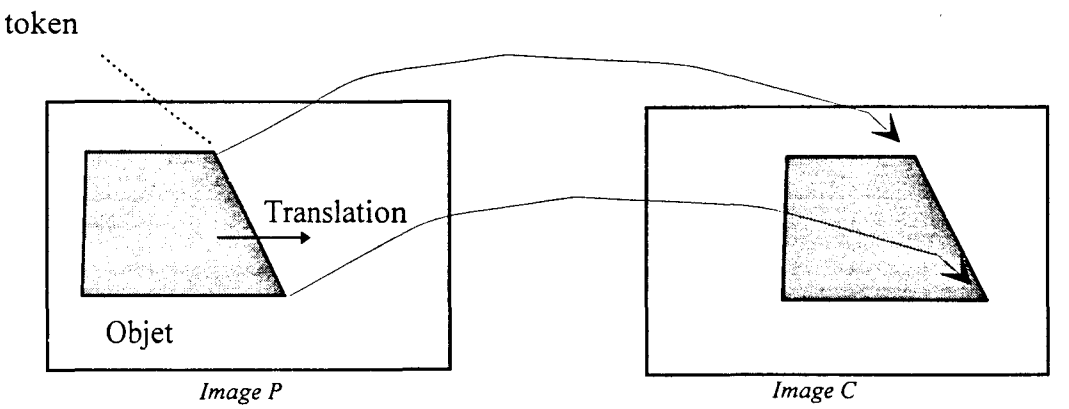


Figure III-2 : Appariement de coins

La plupart des algorithmes de suivi utilisant des éléments caractéristiques reposent sur les hypothèses suivantes :

- les objets sont supposés rigides;
- le mouvement des objets est supposé continu.

Ces hypothèses permettent d'utiliser un certain nombre de contraintes sur la position relative des éléments caractéristiques afin de reconstruire les objets auxquels ils appartiennent. Le principal problème qui se pose pour réaliser l'appariement est que le nombre des primitives n'est généralement pas constant d'une image à la suivante. On ne peut donc utiliser que les primitives que l'on retrouve dans toutes les images. Ainsi, ce type de méthodes ne permet de connaître le mouvement que d'un nombre réduit de primitives, qui de plus sont souvent trop éparses pour être significatives.

Parmi les différents algorithmes proposés pour réaliser l'appariement temporel, certains utilisent des méthodes de relaxation. Citons dans ce cadre l'algorithme de Sethi *et al.* [SET-88] pour lequel les éléments caractéristiques sont des coins. Parmi tous les coins extraits, on ne garde pour la procédure d'appariement que ceux qui sont en mouvement. Pour cela, on utilise les différences entre les images successives de la séquence prises deux à deux. Les coins présents dans l'image différence sont ceux qui appartiennent à un objet en mouvement. Les auteurs obtiennent ainsi en général une vingtaine d'éléments caractéristiques en mouvement par image. Ceux-ci sont ensuite mis en correspondance entre les images successives, en se basant sur leur position. Une certaine confiance initiale est accordée aux appariements ainsi réalisés. L'algorithme de relaxation modifie cette confiance en augmentant celle des éléments appariés qui ont un mouvement continu dans au moins trois images successives de la séquence.

Cette procédure est réitérée jusqu'à ce que la valeur de la confiance accordée à chaque appariement soit égale à zéro (appariement faux) ou égale à 1 (appariement correct). A titre indicatif, seules 11 primitives sur 19 ont ainsi été mises en correspondance dans la séquence traitée pour l'article [SET-88], le nombre de primitives retenues n'étant pas constant dans toutes les images.

D'autres méthodes de suivi utilisent des modèles " iconiques " des images, ou " templates " en anglais [AGG-81]. Les zones de l'image supposées contenir un objet en mouvement sont tout d'abord détectées. Chacune de ces zones fait l'objet d'une représentation " iconique " et ce pour toutes les images successives de la séquence.

Ces " icônes " sont en fait des sous-images qui contiennent un seul objet en mouvement. Elles sont utilisées soit après avoir été binarisées, soit après l'application d'un détecteur de contours suivi d'une binarisation destinée à extraire les points qui appartiennent à

un objet. Des mesures de similarité entre sous-images extraites de deux images successives de la séquence fournissent des appariements.

Il existe également des méthodes de suivi utilisant une représentation pyramidale des images [MEY-94], c'est à dire que la même image est utilisée plusieurs fois, mais à différentes résolutions. Ces méthodes sont dites hiérarchiques. La représentation à faible résolution est utilisée pour calculer une estimation grossière des paramètres du mouvement. Cette estimation est ensuite affinée au fur et à mesure que l'on augmente la résolution des images.

2. L'approche locale

Les méthodes de suivi ne sont pas toutes basées sur l'appariement d'éléments caractéristiques. Certaines utilisent directement la fonction intensité de l'image : ce sont les méthodes basées sur le concept de gradient spatio-temporel [SUB-89]. Ce type de méthodes utilise le fait que sous un éclairage connu, ou mieux constant, les changements spatiaux et temporels de l'intensité lumineuse des points des objets en mouvement sont liés par la relation :

$$V_x G_x + V_y G_y = -D_t \qquad (1)$$

avec :

- (V_x, V_y) les coordonnées du vecteur vitesse
- (G_x, G_y) le gradient spatial de l'intensité
- D_t la variation du niveau de gris du point (x, y) d'une image à la suivante.

G_x, G_y, D_t étant accessibles à la mesure, il reste à déterminer V_x et V_y. Malheureusement, on dispose ici d'une seule relation pour deux inconnues, ce qui rend le problème insoluble dans le cas général, comme l'illustre la figure III-3 :

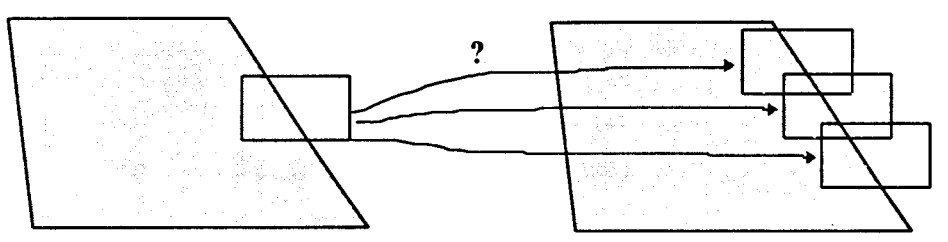


Figure III-3 : Suivi d'une zone d'un objet.

Cependant, il existe des solutions pour un certain nombre de cas particuliers. Citons notamment les travaux de Fennema [FEN-79] et Horn [HOR-81] qui supposent que la vitesse est la même pour tous les points d'un objet. Cette hypothèse est valable lorsque les mouvements étudiés sont des translations d'objets rigides dans un plan parallèle à celui de l'image. Dans le cas général, des modèles des variations de la vitesse peuvent être adaptés, mais les résultats ne sont obtenus qu'au prix de très longs calculs.

D'autres méthodes utilisent les variations de la fonction intensité de l'image entre des images successives pour calculer un flot optique, qui représente l'ensemble des vecteurs vitesses des points de l'image qui se déplacent entre deux images. On se reportera pour ce type de méthodes à [AIS-89], [DUN-92] et [HOR-81].

3. L'approche synthétique

Une troisième approche qui bénéficie à la fois de la rapidité des méthodes globales et de la précision des méthodes locales a été proposée par Yashida [YAS-83]. Il s'agit d'estimer la vitesse de certains points caractéristiques, puis de propager l'information aux autres points en utilisant une contrainte de proximité. Formulé de façon mathématique, la méthode consiste à chercher pour tout point de coordonnées (x, y), le vecteur vitesse (V_x, V_y) qui minimise :

$$E^2=a^2E_1^2+E_2^2,$$

avec :

$E_1=V_xG_x+V_yG_y+D_t,$

$E_2 = \sqrt{(\overline{V}_x - V_x)^2 + (\overline{V}_y - V_y)^2}$

$a \in \Re^+$

E_1 résulte de l'équation (1) du paragraphe précédent, E_2 résulte d'une hypothèse de rigidité que l'on applique aux objets en mouvement, $(\overline{V}_x, \overline{V}_y)$ étant la vitesse moyenne des points voisins, et $1/a^2$ représente le poids de la contrainte de rigidité.

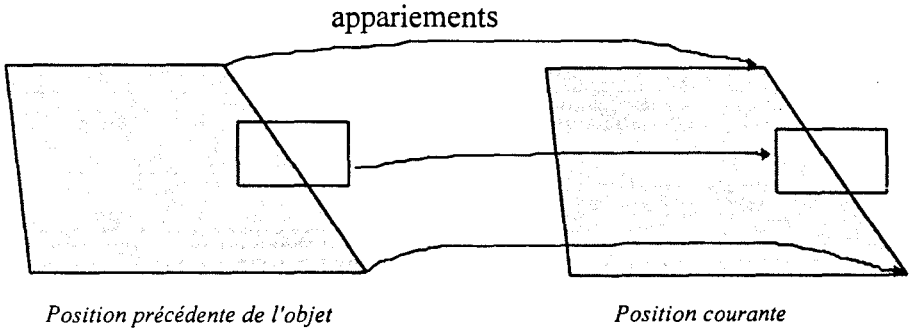


Figure III-4 : L'approche synthétique

La dérivation partielle de E par rapport à V_x et V_y permet d'exprimer les coordonnées de la vitesse en fonction notamment de la vitesse moyenne ($\overline{V}_x$, $\overline{V}_y$) des points voisins. Cette vitesse est *a priori* aussi inconnue que (V_x, V_y). Cependant, en utilisant la vitesse disponible pour certains points caractéristiques, on peut initialiser la relation et propager la détermination de (V_x, V_y) à partir de ces points.

4. Conclusion

Toutes ces méthodes de suivi, y compris l'approche synthétique, utilisent des représentations partielles des objets. Lorsque l'on dispose d'une représentation complète des objets, il est évidemment plus simple de réaliser le suivi de ces objets. Ceci est le cas dans l'approche que nous avons retenue, à savoir l'utilisation des modèles de contours actifs, que nous présenterons dans le chapitre suivant. Nous allons maintenant décrire le second type de techniques de mise en correspondance, à savoir l'appariement stéréoscopique.

C. L'appariement stéréoscopique

1. Introduction

Comme nous venons de le voir, l'analyse du mouvement s'effectue par des procédures de suivi des objets en mouvement. L'analyse obtenue est une projection des paramètres du mouvement 3D sur le plan 2D correspondant au plan image. Si l'on ne possède ni point de repère dans la scène, ni informations *a priori* sur la taille des objets étudiés, une possibilité de retrouver l'information profondeur consiste, à l'instar de la vision humaine et animale en général, à utiliser deux points de vue simultanés de la scène [BOY-88], [BUR-80]. On réalise alors des appariements stéréoscopiques afin de retrouver dans les deux images de la scène les projections d'un même objet physique [MAR-79], [MAY-81], [NIN-94]. Ainsi, à partir des positions respectives des projections des objets dans chacune des deux images, il est possible de calculer leurs positions par rapport aux caméras par triangulation.

Cette méthode de stéréovision à l'aide de deux caméras, ou plus, où l'on se contente d'analyser les images reçues sans agir sur l'environnement, est dite passive. Par opposition, les méthodes où l'on agit sur l'environnement sont dites actives. C'est le cas de la télémétrie laser où les objets sont localisés en envoyant des faisceaux laser dans leur direction, et en analysant les rayons réfléchis [ALI-90].

Nous nous intéressons uniquement à la stéréovision passive.

a) Principes généraux de la stéréovision

Le processus de stéréovision peut être décomposé en quatre étapes : deux étapes préalables à l'analyse des images, et deux étapes réalisées pour chaque scène à analyser. Les deux étapes préalables sont :

- **le calibrage du stéréoscope.**

  Le calibrage des caméras est nécessaire lorsque l'on désire employer des procédures d'appariement exploitant la géométrie du système stéréoscopique. C'est le cas de la grande majorité des procédures d'appariement. Les algorithmes n'exploitant pas la géométrie du stéréoscope [GOL-92] ne nécessitent pas, quant à eux, un calibrage préalable des caméras. Cette étape, non systématique, est exposée en Annexe B.

- **le choix des primitives.**

  Les primitives sont les entités que l'on souhaite apparier. Les points images ne constituent pas des primitives idéales. En effet, sans contraintes particulières et pour des images de taille 512x512 pixels, on dénombre $(512x512)^2$ soit environ 68 milliards d'appariements de couples de points possibles.

En général, les images à analyser contiennent une quantité d'information beaucoup trop importante par rapport à la quantité d'information nécessaire pour obtenir le résultat souhaité. Le choix des primitives est donc une décision importante, conditionnée par le type d'images à traiter et l'information que l'on désire en extraire.

Ayache [AYA-89] recense les propriétés que doivent posséder les primitives, ou indices visuels, pour qu'elles contiennent une information **pertinente**. Elles doivent être :

- **compactes**, afin de fournir une représentation de l'image aussi concise que possible;
- **intrinsèques**, c'est à dire correspondre à la projection dans l'image d'objets physiques;
- **robustes**, afin d'être peu sensibles aux petites variations d'intensité lumineuse;
- **discriminantes**, de telle sorte que les attributs qui leur sont associés permettent de les discriminer, afin de faciliter leur mise en correspondance ultérieure;
- **précises**, car la précision des résultats obtenus dépend de la précision de la localisation des primitives;

- **denses**, la densité des indices visuels devant être suffisante pour représenter tous les objets pertinents de la scène.

Les principales primitives utilisées en stéréovision sont :

* ***la fenêtre***, on divise l'image en sous-images appelées fenêtres que l'on cherche à apparier à l'aide de mesures de ressemblance entre les fenêtres des images gauche et droite (block-matching) [KAN-94].

* ***la région***, qui est une zone homogène de l'image [GAG-89].

* ***le point***, qui est la primitive la plus élémentaire que l'on puisse choisir. On se restreint généralement aux seuls points de contours car ils contiennent plus d'information que tout autre point, et sont en nombre limité. Parmi ces points, on retient souvent les coins, qui ont un fort contenu informationnel [PRA-85], ou les points de Moravec [MOR-79].

* ***le segment de contour*** dont l'utilisation permet de diminuer le nombre de primitives, et d'exploiter pour l'appariement les caractéristiques géométriques qui le décrivent, telles que sa longueur ou son orientation [AYA-89], [MED-85].

Les deux étapes qui sont effectuées pour chaque image de la séquence sont l'extraction des primitives, et l'appariement des primitives et le calcul de la profondeur :

- **l'extraction des primitives.**

  La qualité des résultats de l'appariement dépend des primitives choisies, et de la qualité de l'extraction. Selon l'approche retenue, les phases d'extraction et d'appariement peuvent être plus ou moins interdépendantes.

- **l'appariement des primitives et le calcul de la profondeur.** L'appariement est l'étape la plus délicate du processus de stéréovision. Le calcul de la profondeur par contre s'effectue facilement par simple triangulation (*cf.* Annexe A).

Les processus de stéréovision se différencient essentiellement au niveau du choix des primitives, celles-ci conditionnant fortement les algorithmes d'extraction et d'appariement. Le calcul de la profondeur est, quant à lui, identique quelle que soit l'approche retenue.

b) Choix des caméras

Une caméra est constituée d'un ensemble de cellules CCD. La disposition de ces cellules permet de distinguer deux grands types de caméras : les caméras matricielles et les caméras linéaires. Les caméras linéaires ne comportent qu'une seule barrette CCD, c'est à dire

une ligne de pixels. Cette ligne unique comporte beaucoup plus de pixels que les lignes d'une caméra matricielle. Actuellement, les caméras linéaires standards comportent environ 2500 pixels. La précision obtenue sur la position des objets est donc meilleure qu'avec des caméras matricielles [BRU-94]. Par contre, il n'est pas possible de retrouver la forme ni la surface des objets, puisque l'on ne dispose que d'une "tranche" des objets .

Les caméras matricielles fournissant une image à deux dimensions. Elles permettent donc de retrouver des paramètres de forme et de surface qui caractérisent les objets. Pour des raisons de coût et de temps de traitement, la taille d'un capteur d'une caméra matricielle standard est de l'ordre de 700x500 pixels. La précision du calcul de la profondeur est donc environ cinq fois moins grande que celle obtenue à l'aide des caméras linéaires.

Comme nous désirons non seulement connaître la distance des objets par rapport aux caméras mais également avoir des informations sur la géométrie des objets, nous avons opté pour l'utilisation de caméras matricielles, malgré leur faible résolution. Le processus de stéréovision décrit dans ce chapitre concerne donc uniquement la stéréovision matricielle.

La procédure d'appariement étant en partie conditionnée par la géométrie de la stéréovision, il est important de la choisir judicieusement. C'est pourquoi le paragraphe suivant s'attache à la présentation de la géométrie d'un stéréoscope et à la justification de la géométrie adaptée à notre cas.

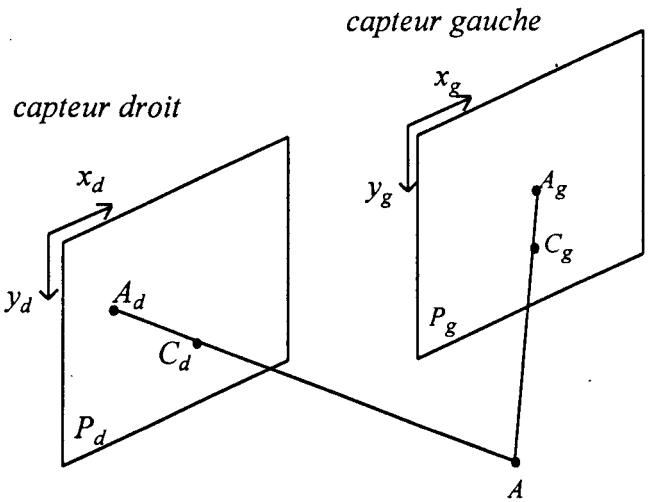
2. Description géométrique d'un stéréoscope

Soient P_d et P_g les plans images des caméras droite et gauche d'un stéréoscope (*cf.* figure III-5). Soit A un point objet de la scène dont on note A_d l'image sur le capteur de la caméra droite et A_g l'image du même point sur le capteur de la caméra gauche. Les points A_d et A_g sont dits points homologues car images d'un même point objet. A partir des images formées sur les capteurs P_d et P_g, la position dans la scène du point objet A est déterminée par l'intersection des rayons issus des deux points homologues A_d et A_g et passant par les centres optiques des caméras qui leur sont associées.

A partir d'une paire d'images stéréoscopiques, la détermination des couples de points homologues permet de calculer les positions des points réels correspondants. La recherche des points homologues constitue la phase d'appariement.

L'homologue d'un point de l'une des deux images peut être situé en n'importe quel point de l'autre image. En limitant l'espace de recherche à une partie de l'espace, on peut réduire considérablement le risque d'ambiguïté ainsi que le temps de calcul. C'est pourquoi de

nombreux algorithmes utilisent les propriétés de la configuration du stéréoscope pour établir des contraintes. La contrainte la plus avantageuse est la contrainte épipolaire car elle permet de réduire l'espace de recherche à une droite.



Légende :

A	*point objet.*
A_d, A_g	*points images de A sur les capteurs des caméras droite et gauche respectivement*
C_d, C_g	*centres optiques des caméras droite et gauche respectivement*
(C_dC_g)	*entraxe des caméras*
P_d, P_g	*plan image des caméras droite et gauche respectivement*

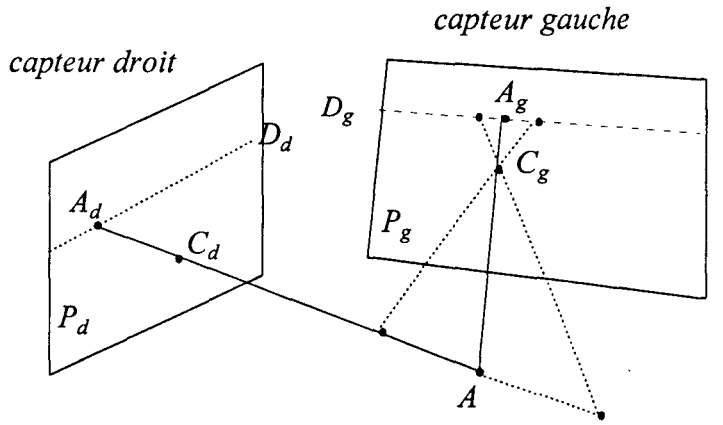
Figure III-5 : Principe du stéréoscope.

a) La contrainte épipolaire

Cette contrainte est établie à partir du modèle du sténopé (voir Annexe B). Considérons un point A et ses deux images A_d et A_g sur chacun des capteurs. Soient C_d et C_g les centres optiques des caméras droite et gauche. Le point A_d se trouve à l'intersection du plan image droit et de la droite (AC_d). De même, le point A_g se trouve à l'intersection du plan image gauche et de la droite (AC_g). Tous les points A situés sur la droite (A_dC_d) ont pour image A_d (*cf.* figure III-6).

Par définition, l'image de la droite (A_dC_d) dans le plan image gauche est appelée la droite **épipolaire**, notée D_g, associée à A_d. Par raison de symétrie, l'image de la droite (A_gC_g) dans le plan image droite est la droite épipolaire, notée D_d, associée à A_g. Ces deux épipôles D_d et D_g sont dites épipôles conjuguées. Les épipôles conjuguées D_d et D_g sont donc les droites d'intersection du plan défini par les points (A, C_d, C_g) avec chacun des deux plans images.

Par conséquent, le point A_g **homologue** de A_d se trouve nécessairement sur la droite épipolaire D_g. Cette contrainte géométrique est donc très importante car elle permet de limiter la recherche des homologues à une droite au lieu de la totalité du plan image.



Légende :

A	*point objet.*
A_d, A_g	*points images de A sur les capteurs des caméras droite et gauche respectivement*
C_d, C_g	*centres optiques des caméras droite et gauche respectivement*
(C_dC_g)	*entraxe des caméras*
P_d, P_g	*plan image des caméras droite et gauche respectivement*
D_d, D_g	*épipôles droite et gauche*

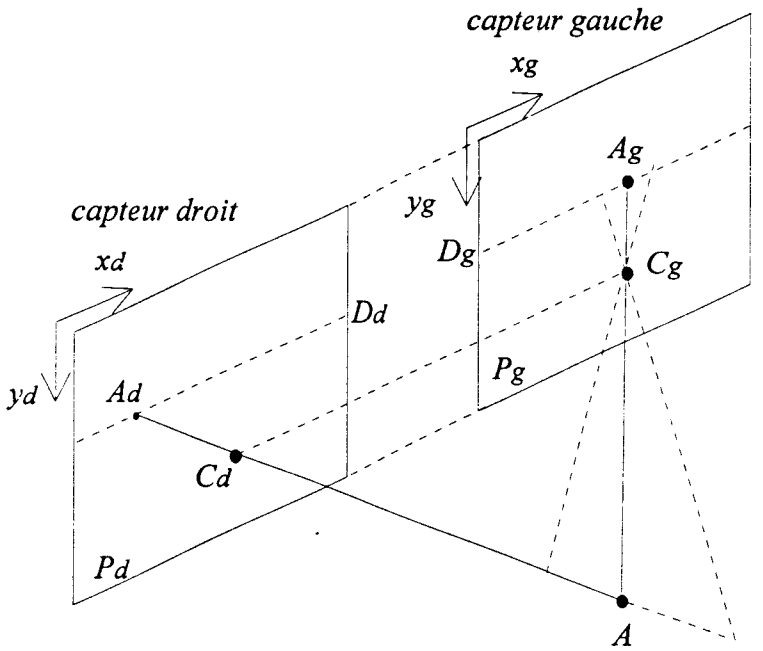
Figure III-6 : La géométrie épipolaire.

b) La contrainte épipolaire idéale

Dans le cas d'une configuration quelconque du stéréoscope, il est nécessaire de calculer pour chaque point de l'image droite l'équation de l'épipôle conjuguée dans l'image gauche et réciproquement. C'est pourquoi on recherchera la configuration particulière de la figure III-7.

Dans cette configuration, les plans images sont coplanaires et parallèles à la droite (C_dC_g) reliant les centres optiques. Les épipôles conjuguées D_d sont parallèles à la droite (C_dC_g) donc parallèles entre elles. Par conséquent, si l'on choisit l'axe des abscisses des repères des plans images parallèle à (C_dC_g), donc à D_d et D_g, **les points homologues A_d et A_g ont même ordonnée** dans les deux images, les épipôles étant confondues.

Cette configuration idéale réduit donc considérablement les calculs de la procédure d'appariement. Elle est réalisée lors de l'étape de calibrage des caméras, décrite en annexe B.



Légende :

A	*point objet*
A_d, A_g	*points images de A sur les capteurs des caméras droite et gauche respectivement*
C_d, C_g	*centres optiques des caméras droite et gauche respectivement*
(C_dC_g)	*entraxe des caméras*
P_d, P_g	*plan image des caméras droite et gauche respectivement*
x_d, y_d	*repère lié au capteur droit*
x_g, y_g	*repère lié au capteur gauche*
D_d, D_g	*épipôles droite et gauche*

Figure III-7 : Configuration particulière du stéréoscope

c) Les autres contraintes géométriques

Des contraintes géométriques autres que la contrainte épipolaire sont également utilisées pour réaliser les appariements et lever les ambiguïtés. Ce sont :

- **la contrainte d'unicité** : de même qu'un point de l'image droite ne peut avoir qu'un homologue dans l'image gauche, une primitive de l'image droite a au plus un homologue dans l'image gauche.
- **la contrainte d'ordre :** il y a conservation de l'ordre des points homologues le long de deux droites épipolaires (sous réserve que la scène ne contienne pas d'objets transparents faiblement inclinés par rapport aux capteurs des caméras [HOR-93]).
- **la contrainte de continuité de la disparité** : l'évolution des valeurs de la disparité doit être continue le long d'un même contour.

Ces contraintes sont couramment utilisées pour réaliser les appariements dans le cadre de la relaxation, ou de l'utilisation des réseaux de neurones.

La relaxation consiste à affecter à chaque primitive une valeur de la disparité, puis à faire évoluer cette valeur jusqu'à ce que tous les appariements réalisés vérifient les contraintes précédentes. Les valeurs de la disparité sont modifiées en parallèle pour toutes les primitives. Il existe essentiellement deux types de méthodes de relaxation : celles basées sur des calculs de probabilités et celles utilisant des processus d'optimisation.

Les méthodes de relaxation utilisant les probabilités consistent à actualiser itérativement les valeurs des probabilités associées à chaque appariement, afin d'assurer une homogénéité globale des appariements. A chaque itération, la probabilité est ajustée en fonction de sa valeur courante et des informations disponibles sur le voisinage des primitives. Les itérations sont arrêtées lorsque les valeurs des probabilités n'évoluent plus, ou lorsqu'un critère d'arrêt est vérifié [BAR-82], [KIM-87].

Les méthodes d'optimisation consistent à minimiser itérativement une fonction d'énergie par propagation des contraintes à tous les appariements. Les meilleurs appariements sont ceux qui correspondent à l'énergie minimale. Un exemple de ce type de méthodes est le « recuit simulé » [BAR-86], [TAN-91].

Les réseaux de neurones, tels que les réseaux de Hopfield, sont également bien adaptés à la recherche des meilleurs appariements puisqu'ils atteignent un état stable en minimisant une fonction d'énergie. La difficulté principale réside dans la définition de la fonction d'énergie, qui doit intégrer les différentes contraintes que doivent vérifier les appariements. Les performances de la méthode sont directement liées au choix de la fonction d'énergie [LEE-94].

Le principe de base des réseaux de neurones consiste à faire coopérer chaque neurone à la prise de décision. Pour cela, chaque neurone reçoit des informations et en communique aux autres, la sortie d'un neurone affectant l'entrée de tous les autres [NAS-92], [KHO-93].

3. Appariement des primitives

Nous présentons ici différents algorithmes de stéréovision utilisant les primitives les plus représentatives mentionnées précédemment, à savoir la fenêtre, la région, le point, et le segment.

a) Appariement de fenêtres.

Les méthodes basées sur l'appariement de fenêtres de l'image n'utilisent pas des éléments particuliers préalablement extraits de l'image. Elles consistent à apparier des

ensembles de pixels connexes, appelés fenêtres de l'image, par des mesures de ressemblance telles que la corrélation. Le problème de ces méthodes réside dans le choix de la taille des fenêtres. 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, et donc les variations de la disparité sont trop importantes. La taille des fenêtres peut être fixe, ou adaptative [KAN-94]. Ces algorithmes procèdent par une estimation initiale de la disparité en chaque point, grâce à l'utilisation d'un modèle statistique de la disparité, puis par une actualisation itérative de la disparité en chaque point jusqu'à la convergence de l'algorithme.

b) Appariement de régions

Les régions constituent des primitives *a priori* séduisantes car elles contiennent plus d'informations que les points ou les segments de contours. Pour faciliter la mise en correspondance de régions, on peut utiliser différents paramètres : géométriques, morphologiques, photométriques ou topologiques.

Les paramètres géométriques peuvent être :

- la surface
- le périmètre
- les coordonnées du rectangle circonscrit à la région considérée.

Les paramètres morphologiques peuvent être :

- la compacité
- les moments d'inertie
- les coordonnées du centre d'inertie.

Les valeurs des paramètres géométriques et morphologiques dépendent fortement de la méthode d'extraction. La segmentation en régions ne fournit pas nécessairement des régions de mêmes caractéristiques géométriques et morphologiques pour chacune des images stéréoscopiques, ni nécessairement le même nombre de régions dans les images droite et gauche.

D'autres paramètres moins sujets à variations de l'image droite à l'image gauche peuvent être utilisés, ce sont les **paramètres photométriques** [WRO-88] tels que :

- la variance des niveaux de gris
- la moyenne des niveaux de gris

- les niveaux de gris minimum et maximum.

Les paramètres topologiques exploitent quant à eux les relations de voisinage entre les régions pour les mettre en correspondance. C'est le cas de la connexité d'une région, qui est le nombre de régions adjacentes à la région considérée. Une mise en correspondance de deux régions est validée si leurs voisinages ont également été mis en correspondance [WRO-88], [GAG-89].

La diversité des paramètres est importante car, comme nous l'avons déjà signalé, leur variabilité est extrêmement forte entre les deux images stéréoscopiques. La conséquence d'une telle diversité est évidemment l'alourdissement du coût en calcul, tant au niveau de la procédure d'extraction, qu'au niveau de la procédure d'appariement. Les algorithmes d'appariement de régions sont donc généralement très lents.

c) Appariement de points de contours

Quoique différentes primitives de type points puissent être choisies, comme par exemple les jonctions triples, la plupart des algorithmes utilisent les points de contours [KIM-88].

Les points ne possédant pas de dimension géométrique, ce type d'algorithme ne peut s'appuyer sur aucune considération géométrique. C'est donc essentiellement le niveau de gris des pixels qui est utilisé pour réaliser l'appariement stéréoscopique.

A titre d'illustration, nous décrivons ci-dessous deux algorithmes assez récents, celui de Benshraïr [BEN-92] et celui de Lew [LEW-94].

(1) L'algorithme de Benshraïr

L'algorithme de Benshraïr fournit un exemple de ce type d'appariement [BEN-92]. Pour simplifier la procédure d'appariement, la configuration idéale du stéréoscope décrite précédemment est utilisée (*cf.* figure III-7). Comme nous l'avons vu, cette configuration permet de limiter la recherche aux paires de points de même ordonnée.

L'extraction des points de contours s'effectue ligne par ligne : après une analyse statistique des niveaux de gris de la ligne, un seuillage auto-adaptatif est effectué afin d'éliminer les fausses détections dues au bruit.

Pour chaque ligne dans les images droite et gauche, les points de contours détectés sont regroupés dans une structure de données qui rassemblent :

- leurs abscisses

- le niveau de gris des trois voisins de gauche et de droite du point considéré.

Ces niveaux de gris sont utilisés pour calculer une similarité photométrique entre deux points candidats à l'appariement : on mesure la similarité des niveaux de gris entre les 3 voisins de droite (respectivement gauche) d'un point de l'image gauche et les niveaux de gris des 3 voisins de droite (respectivement droit) d'un point de l'image droite candidat à l'appariement.

Des hypothèses d'appariement sont générées en utilisant le fait que la disparité entre deux points appariés ne peut dépasser une valeur maximale qui dépend des caractéristiques physiques du stéréoscope. Un poids est ensuite affecté à chaque couple prédit en fonction de la distance photométrique entre les points de ce couple. Le couple de points homologues retenu est celui qui correspond au poids maximal.

Cet algorithme a été appliqué à des paires d'images stéréoscopiques de taille 512x512x8 bits. Le temps de calcul est de 5 ms pour apparier deux lignes contenant 30 points de contours et de 4,5s pour apparier deux images complètes. Ces temps de calcul ont été obtenus sur un PC 486 à 33 Mhz.

(2) L'algorithme de Lew

L'algorithme de Lew [LEW-94] utilise d'autres paramètres que la seule intensité au voisinage du point considéré, comme par exemple :

- le niveau de gris
- la norme du gradient
- l'orientation du gradient
- le laplacien.

Une phase d'apprentissage permet de choisir les paramètres les plus discriminants pour la phase d'appariement parmi l'ensemble des paramètres calculés. La norme du gradient et le laplacien ne comportant pas d'information directionnelle, ces paramètres sont très peu discriminants et sont par conséquent peu utilisés.

Cet algorithme a été testé sur différents types d'images réelles. La fiabilité de la procédure est de 60% à 99% selon les images. Le temps nécessaire à l'appariement **d'un point** est de 0,4 seconde sur un PC 486 à 50 Mhz.

Remarquons que pour tous ces algorithmes utilisant les niveaux de gris des points pour l'appariement, il est nécessaire que les capteurs de chacune des caméras fournissent le même niveau de gris en réponse à une intensité lumineuse donnée.

d) Appariement de segments de contours

Les contours des objets répondent bien aux différentes propriétés généralement requises pour les primitives (*cf.* C-1a). Par contre, les primitives de type régions ne satisfont pas les propriétés de robustesse et ne permettent pas de reconstituer une représentation des objets. Ce sont donc les primitives de type contour qui sont le plus généralement retenues.

On a vu précédemment que les appariements de primitives de type points de contour étaient longs et complexes. C'est pourquoi on préfère travailler avec des primitives moins nombreuses, et plus discriminantes que les seuls points de contour. Généralement, on regroupe les points de contour en segments. Nous allons donner un exemple d'algorithme d'appariement de segments de contours [AYA-89].

Les points de contours appartenant au contour réel d'un même objet sont regroupés dans des chaînes de contours, qui sont ensuite modélisées par des polygones (*cf.* Chapitre I C). Notons que cette étape introduit un biais dans la localisation des contours. Eventuellement, un graphe de voisinage est ensuite construit, afin de structurer les données et de connaître les relations de proximité entre les différents segments.

Les résultats de la segmentation pouvant être différents dans les images droite et gauche, on ne peut pas utiliser la contrainte épipolaire rigoureusement. En pratique, on préfère donc utiliser une forme simplifiée de cette contrainte, permettant d'apparier des segments plutôt que des points de contour. Nous donnons ci-dessous un exemple de simplification de l'utilisation de la contrainte épipolaire [AYA-89].

(1) La contrainte épipolaire simplifiée.

Soit S_d un segment de contour de l'image droite et soit S_g un segment de contour de l'image gauche. Par définition, le couple de segments (S_d, S_g) vérifie la contrainte épipolaire simplifiée si et seulement si la droite épipolaire passant par le milieu de S_d coupe S_g.

Si les épipôles sont horizontales (cas de la configuration idéale), cela correspond à vérifier que l'ordonnée du milieu de S_d est comprise entre les ordonnées des extrémités de S_g (cf. figure III-8).

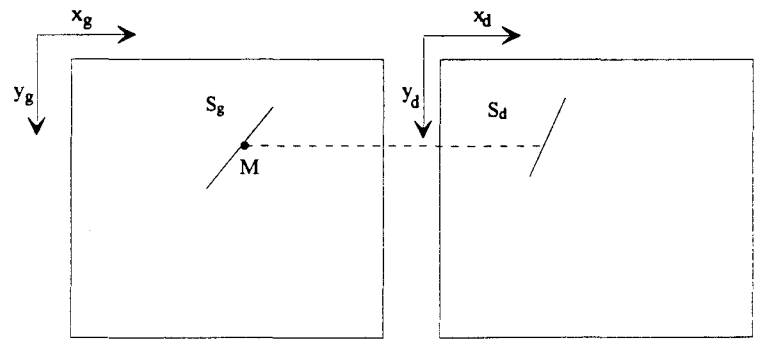
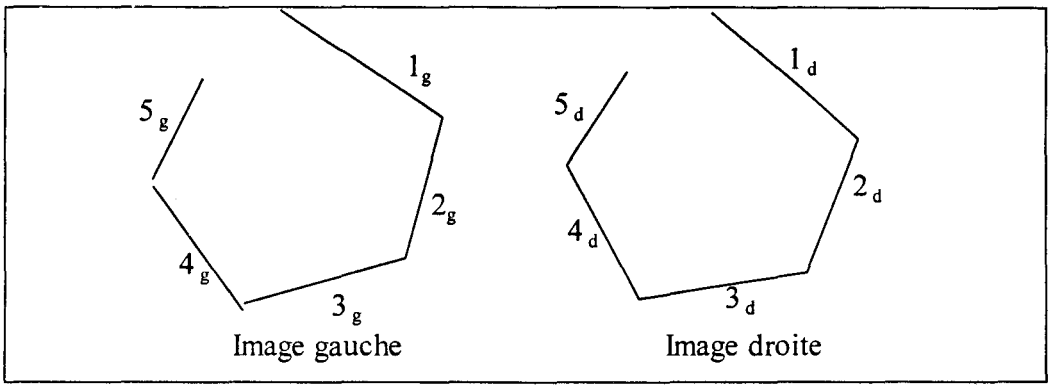


Figure III-8 : Contrainte épipolaire simplifiée

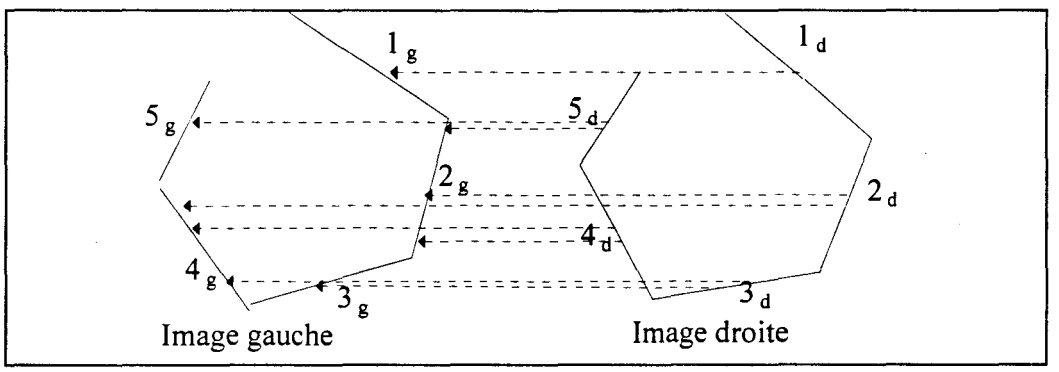
Pour trouver les segments qui vérifient cette contrainte épipolaire simplifiée, on utilise deux « fonctions » d'appariement : f_d et f_g. La « fonction » f_d associe à chaque segment S_d de l'image droite le ou les segments S_g qui contiennent le point homologue du point milieu de S_d. La fonction f_g réalise la même opération : elle associe aux segments de l'image gauche les segments de l'image droite qui vérifient la contrainte épipolaire simplifiée. S_d et S_g ne sont considérés comme des segments homologues que si ils ont été appariés à la fois par f_d et par f_g.

Illustrons cette procédure par un exemple. Considérons les deux ensembles de segments de contours de la figure III-9 a. La fonction f_d fournit les couples de segments appariés suivants : $\{(1_d,1_g), (2_d,2_g), (2_d,4_g), (3_d,3_g), (3_d,4_g), (4_d,4_g), (4_d,2_g), (5_d,5_g), (5_d,2_g)\}$, illustrés par la figure III-9 b, où les droites horizontales représentent les droites épipolaires associées aux milieux des segments de l'image droite. On constate qu'un même segment peut être apparié plusieurs fois.Symétriquement, la fonction f_g fournit les couples de segments appariés suivant : $\{(1_g,1_d), (2_g,2_d), (2_g,4_d), (3_g,3_d), (3_g,4_d), (4_g,2_d), (4_g,4_d), (5_g,2_d), (5_g,5_d)\}$, illustrés figure III-9 c. En ne conservant que les couples de segments appariés à la fois par les fonctions f_d et fg, le résultat final de l'appariement est : $\{ (1_g,1_d), (2_g,2_d), (2_g,4_d), (3_g,3_d), (4_g,4_d), (4_g,2_d), (5_g,5_d)\}$ (*cf.* figure III-9 d).

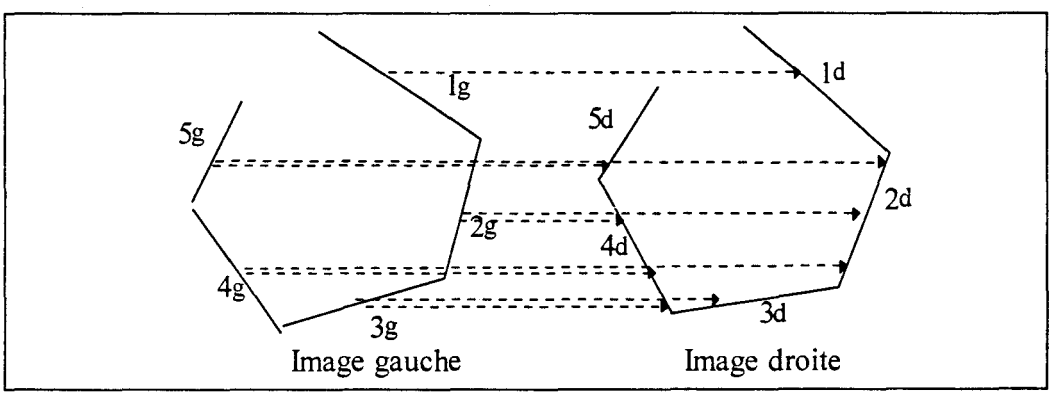
Les segments 2 et 4 ne peuvent pas être appariés car ils ont plusieurs homologues possibles, que la seule contrainte épipolaire ne permet pas de discriminer.



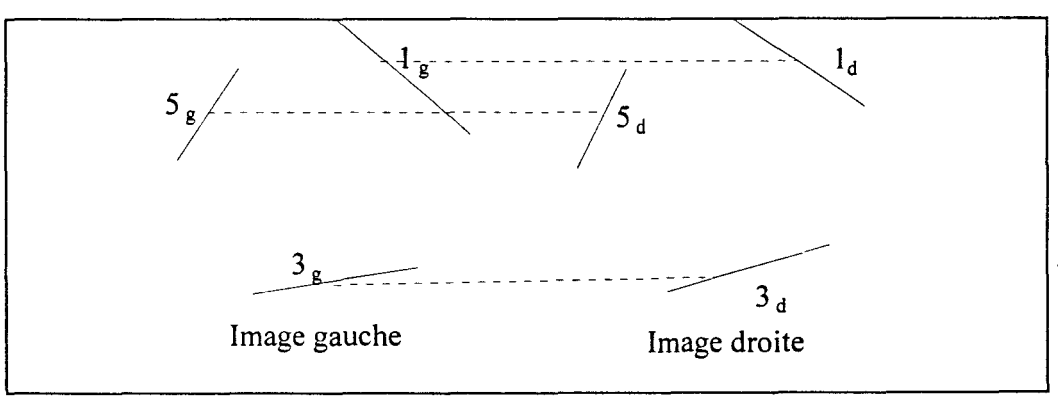
a) Ensembles des segments de contours.



b) Appariement des segments droits avec les segments gauches



c) Appariement des segments gauches avec les segments droits



d) Résultat final de l'algorithme d'appariement

Figure III-9 : Appariement de segments de contours

Afin de réduire le nombre d'appariements multiples, les couples de segments homologues doivent vérifier, outre la contrainte épipolaire simplifiée, d'autres contraintes. Ces contraintes, que nous présentons brièvement, sont les suivantes :

(2) L'orientation

Soient A_d et B_d les points situés aux extrémités du segment S_d dans l'image droite (*cf.* figure III-10). Les points homologues de A_d et de B_d, soient A_g et B_g, sont nécessairement situés sur les droites épipolaires associées à A_d et à B_d dans l'image gauche. Soit (A_{g1},A_{g2}) la droite épipolaire dans l'image gauche qui est associée au point A_d et soit (B_{g1},B_{g2}) celle associée au point B_d. Comme l'illustre la figure III-10, l'orientation du segment S_g, homologue de S_d, est nécessairement comprise entre les orientations des segments $[A_{g1}B_{g2}]$ et $[B_{g1}A_{g2}]$.

Soit Φ l'angle que fait S_g avec l'horizontale. Le segment S_g candidat à l'appariement doit vérifier : $\Phi_1 < \Phi < \Phi_2$, Φ_1 étant la valeur minimale acceptable de l'angle Φ et Φ_2 la valeur maximale.

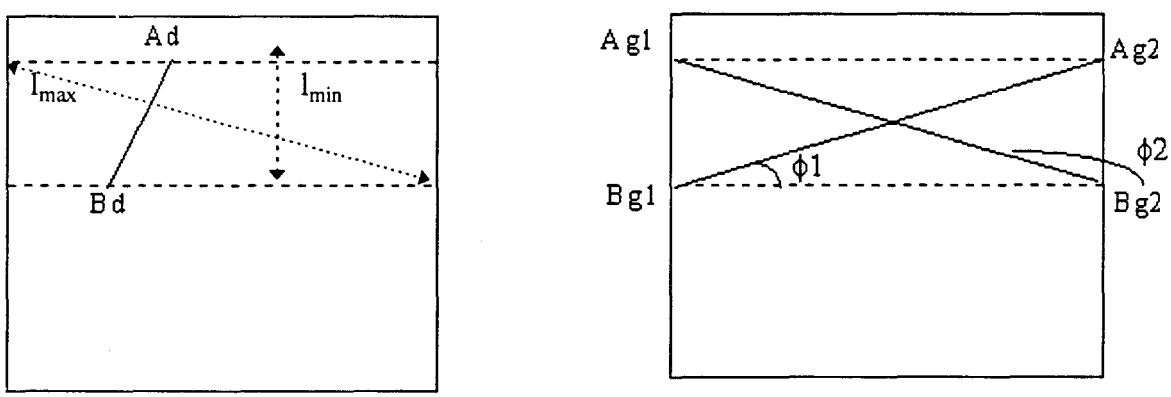


Figure III-10 : La contrainte d'orientation.

(3) La longueur des segments

On peut également calculer l'intervalle $[l_{min}, l_{max}]$ de variation possible de la longueur des segments : l_{min} est la distance minimale entre deux points appartenant respectivement aux segments de droite $[A_{g1}A_{g2}]$ et $[B_{g1}B_{g2}]$, et l_{max} est la distance maximale entre deux points appartenant respectivement aux segments de droite $[A_{g1}A_{g2}]$ et $[B_{g1}B_{g2}]$ (*cf.* figure III-10).

On a vu précédemment que l'approximation polygonale des chaînes de contours était instable, de telle sorte que des segments homologues n'ont pas les mêmes caractéristiques de longueur et d'orientation dans les images droites et gauche, ce qui entraîne des difficultés pour les apparier.

Le problème de l'instabilité de l'approximation se fait d'autant plus sentir que la courbure des contours est importante. Dans ce cas, il faut augmenter le nombre de segments pour pouvoir modéliser correctement la courbe, d'où une augmentation du temps de calcul de l'appariement.

D. Conclusion

Les temps de calcul des différents algorithmes varient de quelques secondes à plusieurs minutes pour apparier toutes les primitives contenues dans une paire d'images stéréoscopiques. Ces temps de calcul sont donc très importants, quelles que soient les primitives utilisées. Ceci est principalement du au nombre important de primitives et au nombre important de paramètres qui les caractérisent. D'autre part, la robustesse des algorithmes est parfois insuffisante, notamment dans le cas des primitives de type régions et points de contours.

En passant en revue la plupart des méthodes classiques d'appariement temporel et stéréoscopique, on a pu constater que les primitives utilisées constituent toujours une représentation partielle des objets et que la procédure d'appariement ne peut par conséquent que difficilement prendre en compte la notion d'objet. Les primitives utilisées ne constituent pas une représentation intrinsèque des objets. L'utilisation d'un modèle global des objets comme primitive faciliterait grandement les procédures d'appariement, aussi bien statiques que dynamiques.

C'est pourquoi nous avons choisi de privilégier une approche de la segmentation par les modèles de contours actifs, nous permettant de disposer d'un modèle global des objets à localiser. En outre, ils permettent de pallier en partie le problème des discontinuités apparaissant dans le contour extérieur des objets en mouvement extraits par l'opérateur de Vieren (*cf.* Chapitre II).

Dans le chapitre suivant, nous présenterons l'approche par le modèle de contour actif introduit par Kass et *al.* Le chapitre V s'attachera à exposer les adaptations que nous avons réalisées afin d'appliquer cette approche au suivi et à la localisation 3D d'objets en mouvement dans des séquences d'images stéréoscopiques.

Chapitre IV

Les contours actifs

Comme nous l'avons vu dans le premier chapitre, les méthodes classiques de segmentation se décomposent en plusieurs étapes distinctes. Dans le cas de l'extraction des segments de contours, trois étapes sont nécessaires : l'extraction des points de contour, le chaînage de ces points, et enfin la modélisation en segments.

Chacune de ces procédures apporte sa part d'erreurs et son lot de paramètres, qu'il convient d'ajuster d'autant plus judicieusement que le résultat d'une procédure influe directement sur le résultat de la suivante. Ce découpage rend le processus d'analyse du mouvement peu souple et délicat à adapter aux différents types d'images étudiées.

Les modèles obtenus constituent d'autre part une représentation partielle des objets en mouvement. L'utilisation des contours actifs permet une approche beaucoup plus globale ne nécessitant pas un découpage en cette multitude de procédures successives.

Nous allons tout d'abord présenter le modèle des contours actifs initialement introduit par Kass et *al.* en 1987 [KAS-87] sous la forme d'une courbe dont l'évolution est régie par la minimisation de l'énergie qui lui est associée. Par la suite, diverses modifications ont été apportées à ce modèle, afin

- d'améliorer la convergence du processus d'évolution, grâce à différentes méthodes de minimisation : la descente du gradient, la programmation dynamique et la méthodes des éléments finis que nous présentons au paragraphe 2.
- d'élargir son champ d'application à des domaines autres que la segmentation, en modifiant la formulation mathématique de l'énergie. Ces différentes formulations ont permis l'application des contours actifs à :
    - ♦ la segmentation d'images couleurs [RON-91]
    - ♦ l'extraction d'éléments caractéristiques dans les images [BAS-92]
    - ♦ la reconnaissance d'objets [LAI-94]
    - ♦ la squelettisation [LEY-92]
    - ♦ la reconstruction 3D [TER-88]
    - ♦ le suivi d'objets déformables [LEY-93].

Nous présenterons ces différentes applications au paragraphe 3. Nous avons adapté les modèles de contours actifs au suivi d'objets en mouvement [SEL-93] et à leur localisation dans l'espace 3D [SEL-95a], [SEL-95b], [SEL-95c]. Ces deux applications sont traitées en détail dans les chapitres V et VI.

A. Présentation du modèle initial

Les modèles de contours actifs ou « snakes » ont été introduits par Kass, Witkin et Terzopoulos en 1987 [KAS-87]. L'idée de base consiste à approcher le contour d'un objet par une courbe continue, fermée ou non. Cette courbe est constituée d'un ensemble de points de contrôle, appelés snaxels.

L'initialisation de cette courbe se fait en exploitant le fait qu'un opérateur humain localise de façon approximative les contours pertinents contenus dans une image, beaucoup plus facilement et rapidement que n'importe quel algorithme. La méthode de recherche des contours proposée s'effectue donc en deux étapes successives :

- première étape : initialisation manuelle grossière du contour actif au voisinage d'un contour à extraire, définissant ainsi un contour initial C^0.
- deuxième étape : évolution automatique de la courbe C^0 vers le contour à extraire de manière plus fine. La courbe finale résultant de cette évolution sera notée C^f

Par exemple, dans la figure IV-1 constituée de segments de droite disposés en cercle, notre système visuel nous suggère la présence de deux cercles : l'un passant par les extrémités intérieures, l'autre passant par les extrémités extérieures des segments.

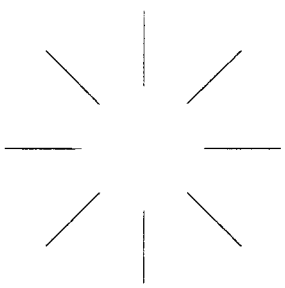


Figure IV-1 : Contours subjectifs

En fonction de la position initiale du contour actif, on peut obtenir l'un des résultats de la figure IV-2.

Figure IV-2 : Modélisation des contours subjectifs par des contours actifs

Pour générer et contrôler l'évolution du contour actif, on définit une fonctionnelle **E**(C) présentant un minimum lorsque le contour actif coïncide avec le contour recherché :

$$\inf_C (\mathbf{E}(C)) = \mathbf{E}(C^f)$$

Dans une image, un contour est caractérisé par le fait que le module du gradient de l'intensité image I est maximal le long du contour. L'opposé du module du gradient est donc quant à lui minimal le long du contour.

Soit s l'abscisse curviligne d'un point le long de la courbe représentant ce contour ($s \in [0,1]$). La fonctionnelle **E**(C) à minimiser doit donc contenir un terme caractérisant le contour recherché et minimum sur ce contour, donc un terme de la forme :

$$\int_0^1 -\|\nabla I(s)\| ds,$$

où $\|\nabla I(s)\|$ est le gradient de l'intensité image I au point d'abscisse curviligne s.

L'inconvénient d'une telle fonctionnelle est d'être très sensible aux discontinuités du gradient liées à l'aspect discret de l'image. Il n'y a aucune atténuation des points de fort gradient dus au bruit. Ceci conduit à une évolution « anarchique » de chaque point du contour initial, réduisant les chances de stabilisation vers un minimum global.

Pour éviter ce phénomène, on ajoute à la fonctionnelle un terme destiné à lisser la courbe, pour tenir compte du fait que les frontières d'un objet sont en principe régulières. Une courbe "lisse" étant caractérisée par de faibles variations d'amplitude de la dérivée seconde, ce second terme est proportionnel à cette dérivée d'ordre 2, afin de minimiser ses variations. Pour améliorer encore le contrôle de la régularité de la courbe, on ajoute également un terme proportionnel à la dérivée première.

- soit s l'abscisse curviligne paramétrant le contour actif, et soit $\vec{v}(s)$ le vecteur tangent au contour actif au point d'abscisse s.
- soit $\vec{v}_s(s)$ la dérivée première de $\vec{v}(s)$ par rapport à s.
- soit $\vec{v}_{ss}(s)$ la dérivée seconde de $\vec{v}(s)$ par rapport à s.

A ce stade, **E** est de la forme:

$$\mathbf{E} = \int_0^1 -\|\nabla I(s)\| \, ds + \int_0^1 \alpha(s)\|\vec{v}_s(s)\|^2 \, ds + \int_0^1 \beta(s)\|\vec{v}_{ss}(s)\|^2 \, ds,$$

où $\alpha(s)$ et $\beta(s)$ sont des coefficients de pondération destinés à choisir l'importance relative accordée à chacun des termes.

Cette fonctionnelle peut s'interpréter en terme d'énergie : le contour actif possède alors une énergie minimale lorsqu'il a atteint sa position d'équilibre C^f. L'énergie du contour actif, notée $\mathbf{E}_{\text{contour actif}}$, est donc composée de deux termes :

- un terme dépendant uniquement des caractéristiques de l'image, nommé pour cette raison énergie externe :

$$\mathbf{E}_{ext} = -\int_0^1 \|\nabla I(s)\| ds$$

- un terme dépendant uniquement des positions relatives des points du contour actif, nommé énergie interne :

$$\mathbf{E}_{int} = \int_0^1 \left\{ \alpha(s)\|\vec{v}_s(s)\|^2 + \beta(s)\|\vec{v}_{ss}(s)\|^2 \right\} ds$$

L'**énergie interne** régit les interactions entre les points du contour actif et contrôle la régularité de la courbe. Par contre, l'**énergie externe** ou énergie image régit les interactions entre les points de l'image et les points du contour actif, afin que ces derniers soient attirés par les contours de l'image. On a donc:

$$\boxed{\mathbf{E}_{\text{contour actif}} = \mathbf{E}_{int} + \mathbf{E}_{ext}}$$

Remarquons que l'expression de l'énergie interne est identique à l'expression de l'énergie de déformation d'une poutre soumise à un champ de forces. Dans ce cas, $\|\vec{v}_s(s)\|$ représente la force de tension appliquée en chaque point de la poutre et $\|\vec{v}_{ss}(s)\|$ la force de flexion au point d'abscisse s [BAR-84], [BOO-89]. $\|\vec{v}_{ss}(s)\|$ représente également la courbure locale du contour actif.

Par analogie avec cette énergie de déformation, la valeur de $\alpha(s)$ régit donc la tension entre les points de la courbe tandis que la valeur de $\beta(s)$ régit la flexion de la courbe. En notant $E_{int}(\vec{v}(s))$ et $E_{ext}(\vec{v}(s))$ les énergies locales et en caractères gras les énergies globales $\mathbf{E}_{int}$ et $\mathbf{E}_{ext}$, l'énergie du contour actif s'écrit :

$$\mathbf{E}_{\text{contour actif}} = \int_0^1 E_{\text{contour actif}}(\vec{v}(s))ds = \int_0^1 \left\{ E_{int}(\vec{v}(s)) + E_{ext}(\vec{v}(s)) \right\} ds \qquad (1)$$

Un contour actif défini de la sorte est appelé 2D-snake. Les expressions $E_{int}(\vec{v}(s))$ et $E_{ext}(\vec{v}(s))$ correspondent ainsi aux valeurs de l'énergie interne et de l'énergie externe au point

d'abscisse s, alors que les expressions $\mathbf{E}_{int}$ et $\mathbf{E}_{ext}$ correspondent à la somme sur tous les points du contour actif des énergies E_{int} et E_{ext} respectivement.

Le principal problème lié à l'utilisation des contours actifs est celui de leur initialisation. C'est pourquoi de nombreuses applications nécessitent encore une initialisation manuelle. Afin d'améliorer le comportement du contour actif et de rendre possible sa convergence vers un contour réel à extraire même lorsqu'il a été initialisé relativement loin de celui-ci, Cohen a introduit la notion de modèle de contour actif de type « ballon » [COH-90].

B. Le modèle du « ballon »

Dans ce modèle du ballon, une force d'expansion a été ajoutée au modèle initial afin que le contour actif se comporte comme un « ballon que l'on gonfle ». Quand au cours de son expansion, le contour actif passe sur des contours de l'image, ceux-ci exercent une force qui a tendance à le stabiliser sur ces contours. En revanche, les points de contour isolés dus au bruit n'exercent pas une force suffisante pour arrêter l'expansion du contour actif. Ce modèle est ainsi moins sensible au bruit contenu dans les images [COH-90].

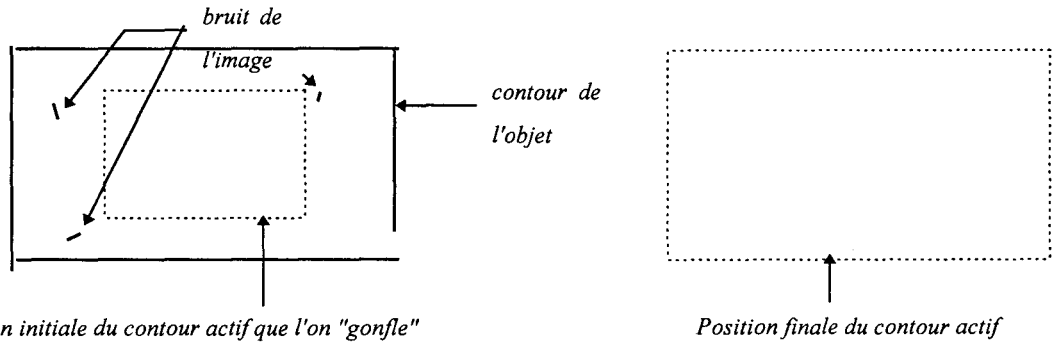


Figure IV-3 :Exemple d'utilisation du modèle du ballon.

Lorsqu'un contour actif n'est soumis à aucune force externe, c'est à dire lorsqu'il n'y a pas de contours dans son voisinage, il a tendance à se rétracter sur lui-même. En effet, l'énergie interne est alors prépondérante et la tension qui s'exerce sur les points du contour actif sans contrepartie fait que le contour actif se rétracte.

Pour empêcher ce phénomène, Rougon [ROU-91] a ajouté un terme à l'énergie interne du modèle de Kass *et al.*, terme très voisin de la force d'expansion du modèle du ballon. Ce terme d'énergie permet de contrôler l'expansion ou la contraction du contour actif, suivant le signe du coefficient qui le pondère. Cette modification sera décrite plus en détail dans le chapitre suivant.

C. Les différentes méthodes de minimisation de l'énergie.

On a vu précédemment (cf. §A) que l'on disposait d'un contour initial C^0 et que l'évolution du contour actif vers le contour de l'objet le plus proche, à partir de ce contour initial, était régie par la minimisation de son énergie $\mathbf{E}_{\text{contour actif}} = \int_0^1 E_{\text{contour actif}}(\vec{v}_s)ds$. Nous allons maintenant décrire trois méthodes différentes de minimisation de l'énergie du contour actif vers le contour final C^f, à savoir :

- la méthode de la descente du gradient,
- la méthode des éléments finis,
- la programmation dynamique.

On considérera par la suite que les fonctions $\alpha(s)$ et $\beta(s)$ sont choisies constantes, et ce pour toutes les méthodes présentées : $\alpha(s)=\alpha$ et $\beta(s)=\beta$ $\forall s$.

1. La méthode de la descente du gradient

a) Présentation de la méthode

Cette méthode est une méthode itérative [PRE-88]. A chaque itération n correspond une courbe C^n définie par un ensemble de vecteurs position :

$$\vec{v}^n(s) = \begin{pmatrix} x^n(s) \\ y^n(s) \end{pmatrix},$$

où $x^n(s)$ et $y^n(s)$ sont les coordonnées d'un point de C^n dans le plan image, et où la suite des vecteurs $\vec{v}^n(s)$ est telle que $\lim_{n\to\infty} \vec{v}^n = \vec{v}^f$, l'ensemble des vecteurs $\vec{v}^0(s)$ étant déterminé par la procédure d'initialisation et l'ensemble des vecteurs $\vec{v}^f(s)$ définissant la courbe C^f.

Pour que la méthode converge le plus rapidement possible, la différence d'énergie entre deux termes successifs de la suite doit être la plus grande possible. On cherche donc à maximiser la différence :

$$\mathbf{E}_{\text{contour actif}}(\vec{v}^{n+1}) - \mathbf{E}_{\text{contour actif}}(\vec{v}^n) = \int_0^1 E_{\text{contour actif}}(\vec{v}^{n+1}(s))\,ds - \int_0^1 E_{\text{contour actif}}(\vec{v}^n(s))\,ds$$

Or, par définition du gradient, on a:

$$E_{\text{contour actif}}(\vec{v}^n(s) + \vec{w}(s)) = E_{\text{contour actif}}(\vec{v}^n(s)) + \vec{\nabla}E_{\text{contour actif}}(\vec{v}^n(s)) \cdot \vec{w}(s) + \|\vec{w}(s)\| \cdot \varepsilon(\vec{w}(s)) \quad (2)$$

où $\vec{w}(s)$ est un vecteur vérifiant $\lim_{\vec{w}(s)\to 0} \varepsilon(\vec{w}(s)) = 0$.

L'équation (2) équivaut à :

$$E_{\text{contour actif}}(\vec{v}^n(s) + \vec{w}(s)) - E_{\text{contour actif}}(\vec{v}^n(s)) = \vec{\nabla}E_{\text{contour actif}}(\vec{v}^n(s)) \cdot \vec{w}(s) + \|\vec{w}(s)\| \cdot \varepsilon(\vec{w}(s)) \quad (3)$$

Si on intègre les deux membres de cette égalité le long de la courbe, on obtient :

$$\int_0^1 E_{\text{contour actif}}(\vec{v}^n(s) + \vec{w}(s)) - E_{\text{contour actif}}(\vec{v}^n(s))\,ds = \int_0^1 \vec{\nabla}E_{\text{contour actif}}(\vec{v}^n(s)) \cdot \vec{w}(s) + \|\vec{w}(s)\| \cdot \varepsilon(\vec{w}(s))\,ds \quad (4)$$

En posant $\vec{v}^{n+1}(s) = \vec{v}^n(s) + \vec{w}(s)$, on peut écrire l'équation (4) sous la forme :

$$\mathbf{E}_{\text{contour actif}}(\vec{\mathbf{v}}^{\mathbf{n+1}}) - \mathbf{E}_{\text{contour actif}}(\vec{\mathbf{v}}^{\mathbf{n}}) = \int_0^1 \vec{\nabla}E_{\text{contour actif}}(\vec{v}^n(s)) \cdot \vec{w}(s) + \|\vec{w}(s)\| \cdot \varepsilon(\vec{w}(s))\,ds \quad (5)$$

Comme on l'a vu précédemment, on souhaite que la différence d'énergie du contour actif entre deux itérations successives soit maximale, c'est à dire que le second membre de l'équation (5) soit maximal. Or, le produit scalaire

$$\vec{\nabla}E_{\text{contour actif}}(\vec{v}^n(s)) \cdot \vec{w}(s)$$

est maximal quand le vecteur $\vec{w}(s)$ est colinéaire au gradient de $E_{\text{contour actif}}$ au point d'abscisse s.

Comme l'on souhaite que l'énergie de la courbe diminue au cours des itérations :

$$\mathbf{E}_{\text{contour actif}}(\vec{\mathbf{v}}^{\mathbf{n+1}}) - \mathbf{E}_{\text{contour actif}}(\vec{\mathbf{v}}^{\mathbf{n}}) \le 0$$

on doit progresser dans la direction opposée au gradient à chaque itération, ce qui conduit à poser : $\gamma \cdot \vec{w} = -\vec{\nabla}E$. γ est le pas de calcul, que nous choisissons constant.

Cela revient à dire que l'on descend le long de la surface représentant l'énergie en fonction des coordonnées x et y afin d'atteindre un minimum local, c'est à dire une vallée de la fonction énergie. On descend selon la direction de plus grande pente locale définie par l'opposé du gradient de l'énergie au point considéré. La figure IV-4 illustre ceci avec la fonction énergie $E=x^2+y^2$.

On connaît donc la direction de déplacement de chaque point du contour actif entre deux itérations. Sachant que $\vec{v}^{n+1}(s) = \vec{v}^n(s) + \vec{w}(s)$ avec $\vec{w}(s) = -\frac{1}{\gamma} \cdot \vec{\nabla}E_{\text{contour actif}}(\vec{v}^n(s))$, on obtient:

$$\vec{v}^{n+1}(s) = \vec{v}^n(s) + \vec{w}(s) = \vec{v}^n(s) - \frac{1}{\gamma} \cdot \vec{\nabla}E_{\text{contour actif}}(\vec{v}^n(s)) \quad (6)$$

Cette dernière équation est l'équation qui régit l'évolution du contour actif. Cette méthode de recherche du minimum d'une fonctionnelle est la méthode de la descente du gradient à pas constant. Le calcul du gradient de l'énergie s'effectue grâce aux équations d'Euler.

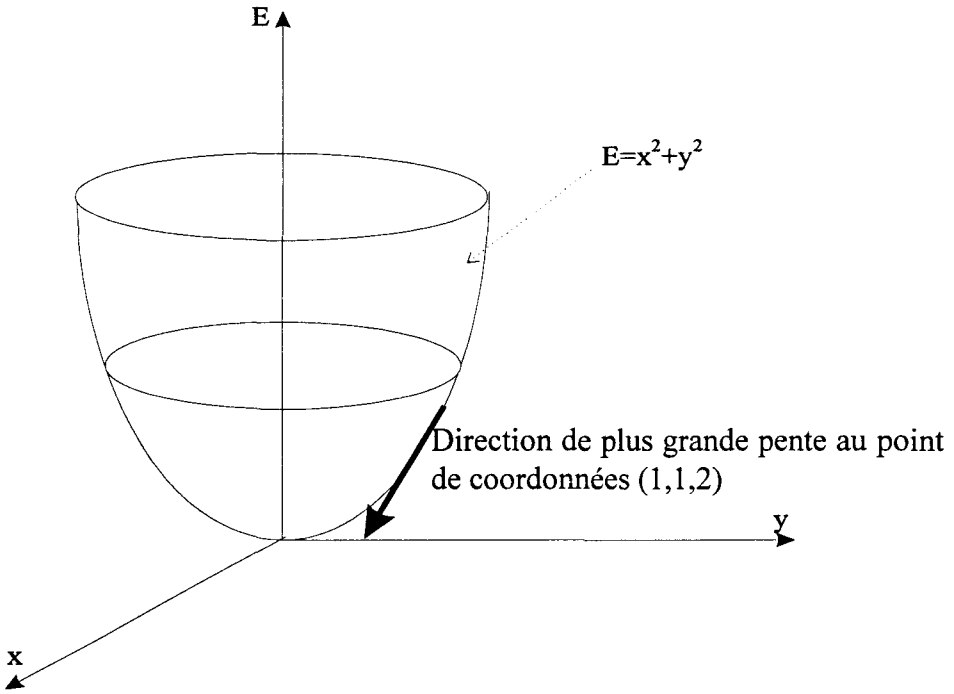


Figure IV-4 : Gradient d'une fonction énergie.

b) Les équations d'Euler

Les équations d'Euler du calcul des variations permettent de remplacer l'exigence selon laquelle une intégrale donnée est extrémale (dans notre cas, l'énergie **E** doit être minimale) par des équations différentielles pour cette fonction.

En utilisant les coordonnées cartésiennes x(s) et y(s) telles que $\vec{v}(s) = \begin{pmatrix} x(s) \\ y(s) \end{pmatrix}$, et en utilisant la même notation indicielle que précédemment pour les dérivées, on a :

$$\|\vec{v}_s\|^2 = x_s{}^2 + y_s{}^2 \text{ et } \|\vec{v}_{ss}\|^2 = x_{ss}{}^2 + y_{ss}{}^2$$

De plus, en choisissant les coefficients de tension α et de courbure β constants, l'énergie interne du contour actif s'écrit:

$$E_{int}(\vec{v}_s) = \alpha \cdot \|\vec{v}_s\|^2 + \beta \cdot \|\vec{v}_{ss}\|^2 = \alpha \cdot (x_s^2 + y_s^2) + \beta \cdot (x_{ss}^2 + y_{ss}^2).$$

Rappelons que $E_{contour\ actif} = E_{int} + E_{ext}$.

La définition du gradient de l'énergie du contour actif est la suivante :

$$\vec{\nabla}E_{\text{contour actif}} = \begin{pmatrix} \dfrac{\partial E_{\text{contour actif}}}{\partial x} \\ \dfrac{\partial E_{\text{contour actif}}}{\partial y} \end{pmatrix}$$

En utilisant les équations d'Euler pour calculer ce gradient, on obtient :

$$\begin{cases} \dfrac{\partial E_{\text{contour actif}}}{\partial x} = \dfrac{\partial E_{ext}}{\partial x} - \dfrac{d}{ds}\left(\dfrac{\partial E_{int}}{\partial x_s}\right) + \dfrac{d^2}{ds^2}\left(\dfrac{\partial E_{int}}{\partial x_{ss}}\right) \\ \dfrac{\partial E_{\text{contour actif}}}{\partial y} = \dfrac{\partial E_{ext}}{\partial y} - \dfrac{d}{ds}\left(\dfrac{\partial E_{int}}{\partial y_s}\right) + \dfrac{d^2}{ds^2}\left(\dfrac{\partial E_{int}}{\partial y_{ss}}\right) \end{cases} \quad (7)$$

Les différents termes qui apparaissent dans la première équation de ce système valent :

$$\frac{\partial E_{int}}{\partial x_s} = \alpha x_s \qquad \frac{d}{ds}\left(\frac{\partial E_{int}}{\partial x_s}\right) = \alpha x_{ss}$$

d'où $$\frac{\partial E_{int}}{\partial x_{ss}} = \beta x_{ss} \qquad \frac{d^2}{ds^2}\left(\frac{\partial E_{int}}{\partial x_{ss}}\right) = \beta x_{ssss}$$

De même pour la seconde équation du système (7), on trouve :

$$\frac{\partial E_{int}}{\partial y_s} = \alpha y_s \qquad \frac{d}{ds}\left(\frac{\partial E_{int}}{\partial y_s}\right) = \alpha y_{ss}$$

$$\frac{\partial E_{int}}{\partial y_{ss}} = \beta y_{ss} \qquad \frac{d^2}{ds^2}\left(\frac{\partial E_{int}}{\partial y_{ss}}\right) = \beta y_{ssss}$$

On obtient finalement un système de deux équations indépendantes :

$$\begin{cases} \dfrac{\partial E_{\text{contour actif}}}{\partial x} = \dfrac{\partial E_{ext}}{\partial x} - \alpha x_{ss} + \beta x_{ssss} \\ \dfrac{\partial E_{\text{contour actif}}}{\partial y} = \dfrac{\partial E_{ext}}{\partial y} - \alpha y_{ss} + \beta y_{ssss} \end{cases} \quad (8)$$

c) Discrétisation des équations

D'un point de vue numérique, une image est un ensemble discret de points. Tout contour dans une image est donc lui aussi un ensemble discret de points. Le contour actif qui, d'un point de vue mathématique, est une courbe continue, sera considéré dans ce qui suit comme une courbe échantillonnée de p points.

Comme nous avons choisi de ne travailler qu'avec des courbes fermées, afin de disposer d'un modèle des objets, les points du contour actif sont choisis de telle sorte que le premier point coïncide avec le dernier.

$$\vec{v}_i = \begin{pmatrix} x_i \\ y_i \end{pmatrix} = \begin{pmatrix} x(ih) \\ y(ih) \end{pmatrix} \text{ avec } \vec{v}(0) = \vec{v}(p).$$

Les dérivées sont calculées de façon approchée en utilisant les différences finies :

$$x_s = \frac{x_{i+1} - x_i}{h} = x_{i+1} - x_i$$

$$x_{ss} = \frac{x_{i-1} - 2x_i + x_{i+1}}{h^2} = x_{i-1} - 2x_i + x_{i+1}$$

En posant: $\begin{cases} f_x(i) = \left(\dfrac{\partial E_{ext}}{\partial x}\right)_{x_i} \\ f_y(i) = \left(\dfrac{\partial E_{ext}}{\partial y}\right)_{y_i} \end{cases}$,

on obtient pour les équations d'Euler discrètes:

$$\begin{cases} \alpha \cdot (-x_{i-1} + 2x_i - x_{i+1}) + \beta \cdot (x_{i-2} - 4x_{i-1} + 6x_i - 4x_{i+1} + x_{i+2}) + f_x(i) = \left(\dfrac{\partial E_{snake}}{\partial x}\right)_{x_i} \\ \alpha \cdot (-y_{i-1} + 2y_i - y_{i+1}) + \beta \cdot (y_{i-2} - 4y_{i-1} + 6y_i - 4y_{i+1} + y_{i+2}) + f_y(i) = \left(\dfrac{\partial E_{snake}}{\partial y}\right)_{y_i} \end{cases}$$

ou encore :

$$\begin{cases} \beta \cdot x_{i-2} + (-\alpha - 4\beta) \cdot x_{i-1} + (2\alpha + 6\beta) \cdot x_i + (-\alpha - 4\beta) \cdot x_{i+1} + \beta \cdot x_{i+2} + f_x(i) = \left(\dfrac{\partial E_{snake}}{\partial x}\right)_{x_i} \\ \beta \cdot y_{i-2} + (-\alpha - 4\beta) \cdot y_{i-1} + (2\alpha + 6\beta) \cdot y_i + (-\alpha - 4\beta) \cdot y_{i+1} + \beta \cdot y_{i+2} + f_y(i) = \left(\dfrac{\partial E_{snake}}{\partial y}\right)_{y_i} \end{cases}$$

qui permet d'écrire ces équations sous une forme matricielle.

d) Passage à la forme matricielle

Ce système peut se mettre sous la forme matricielle suivante :

$$\begin{cases} A \cdot X + f_X(X, Y) = \dfrac{\partial E_{snake}}{\partial X} \\ A \cdot Y + f_Y(X, Y) = \dfrac{\partial E_{snake}}{\partial Y} \end{cases} \quad (9)$$

avec :

- A une matrice circulante symétrique

$$A = \begin{bmatrix} 2\alpha+6\beta & -\alpha-4\beta & \beta & 0 & \cdots & 0 & \beta & -\alpha-4\beta \\ -\alpha-4\beta & 2\alpha+6\beta & -\alpha-4\beta & \beta & 0 & \cdots & 0 & \beta \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \\ -\alpha-4\beta & \beta & 0 & \cdots & 0 & \beta & -\alpha-4\beta & 2\alpha+6\beta \end{bmatrix}$$

- X le vecteur composé des abscisses des points du contour actif : $X = \begin{pmatrix} x_0 \\ x_1 \\ \vdots \\ x_p \end{pmatrix}$,

- Y le vecteur composé des ordonnées des points du contour actif : $Y = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_p \end{pmatrix}$,

- $f_X(X,Y) = \begin{pmatrix} f_{x(0)} \\ f_{x(1)} \\ \vdots \\ f_{x(p)} \end{pmatrix}$ $\qquad f_Y(X,Y) = \begin{pmatrix} f_{y(0)} \\ f_{y(1)} \\ \vdots \\ f_{y(p)} \end{pmatrix}$

- $\dfrac{\partial E_{\text{contour actif}}}{\partial X} = \begin{pmatrix} \left(\dfrac{\partial E_{\text{contour actif}}}{\partial x}\right)_{x_0} \\ \left(\dfrac{\partial E_{\text{contour actif}}}{\partial x}\right)_{x_1} \\ \vdots \\ \left(\dfrac{\partial E_{\text{contour actif}}}{\partial x}\right)_{x_p} \end{pmatrix}$ et $\dfrac{\partial E_{\text{contour actif}}}{\partial Y} = \begin{pmatrix} \left(\dfrac{\partial E_{\text{contour actif}}}{\partial y}\right)_{y_0} \\ \left(\dfrac{\partial E_{\text{contour actif}}}{\partial y}\right)_{y_1} \\ \vdots \\ \left(\dfrac{\partial E_{\text{contour actif}}}{\partial y}\right)_{y_p} \end{pmatrix}$

Utilisons cette notation matricielle pour réécrire l'équation (6) d'évolution du contour actif :

$$\begin{cases} X^{n+1} = X^n - \dfrac{1}{\gamma}\dfrac{\partial E_{\text{contour actif}}}{\partial X} \\ Y^{n+1} = Y^n - \dfrac{1}{\gamma}\dfrac{\partial E_{\text{contour actif}}}{\partial Y} \end{cases}$$

soit :

$$\begin{cases} \dfrac{\partial E_{\text{contour actif}}}{\partial X} = -\gamma \cdot (X^{n+1} - X^n) \\ \dfrac{\partial E_{\text{contour actif}}}{\partial Y} = -\gamma \cdot (Y^{n+1} - Y^n) \end{cases}$$

En remplaçant les coordonnées du gradient de l'énergie par leurs expressions issues des équations d'Euler-Lagrange, on obtient:

$$\begin{cases} A \cdot X^{n+1} + f_X(X^{n+1}, Y^{n+1}) = -\gamma \cdot (X^{n+1} - X^n) \\ A \cdot Y^{n+1} + f_Y(X^n, Y^{n+1}) = -\gamma \cdot (Y^{n+1} - Y^n) \end{cases} \quad (10)$$

Les inconnues X^{n+1} et Y^{n+1} ne peuvent pas être calculées puisque $f_X(X^{n+1},Y^{n+1})$ et $f_Y(X^{n+1},Y^{n+1})$ ne sont pas connues à l'itération n+1. On fait par conséquent l'approximation suivante : supposant que les valeurs de f_X et f_Y varient peu entre deux itérations, on assimile $f_X(X^{n+1},Y^{n+1})$ à $f_X(X^n,Y^n)$ et $f_Y(X^{n+1},Y^{n+1})$ à $f_Y(X^n,Y^n)$. De la sorte :

$$\begin{cases} A \cdot X^{n+1} + f_X(X^n, Y^n) = -\gamma \cdot (X^{n+1} - X^n) \\ A \cdot Y^{n+1} + f_Y(X^n, Y^n) = -\gamma \cdot (Y^{n+1} - Y^n) \end{cases} \quad (11)$$

d'où

$$\begin{cases} X^{n+1} = (A + \gamma I)^{-1} \cdot (\gamma X^n - f_X(X^n, Y^n)) \\ Y^{n+1} = (A + \gamma I)^{-1} \cdot (\gamma Y^n - f_Y(X^n, Y^n)) \end{cases} \quad (12)$$

avec I la matrice identité : $I = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \ddots & \ddots & 0 \\ 0 & \ddots & \ddots & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$.

On obtient ainsi les nouveaux vecteurs X et Y à l'itération n+1 en fonction des vecteurs X et Y à l'itération n, sous réserve que la matrice A+γI soit inversible. Nous allons montrer que, sous certaines hypothèses, A+γI est bien inversible.

e) Inversion de la matrice A+γI

Pour que A+γI soit inversible, il faut et il suffit qu'aucune de ses valeurs propres soit nulle. D. Terzopoulos propose de calculer l'inverse de A+γI par décomposition LU (Lower Upper) [KAS-87].

Dans notre cas, A est une matrice circulante symétrique. Rajouter γI à A revient à remplacer le terme diagonal de A, qui est une constante : 2α+6β par une autre constante :

$2\alpha+6\beta+\gamma$. $A+\gamma I$ est donc également une matrice circulante symétrique. Calculons ses valeurs propres. Soit Γ une matrice circulante :

$$\Gamma = \begin{bmatrix} a_0 & a_1 & \cdots & a_{p-1} \\ a_{p-1} & a_0 & \cdots & a_{p-2} \\ \vdots & \vdots & \ddots & \vdots \\ a_1 & a_2 & \cdots & a_0 \end{bmatrix}$$

Les valeurs propres de Γ sont données par :

$$\lambda_i = a_0 + \xi^i a_1 + \cdots + \xi^{(p-1)i} a_{p-1}.$$

où ξ est une racine $p^{ième}$ de l'unité : $\xi = e^{\left(\frac{j2\Pi}{p}\right)}$.

$A+\gamma I$ étant de plus symétrique ($a_{p-i}=a_i$), on obtient comme valeurs propres :

$$\lambda_i = 2\alpha + 6\beta + \gamma + 2(-\alpha - 4\beta)\cos\left(\frac{2\pi i}{p}\right) + 2\beta\cos\left(\frac{4\pi i}{p}\right)$$

On constate que λ_i ne dépend que de i et de p. En étudiant les variations de λ_i, on obtient le résultat suivant pour α et β positifs ou nuls :

La valeur propre λ_i est minimale pour les valeurs de i et de p qui vérifient :

$$\frac{2\Pi i}{p} = 2q\Pi, \quad q \in N$$

et ce minimum correspond à :

$$\boxed{\lambda_i=\lambda_0=\gamma}$$

Or, γ étant le pas de calcul, il est toujours choisi strictement positif. On peut donc énoncer le résultat suivant :

Si les coefficients α et β sont positifs ou nuls, alors toutes les valeurs propres de $A+\gamma I$ sont strictement positives et supérieures ou égales à $\lambda_0=\gamma>0$. **La matrice $A+\gamma I$ est donc inversible.**

L'inversion numérique de la matrice $A+\gamma I$ est d'autant plus difficile que γ est petit, puisque l'une de ses valeurs propres est alors proche de 0. En effet, on peut remarquer que les valeurs propres μ_i de A sont données par :

$$\mu_i = 2\alpha + 6\beta + 2(-\alpha - 4\beta)\cos\left(\frac{2\pi i}{p}\right) + 2\beta\cos\left(\frac{4\pi i}{p}\right)$$

D'où $\mu_0=0$, et par conséquent la matrice **A est non inversible.**

On a ainsi intérêt à choisir une valeur de γ élevée, ce qui améliore de plus le comportement du contour actif qui évolue plus lentement en oscillant moins. Par analogie avec la physique, on assimile γ à la "masse" de la courbe [BER-91].

Cependant, si on choisit une valeur de γ trop élevée, la matrice $A+\gamma I$ tend vers γI, et on perd toute l'information liée au contour actif. Il faut donc trouver un compromis acceptable pour γ.

L'inverse de $A+\gamma I$ est donnée par :

$$(A+\gamma I)^{-1} = \begin{bmatrix} V_0 & V_1 & \cdots & V_{p-1} \end{bmatrix} \begin{bmatrix} \lambda_0 & 0 & \cdots & 0 \\ 0 & \lambda_1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_{p-1} \end{bmatrix} \begin{bmatrix} V_0 & V_1 & \cdots & V_{p-1} \end{bmatrix}^{-1}$$

où les vecteurs $V_k = \begin{pmatrix} 1 \\ \xi^k \\ \xi^{2k} \\ \vdots \\ \xi^{(p-1)k} \end{pmatrix}, k \in [0, p-1]$ sont les vecteurs propres de la matrice $A+\gamma I$.

Connaissant l'inverse de $A+\gamma I$, on peut donc calculer grâce à l'équation (12) les coordonnées des points du contour actif à chaque itération.

2. La méthode des éléments finis

La méthode décrite précédemment utilise les différences finies pour discrétiser les équations d'évolution du modèle de contour actif. Avec les différences finies, on ne connaît les différentes fonctions qu'en des points discrets d'une subdivision de l'image, et on ne dispose d'aucune information entre ces points.

Au contraire, avec la méthode des éléments finis, on travaille avec des fonctions continues quelle que soit la taille de la grille. La fonction est connue partout dans l'image, indépendamment de la discrétisation choisie. Il en résulte une meilleure stabilité numérique de l'évolution du contour actif [COH-93].

Cette méthode est surtout avantageuse pour des snakes 3D car elle permet de diminuer sensiblement le nombre de points de discrétisation. Les snakes 3D sont des surfaces déformables sous l'action de forces internes et externes. Ces surfaces, définies comme des séries de courbes 2D ont été utilisées par Cohen pour reconstruire la région du coeur à partir

de coupes obtenues par résonance magnétique. En revanche, pour des snakes 2D, elle apporte très peu d'améliorations, c'est pourquoi nous ne la détaillerons pas.

3. La programmation dynamique [AMI-90]

La fonction énergie est choisie initialement comme une fonction discrète de la forme : $E_{\text{contour actif}}(v_1,...,v_i,..,v_p)$, où v_i représente le vecteur des coordonnées du ième point du contour actif, $i \in [0,p]$. Cette énergie est égale, comme précédemment, à la somme de l'énergie interne et de l'énergie externe sur tous les points. L'expression de l'énergie interne au point d'indice i est la suivante :

$$E_{int}(\vec{v}_{i-1},\vec{v}_i,\vec{v}_{i+1}) = \alpha|\vec{v}_i - \vec{v}_{i-1}|^2 + \beta|\vec{v}_{i-1} - 2\vec{v}_i + \vec{v}_{i+1}|^2$$

On a donc :

$$\mathbf{E}_{\text{contour actif}}(\vec{v}_1,...,\vec{v}_i,..,\vec{v}_p) = \sum_{i=1}^{i=p-1} E_{int}(\vec{v}_{i-1},\vec{v}_i,\vec{v}_{i+1}) + E_{ext}(\vec{v}_i) = \sum_{i=1}^{i=p-1} E_i(\vec{v}_{i-1},\vec{v}_i,\vec{v}_{i+1})$$

où E_i est la somme de l'énergie interne et de l'énergie externe au point d'indice i. E_i est fonction du point courant d'indice i et de ses deux voisins d'indices (i-1) et (i+1). Avec cette notation, l'énergie totale s'écrit :

$$\mathbf{E}_{\text{contour actif}} = E_2(\vec{v}_1,\vec{v}_2,\vec{v}_3) + \cdots + E_{p-1}(\vec{v}_{p-2},\vec{v}_{p-1},\vec{v}_p)$$

Une méthode de minimisation exhaustive, difficilement envisageable en raison de son temps de calcul rédhibitoire, consisterait à trouver toutes les courbes de p points de l'image puis à calculer la fonction énergie totale associée à chacune de ces courbes afin de trouver la courbe correspondant à l'énergie minimale.

La programmation dynamique permet d'atteindre le minimun de l'énergie sans avoir à explorer toutes les courbes possibles [AMI-90]. On cherche le minimum de $E_2(\vec{v}_1,\vec{v}_2,\vec{v}_3)$ par rapport à $\vec{v}_1$, lorsque $\vec{v}_2$ et $\vec{v}_3$ sont fixés. Pour cela, on calcule E_2 pour différentes positions de $\vec{v}_1$, et on garde la position $\vec{v}_{1\min}$ de $\vec{v}_1$ qui donne le minimum de E_1. Seules m positions de $\vec{v}_1$ sont explorées, par exemple celles obtenues par des déplacements horizontaux de un pixel à gauche et de un pixel à droite (m=3). Soit :

$$S_2(\vec{v}_1,\vec{v}_2) = \min_{\vec{v}_1} E_2(\vec{v}_1,\vec{v}_2,\vec{v}_3) = E_2(\vec{v}_{1\min},\vec{v}_2,\vec{v}_3)$$

On minimise ensuite S_2+E_2 par rapport à $\vec{v}_1$, on cherche le minimum de :

$$S_3(\vec{v}_2,\vec{v}_3) = \min_{\vec{v}_2} \{S_2(\vec{v}_1,\vec{v}_2) + E_3(\vec{v}_2,\vec{v}_3,\vec{v}_4)\}$$

On crée ainsi une suite de fonctions : $\vec{v}_{i+1}$ et $\vec{v}_i$ étant fixés, on cherche :

$$S_{i+1}(\vec{v}_i, \vec{v}_{i+1}) = \min_{\vec{v}_{i-1}} \{S_i(\vec{v}_{i-1}, \vec{v}_i) + E_{i+1}(\vec{v}_i, \vec{v}_{i+1}, \vec{v}_{i+2})\}$$

Lorsque la fonction S_p est calculée, on réitère l'algorithme jusqu'à ce que la valeur de l'énergie totale ne varie plus entre deux itérations. On a alors atteint le minimum global.

Cet algorithme possède des avantages par rapport à la méthode de minimisation utilisant la descente du gradient. En particulier, il ne nécessite aucune inversion de matrice, ce qui élimine tous les problèmes liés au conditionnement des matrices à inverser. De plus, il permet d'intégrer de façon naturelle des contraintes sur la position des points. On peut par exemple imposer une distance minimale entre les points du contour actif, afin d'éviter des agglomérations de points en certains endroits. Cette contrainte est nécessaire pour garantir une répartition régulière des points du contour actif afin d'obtenir une modélisation précise des objets. Avec les autres méthodes de minimisation, une telle contrainte ne peut être appliquée qu'indépendamment de la procédure de minimisation.

Cependant, le temps de calcul est proportionnel à pm^3, où p représente le nombre de points du contour actif et m le nombre de positions autorisées pour chacun de ces points. Or, pour être intéressant, cet algorithme nécessite des valeurs de m supérieures à 3, ce qui entraîne un temps de calcul rédhibitoire.

D. Les diverses applications des modèles de contours actifs.

Les applications des contours actifs se sont élargies à des domaines autres que la segmentation statique dans des images en niveaux de gris, qui était son premier champ d'application. Nous donnons un panaroma de ces nouvelles applications. On peut également citer une nouvelle approche qui consiste à faire coopérer les contours actifs avec les techniques multirésolution. Un contour actif est initialisé autour de l'objet à modéliser dans une image à très faible résolution, puis le modèle ainsi obtenu sert d'initialisation dans les images de résolution de plus en fine, jusqu'à obtenir le modèle dans la résolution réelle de l'image. Cette technique simplifie notamment l'initialisation, qui reste cependant manuelle [BAJ-89].

1. La segmentation des images en couleur

Comme nous l'avons vu précédemment, les modèles de contours actifs ont été proposés initialement pour réaliser la segmentation d'images statiques. Dans ce cadre, Ronfard [RON-91], [RON-94] a modifié le modèle initial afin de pouvoir l'utiliser pour la segmentation d'images en couleur. Il a ainsi défini une « énergie de fusion » à partir d'un vecteur « de couleur » qui permet au contour actif de se déformer afin de modéliser des contours d'objets colorés. Cette énergie de fusion joue le même rôle que l'énergie externe.

2. L'extraction d'éléments caractéristiques

Les portions de contours sont des éléments caractéristiques facilement reconnaissables dans les images successives d'une séquence, ou dans des couples d'images stéréoscopiques. Ils sont donc très utiles pour les procédures d'appariement. Bascle et Deriche [BAS-92] proposent d'utiliser les contours actifs pour extraire ces éléments caractéristiques des images, tels que des coins, des jonctions triples (point de jonction de trois segments de droites), des segments ou même des ellipses. Les coins et les jonctions triples peuvent être modélisés grâce à des droites, alors que les segments et les ellipses peuvent être modélisés par des splines rationnelles de degré supérieur à 1. Par ailleurs, afin de modéliser précisément ces éléments par des contours actifs, l'énergie interne ne comporte pas de terme de lissage.

3. La reconnaissance

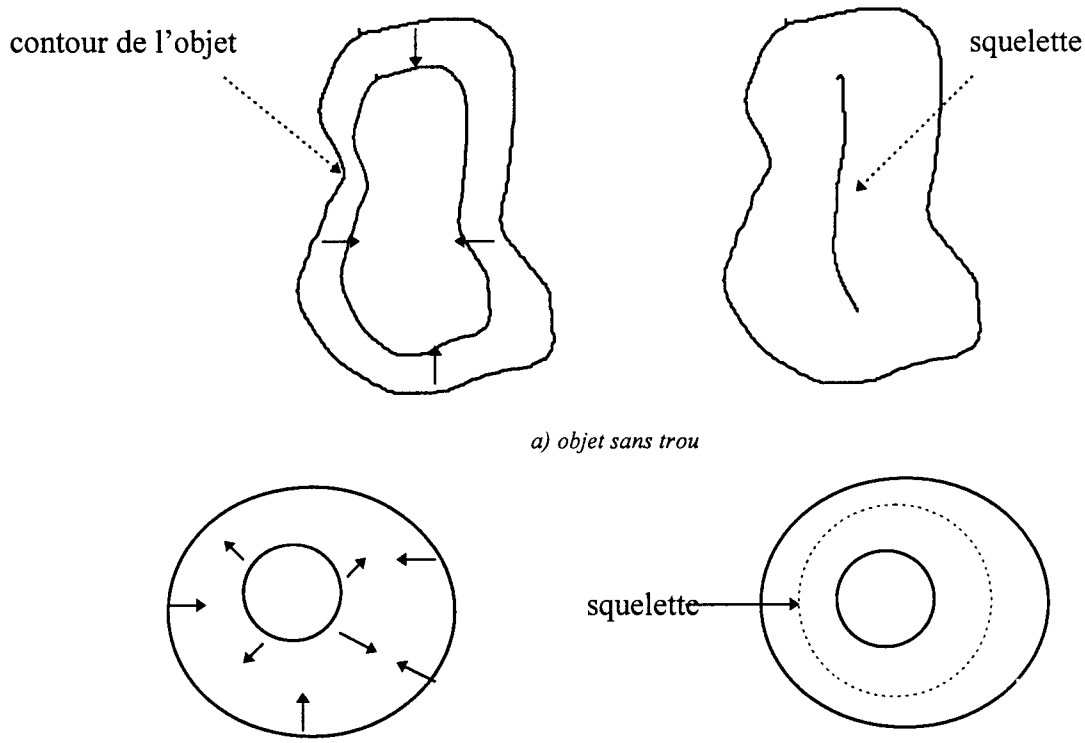
Les contours actifs ont également été utilisés par Lai [LAI-94] pour retrouver des objets dans des images bruitées. L'utilisation d'un modèle global des objets permet en effet de s'affranchir du bruit, ce qui est très utile, notamment lorsque l'on travaille avec des images médicales qui sont très souvent bruitées. Le contour actif est initialisé par la transformée de Hough, puis il se déforme jusqu'à épouser parfaitement le contour de l'objet. Il est ensuite comparé à un modèle, en mesurant la quantité de déformation nécessaire pour retrouver exactement la forme du modèle.

4. La squelettisation

Les modèles de contours actifs s'inspirent de phénomènes physiques et de leurs formulations mathématiques. Ainsi, l'énergie interne du modèle de Kass et *al.* est analogue à l'énergie de déformation d'une poutre. En s'inspirant d'autres phénomènes physiques, on peut

induire d'autres comportements pour les contours actifs, permettant de les utiliser pour faire de la morphologie mathématique. Leymarie notamment [LEY-92] s'est inspiré de la façon dont le feu se propage dans une prairie pour réaliser la squelettisation d'objets plans. Le « feu » commence à prendre à la frontière de l'objet, à l'endroit où est initialisé le contour actif. Le « feu » se propage ensuite à vitesse constante, ce qui correspond à déplacer les points à vitesse constante vers l'intérieur de l'objet. Lorsque les fronts se rencontrent, les points d'intersection sont retenus comme appartenant au squelette de l'objet. Cette méthode est schématisée figure IV-5a.

Si l'objet contient des trous, on place un contour actif sur sa frontière extérieure, et un autre autour de sa frontière intérieure. Ces deux contours actifs se déplacent l'un vers l'autre, et leur intersection constitue le squelette de l'objet (figure IV-5b).



a) objet sans trou

b) objet comportant un trou

Figure IV-5 :Squelettisation par contour actif

5. La reconstruction 3D

Les contours actifs ont également été utilisés pour la reconstruction d'objets à partir de leur silhouette ou à partir de coupes 2D. Le modèle de contours actifs s'appuie dans ce cas sur un axe de symétrie de l'objet [TER-88] pour modéliser le contour extérieur de chacune des coupes, et permettre une représentation de l'objet en trois dimensions par le regroupement de

ces coupes successives. Cette technique peut même être utilisée si la symétrie des objets n'est pas parfaite, en raison de sa faculté d'adaptation. La position approximative de l'axe de symétrie doit être localisée par un opérateur.

6. Le suivi d'objets déformables

Leymarie utilise les modèles de contours actifs pour réaliser en une seule étape l'extraction de contours et le suivi de cellules vivantes vus à travers un microscope [LEY-93]. Les cellules se déforment au cours de leurs déplacements et peuvent adopter des formes quelconques. Les propriétés de déformation des contours actifs font qu'ils sont particulièrement bien adaptés pour traiter ce type de problème. Ils permettent également de quantifier les changements de forme des cellules par des différences d'énergie, ce que les autres méthodes de suivi ne permettent pas. D'autre part, ils permettent de passer outre les discontinuités éventuelles du contour, ce qui est appréciable avec des images médicales qui sont très souvent bruitées et peu contrastées.

Par contre, comme toujours avec l'utilisation des contours actifs, l'initialisation pose une réelle difficulté. Si le déplacement des objets entre deux images n'est pas trop important, le problème de l'initialisation ne se pose que pour la première image de la séquence. En effet, la position du contour actif dans la première image, après convergence du processus d'évolution, peut servir de position initiale pour l'image suivante. Mais l'initialisation du contour dans la première image ne peut que s'effectuer manuellement avec cette méthode.

On peut noter que Terzopoulos et Metaxas ont également utilisé les contours actifs pour l'estimation du mouvement dans des images de synthèse [MET-93].

Il existe également une méthode de suivi d'objets 2D pour laquelle l'initialisation des contours actifs n'est pas automatique [DEL-94]. Cette méthode permet de s'affranchir du fond texturé sur lequel se déplacent les objets, à condition de supposer que les objets ont une texture uniforme.

7. Le suivi automatique d'objets en mouvement

Il n'existe à notre connaissance que deux méthodes traitant du suivi automatique d'objets en mouvement à l'aide de contours actifs [DEN-94], [KOL-94]. Ces méthodes seront exposées dans le chapitre suivant qui est consacré à la méthode que nous proposons pour le

suivi à l'aide de contours actifs. Nous pourrons ainsi établir des comparaisons entre notre nouvelle approche et les méthodes existantes.

8. L'appariement de contours actifs

Brint et Brady [BRI-90] ont proposé une méthode d'appariement de contours actifs qui sera également décrite dans le chapitre VI car elle concerne un problème que nous avons également traité.

E. Conclusion

Nous venons de voir dans ce chapitre que les contours actifs permettent une approche globale de la segmentation et qu'ils peuvent être adaptés à de nombreux problèmes d'analyse d'images. C'est la raison pour laquelle nous avons choisi de les utiliser.

Nous allons décrire dans le chapitre suivant notre approche de la modélisation et du suivi des objets en mouvement par les contours actifs. Notre algorithme devant être rapide, nous avons choisi d'utiliser la méthode de la descente du gradient pour minimiser l'énergie des contours actifs plutôt que la programmation dynamique.

Nous allons également montrer qu'une fois que l'on dispose d'un modèle des objets, il est intéressant de réaliser des appariements stéréoscopiques en utilisant ces modèles comme primitives, et ainsi de localiser les objets dans l'espace 3D.

Chapitre V

Suivi automatique d'objets en mouvement à l'aide de modèles de contours actifs

Ce chapitre est consacré à la description et à l'utilisation des modèles de contours actifs dans le cadre du suivi d'objets en mouvement pour l'analyse de séquences d'images. Comme les modèles de contours actifs constituent une représentation globale des objets, ils vont être utilisés pour effectuer les appariements temporels et stéréoscopiques. L'appariement stéréoscopique, qui permet la localisation dans l'espace 3D des objets en mouvement, sera traité dans le chapitre suivant.

L'utilisation du modèle original de Kass et *al.* pour le suivi est possible sans aucune modification dans des cas simples, où les objets se déplacent relativement peu entre deux images et en acceptant que l'initialisation des contours actifs soit manuelle. Dans ce cas, le traitement d'une séquence s'effectue comme suit :

- on positionne un contour actif dans le voisinage de l'objet à suivre.
- première image : on itère le processus de minimisation de l'énergie du contour actif, afin que celui-ci se déforme à partir de sa position initiale pour modéliser le contour de l'objet.
- image suivante : on réitère le processus de minimisation en utilisant pour la première itération la position du contour actif dans l'image précédente, et l'on procède de même pour toutes les images suivantes de la séquence.

Cependant, cette méthode n'est pas automatique, puisqu'elle nécessite l'intervention d'un opérateur pour la phase d'initialisation. D'autre part, elle ne permet que le suivi d'objets dont les déplacements sont relativement faibles entre deux images.

Il est donc nécessaire d'apporter diverses modifications au modèle de Kass et *al.* pour l'adapter au suivi automatique d'objets dont les déplacements peuvent être importants entre deux images successives.

A notre connaissance, deux méthodes ont été proposées pour adapter les contours actifs au suivi d'objets en mouvement. Après les avoir examinées en mettant leurs limites en évidence, nous présenterons notre approche.

A. L'algorithme de Denzler et Niemann

Denzler et Niemann [DEN-94] proposent un algorithme de suivi en temps réel, sous l'hypothèse restrictive que la scène ne comporte **qu'un seul objet** en mouvement. Le suivi s'effectue en deux étapes :

1. l'objet en mouvement est détecté et le contour actif initialisé
2. l'objet est suivi par le contour actif.

Il se pose comme toujours le problème de l'initialisation. Les auteurs ont choisi de détecter les points en mouvement par simple différence entre deux images successives de la séquence, avec l'hypothèse que la caméra est fixe et donc que les points de l'image différence appartiennent à l'objet en mouvement ou sont dus au bruit.

Pour éliminer ces détections parasites dues au bruit, les auteurs effectuent une binarisation de l'image différence. Le résultat est un ensemble de régions plus ou moins connexes. Les régions de l'image binarisées dont la taille est inférieure à un certain seuil sont éliminées afin de ne conserver qu'une seule région d'intérêt qui doit contenir les points de l'objet en mouvement.

Le chaînage des points restants permet d'initialiser automatiquement le contour actif. Celui-ci converge ensuite vers le contour de l'objet en mouvement, même si l'initialisation n'était pas très précise. Le suivi de l'objet s'effectue en utilisant la position précédente comme initialisation de la position du contour actif dans l'image courante. Aucune prédiction de la position du contour actif n'est réalisée, ce qui impose un déplacement faible de l'objet entre deux images successives. Cet algorithme présente donc les limitations suivantes :

- un seul objet en mouvement dans la scène,
- déplacements faibles de l'objet.

B. L'algorithme de Koller et al.

Le problème traité par les auteurs est la surveillance du trafic sur des routes encombrées [KOL-94]. Pour cela, ils doivent extraire et décrire les objets en mouvement à partir de séquences d'images, et ce en tenant compte des occlusions qui peuvent se produire entre les véhicules.

Segmentation des images et modélisation des objets

La segmentation des images s'effectue par différence entre l'image courante et une image de référence réactualisée en utilisant un filtre de Kalman. Les contours des régions en mouvement sont obtenus par seuillage du gradient de cette image différence. Des dodécagones convexes entourant les contours des régions en mouvement servent alors d'initialisation à la description des objets en mouvement. Des splines cubiques établies à partir des sommets de ces dodécagones sont ensuite utilisées pour lisser les contours des polygones.

Ces splines sont des contours actifs dont on a retiré le terme d'énergie lié à l'image, c'est à dire l'énergie externe. Un second filtre de Kalman est utilisé pour estimer et prédire les coordonnées des points de contrôle dans l'image suivante. Les points des contours actifs sont ensuite utilisés pour calculer la vitesse des objets ainsi que leurs trajectoires.

Traitement des occlusions

Un traitement des occlusions est intégré à l'algorithme. Sachant que les objets proches de la caméra vont apparaître en bas du plan image et risquent de masquer ainsi les objets plus lointains, les contours des objets sont recherchés en partant du bas de l'image : lorsqu'il y a occlusion, c'est à dire lorsqu'un objet masque en partie un autre, la zone commune aux deux objets n'est pas utilisée dans le calcul du modèle. Ce sont les deux modèles issus des images précédentes qui sont repris pour cette zone.

Résultats

La caméra est fixée sur un pont au dessus de la route afin de couvrir un large champ et de réduire les risques d'occlusions. Notons que cet algorithme nécessite 5 secondes par image sur Sun Sparc Station 10 pour suivre simultanément dix véhicules.

Les contours actifs ne sont utilisés dans cette procédure de suivi que pour lisser les contours des objets en mouvement obtenus par une autre procédure de segmentation. Une connaissance *a priori* de la scène est utilisée d'une part pour le calcul de la vitesse des objets

qui doit être nécessairement dans le plan de la route, d'autre part pour le traitement des occlusions en considérant que les objets les plus proches apparaissent en bas de l'image.

C. Proposition d'un modèle de contour actif « dynamique »

Les deux algorithmes présentés ci-dessus permettent le suivi automatique d'objets en mouvement. Le premier algorithme proposé fonctionne en temps réel mais ne permet que le suivi d'un seul objet dont le déplacement entre deux images successives doit être relativement faible. Le second algorithme n'utilise que peu les modèles de contours actifs et demande une connaissance *a priori* de la scène.

Le modèle de contour actif que nous proposons résout partiellement les limitations de ces deux approches. Nous l'avons qualifié de « dynamique » car il est adapté au suivi d'objets en mouvement dans des séquences d'images. Rappelons que pour utiliser le modèle des contours actifs pour le suivi d'objets en mouvement, il est d'abord nécessaire de rendre son initialisation automatique. Nous présentons cette modification et ses conséquences dans le paragraphe suivant.

Ensuite, nous avons utilisé une estimation de la position des points du contour actif dans l'image suivante à partir de leurs positions précédentes afin que le contour actif puisse suivre des objets dont les mouvements sont rapides.

1. L'initialisation automatique

Dans la quasi totalité des applications proposées dans la littérature, les contours actifs sont initialisés de façon manuelle ou semi-automatique à proximité immédiate des contours des objets à modéliser. Or ce type d'initialisation ne peut être retenu dans le cadre d'applications devant être totalement automatiques.

Seules deux méthodes d'initialisation automatique ou semi-automatique ont été proposées récemment. L'initialisation automatique décrite dans [LAI-94] consiste à calculer la transformée de Hough de l'image et à se servir des contours ainsi obtenus comme contours initiaux des contours actifs. Malheureusement, cette méthode est très coûteuse en temps de calcul. Elle n'est donc pas utilisable pour le suivi d'objets en mouvement en temps réel.

L'initialisation semi-automatique a été proposée par Berger [BER-91]. Un contour actif ouvert comportant très peu de points est placé par l'utilisateur au voisinage du contour à segmenter. Ce contour actif s'allonge ensuite au fur et à mesure des itérations dans la

direction du maximum du gradient. Cette méthode a pour inconvénient de faire perdre aux contours actifs deux propriétés importantes :

- le contour actif ne peut plus modéliser des objets dont les contours présentent des discontinuités, l'allongement du contour actif le long du contour étant limité par la présence de ces discontinuités;
- le contour actif devient très sensible au bruit car l'effet de lissage qu'assurait l'énergie interne et qui permettait de passer outre le bruit est occulté par la méthode de croissance du contour actif.

Initialisation proposée :

Dans les applications que nous avons envisagées, nous considérons toujours que les objets en mouvement pénètrent dans la scène par les bords de l'image. C'est le cas notamment lorsque l'on surveille le trafic routier : les véhicules pénètrent dans la zone surveillée par les bords du champ visé.

C'est pourquoi nous proposons de placer un contour actif initial sur les bords de l'image. Ce contour actif doit permettre de « capturer » les objets mobiles qui pénètrent dans la scène. Dès que ce contour actif se referme derrière un objet en mouvement, une procédure de scission intervient, créant deux contours actifs : un contour actif qui modélise et suit l'objet en mouvement durant sa traversée du champ et un contour actif périphérique qui revient à sa position d'origine afin de capturer de nouveaux objets. Une procédure similaire est mise en oeuvre lorsqu'un objet quitte la scène. Le contour actif associé à cet objet fusionne alors avec le contour actif périphérique, par détection des intersections entre le contour actif associé à l'objet et le contour actif périphérique

La figure V-1 illustre ce processus sur une scène de trafic urbain. Les différentes étapes de la procédure de suivi sont reprises en détail dans les paragraphes suivants.

a) La procédure de ré-échantillonnage des snaxels

Pour alléger l'écriture, nous utiliserons désormais le mot « snaxels » pour désigner les points du contour actif, ainsi que les expressions $E_{tension}$ et $E_{flexion}$ pour désigner les termes suivants de l'énergie interne :

$$E_{tension} = \alpha\|v_s(s)\|,\ E_{flexion} = \beta\|v_{ss}(s)\|.$$

Quand la longueur du contour actif augmente, la distance inter-snaxels s'accroît, et rend de plus en plus prépondérante l'influence de $E_{tension}$. Ceci freine l'extension du contour

actif périphérique, l'empêchant de se refermer autour des objets pénétrant dans la scène. C'est pourquoi nous effectuons un ré-échantillonnage périodique des snaxels afin de maintenir la distance inter-snaxels constante et égale à la valeur initiale choisie.

Cette modification est cependant encore insuffisante. C'est pourquoi nous avons introduit un terme d'énergie permettant encore d'accélérer cette fermeture. Cette modification est décrite dans le paragraphe suivant.

L'initialisation automatique du contour actif est illustrée figure V-1. Les images sont issues d'une séquence monoculaire de trafic urbain. La caméra était placée à l'aplomb du carrefour. Les contours des objets en mouvement ont été extraits par l'opérateur de Vieren, puis moyennés afin de les lisser.

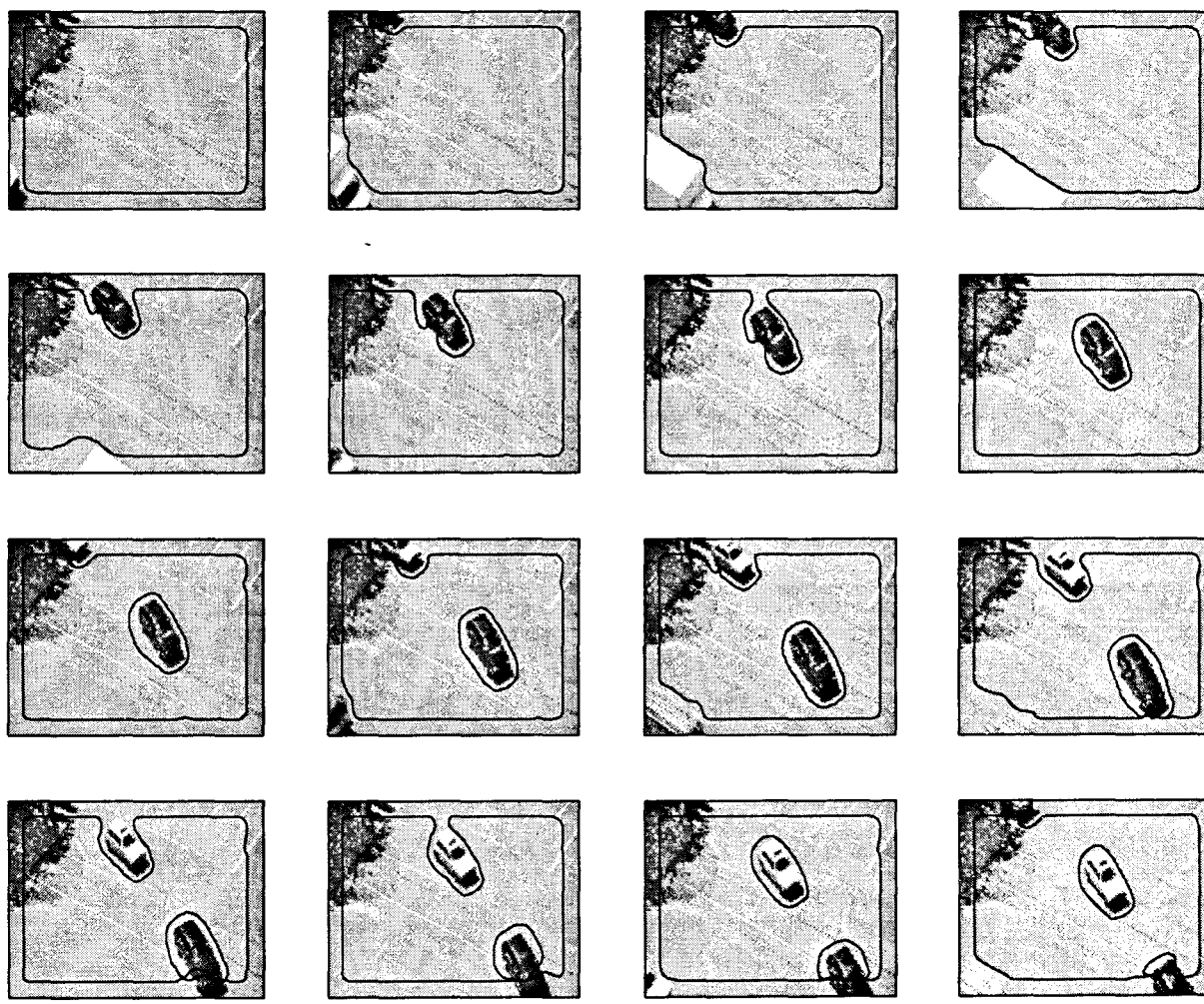


Figure V-1 : Processus d'initialisation automatique pour le suivi d'objets en mouvement.

b) L'accélération du processus de fermeture

Le modèle initial proposé par Kass et *al.* [KAS-87] n'est pas directement utilisable pour notre application. En effet, le contour actif périphérique subit des déformations très importantes lorsqu'un objet pénètre dans la scène.

Afin d'accélérer la fermeture de ce contour autour de l'objet pénétrant dans la scène, nous avons introduit dans le modèle initial un terme favorisant l'augmentation de la surface du contour actif périphérique. Ainsi, ce dernier se referme beaucoup plus rapidement et permet l'obtention du modèle de l'objet dès sa pénétration totale dans la scène.

L'énergie interne est alors composée de trois termes. Les deux premiers sont ceux vus précédemment: un terme de tension qui régit la distance entre les snaxels et un terme de flexion qui régit la courbure du contour actif.

Ce troisième terme, noté $E^*_{surface}$, et introduit par Rougon [ROU-91], s'exprime sous la forme :

$$E^*_{surface} = \int_0^1 \left(\vec{\delta}(\vec{v}(s)) \wedge \vec{v}_s \right) \vec{k} \, ds$$

où $\vec{v}(s)$ est le vecteur position d'un point du contour actif d'abscisse curviligne s et $\vec{v}_s$ sa dérivée. $\vec{k}$ est le troisième vecteur normé de la base des coordonnées des points du contour actif $(O, \vec{i}, \vec{j}, \vec{k})$. Les contours actifs sont dans le plan $(O, \vec{i}, \vec{j})$, où $\vec{i}$ est le vecteur unitaire des abscisses et $\vec{j}$ celui de l'axe des ordonnées.

La fonction vectorielle $\vec{\delta}(\vec{v})$ permet de contrôler l'expansion ou la contraction du contour actif (suivant son signe) dans la direction normale au contour. Dans notre cas, nous choisissons $\vec{\delta}(\vec{v}) = \delta \times \vec{v}$, avec δ constant. Si l'on discrétise cette expression, on obtient :

$$E^*_{surface} = -\delta \cdot \sum_{i=0}^{i=p-1} \left(x_i \cdot (y_{i+1} - y_i) - y_i \cdot (x_{i+1} - x_i) \right) = -\delta \cdot \sum_{i=0}^{i=p-1} \left(x_i \cdot y_{i+1} - y_i \cdot x_{i+1} \right)$$

Or, la surface S d'un polygone dont les coordonnées des sommets sont notées (x_i,y_i), avec $i \in [0,p]$, est donnée par [SON-94] :

$$S = \frac{1}{2} \left| \sum_{i=0}^{p-1} x_i y_{i+1} - x_{i+1} y_i \right|$$

On en déduit que $E^*_{surface} = -2\varepsilon\delta S$, où le coefficient ε, qui prend les valeurs +1 ou -1, indique l'orientation de la surface. L'énergie $E^*_{surface}$ est donc proportionnelle à la surface

délimitée par le contour actif. L'augmentation de la surface délimitée par le contour actif conduit donc à la diminution de l'énergie du contour actif.

L'expression de l'énergie interne devient ainsi :

$$E_{int}(s) = \alpha \cdot \left\|\vec{v}_s(s)\right\|^2 + \beta \cdot \left\|\vec{v}_{ss}(s)\right\|^2 - (\delta \cdot \vec{v} \wedge \vec{v}_s) \cdot \vec{k}$$

Résolution du nouveau système :

Nous avons choisi la méthode de la descente du gradient pour minimiser l'énergie du contour actif car c'est la moins coûteuse en temps de calcul. Dans ce cadre (*cf.* Chapitre IV,§A), les équations d'Euler deviennent :

$$\begin{cases} \alpha x_{ss} + \beta x_{ssss} + \delta y_s + f_x = \dfrac{\partial E_{\text{contour actif}}}{\partial x} \\ \alpha y_{ss} + \beta y_{ssss} - \delta x_s + f_y = \dfrac{\partial E_{\text{contour actif}}}{\partial y} \end{cases}, \text{ où } f_x = \frac{\partial E_{ext}}{\partial x}, \text{ et } f_y = \frac{\partial E_{ext}}{\partial y}.$$

En approchant les dérivées par des différences finies, on obtient:

$$\begin{cases} \alpha \cdot (-x_{i-1} + 2x_i - x_{i+1}) + \beta \cdot (x_{i-2} - 4x_{i-1} + 6x_i - 4x_{i+1} + x_{i+2}) + \delta(y_{i-1} - y_{i+1}) + f_x(i) = \left(\dfrac{\partial E_{\text{contour actif}}}{\partial x}\right) \\ \alpha \cdot (-y_{i-1} + 2y_i - y_{i+1}) + \beta \cdot (y_{i-2} - 4y_{i-1} + 6y_i - 4y_{i+1} + y_{i+2}) - \delta(x_{i-1} - x_{i+1}) + f_y(i) = \left(\dfrac{\partial E_{\text{contour actif}}}{\partial y}\right) \end{cases}$$

Soit, en regroupant les points de même indice :

$$\begin{cases} \beta \cdot x_{i-2} + (-\alpha - 4\beta) \cdot x_{i-1} + (2\alpha + 6\beta) \cdot x_i + (-\alpha - 4\beta) \cdot x_{i+1} + \beta \cdot x_{i+2} + \delta(y_{i-1} - y_{i+1}) + f_x(i) = \left(\dfrac{\partial E_{\text{contour actif}}}{\partial x}\right) \\ \beta \cdot y_{i-2} + (-\alpha - 4\beta) \cdot y_{i-1} + (2\alpha + 6\beta) \cdot y_i + (-\alpha - 4\beta) \cdot y_{i+1} + \beta \cdot y_{i+2} - \delta(x_{i-1} - x_{i+1}) + f_y(i) = \left(\dfrac{\partial E_{\text{contour actif}}}{\partial y}\right) \end{cases}$$

Ce système peut se mettre sous la forme matricielle suivante :

$$\begin{cases} A \cdot X + B \cdot Y + f_X(X, Y) = \dfrac{\partial E_{\text{contour actif}}}{\partial X} \\ A \cdot Y - B \cdot X + f_Y(X, Y) = \dfrac{\partial E_{\text{contour actif}}}{\partial Y} \end{cases}$$

où la matrice A est identique à la matrice A du chapitre IV :

$$A = \begin{bmatrix} 2\alpha+6\beta & -\alpha-4\beta & \beta & 0 & \cdots & 0 & \beta & -\alpha-4\beta \\ -\alpha-4\beta & 2\alpha+6\beta & -\alpha-4\beta & \beta & 0 & \cdots & 0 & \beta \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \\ -\alpha-4\beta & \beta & 0 & \cdots & 0 & \beta & -\alpha-4\beta & 2\alpha+6\beta \end{bmatrix}$$

et où la matrice B vaut :

$$B = \begin{bmatrix} 0 & \delta & 0 & \cdots & \cdots & 0 & -\delta \\ -\delta & 0 & \delta & 0 & \cdots & \cdots & 0 \\ 0 & -\delta & 0 & \delta & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \ddots & \delta \\ \delta & 0 & \cdots & \cdots & 0 & -\delta & 0 \end{bmatrix}$$

La méthode de résolution par recherche des matrices inverses est rendue très délicate dans ce cas car chacune des deux équations du système dépend à la fois de X et de Y.

Pour résoudre le système, nous utiliserons donc la même procédure itérative que celle décrite au chapitre IV,§A sans faire appel cependant à l'inversion matricielle. En notant (x_i^n, y_i^n) les coordonnées du $i^{ème}$ point du contour actif à l'itération n, le système devient (pour ce $i^{ème}$ point) :

$$\begin{cases} \beta\cdot x_{i-2}^{n-1} + (-\alpha-4\beta)\cdot x_{i-1}^{n-1} + (2\alpha+6\beta)\cdot x_i^{n-1} + (-\alpha-4\beta)\cdot x_{i+1}^{n-1} + \beta\cdot x_{i+2}^{n-1} + \delta\cdot(y_{i-1}^{n-1} - y_{i+1}^{n-1}) + f_{x_{n-1}}(i) = \gamma\cdot\left(x_i^{n-1} - x_i^n\right) \\ \beta\cdot y_{i-2}^{n-1} + (-\alpha-4\beta)\cdot y_{i-1}^{n-1} + (2\alpha+6\beta)\cdot y_i^{n-1} + (-\alpha-4\beta)\cdot y_{i+1}^{n-1} + \beta\cdot y_{i+2}^{n-1} + \delta\cdot(x_{i-1}^{n-1} - x_{i+1}^{n-1}) + f_{y_{n-1}}(i) = \gamma\cdot\left(y_i^{n-1} - y_i^n\right) \end{cases}$$

On obtient alors l'expression des coordonnées du point (x_i^n, y_i^n) en fonction de ses coordonnées à l'itération précédente (x_i^{n-1}, y_i^{n-1}) :

$$\begin{cases} x_i^n = x_i^{n-1} - \frac{1}{\gamma}\left(\beta\cdot x_{i-2}^{n-1} + (-\alpha-4\beta)\cdot x_{i-1}^{n-1} + (2\alpha+6\beta)\cdot x_i^{n-1} + (-\alpha-4\beta)\cdot x_{i+1}^{n-1} + \beta\cdot x_{i+2}^{n-1} + \delta\cdot(y_{i-1}^{n-1} - y_{i+1}^{n-1}) + f_{x_{n-1}}(i)\right) \\ y_i^n = y_i^{n-1} - \frac{1}{\gamma}\left(\beta\cdot y_{i-2}^{n-1} + (-\alpha-4\beta)\cdot y_{i-1}^{n-1} + (2\alpha+6\beta)\cdot y_i^{n-1} + (-\alpha-4\beta)\cdot y_{i+1}^{n-1} + \beta\cdot y_{i+2}^{n-1} - \delta\cdot(x_{i-1}^{n-1} - x_{i+1}^{n-1}) + f_{y_{n-1}}(i)\right) \end{cases}$$

c) La procédure de scission du contour initial

Lorsque le contour actif périphérique se referme autour d'un objet en mouvement, il y a nécessairement deux intersections entre quatre de ses segments pris deux à deux, comme l'illustre la figure V-2. On constate que la partie A du contour actif modélise l'objet « capturé ». La recherche de l'intersection I_1 permet donc de créer un contour actif associé à l'objet à poursuivre. Afin de pouvoir « capturer » d'autres objets, le contour actif périphérique sera quant à lui reconstitué à partir de la partie B par recherche de l'intersection I_2.

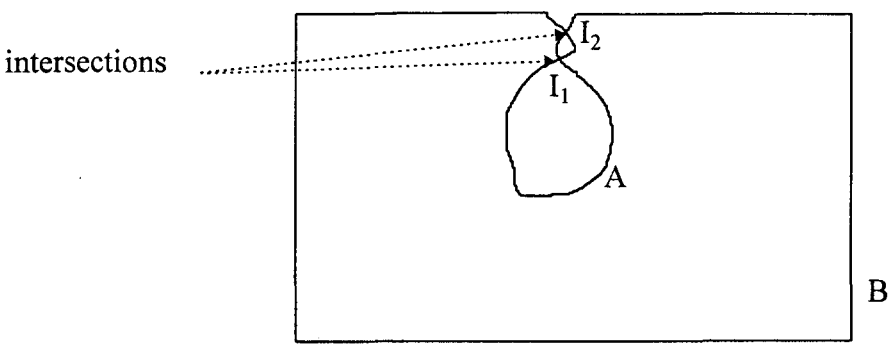


Figure V-2 : Recherche des intersections sur le contour actif périphérique.

Pour déterminer si les deux segments AB et CD se coupent, il suffit de savoir si les points A et B sont de part et d'autre de la droite support du segment CD et, réciproquement, si les points C et D sont de part et d'autre de la droite support du segment AB (*cf.* figure V-3).

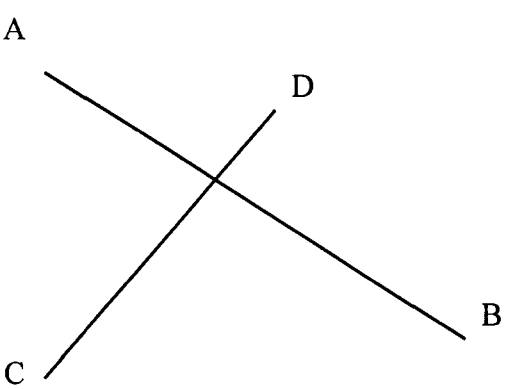


Figure V-3 : Intersection de deux segments.

L'équation de la droite AB est donnée par :

$$(y_B-y_A)x + (x_A-x_B)y + (y_Ax_B-x_Ay_B)=0,$$

celle de la droite CD est donnée par :

$$(y_D-y_C)x + (x_C-x_D)y + (y_Cx_D-x_Cy_D)=0.$$

Le fait que les points A et B soient de part et d'autre des points C et D s'exprime par [LUC-82] :

$$\{(y_D-y_C)x_A + (x_C-x_D)y_A + (y_Cx_D-x_Cy_D)\}.\{(y_D-y_C)x_B + (x_C-x_D)y_B + (y_Cx_D-x_Cy_D)\}<0.$$

Dans un souci d'économie de calculs, ce test est restreint aux segments du contour actif ne se trouvant plus à leur position initiale du fait de la pénétration de l'objet dans la scène. Ceci est illustré figure V-4 : la procédure de scission ne prend en compte que les points du contour actif périphérique représentés en pointillés. La complexité de cet algorithme est en $O(n^2)$, où n est le nombre de segments testés.

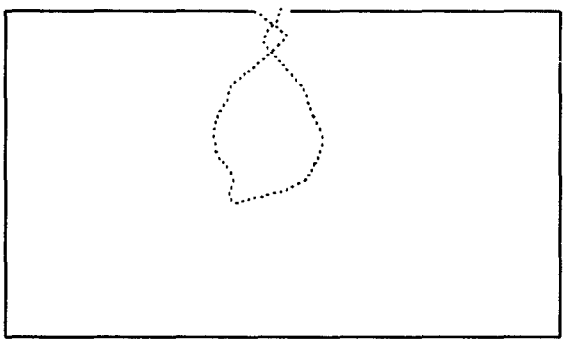


Figure V-4 : Ensemble des segments sur lesquels est réalisé le test d'intersection.

2. Le suivi des modèles

Pour que le contour actif puisse suivre des objets en mouvement, la différence de position d'un même objet entre deux images successives doit rester faible, puisque c'est à partir de la position de l'objet dans l'image précédente que le contour actif se déforme pour converger vers la position de l'objet dans l'image courante. Ainsi, le modèle des contours actifs ne peut être utilisé tel quel pour le suivi d'objets en mouvement que si ces derniers se déplacent très peu entre deux images.

Pour pallier cet inconvénient, il est nécessaire de prédire la position approximative de l'objet dans l'image courante. Pour cela, on calcule la vitesse du contour actif associé à cet objet l'instant (t-1) à partir de ses positions successives dans les deux images précédentes.

En supposant le mouvement de l'objet uniforme, on prédit donc la position de chaque snaxel en supposant que leur vitesse à l'instant t est la même qu'à l'instant (t-1). C'est à partir de cette position prédite (*cf.* figure V-5) que l'on réitère le processus de minimisation de l'énergie dans l'image courante.

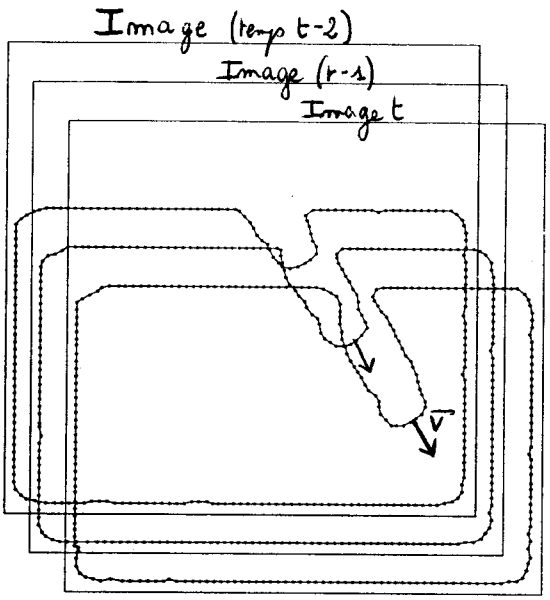


Figure V-5 : Estimation de la position courante d'un objet à partir de ses positions précédentes.

En réalisant une prédiction basée sur la vitesse de chaque snaxel plutôt que sur la vitesse globale du contour actif, on prend en compte les problèmes de perspective. En effet, dans l'image, les points d'un objet n'ont pas tous la même vitesse suivant leurs distances par rapport aux caméras.

La procédure de prédiction de la position des objets entre deux images successives permet au contour actif de suivre des objets en déplacement rapide. Si la prédiction de la position courante ne peut être réalisée (mouvements aléatoires), il est nécessaire d'utiliser une fréquence d'échantillonnage des images suffisamment élevée pour obtenir des déplacements entre deux images compatibles avec l'approche contours actifs.

3. La méthodologie d'ajustement des différents paramètres du contour actif.

L'ajustement des paramètres du contour actif, qui s'effectue par essais successifs, est rendu difficile par leur interdépendance [RAN-92]. Cependant, on peut dégager une méthodologie permettant de simplifier leur détermination. Tout d'abord, il est nécessaire de fixer la distance inter-snaxels, car les valeurs des autres paramètres sont fonctions de cette distance.

Ensuite, on ajuste les paramètres de l'énergie interne : le coefficient α est celui qui a le plus d'importance car il pondère la dérivée première du vecteur position. Il est donc fixé en premier. Les coefficients β et δ sont ajustés facilement ensuite.

Enfin, on pondère l'énergie externe en fonction des images traitées et des coefficients choisis pour l'énergie interne. L'influence des différents paramètres est illustrée par quelques séquences d'images de synthèse.

a) Ajustement de la distance inter-snaxels.

Comme nous l'avons vu précédemment, nous avons choisi de contrôler la distance entre les points du contour actif en la maintenant constante à l'aide d'une procédure de ré-échantillonnage. Les valeurs des différents paramètres sont liées directement à cette distance inter-snaxels, qui est très simple à ajuster. Plus la distance entre deux points du contour actif est faible et plus la détection et le suivi sont précis. La distance inter-snaxels doit également être choisie en fonction de la taille des objets en mouvement, afin que le contour actif périphérique puisse les « capturer » : plus les objets sont petits et plus la distance inter-snaxels doit être faible afin que les objets ne passent pas entre deux snaxels.

La valeur choisie réalise donc un compromis entre la précision et le temps de calcul, ce dernier étant inversement proportionnel à la distance inter-snaxels. Ce paramètre étant fixé, nous pouvons passer à l'ajustement des paramètres intervenant dans l'expression de l'énergie interne.

b) Ajustement des paramètres de l'énergie interne

(1) Ajustement du coefficient de tension α

Plus la valeur du coefficient de tension α est élevée, plus le contour actif est rigide. A une valeur faible correspond une courbe qui épouse bien le contour des objets mais dont l'effet de lissage est trop faible pour empêcher le contour actif de pénétrer à l'intérieur des objets délimités par des contours présentant des discontinuités. Pour une valeur élevée, l'effet de lissage du contour actif est très important, de telle sorte que les contours extérieurs des objets sont modélisés de façon grossière, sans respect des détails. En revanche, le contour actif ne pénètre pas à l'intérieur du contour de l'objet au niveau de ses discontinuités.

Notons que la valeur du paramètre α a également une influence sur la rapidité de la fermeture du contour actif périphérique autour de l'objet. Cette influence est illustrée par la figure V-6 à partir d'une séquence d'images de synthèse : les objets en mouvement sont des disques se déplaçant à vitesse constante et dont les contours présentent des discontinuités. Nous avons sélectionné quelques images de la séquence afin d'alléger l'illustration tout en

présentant les phases essentielles. Dans cette séquence, les contours actifs qui modélisent les objets en mouvement sont visualisés en trait plein afin de les distinguer du contour actif périphérique qui est visualisé par la succession de ses snaxels.

Les résultats de la figure V-6a ont été obtenus pour α=1, tandis que ceux de la figure V-6b correspondent à l'ajustement de α à la valeur 1,8. On constate que le contour actif périphérique se referme plus rapidement autour des objets pour α=1 que pour α=1,8 : il se referme derrière le disque du haut à l'image numéro 31 alors qu'il ne se referme autour de ce disque qu'à l'image 35 pour α=1,8.

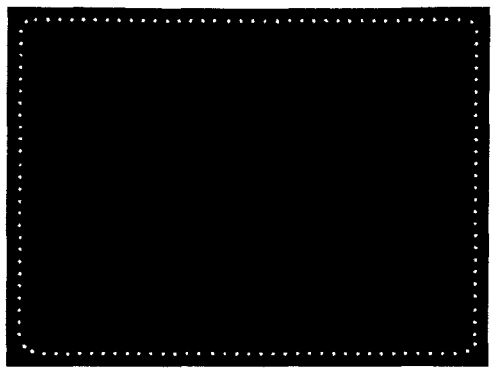


Image 0

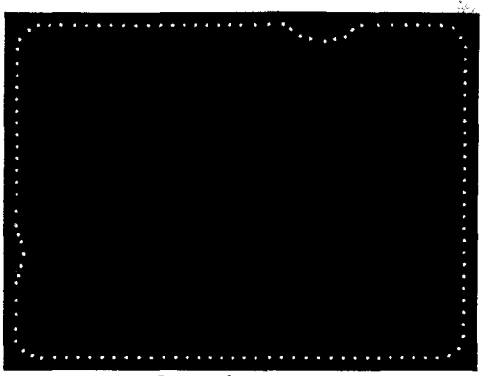


Image 6

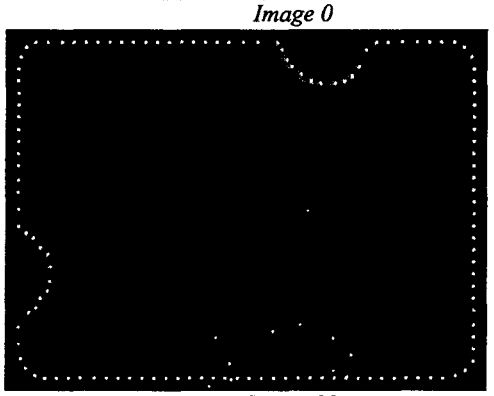


Image 11

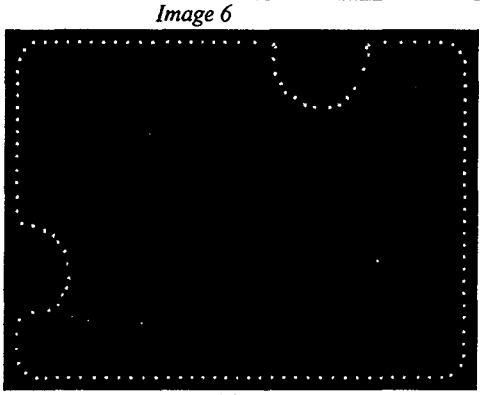


Image 16

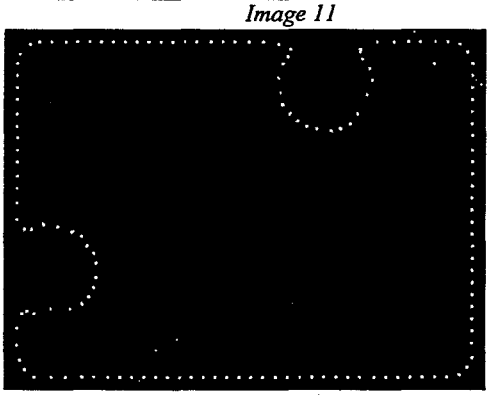


Image 21

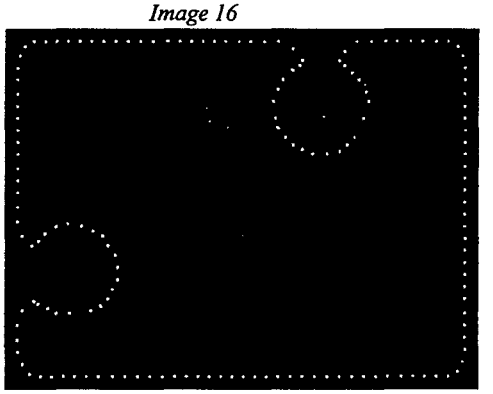


Image 26

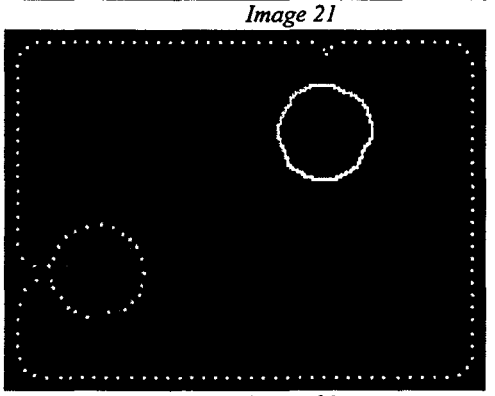


Image 31

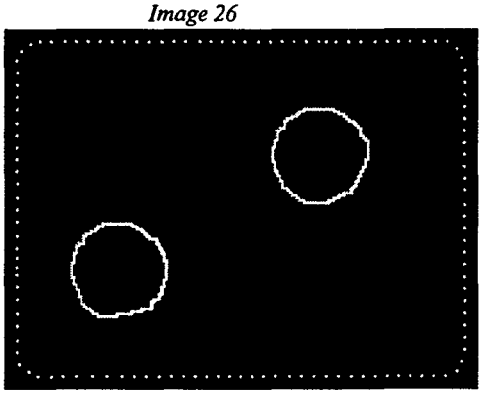


Image 36

a) α=1

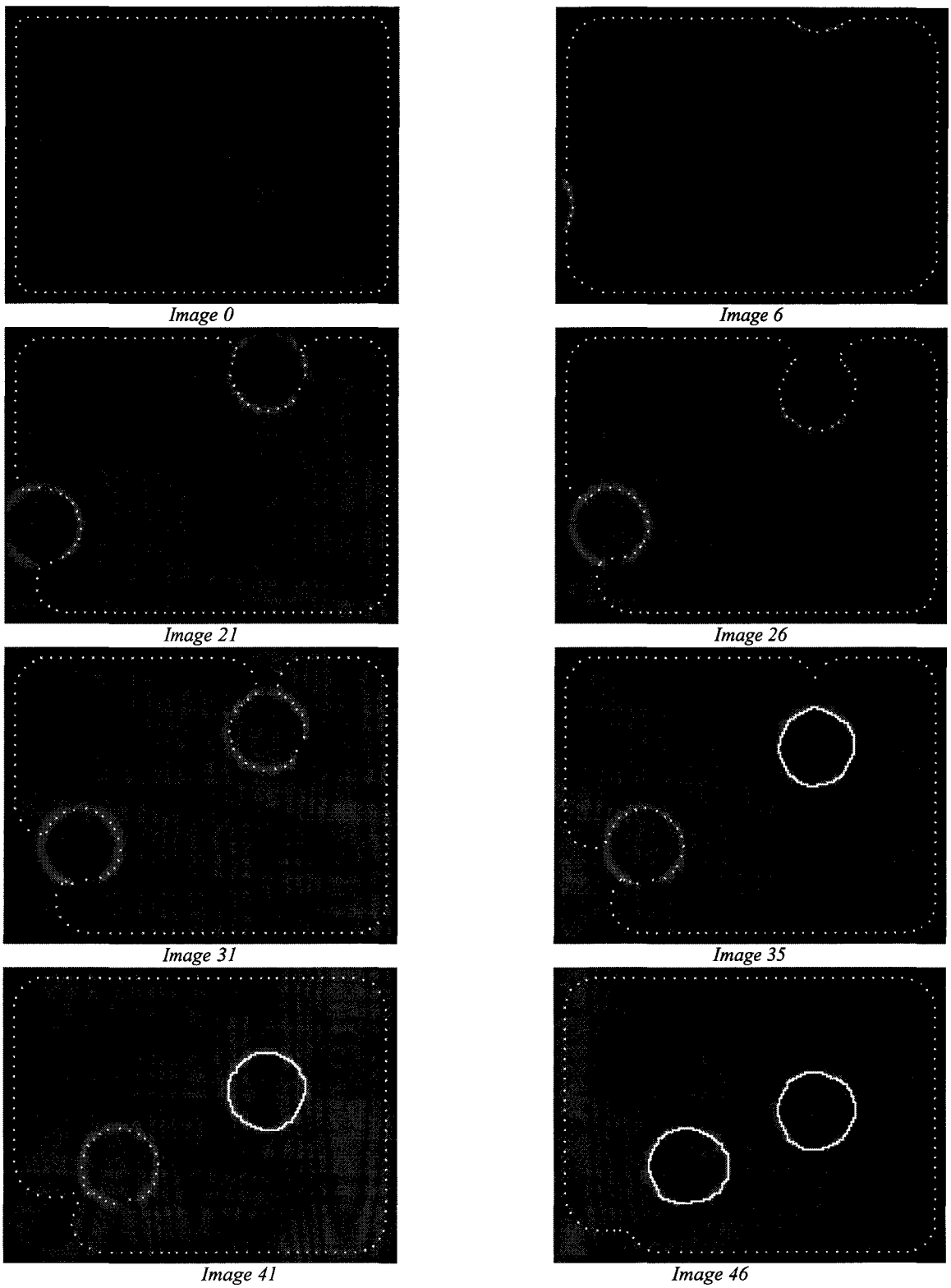


Image 0 *Image 6*

Image 21 *Image 26*

Image 31 *Image 35*

Image 41 *Image 46*

b) $\alpha=1,8$

Figure V-6 : Influence du paramètre α *sur le comportement du contour actif périphérique*

(2) Ajustement du paramètre de flexion β

La valeur du paramètre de flexion β doit être adaptée à la courbure des contours des objets à suivre. Plus cette courbure est élevée et plus la valeur de β doit être grande. Cependant, si la valeur de β est trop élevée, le contour actif n'épouse plus correctement les contours et présente des variations erratiques. Ceci est illustré par la figure V-7 où l'on a choisi β=2,45.

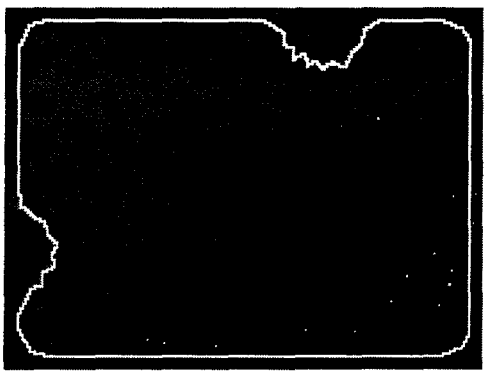


Image 12

Figure V-7 : Effet d'une valeur trop élevée du paramètre β *(*β*=2,45)*

L'ajustement du paramètre β n'est pas critique : sa plage de variation autorisant un suivi correct est grande. A titre indicatif, il est couramment choisi entre 0,1 et 2.

(3) Ajustement du paramètre d'augmentation de surface δ

Comme nous l'avons vu au paragraphe B-1 de ce chapitre, le terme d'augmentation de surface a été introduit pour accélérer la fermeture du contour actif périphérique autour des objets qui pénètrent dans la scène. Plus le paramètre δ est grand et plus cette fermeture intervient rapidement dans la séquence. Par exemple, pour δ=0,1, le contour actif périphérique se referme autour du premier disque à l'image 31 et autour du second à l'image 34 (*cf.* figure V-8a), alors que pour δ=0,35, il se referme autour du premier disque à l'image 28 et autour du second à l'image 30 (*cf.* figure V-8b). Cependant, l'ajustement de ce paramètre n'est pas très critique pour la fiabilité du suivi. Sa plage d'ajustement acceptable s'étend de 0 à 0,5.

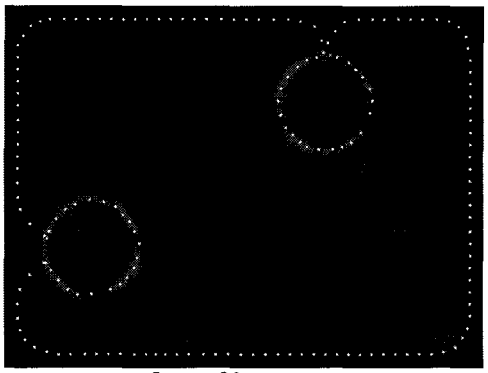


Image 30

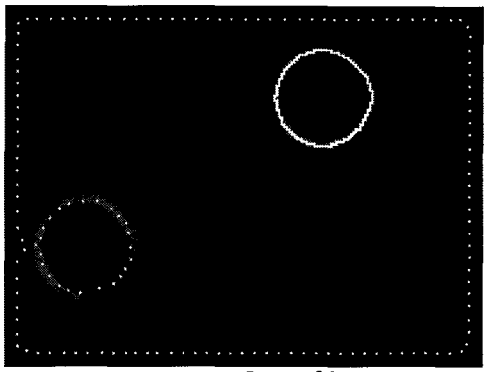


Image 31

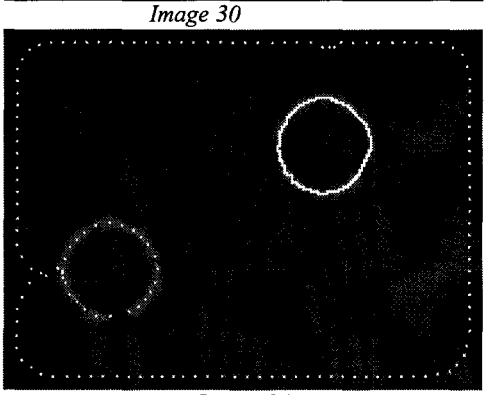


Image 34

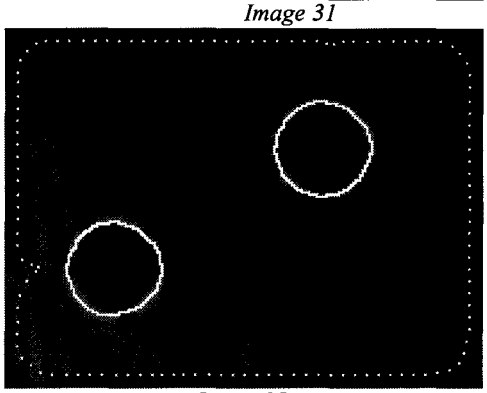


Image 35

a) δ=0,1

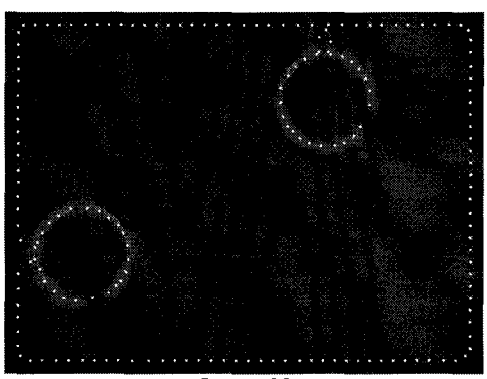


Image 28

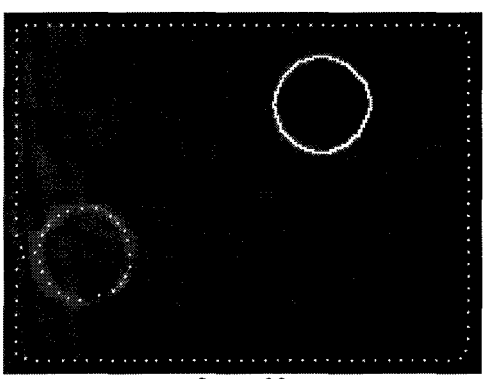


Image 29

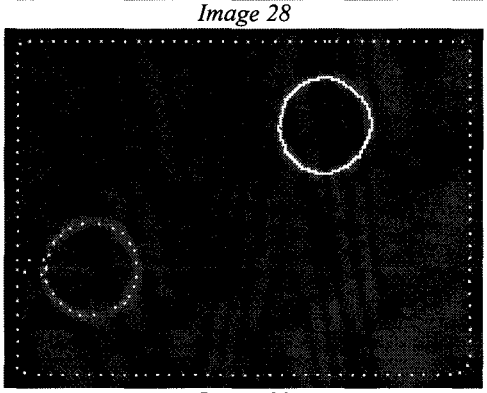


Image 30

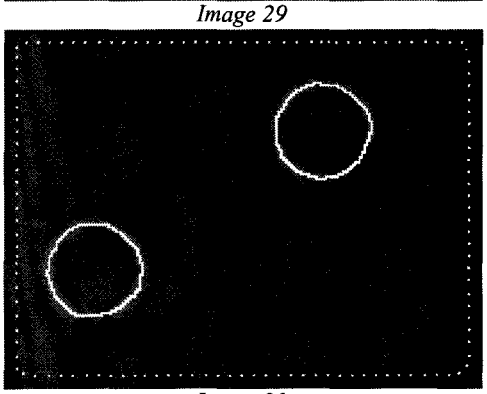


Image 31

b) δ=0,35

Figure V-8 : Influence du paramètre δ

c) Ajustement du coefficient de pondération de l'énergie externe

Les paramètres précédents étaient tous associés aux différents termes de l'énergie interne. On pondère ensuite l'énergie externe, c'est à dire l'énergie associée à l'image, en fonction du type d'images mais également des valeurs données aux paramètres de l'énergie interne. On compensera des valeurs élevées des paramètres de l'énergie interne par un coefficient de pondération de l'énergie externe important. En effet, comme l'on travaille avec des images gradients lissées, les valeurs des niveaux de gris peuvent être assez faibles.

Si les discontinuités dans les contours sont très importantes, on choisira un coefficient de pondération de l'énergie externe négatif, ce qui a pour effet de stabiliser le contour actif autour de l'objet et non pas directement sur son contour. Le contour actif est repoussé par le contour de l'objet, il ne peut donc pas pénétrer à l'intérieur des discontinuités. L'énergie interne permet alors de maintenir le contour actif autour de l'objet, et l'empêche de s'en éloigner.

d) Ajustement du pas de calcul γ

Le pas de calcul γ est utilisé pour la minimisation de l'énergie des contours actifs par la méthode de la descente du gradient (*cf.* Chapitre IV*)*. Son ajustement doit permettre de réaliser un compromis entre la stabilité de la position des snaxels et le temps de calcul. Plus la valeur du pas de calcul γ est élevée, et plus l'évolution du contour actif vers le contour de l'objet en mouvement est rapide. Ceci se fait au détriment de la qualité du modèle, et peut également nuire fortement à la fiabilité du suivi.

Sur la figure V-9 où γ vaut 0,1, on voit que la qualité du modèle est moins bonne que pour les séquences précédentes où γ valait 0,04.

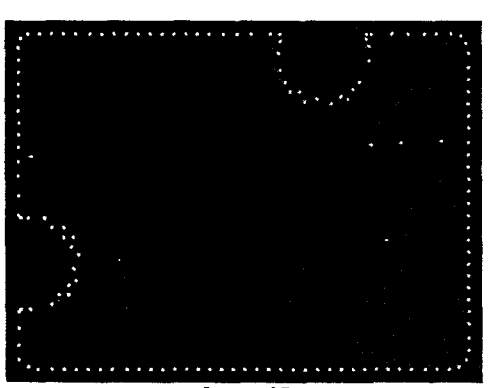


Image 17

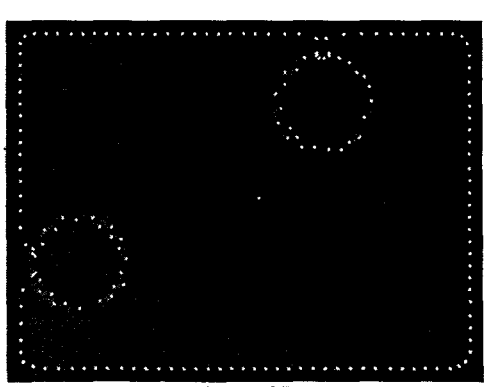


Image 27

Figure V-9 : Mauvaise qualité du modèle(par rapport à la figure V-8) due au choix de γ =0,1

e) Ajustement du nombre d'itérations

Le nombre d'itérations doit être ajusté de manière à permettre la convergence du contour actif sur le contour de l'objet à chaque image. Il est choisi de façon à réaliser un compromis entre précision de la localisation du contour et temps de calcul. Pour notre part, nous le choisissons constant, et suffisamment élevé pour assurer cette convergence. Nous choisissons en général 10 itérations par image.

D'autres auteurs utilisent un critère d'arrêt : lorsque la différence d'énergie du contour actif entre les itérations n et n-1 est inférieure à un certain seuil, le processus de convergence est arrêté.

D. Conclusion

Le nombre important de paramètres, leur interdépendance et la manière empirique de les ajuster sont souvent évoqués pour critiquer les contours actifs. Comme nous venons de le voir, en ajustant ces différents paramètres selon l'ordre proposé et en maîtrisant le rôle joué par chacun d'eux, il devient plus aisé de les choisir pour obtenir un contour actif modélisant précisément les contours des objets. Notons que cela ne signifie pas, bien entendu, qu'il existe toujours un ensemble de valeurs des paramètres permettant d'atteindre l'objectif visé.

Nous allons présenter dans le chapitre suivant l'utilisation du modèle de contours actifs « dynamique » pour la localisation 3D des objets en mouvement dans des séquences d'images stéréoscopiques.

Chapitre VI

Appariement stéréoscopique des contours actifs

La procédure de suivi décrite au chapitre précédent est utilisée pour chacune des séquences d'images droite et gauche obtenue avec le stéréoscope. Ainsi nous disposons d'un modèle de contour actif pour chacun des objets en mouvement présents dans les images droites et gauches de la séquence stéréoscopique. Ces modèles peuvent être utilisés en tant que primitives pour la procédure d'appariement stéréoscopique. En effet, ils constituent une représentation compacte, au sens où le nombre de primitives est réduit, et unique de chaque objet. L'appariement stéréoscopique va donc nous permettre la localisation 3D des objets en mouvement.

A. Appariement des contours actifs

1. Attributs associés aux primitives

Le modèle de contour actif associé à chaque objet en mouvement est utilisé comme primitive. Nous proposons de caractériser ce modèle par trois attributs géométriques : sa forme et sa surface que nous quantifierons directement à partir de la formulation mathématique des contours actifs, ainsi que par les coordonnées de son barycentre. Ce sont ces trois attributs qui seront utilisés pour réaliser la mise en correspondance.

Nous allons montrer sur quelques exemples concrets que l'énergie des contours actifs permet effectivement de caractériser la forme des objets.

Reprenons l'expression de l'énergie interne des contours actifs :

$$\mathbf{E}_{int} = \underbrace{\int_0^1 \left(\alpha \cdot \|\vec{v}_s(s)\|^2 + \beta \cdot \|\vec{v}_{ss}(s)\|^2\right) ds}_{1^{er}\text{ terme}} - \underbrace{\int_0^1 \left(\delta \cdot \vec{v} \wedge \vec{v}_s\right) \cdot \vec{k}\, ds}_{2^{ème}\text{ terme}}$$

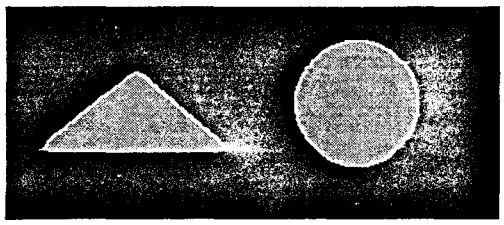
où $\vec{v}(s) = (x(s), y(s))^T$ représente un point du contour actif d'abscisse curviligne s, et où $\vec{v}_s(s)$ et $\vec{v}_{ss}(s)$ sont respectivement les dérivées première et seconde de $\vec{v}(s)$ par rapport à s, comme nous l'avons vu au chapitre V. Rappelons que le premier terme de l'énergie interne caractérise en quelque sorte la forme du modèle, car il contrôle sa longueur et sa courbure alors que le second terme de l'énergie interne, noté $\mathbf{E}_{surface}$, est directement proportionnel à la surface comprise à l'intérieur du modèle de contour actif. Nous nommerons désormais le premier terme de l'énergie interne $\mathbf{E}_{forme}$, $\mathbf{E}_{forme}$ étant la somme de $\mathbf{E}_{tension}$ et de $\mathbf{E}_{flexion}$:

$$\mathbf{E}_{int} = \mathbf{E}_{forme} + \mathbf{E}_{surface}$$

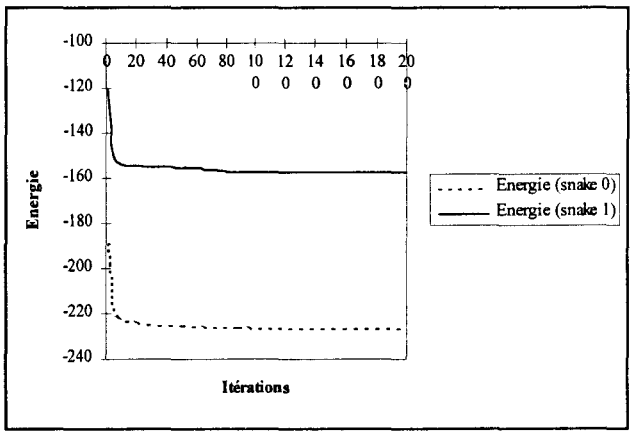
$$\text{avec } \mathbf{E}_{\text{forme}} = \int_0^1 \alpha \cdot \|v_s(s)\| + \beta \cdot \|v_{ss}(s)\| ds \text{ et } \mathbf{E}_{\text{surface}} = -\int_0^1 (\delta \cdot \vec{v} \wedge \vec{v}_s) \cdot \vec{k}\, ds$$

Nous allons étudier de façon systématique les variations des différents termes d'énergie. Pour que cette étude soit valable, il faut s'assurer que les contours actifs utilisés se sont stabilisés sur les contours des objets qu'ils modélisent. Toutes les valeurs données par la suite sont obtenues après 20 itérations, car on constate que l'énergie se stabilise avant ce nombre d'itérations, comme l'illustre la figure VI-1b. L'écart avec la valeur finale obtenue après 200 itérations n'est que de 1,3% pour le contour actif n°0 et de 1,8% pour le contour actif n°1.

Cette stabilisation après 20 itérations est obtenue en initialisant le contour actif au voisinage immédiat du contour des objets, comme pour toutes les autres images de ce chapitre.



a) Objets de synthèse étudiés.



b) Evolution de l'énergie associée aux objets de synthèse

Figure VI-1 : Variation de l'énergie des contours actifs en fonction des itérations.

La figure VI-1a contient un triangle et un cercle, modélisés chacun par un contour actif, représenté en blanc. Le contour actif numéro 0 correspond au cercle, et le contour actif numéro 1 au triangle. Les paramètres utilisés sont : α=4, β=0,5 et δ=0,3. Les énergies interne et externe ont le même poids, et la distance entre snaxels est de 3 pixels.

a) Propriétés de l'attribut surface

La surface est un attribut classique des primitives. On ne détaillera donc pas son utilisation.

b) Propriétés de l'attribut E_{forme}

Considérons l'exemple de la figure VI-2. Cette image contient des objets de formes et de surface différentes, qui sont modélisés par des contours actifs représentés en blanc.

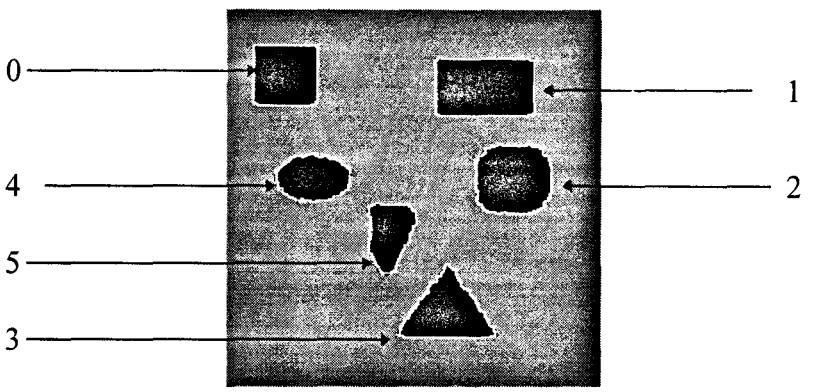
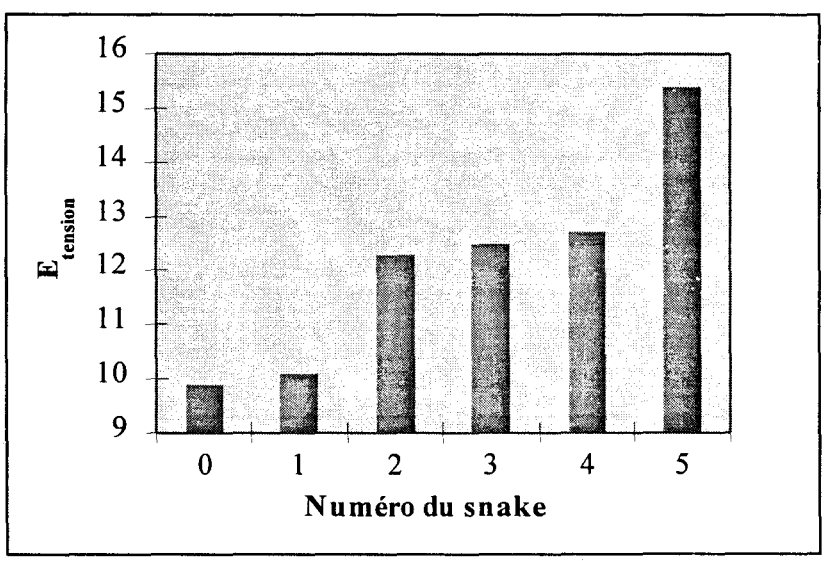


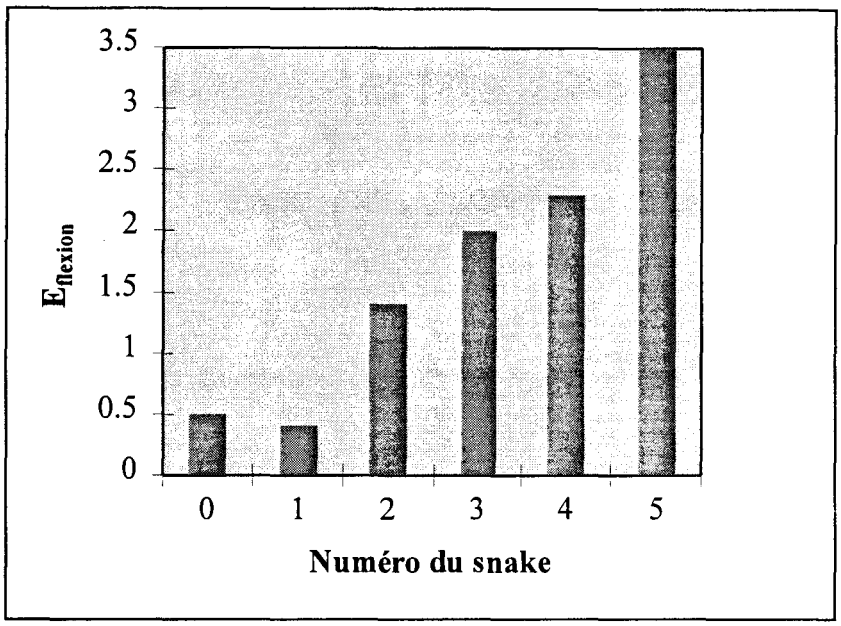
Figure VI-2 : Objets de formes différentes modélisés par des contours actifs

Nous avons représenté les différentes valeurs des termes d'énergie de chacun de ces contours actifs (*cf.* figure VI-3). Ces valeurs ont été divisées par le nombre de points des contours actifs correspondants, afin de pouvoir les comparer.

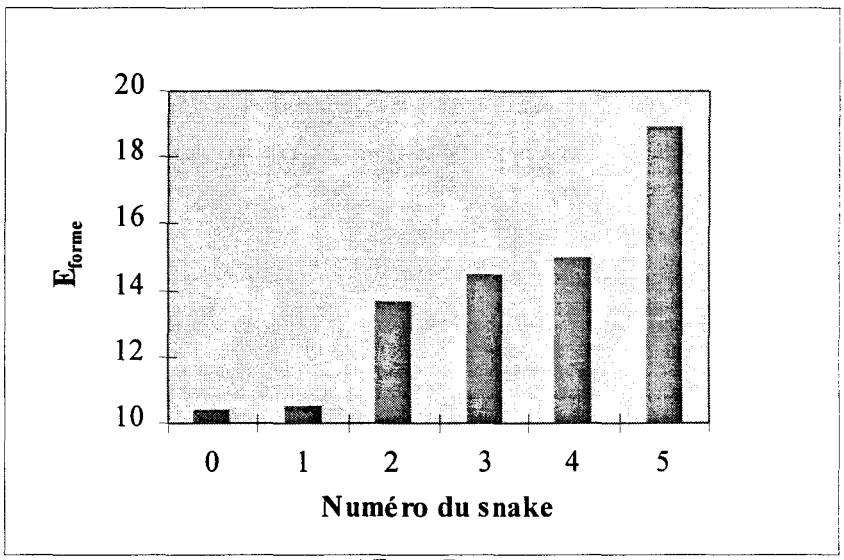
Nous constatons que le terme $E_{flexion}$ joue un rôle important : sur cet exemple, il semble être plus discriminant que le terme $E_{tension}$. L'énergie totale ne peut pas être utilisée comme attribut. En effet, ses valeurs dépendent du contraste des objets, puisque l'énergie externe est fonction du gradient de l'intensité de l'image. On pourrait donc avoir deux objets de même énergie totale mais de formes différentes.



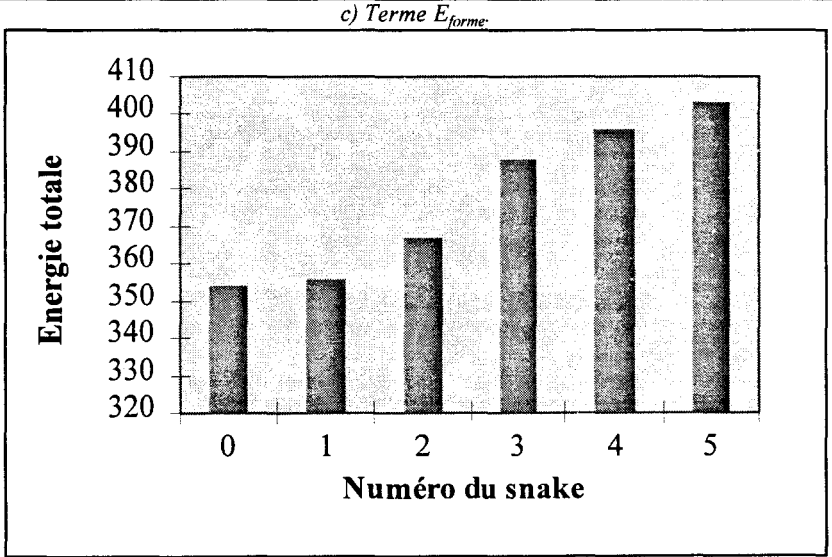
a) Terme $E_{tension}$



b) Terme $E_{flexion}$



c) Terme E_{forme}.



d) Energie totale.

Figure VI-3: Valeurs des différents termes d'energie.

Après avoir vu sur un exemple les variations de l'énergie pour des objets de formes différentes, nous allons maintenant étudier les propriétés de l'attribut E_{forme}. Nous étudierons tout d'abord ses propriétés d'invariance géométrique, puis ses propriétés discriminantes. Cette étude est menée sur des images de synthèse contenant un triangle, auquel on fait subir différentes transformations.

(1) Invariance géométrique

Nous allons étudier les propriétés d'invariance géométrique de l'énergie E_{forme} par rapport aux transformations usuelles (translation, rotation, homothétie).

(a) Invariance en translation.

Le triangle est translaté d'un vecteur (10 pixels, 10 pixels) dans 8 positions successives. On constate que l'énergie E_{forme} du contour actif ne varie quasiment pas lors de la translation (*cf.* figure VI-4). Il n'y a que 5% de variations entre les valeurs extrêmes.

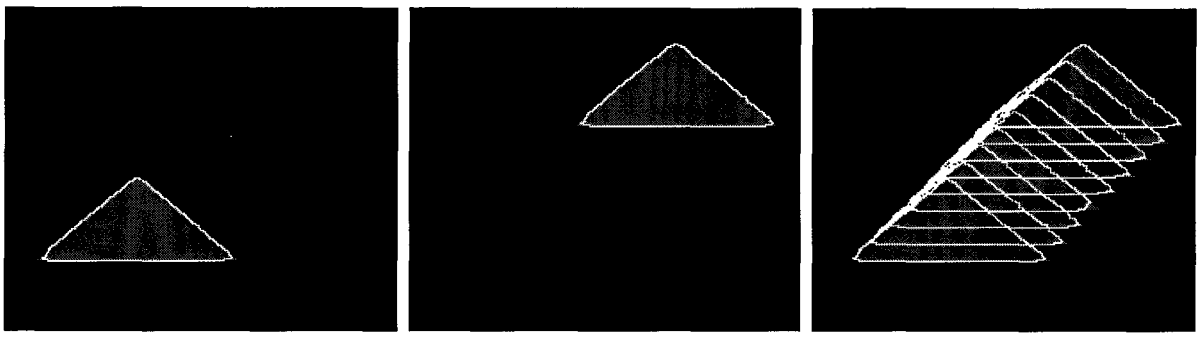
(b) Invariance en rotation.

Afin d'étudier les effets de la rotation sur l'énergie E_{forme}, nous faisons tourner l'objet par pas de 10° jusqu'au tour complet. Un extrait de la série d'images est donné figure VI-5.

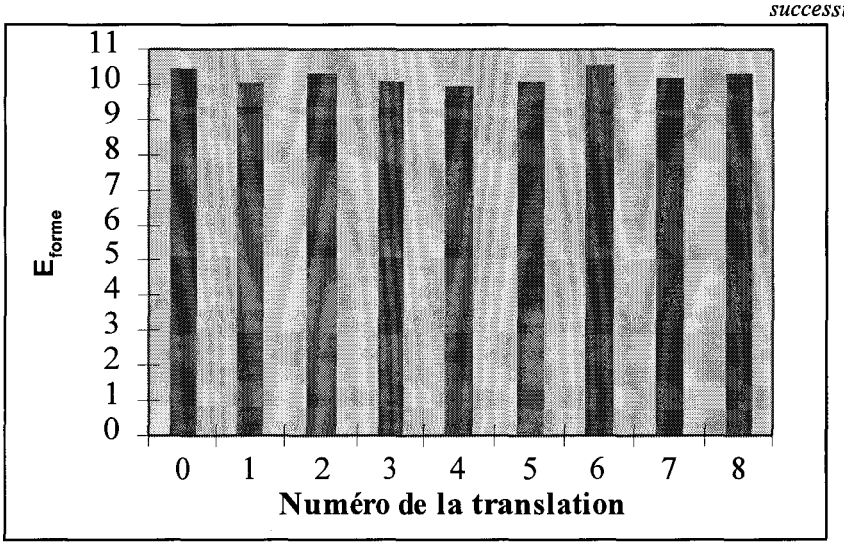
Les variations de l'énergie E_{forme} en fonction de l'angle de rotation sont représentées figure VI-5d. L'énergie E_{forme} n'est donc quasiment pas sensible à l'orientation de l'objet, puisque l'écart entre les deux valeurs extrêmes est inférieur à 10%.

(c) Invariance par rapport à l'homothétie.

Pour étudier les propriétés d'invariance de E_{forme} par rapport à l'homothétie, nous avons créé une série d'images contenant un triangle dont la taille augmente. Ce triangle est inscrit dans un cercle dont le rayon varie de 50 à 90 pixels par pas de 5 pixels. Le premier et le dernier triangle de la série sont représentés sur la figure VI-6, ainsi que les variations de l'énergie E_{forme}. On constate que cette énergie est quasiment indépendante de la taille du triangle : ses variations en fonction de la taille du triangle sont inférieures à 10%.

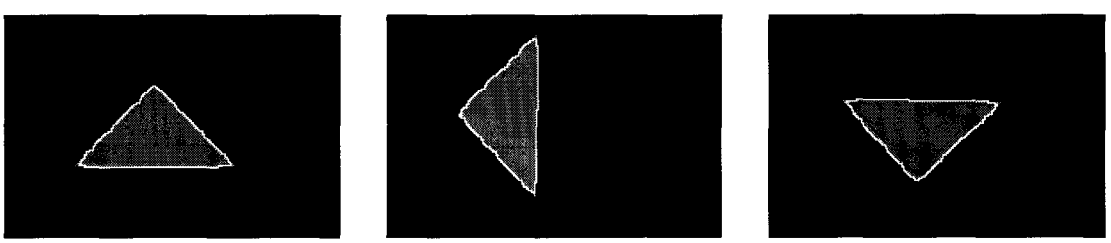


a) Première image *b) Dernière image* *c)Superposition des 9 positions successives*

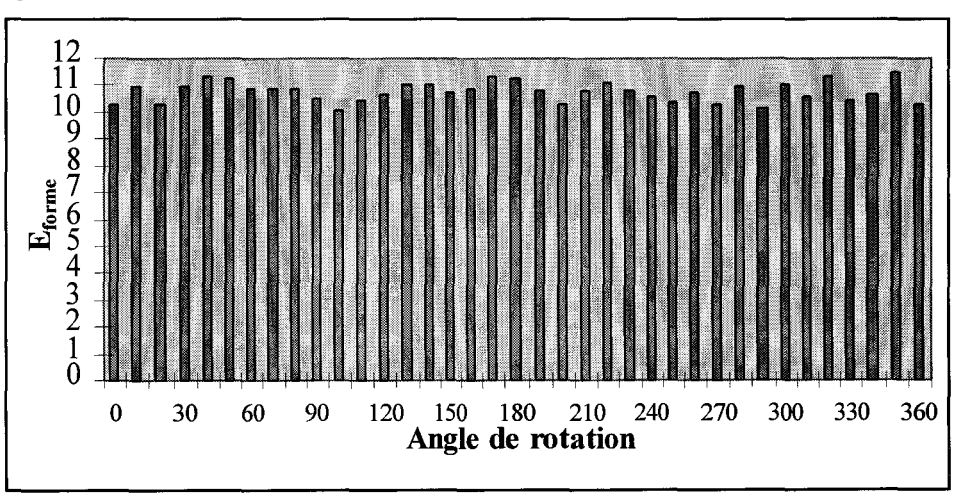


d) Valeurs successives de E_{forme}

Figure VI-4 :Influence de la translation d'un objet sur la valeur de E_{forme}

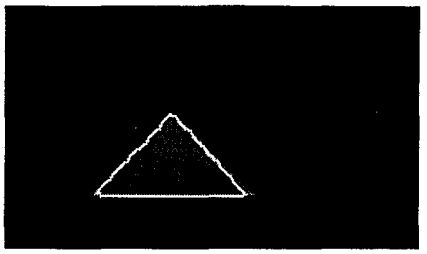


a) Image de référence *b) Rotation de 90°* *c) Rotation de 180°*

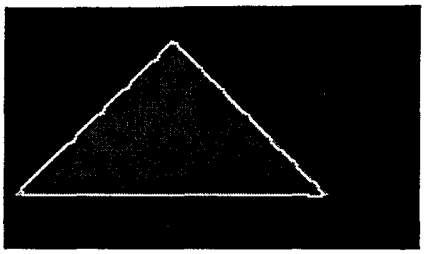


d) Valeurs successives de E_{forme}

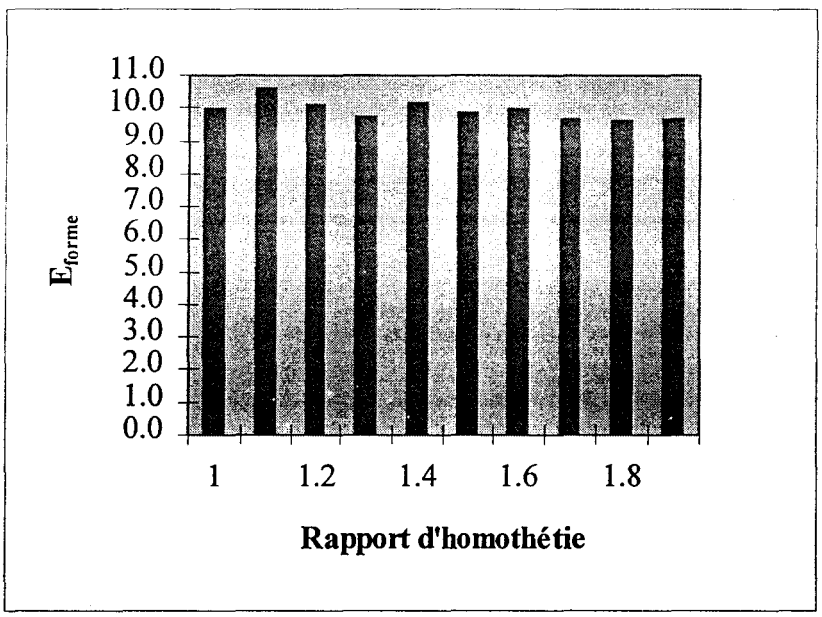
Figure VI-5 :Influence de la rotation d'un objet sur la valeur de E_{forme}



a) Première image



b) Dernière image



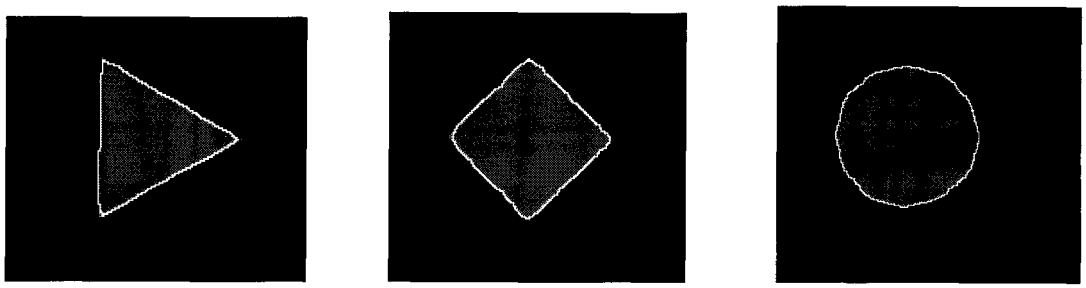
d) Valeurs successives de E_{forme}

Figure VI-6 :Influence de la taille d'un objet sur la valeur de E_{forme}

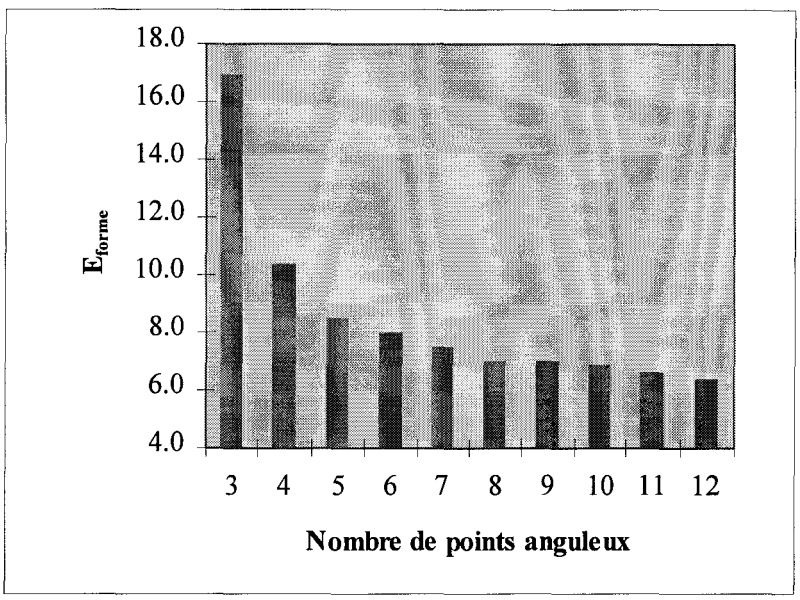
(2) Propriétés discriminantes de l'énergie E_{forme}

Considérant que le nombre de points anguleux d'un objet est un attribut discriminant, nous proposons de mettre en évidence la relation entre la valeur de l'énergie de forme E_{forme} et ce nombre de points. Pour cela, l'objet choisi est un polygone régulier de 3 à 12 côtés, de surface constante (*cf.* figure VI-7).

Notons que cette variation n'est pas linéaire et correspond assez bien à notre perception de la forme des objets : nous différencions plus facilement un triangle d'un carré qu'un polygone de 11 côtés d'un dodécagone (*cf.* figure VI-7d).



a) Triangle *b) Carré* *c) Dodécagone*



d) Valeurs successives de E_{forme}

Figure VI-7 :Influence du nombre de points anguleux d'un objet sur la valeur de E_{forme}

c) Conclusion sur les propriétés de l'attribut forme

Après cette étude, nous pouvons conclure que l'énergie E_{forme} constitue un attribut pertinent des modèles : il caractérise leur forme en restant peu sensible à leur taille, à leur position ou à leur orientation. Il est donc à la fois intrinsèque et discriminant.

2. Sélection des candidats à l'appariement

La procédure d'appariement utilisée comporte un tri préliminaire des modèles d'objets basé sur l'utilisation de la contrainte épipolaire. On a vu au chapitre III que deux points homologues devaient nécessairement se trouver sur la même épipôle. Cette propriété est vérifiée par les deux points homologues correspondant aux barycentres des images des objets. La contrainte épipolaire sera par conséquent appliquée aux barycentres des modèles associés à chaque objet, en tolérant une certaine marge d'incertitude.

On tolère un écart de ±2 pixels sur la position des deux barycentres d'objets homologues. Cette tolérance est nécessaire en raison des erreurs introduites principalement par :

- **la prise de vue :**
  - ♦ **le calibrage,**

    qui doit assurer que les épipôles sont parallèles et confondues avec les lignes horizontales des capteurs, n'est pas d'une fiabilité absolue. Les caméras sont calibrées avant la prise de vue mais elles peuvent subir des chocs et des vibrations durant la prise de vue, décalant légèrement la position des épipôles.
  - ♦ **la distorsion des objectifs**

    entraîne une déformation des épipôles, qui ne sont plus exactement des droites horizontales mais des courbes, surtout aux extrémités du champ. Une des formes de distorsion la plus couramment observée est représentée figure VI-8 .

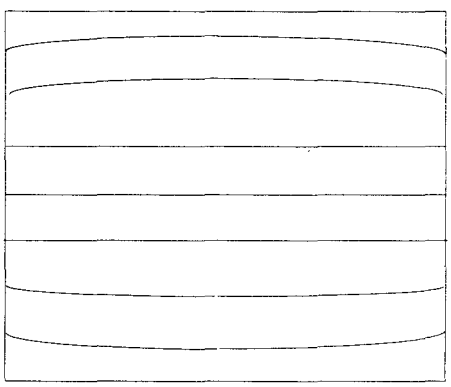


Figure VI-8 :Déformation des épipôles due à la distorsion des objectifs.

La conséquence de cette distorsion est que deux points homologues situés dans les bords du champ de visée des objectifs n'ont pas la même ordonnée dans l'image.

- **le calcul du barycentre :**

  Le calcul des coordonnées des points du contour actif s'appuie, au niveau de l'énergie externe, sur l'image, qui est par définition une fonction discrète. Il en résulte une imprécision de l'ordre du pixel. Cette imprécision se reporte sur le calcul du barycentre. D'autre part, les deux images d'un même objet à travers chacune des caméras du stéréoscope ne sont pas forcément identiques, d'où un décalage dans leurs barycentres. Cependant, cette déformation est d'autant moins importante que les objets sont éloignés du stéréoscope.

3. Calcul de la fonction de coût

Les candidats à l'appariement retenus précédemment ont leur barycentre situé sur la même épipôle. Il faut maintenant trouver parmi ces modèles les paires de modèles homologues correspondant à un même objet dans la scène réelle. Pour ce faire, on associe un coût $M_{coût}$ à chaque appariement potentiel de telle sorte qu'un appariement correct conduise au minimum de ce coût.

Nous avons introduit une fonction de coût $M_{coût}$ qui est une somme pondérée de facteurs de coût. On affecte un poids à chacun des facteurs de coût retenu en fonction de l'importance des caractéristiques qu'ils représentent.

Pour la mise en correspondance des objets, il nous faut donc choisir des facteurs de coût basés sur des caractéristiques discriminantes des objets concernés par l'application. Nous avons choisi d'utiliser leur **forme** et leur **surface,** quantifiées respectivement par les énergies E_{forme} et $E_{surface}$ du contour actif correspondant.

Soit G_j le $j^{ième}$ contour actif de l'image gauche, et soit D_k le $k^{ième}$ contour actif de l'image droite. On choisit comme premier facteur de coût F_{forme} la différence de forme relative entre les deux contours actifs G_j et D_k, que l'on quantifie par :

$$F_{forme}(G_j, D_k) = \left| \frac{E^*_{forme}(G_j) - E^*_{forme}(D_k)}{E^*_{forme}(G_j) + E^*_{forme}(D_k)} \right|$$

Le second facteur de coût $F_{surface}$ est égal à la différence de surface entre les deux contours actifs G_j et D_k :

$$F_{surface}(G_j, D_k) = \left| \frac{E^*_{surface}(G_j) - E^*_{surface}(D_k)}{E^*_{surface}(G_j) + E^*_{surface}(D_k)} \right|$$

L'ordonnée du barycentre du modèle de l'objet intervient également dans la fonction de coût afin d'apparier les modèles dont les barycentres ont les ordonnées les plus proches. On choisit comme troisième facteur de coût $F_{ordonnée}$:

$$F_{ordonné e}(G_j, D_k) = \left| y_{bary}(G_j) - y_{bary}(D_k) \right|$$

Au total, le coût de l'appariement des contours actifs gauche G_j et droit D_k, noté $M_{coût}(G_j, D_k)$, est le suivant :

$$M_{coût}(G_j, D_k) = w_1 F_{forme}(G_j, D_k) + w_2 F_{surface}(G_j, D_k) + w_3 F_{ordonnée}(G_j, D_k)$$

où w_1, w_2 et w_3 sont des coefficients de pondération.

Cependant, cette définition de la fonction de coût ne suffit pas à résoudre tous les cas. En effet, imaginons qu'il y ait dans les images droite et gauche deux modèles situés sur la même épipôle, de forme et de surface différentes ne constituant pas un couple d'objets homologues. Ceci peut se produire par exemple dans la configuration de la figure VI-9, où un objet masque en partie chacun des champs de visée des deux caméras. Si ces modèles sont les seuls candidats à l'appariement sur cette épipôle, ils réalisent nécessairement le minimum de la fonction de coût. L'algorithme doit cependant interdire leur appariement.

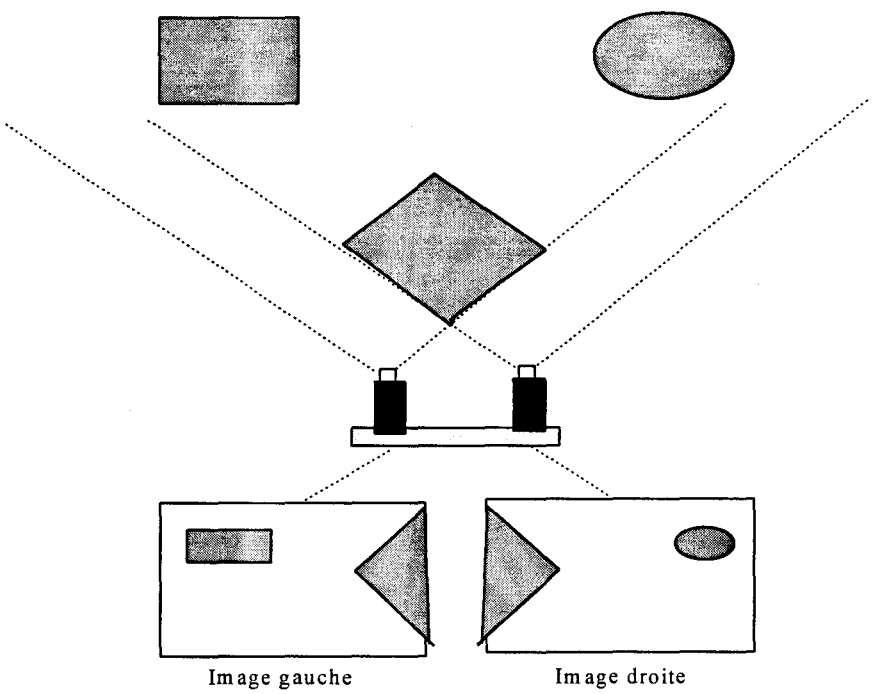


Figure VI-9 :Cas particulier d'obtention d'objets non homologues situés sur la même épipôle.

C'est pourquoi, si les différences de forme F_{forme} (G_j, D_k) et de surface $F_{surface}(G_j, D_k)$ entre les deux contours actifs G_j et D_k sont trop importantes, l'appariement de G_j et D_k est rejeté, même si G_j et D_k sont les deux seuls contours actifs des images droites et gauches situés sur la même épipôle.

4. Suivi temporel des appariements

Pour améliorer la rapidité et la fiabilité de l'algorithme, on ne réitère l'appariement à l'instant (t+1) que si un nouvel objet est entré dans la scène. Dans le cas contraire, on vérifie simplement que les appariements réalisés à l'instant t sont encore corrects, c'est à dire que les valeurs F_{forme} et $F_{surface}$ associées à ces couples sont inférieures au seuil choisi. Si ce n'est pas le cas, on applique l'algorithme d'appariement intégralement.

B. Comparaison avec la méthode de Brint et Brady

Brint et Brady [BRI-90] ont également proposé un algorithme d'appariement stéréoscopique basé sur l'utilisation des contours actifs. Les points de contour obtenus par l'application d'un filtre de Canny sont chaînés pour décrire une courbe : un contour actif est alors initialisé à partir d'un échantillonnage des points de la courbe. Le minimum de l'énergie de ce contour actif est recherché en utilisant la programmation dynamique.

L'appariement des courbes se fait alors en mesurant « la quantité de déformation » nécessaire pour passer du modèle de contour actif dans l'image droite au modèle de contour actif dans l'image gauche. Cette « quantité de déformation » est calculée par l'intermédiaire d'une nouvelle énergie introduite dans la formulation mathématique des contour actifs.

1. Expression de l'énergie des contours actifs

Cette énergie, appelée énergie de croisement et notée E_{cross}, est calculée pour chaque couple de snaxels homologues droits et gauches. L'énergie E_{cross} est composée de deux termes : un terme contenant le gradient de la disparité, noté $E_{disparity\ gradient}$, et un terme de similarité, destiné à favoriser l'appariement des points qui se trouvent sur des contours de formes similaires.

L'expression de $E_{disparity\ gradient}$ est la suivante :

$$E_{disparity\,gradient}(i) = \frac{\beta\left|ep^{\perp} \cdot d_i\right| + \gamma\left|ep \cdot d_i\right|}{\partial s_{left}(i) + \partial s_{right}(i)}$$

où :

- $\delta s_{left}(i)$ est la distance séparant les points d'indices (i) et (i-1) dans l'image gauche.
- $\delta s_{right}(i)$ est la distance séparant les points d'indices (i) et (i-1) dans l'image droite. Ces calculs de distance impliquent que les deux courbes candidates aient le même nombre de points.
- ep est le vecteur parallèle à l'épipôle
- $ep^{\perp}$ est le vecteur orthogonal à ep
- d_i est la disparité calculée au point d'indice i, définie à l'aide de la position du point courant r(i) et du point précédent r(i-1)
- β et γ sont des paramètres constants.

L'énergie de similarité est calculée comme étant la différence des énergies internes des deux courbes à apparier.

2. Appariement des contours actifs

A partir de l'une des deux images stéréoscopiques, on cherche les courbes candidates à l'appariement dans l'autre image comme étant celles qui coupent les épipôles homologues. Si les courbes ont des longueurs voisines, avec une tolérance de 30%, elles sont retenues pour la phase principale de la procédure d'appariement.

Dans cette phase, les énergies internes sont minimisées conjointement avec E_{cross}. Les appariements dont l'énergie $E_{disparity\ gradient}$ est inférieure à un certain seuil sont éliminés. Après cette étape, des appariements multiples peuvent subsister. Ceci provient du fait que les deux images traitées par les auteurs comportent des courbes de contour qui sont des droites verticales, pour lesquelles l'énergie de similarité n'est pas discriminante. Un arbre de recherche est ensuite construit afin de ne garder que les appariements corrects. Pour cela, les contraintes d'unicité et d'ordre sont utilisées.

Notons que cet algorithme prend plusieurs minutes sur station Sun pour traiter une paire d'images stéréoscopiques [BRI-90]. Ce temps de calcul important est à imputer à la minimisation de l'énergie conjointe E_{cross} pour tous les couples de courbes homologues droite et gauche. D'autre part, l'énergie est minimisée en utilisant la programmation dynamique qui, comme on l'a vu précédemment, n'est pas une méthode très rapide. Enfin, cet algorithme n'utilise pas la notion d'objet pour réaliser les appariements, les primitives à apparier étant des portions du contour des objets.

C. Conclusion

L'utilisation des contours actifs pour l'appariement stéréoscopique présente d'importants avantages par rapport aux méthodes d'appariement basées sur des primitives plus élémentaires. En particulier :

- **nombre de primitives très réduit :** dans une image, les points de contours se chiffrent par milliers. On gagne un facteur 10 environ dans le nombre de primitives en passant aux segments de contours, et encore un facteur 10 en utilisant les modèles de contours actifs. Cette réduction du nombre de primitives entraîne un

important gain de temps de calcul. Les primitives retenues constituent donc une représentation compacte des objets.

- **primitives directement exploitables** en tant que modèles des objets physiques. L'étape de reconstruction n'est plus nécessaire.
- **obtention directe des attributs associés aux primitives :** la forme et la surface des modèles de contours actifs se déduisent directement de leur formulation mathématique.

Dans le chapitre suivant, nous présentons les résultats d'appariement et de localisation que nous avons obtenus sur des séquences d'images de synthèse et d'images réelles.

Chapitre VII

Présentation des résultats

Nous présentons dans ce chapitre les résultats de notre algorithme de suivi appliqué à une séquence monoculaire d'images réelles, et les résultats du suivi et de la localisation 3D sur une séquence stéréoscopique d'images de synthèse, ainsi que sur une séquence stéréoscopique d'images réelles.

A. Suivi d'objets en mouvement dans une séquence monoculaire.

Nous avons appliqué notre algorithme de suivi, décrit dans le chapitre V, à une séquence réelle de trafic autoroutier (*cf.* figure VII-1). La caméra utilisée a été placée sur un pont surplombant l'autoroute. Les objets en mouvement ont été extraits de la séquence par l'opérateur de Vieren, puis les images résultats ont été moyennées afin de lisser les contours, et donc de faciliter la convergence de la méthode de la descente du gradient utilisée pour la minimisation de l'énergie des contours actifs.

Au moment où les véhicules pénètrent dans le champ de la caméra, ils sont très éloignés de celle-ci, et sont donc de très petite taille dans l'image. On voit qu'au début de la séquence, trois véhicules pénètrent quasiment en même temps dans le champ de visée. Puis, au fur et à mesure de leur progression, ils se séparent.

La voiture la plus à gauche dans le champ de la caméra est la première a être modélisée, car elle se distingue rapidement des autres. Le contour actif périphérique, puis le contour actif associé à cette voiture semblent éloignés du contour réel de celle-ci, surtout à sa gauche dans le sens de la marche. Cela est du au fait que la scène est fortement éclairée par le soleil, et donc que les ombres des véhicules sont très marquées. Ces ombres sont détectées par l'algorithme de détection des objets en mouvement, au même titre que les véhicules. Les contours actifs modélisent donc à la fois les véhicules et leur ombre, sans qu'il soit possible de faire la distinction.

Les deux autres voitures à droite de l'image sont modélisées par un contour actif qui leur est propre un peu plus tard dans la séquence, car elles sont très proches l'une de l'autre au début de la séquence.

Au moment où une voiture quitte le champ de visée, son contour actif associé intersecte le contour actif périphérique. Par une procédure similaire à celle permettant de créer un modèle associé à un objet en mouvement par détection des intersections sur le contour actif

périphérique, on fusionne le modèle associé à l'objet et le contour actif périphérique lorsque l'objet quitte la scène.

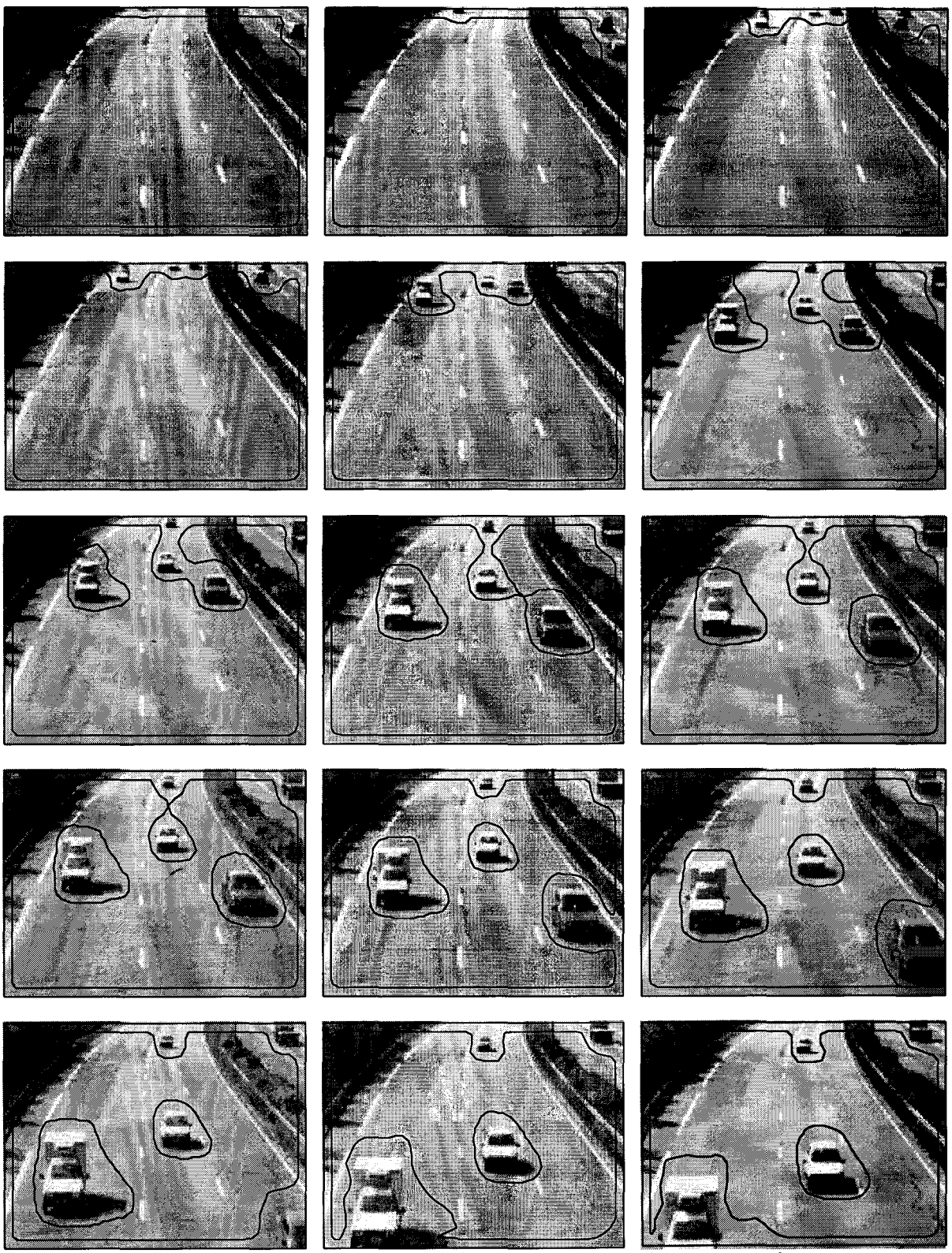


Figure VII-1 :Modélisation et suivi d'objets en mouvement par des contours actifs.

On constate que l'utilisation des contours actifs permet de suivre des objets dont les variations de taille sont très importantes. Ici, ces variations de taille sont dues à la perspective, les voitures étant très éloignées de la caméra lorsqu'elles pénètrent dans son champ de visée, puis très proches lorsqu'elles le quittent.

B. Suivi et localisation 3D d'objets en mouvement

Nous avons également appliqué notre algorithme de modélisation, de suivi et de localisation 3D, décrit dans le chapitre VI, à une séquence d'images de synthèse et à une séquence réelle de trafic routier.

Pour avoir une idée de la précision de la localisation des objets obtenue avec les modèles de contours actifs, nous avons effectué différentes mesures de distance sur des images stéréoscopiques statiques prises en laboratoire. Dans le cas de scènes réelles, nous n'avons pas de moyen de connaître la distance réelle des objets en mouvement par rapport aux caméras. C'est pourquoi nous avons choisi de tester la précision des mesures sur des images statiques, car dans ce cas il nous est aisé de mesurer la distance réelle des objets au stéréoscope.

Les images de laboratoire ont été prises en plaçant des objets plans à des distances de 2,5m et 4m environ du stéréoscope, et à différentes positions dans le champ de visée.

Les contours actifs ont été utilisés pour la segmentation de ces images stéréoscopiques afin d'obtenir une modélisation du contour extérieur des objets dans chacune des images. Les images originales ont été traitées comme suit :

- gradient de Sobel pour extraire les contours des objets.
- moyennage de l'image gradient pour lisser les contours des objets.

On calcule ensuite les coordonnées du barycentre des objets. La disparité entre les abscisses des barycentres permet alors de connaître la position des objets par rapport aux caméras. Les formules de calcul de la distance des objets au stéréoscope sont données en Annexe A.

Puis on compare cette valeur calculée avec la valeur de la distance du milieu de l'entraxe des caméras au barycentre de l'objet, mesurée avec un mètre. Les différentes mesures effectuées donnent toutes un taux d'erreur sur la distance des objets au stéréoscope inférieur à 4%. Elles sont rassemblées dans le tableau VII-1.

coordonnées du barycentre dans l'image gauche (en pixels)	coordonnées du barycentre dans l'image droite (en pixels)	disparité (en pixels)	distance calculée (en m)	distance mesurée (en m)	erreur
x_{gbary}=465,6 y_{gbary}=325,4	x_{dbary}=325,4 y_{dbary}=327,2	140,2	2, 51	2,52	0,3%
x_{gbary}=456,2 y_{gbary}=131,7	x_{dbary}=319,5 y_{dbary}=133,1	136,7	2, 57	2,53	1,7%
x_{gbary}=320,9 y_{gbary}=85,2	x_{dbary}=181,9 y_{dbary}=86,3	139	2, 53	2,57	1,5%
x_{gbary}=190,2 y_{gbary}=97,7	x_{dbary}=54,7 y_{dbary}=97,9	135,5	2,59	2,57	0,8%
x_{gbary}=147,2 y_{gbary}=188,95	x_{dbary}=57,2 y_{dbary}=189,9	90	3,9	3,76	3,7%
x_{gbary}=297,1 y_{gbary}=156,9	x_{dbary}=209,2 y_{dbary}=158,2	87,9	4 m	3,94 m	1,6%

Tableau VII-1

Il semble que la précision de la localisation soit meilleure lorsque l'objet est grand. Cela est probablement lié au nombre de snaxels constituant le modèle. En effet, en maintenant constante la distance inter-snaxels, le nombre de snaxels augmente avec le périmètre de l'objet. Par conséquent, le calcul des coordonnées du barycentre donne un résultat d'autant plus précis que l'objet est grand. Cette constatation est à prendre en compte lors de l'ajustement de la distance inter-snaxels (*cf.* Chapitre V, § C.3.a).

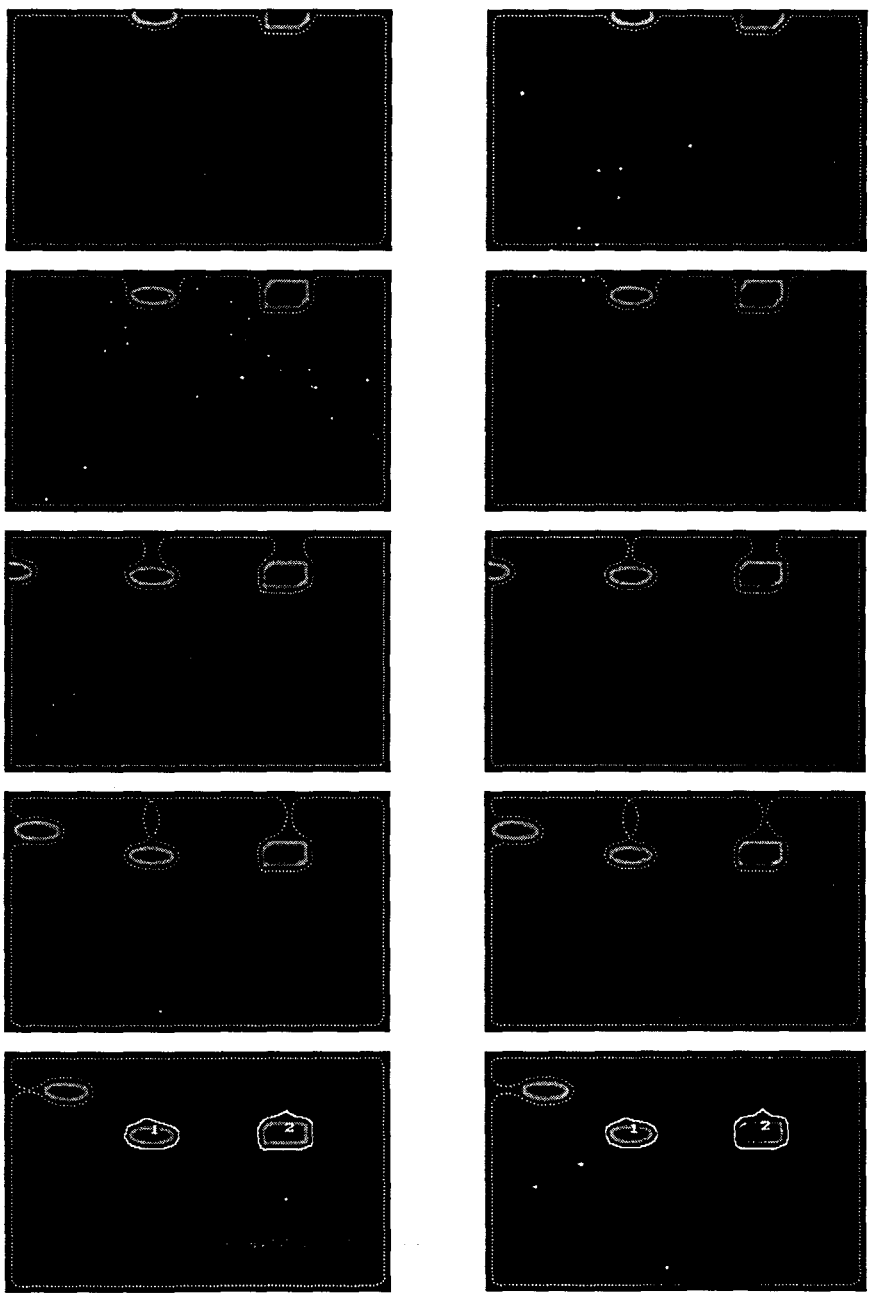
L'erreur de mesure obtenue, tout en restant acceptable pour la localisation des objets, est cependant supérieure à l'erreur théorique établie en Annexe B et résumée dans le tableau B-1. En effet, pour une distance de 4m, l'erreur de localisation n'est que de 0,9%. Nous pouvons imputer ceci aux aberrations géométriques affectant les objectifs, notamment la distorsion, qui fait que les épipôles sont des courbes à la périphérie du champ. Nous allons maintenant présenter les résultats obtenus sur une séquence d'images stéréoscopiques de synthèse, et sur une séquence réelle.

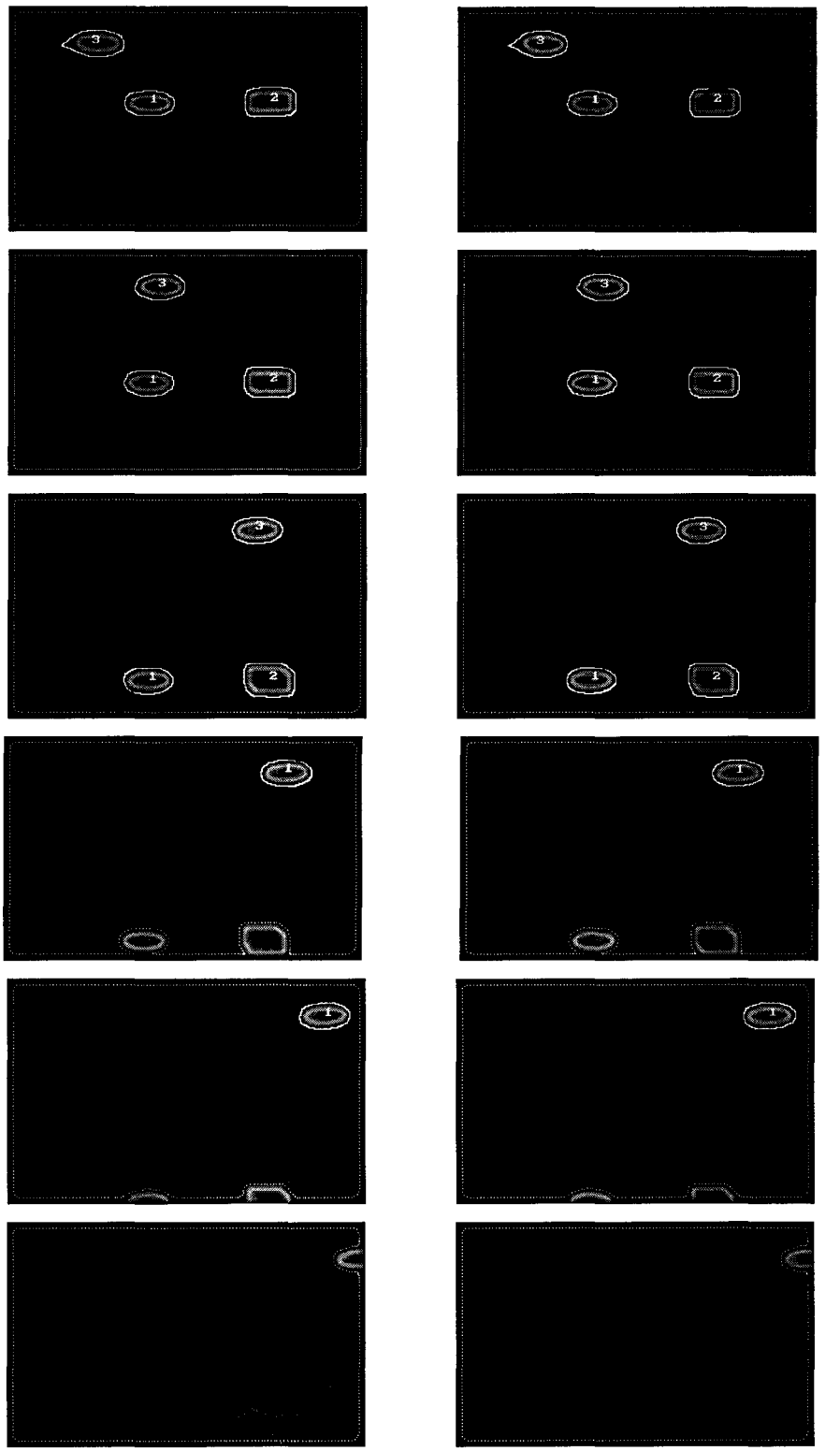
1. Séquence d'images de synthèse

Nous avons tout d'abord travaillé une séquence d'images de synthèse (*cf.* figure VII-2). Les deux premiers objets qui pénètrent dans la scène, un objet de forme ellipsoïdale et un parallélépipède, ont été placés volontairement sur la même épipôle afin de tester l'algorithme d'appariement. Ainsi, seule leur différence de forme permet de réaliser l'appariement correct.

Un troisième objet, de même forme et taille que le premier objet ellipsoïdal, apparaît ensuite dans la séquence.

La procédure d'appariement intervient à partir du moment où la procédure de scission du contour actif initial a créé au moins un modèle associé à un objet dans chaque image stéréoscopique. Lorsqu'un appariement est réalisé, il est visualisé par un chiffre qui s'affiche sur chacun des deux modèles appariés. On constate que les appariements réalisés sont corrects.





Images gauches *Images droites*

Figure VII-2: Suivi et appariement d'objets en mouvement dans une séquence d'images de synthèse.

2. Séquence réelle

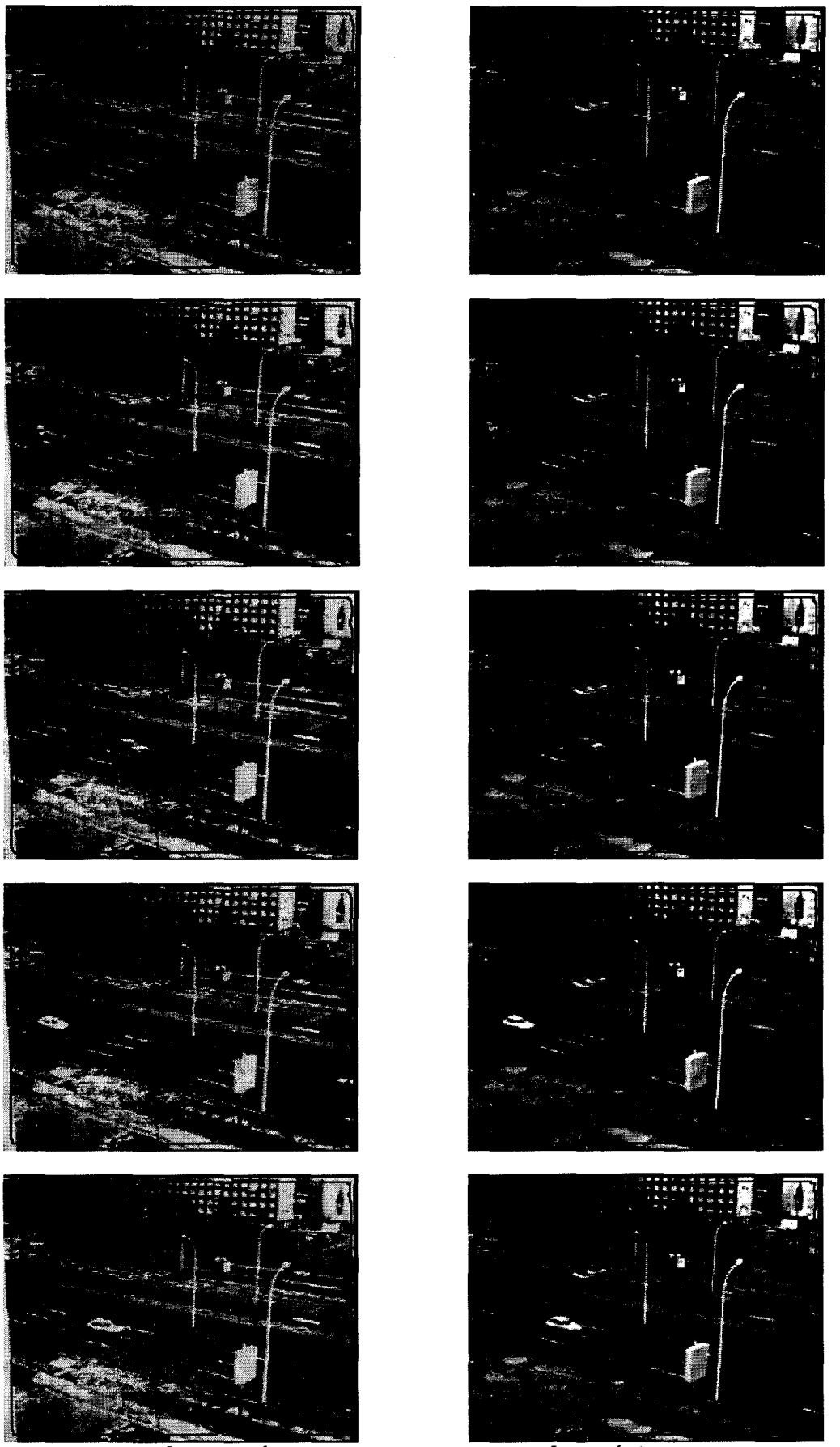
La séquence présentée sur la figure VII-3 est une séquence réelle stéréoscopique de trafic routier, dont les images ont été traitées par l'opérateur de Vieren, afin d'en extraire les contours des véhicules en mouvement, puis lissées par un filtre moyenneur. La scène est initialement exempte d'objets mobiles. Un contour actif périphérique, représenté en noir, est initialisé à la périphérie de chaque image gauche et droite.

Une voiture pénètre ensuite dans la scène et déforme ce contour actif périphérique. Lorsque la voiture est totalement entrée dans la scène, elle est modélisée dans chaque image gauche et droite par un contour actif qui lui est propre. L'appariement de ces deux modèles droit et gauche permet de localiser la voiture à 35 m du stéréoscope.

Au moment où cette voiture quitte la scène, une autre arrive. De la même manière, celle-ci est localisée à 40 m du stéréoscope. Ces deux mesures correspondent à la réalité, estimée d'après la position du stéréoscope dans la scène.

C. Conclusion

Nous avons montré sur une séquence d'images de synthèse et sur une séquence d'images réelles le bon déroulement de notre méthode de modélisation, de suivi et de localisation dans l'espace 3D. L'avantage principal de notre approche par modèles de contours actifs réside dans l'intégration des trois étapes habituellement dissociées de segmentation, de modélisation et de suivi. Le fait de disposer d'un modèle des objets réduit de plus considérablement la complexité algorithmique de l'étape de mise en correspondance.



Images gauches *Images droites*

Figure VII-3: Suivi et localisation d'objets en mouvement dans une séquence d'images réelles

Conclusion générale

Nous avons présenté dans ce mémoire différentes méthodes de suivi et de localisation 3D d'objets en mouvement dans des séquences d'images. La contribution essentielle de ce travail est l'introduction d'une méthode globale qui permet de modéliser, suivre et localiser des objets en mouvement. Il s'agit du modèle de contours actifs qui nous a permis de simplifier considérablement les procédures d'appariement dans le temps et dans l'espace.

La principale contribution de notre travail est l'adaptation du modèle de contours actifs au suivi des objets en mouvement dans des séquences dynamiques. Afin de permettre l'automatisation totale du suivi, nous avons proposé une procédure d'initialisation des modèles de contours actifs associés aux objets à suivre et à localiser. Cette initialisation s'obtient par le positionnement d'un contour actif à la périphérie de l'image, associée à une procédure de scission permettant de créer des modèles de contours actifs propres à chaque objet pénétrant dans la scène.

Nous avons également proposé une amélioration de la procédure de suivi des modèles de contours actifs tout au long de la séquence : en effet, en supposant les objets animés d'un mouvement uniforme, une prédiction de la position de chaque snaxel est établie à partir de ses positions précédentes. Ceci permet de suivre des objets dont les mouvements sont relativement rapides. De plus, nous avons introduit une méthode d'ajustement des paramètres régissant le comportement des contours actifs, afin de les adapter au type de scènes traitées.

Une autre contribution importante de notre travail est l'utilisation des modèles de contours actifs pour réaliser des appariements stéréoscopiques. Les contours actifs fournissent un nombre réduit de primitives, puisqu'on obtient le même nombre de primitives que d'objets en mouvement dans la scène. En outre, ces primitives peuvent être caractérisées par un nombre réduit d'attributs discriminants, à savoir leurs formes et leur surfaces, et les coordonnées de leurs barycentres.

Enfin, nous avons évalué l'approche globale proposée sur plusieurs séquences d'images : images de synthèse et images réelles. Nous avons montré que notre méthode permet d'apparier des objets de formes différentes situés sur la même épipôle. Nous avons traité avec succès quelques séquence d'images de synthèse et d'images réelles de trafic routier.

Enfin, la programmation de la méthode réalisée en langage C n'a pas fait l'objet d'une optimisation. Les traitements proposés permettent actuellement de travailler à une cadence de 7 images par seconde sur un PC 486. Une réécriture de certaines procédures en assembleur,

voire l'élaboration d'un processeur câblé, permettrait vraisemblablement d'atteindre la cadence vidéo.

Une des limitations de notre approche est la prédiction de la position courante des objets, qui est réalisée en supposant que le mouvement des objets est uniforme. L'introduction d'un filtrage de Kalman dans l'étape de prédiction permettrait de suivre des objets dont les déplacements ne se font pas nécessairement à vitesse constante. D'autre part, cela permettrait peut-être de suivre un modèle lorsque celui-ci se trouve momentanément caché par un autre objet en mouvement. En effet, nous n'avons pas traité pour l'instant le cas des occlusions pouvant intervenir entre les objets en mouvement. Ces deux problèmes restent ouverts.

Cependant, ces problèmes apparaissent peu dans le cas de scènes de trafic routier. Il n'y a pas d'occlusions par exemple lorsque la caméra est placée à l'aplomb d'un carrefour pour la surveillance du trafic. D'autre part, le mouvement des véhicules peut toujours être estimée uniforme entre deux instants d'échantillonnage. Une application privilégiée du suivi et de la localisation 3D d'objets en mouvement est donc le domaine de la surveillance des transports et de leur sécurité, domaine pour lequel de nombreuses recherches sont menées par l'INRETS, et auquel le projet européen PROMETHEUS était consacré. Le suivi de véhicules par des contours actifs peut apporter des informations intéressantes, notamment de nature statistique : vitesse des véhicules, nombre pendant une période de temps donnée, etc...

Une autre application possible des contours actifs à envisager est la reconnaissance. En effet, les attributs des contours actifs que sont leurs formes et leurs surfaces peuvent être utiles pour réaliser les appariements stéréoscopiques, mais également pour classer des objets et les comparer à un modèle.

Annexe A

Localisation 3D

La localisation 3D constitue la phase ultime du processus de stéréovision. Disposant d'un couple de points homologues des objets, il est aisé d'établir la distance de ces objets au stéréoscope.

Les formules de calcul données ici sont basées sur une configuration idéale du stéréoscope, c'est à dire que les capteurs des deux caméras sont coplanaires (cf. chapitre III). Pour calculer la distance d'un point au stéréoscope, il faut tout d'abord définir la notion de disparité.

Soient A_d et A_g les coordonnées de l'image du point A sur les capteurs droit et gauche dans les repères relatifs (x_d,y_d) et (x_g,y_g), avec :

$$A_d=\begin{pmatrix} x_d \\ y_d \end{pmatrix} \text{ et } A_g=\begin{pmatrix} x_g \\ y_g \end{pmatrix}.$$

A_d et A_g sont dits points homologues.

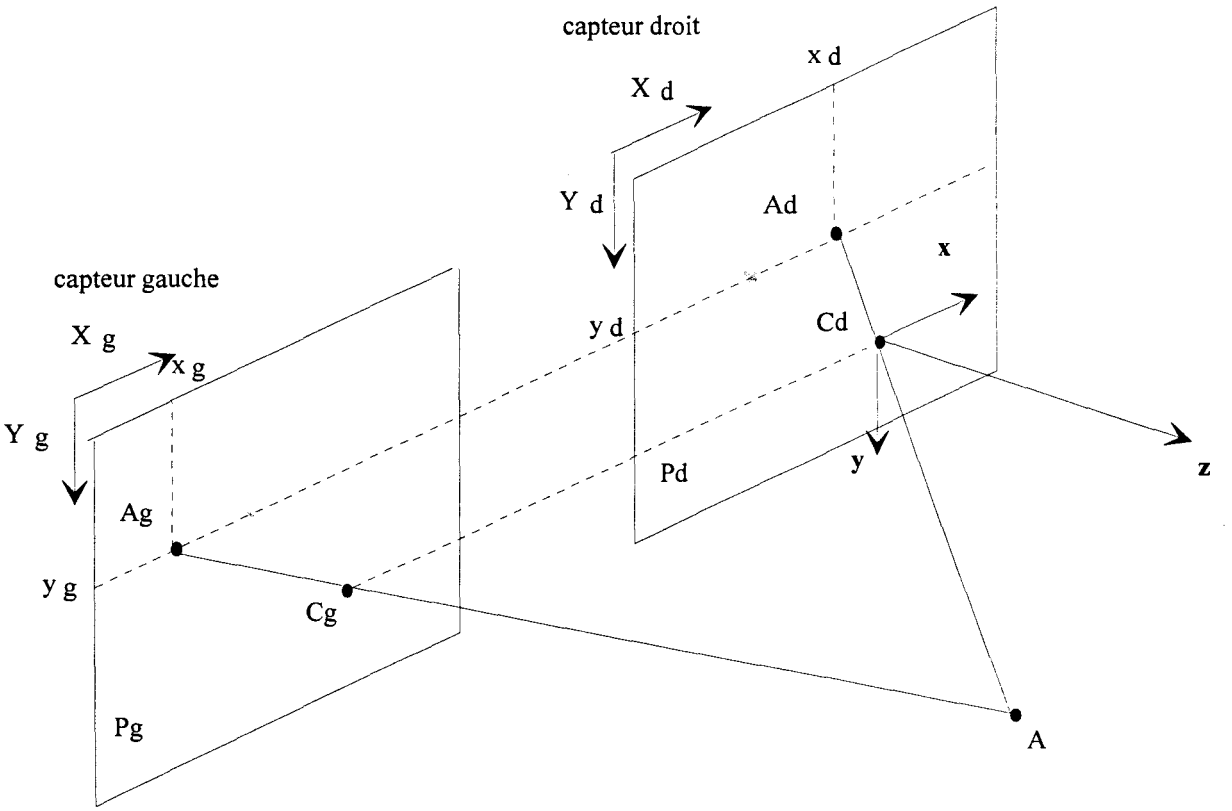


Figure A-1 : Localisation 3D

La disparité entre les points A_d et A_g est égale à la différence de leurs abscisses, puisque dans notre cas de configuration idéale, deux épipôles conjuguées ont mêmes ordonnées : $disp=x_g-x_d$

Soient $\begin{pmatrix} x_A \\ y_A \\ z_A \end{pmatrix}$ les coordonnées du point A dans le repère absolu (C_g,x,y,z). Les formules de calcul des coordonnées de A en fonction de la disparité sont les suivantes:

$$x_A = \frac{x_g \times E}{disp} \qquad y_A = \frac{y_g \times E}{disp} \qquad z_A = \frac{f \times E}{disp}$$

avec:

E: entraxe des caméras (E= distance C_dC_g)

f: distance focale des objectifs

Cette formule n'est valable que si les deux caméras ont des objectifs de même distance focale.

Annexe B

Calibrage du stéréoscope

Afin de déduire la position d'un point de l'espace à partir des positions des images de ce point obtenues par les deux caméras, il est nécessaire d'établir une relation univoque entre chaque point de l'espace vu par les caméras et chaque point de l'image. On remarquera qu'il n'est pas possible d'établir une relation univoque lorsque l'on ne dispose que d'un seul point de vue, sauf si l'on possède une connaissance *a priori* de la scène. Certaines ambiguïtés, certes rares, peuvent cependant persister.

Deux possibilités s'offrent à nous pour établir cette relation :

- à partir d'une position inconnue de la caméra, on établit la correspondance entre chaque point de l'espace et chaque point de l'image par plusieurs mesures effectuées dans la scène et dans l'image.
- on place les caméras dans une configuration connue et cette relation est établie à partir des paramètres intrinsèques et extrinsèques des caméras.

Ces deux catégories de méthodes sont regroupées sous le vocable calibrage.

A. Les différentes méthodes de mesure des paramètres des caméras.

1. Le problème posé

Soit A un point de l'espace, de coordonnées (x,y,z) dans le repère du stéréoscope (C_g,X,Y,Z), illustré figure B-1. Soit $A_d(x_d,y_d)$ le point image de A à travers la caméra droite et soit $A_g(x_g,y_g)$ le point image de A à travers la caméra gauche. Les méthodes de calibrage permettent de calculer les paramètres de la transformation Tr_d qui associe aux coordonnées de tout point A de l'espace les coordonnées de son image A_d obtenue à travers la caméra droite, ainsi que les paramètres de la transformation Tr_g qui associe aux coordonnées de tout point *a* de l'espace les coordonnées de son image A_g obtenue à travers la caméra gauche.

Connaissant Tr_d et Tr_g, on peut alors calculer la transformation inverse T_{inv} qui permet de passer des coordonnées images aux coordonnées objets. Lorsque l'on dispose des coordonnées images de deux points homologues A_d et A_g, la transformation T_{inv} permet de calculer les coordonnées 3D du point objet A : $T_{inv}(A_d, A_g)=A$.

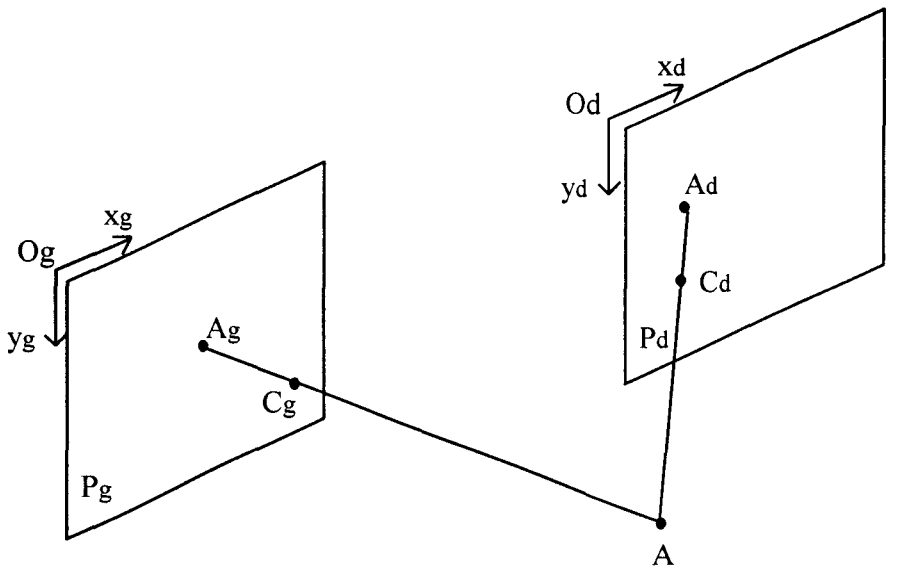


Figure B-1 : Schéma d'un stéréoscope

Légende:
A : point objet.
Ad, Ag : points image de A à travers les caméras droite et gauche
Cd, Cg : centres optiques des caméras droite et gauche
Pd, Pg : plans images des caméras droite et gauche

2. Les différents types de paramètres

Les paramètres géométriques nécessaires au calcul des transformations géométriques Tr_d et Tr_g sont de deux types :

- **les paramètres intrinsèques** :

  Ils correspondent aux caractéristiques propres des caméras, plus précisément à celles du capteur (taille, résolution) et de l'objectif (focale, distorsion, ...). Ces paramètres sont généralement fournis par le fabricant avec une certaine précision. Cette précision est cependant insuffisante pour faire de la métrologie par exemple, car l'erreur sur les résultats de mesure doit souvent être inférieure au micron [ZHO-92].

- **les paramètres extrinsèques** :

  Ils définissent la position et l'orientation de chaque caméra par rapport à la scène.

3. Les différents modèles de caméras

Toutes les méthodes de calibrage n'utilisent pas la même modélisation des caméras comme support des calculs. Les deux modèles les plus courants sont le modèle du sténopé [TSA-87] et le modèle des deux plans [MAR-81].

a) Le modèle du sténopé

Le modèle du sténopé (pin-hole en anglais) se compose :

- d'un centre optique C par lequel passent tous les rayons
- d'un plan image, correspondant au plan du capteur de la caméra
- d'un axe optique contenant C et orthogonal au plan image.

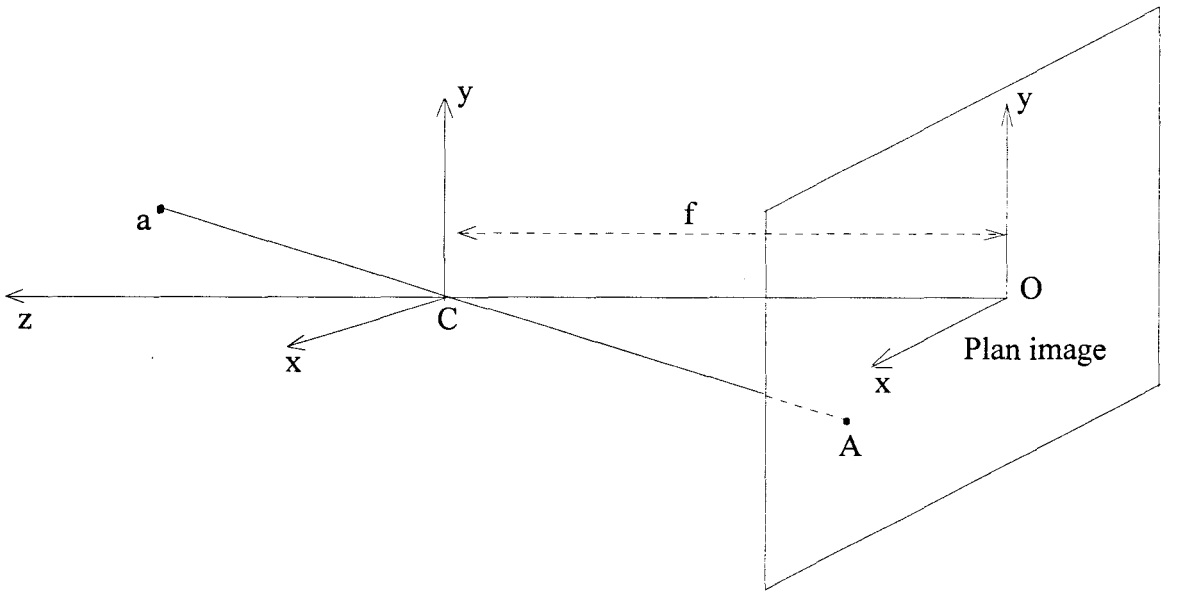


Figure B-2 : Modèle du sténopé

Légende:
a : *point objet*
A : *point image*
C : *centre optique de la caméra*
f : *distance focale de l'objectif*
O : *centre du plan image et origine du repère dans ce plan.*
Oz : *axe optique de la caméra.*

Ce modèle ne tient pas compte de la distorsion de l'objectif : on considère que tous les rayons passant par C ne sont pas déviés. Il existe un modèle de sténopé plus élaboré qui tient compte de la distorsion de l'objectif [TSA-87].

b) Le modèle des deux plans

Contrairement au modèle précédent, le modèle des deux plans n'utilise pas de centre optique par lequel sont supposés passer tous les rayons. Il se compose de deux plans de référence (*cf.* figure B-3).

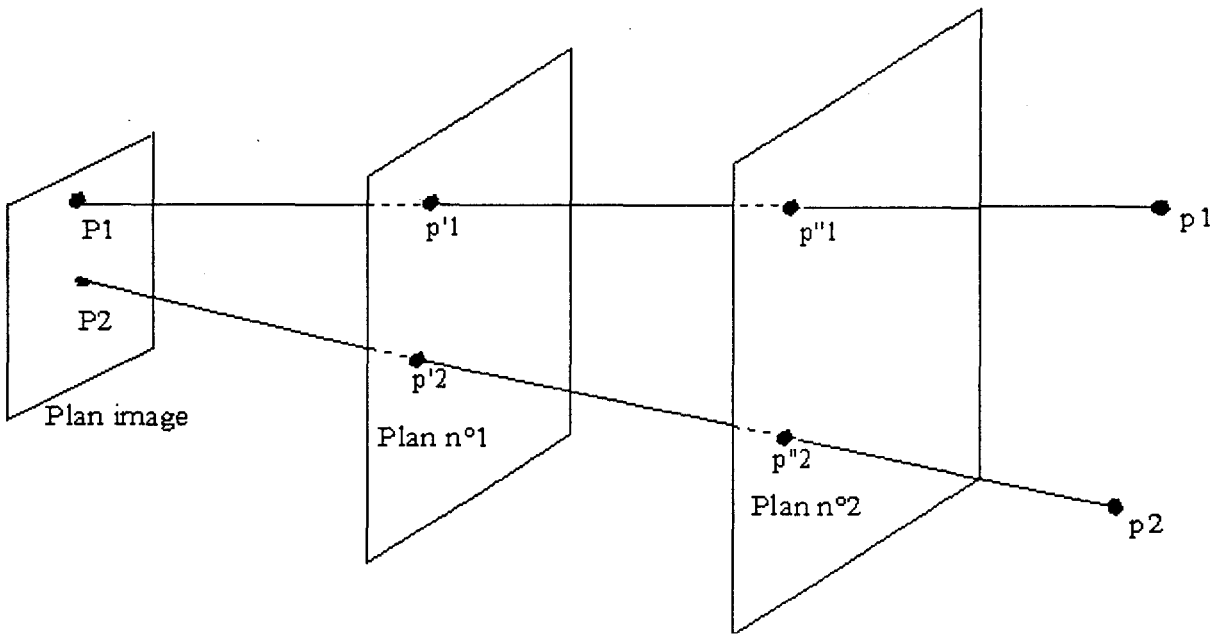


Figure B-3 : Le modèle des deux plans.

Légende:

p_1, p_2 :	*points objets.*
P_1, P_2 :	*points images de p_1 et p_2 dans le plan image de la caméra.*
plan n°1 :	*premier plan de référence.*
plan n°2 :	*second plan de référence.*

A partir d'un point image P_1, on calcule par interpolation deux points qui lui correspondent dans chacun des deux plans de référence : p'1 et p"1. Ces deux points définissent le cheminement du rayon lumineux correspondant.

4. La mise en oeuvre

Pour calibrer les caméras, on utilise des mires de calibrage. Ces mires comportent différentes figures géométriques dont on connaît les positions avec une très grande précision. On recherche les coordonnées de ces figures dans les images de ces mires saisies par les caméras. La connaissance des coordonnées de ces figures géométriques sur les mires et dans leurs images permet alors de calculer les paramètres des transformations Tr_d et Tr_g.

A priori, les caméras ne sont pas dans une configuration particulière. Le nombre de paramètres qui définissent Tr_d et Tr_g est alors très important, ce qui rend les calculs de la transformation inverse T_{inv} longs et complexes.

5. Les différentes méthodes de calcul

Les différentes méthodes de calcul des paramètres des transformations Tr_d et Tr_g sont toutes associées à l'un des deux modèles de caméra vus précédemment [TSA-87], [HAN-92], [MAR-81]. La méthode de calcul est notamment choisie en fonction de la précision que l'on souhaite obtenir sur les mesures de profondeur.

La méthode de calibrage la plus connue est celle de Tsaï [TSA-87], basée sur le modèle de caméra du sténopé. Dans cette méthode, la transformation Tr_d s'exprime comme suit :

$$Tr_d = P_d \times R_d \times T_d$$

où :

- T_d est une matrice de translation de taille 4*4 qui représente la translation du centre du repère image O_d au centre du repère objet O,
- R_d est une matrice de rotation de taille 4*4,
- P_d est une matrice de transformation perspective de taille 4*4 représentant la projection des points objets dans le repère image.

La relation entre les points A et A_d est la suivante :

$$A_d = Tr_d \times A = P_d \times R_d \times T_d \times A$$

avec $A_d = \begin{pmatrix} x_d \\ y_d \\ z_d \\ 1 \end{pmatrix}$ et $A = \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$, la dernière coordonnée de A_d et de A étant simplement un facteur de normalisation. Les matrices P_d, T_d, R_d étant inversibles, on calcule les coordonnées de A grâce à l'équation :

$$A = T_d^{-1} \times R_d^{-1} \times P_d^{-1} \times A_d$$

On dispose des équations analogues pour la caméra gauche. Dans la pratique, les mires de calibrage comportent plus de points que nécessaires au calcul des différentes matrices. On utilise alors des procédures statistiques du type moindres carrés pour calculer les matrices avec plus de précision [HAN-92].

Différentes méthodes de calcul des paramètres des transformations Tr_d et Tr_g sont également associées au modèle des deux plans [MAR-81]. Le modèle d'interpolation linéaire, par exemple, exprime chacun des deux plans comme une combinaison linéaire des coordonnées images. L'interpolation quadratique utilise une approximation du second ordre

pour chaque plan. Il existe également un modèle qui utilise des splines linéaires d'ordre plus élevé pour modéliser les deux plans. Cette dernière méthode est celle qui donne les meilleurs résultats, mais dont le temps de calcul des différents paramètres est le plus long.

B. Configuration du stéréoscope

Le terme de calibrage est également utilisé, non plus pour l'étape de calcul des paramètres physiques du système, mais pour l'étape de configuration du stéréoscope.

Le calibrage est choisi en fonction de contraintes géométriques que l'on s'impose afin de simplifier, d'une part les calculs des coordonnées des points objets A à partir des coordonnées des points images A_d et A_g, et d'autre part l'appariement stéréoscopique..

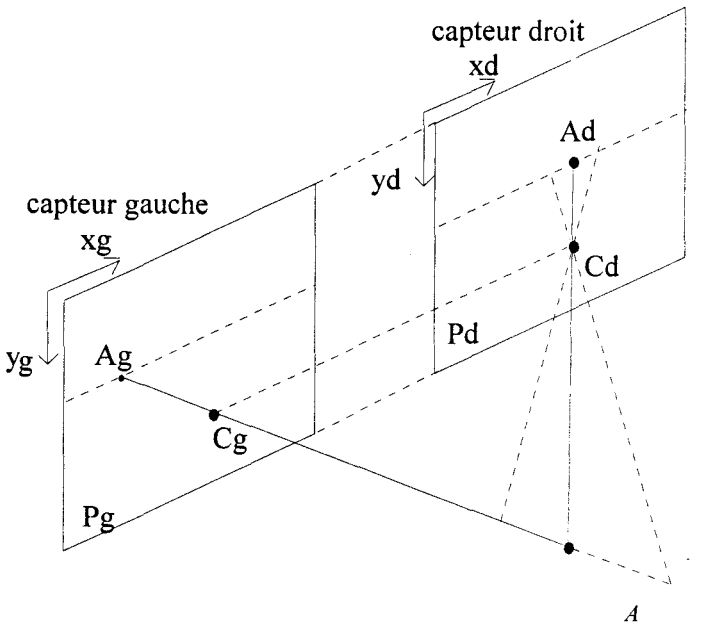
1. Choix de la configuration

Pour notre application, nous allons placer les deux caméras du stéréoscope dans la configuration idéale (*cf.* figure B-4) caractérisée par :

- des axes optiques de caméras parallèles,
- des distances focales des objectifs égales,
- des plans de capteurs coplanaires,
- la droite reliant les centres optiques des objectifs horizontale est parallèle aux lignes du capteur et donc à celles de l'image.

Cette configuration, rappelons-le, permet :

- l'obtention des formules de passage du repère caméra au repère objet les plus simples.
- une simplification de l'utilisation des épipôles lors de la procédure d'appariement, celles-ci étant alors confondues avec les lignes horizontales du capteur.



Légende:
A *point objet.*
A_d, A_g : *point image de A à travers les caméras droite et gauche*
C_d, C_g : *centres optiques des caméras droite et gauche*
C_dC_g : *entraxe des caméras*
P_d, P_g : *plan image des caméras droite et gauche*

Figure B-4 : Configuration particulière du stéréoscope

2. Mise en oeuvre

Pour notre application, nous disposons de 2 caméras qui ont les caractéristiques suivantes :

- f=12.5 mm
- dimension du capteur : 8.8mm*6.6mm
- dimension de l'image : 756*581 pixels
- longueur d'un pixel l_0 = 11.64 µm
- hauteur d'un pixel h_0 = 11.36 µm.

Ces valeurs sont celles fournies par le constructeur. Nous nous contenterons de leur précision pour notre application.

a) Choix de la distance mire capteur D

La distance D entre le stéréoscope et la mire doit être choisie en fonction des conditions d'utilisation prévues pour le stéréoscope. Etant donné que notre stéréoscope sera utilisé en extérieur pour filmer des objets situés à une distance inconnue, on choisira la distance D qui fournit la plus grande profondeur de champ possible, c'est à dire la distance

hyperfocale des objectifs. La distance hyperfocale notée H fournit la plus grande profondeur de champ possible : l'image est nette de l'infini jusqu'à la moitié de H. Les paramètres nécessaires au calcul de la profondeur de champ Pc sont les suivants :

f : distance focale de l'objectif;

Ng : ouverture du diaphragme de l'objectif;

D : distance de mise au point

Dt : diamètre de la tâche acceptable. L'image d'un point P de la scène est un point P' sur le capteur uniquement si P est situé précisément à la distance *D* à laquelle on a effectué la mise au point. Tout point éloigné des capteurs d'une distance supérieure ou inférieure à *D* a pour image une tâche, de diamètre plus ou moins grand. On considère cependant que si cette tâche est suffisamment petite, l'image d'un point reste nette. C'est pourquoi on choisit en général comme valeur du diamètre de la tâche acceptable la largeur d'un pixel du capteur

Dp et *Da* : limites supérieure et inférieure de la profondeur de champ *Pc*. *Dp* est la distance la plus grande pour laquelle les objets sont encore nets et *Da* est la plus petite. Les distances *Da* et *Dp* sont données par :

$$Da = \frac{1}{\left(\frac{1}{D} + \frac{1}{H}\right)} \qquad Dp = \frac{1}{\left(\frac{1}{D} - \frac{1}{H}\right)}$$

H : distance hyperfocale égale à : $H = \frac{f^2}{Ng \times Dt}$

La profondeur de champ est donnée par :

$$Pc = Dp - Da = \frac{2 \times D^2 \times H}{H^2 - D^2}$$

On choisit comme valeur de *Dt* la largeur d'un pixel, d'où *Dt*=11,36 µm. Le diaphragme est choisi pour une luminosité « standard » de scène d'extérieur, soit *Ng*=5,6. On obtient :

H=2,46m.

On fait la mise au point sur l'hyperfocale, *D*=*H*, d'où : *Da*=*H*/2=1,23m

et *Dp*=infini

L'image est donc nette pour des objets situés de **1,23m** à **l'infini**.

b) Choix de l'entraxe des caméras

Les dimensions des champs de visée des deux caméras sont identiques car on utilise des objectifs de même distance focale. Cependant, ces champs de visée ne sont pas confondus en raison de la distance E qui sépare les centres optiques des caméras. Seule la partie commune de ces deux champs de visée est exploitable pour la stéréovision (*cf.* figure B-5).

Le choix des dimensions du champ commun aux deux caméras est donc essentiel. On souhaite que la largeur du champ commun soit la plus importante possible par rapport à la largeur du champ total des deux caméras, afin d'utiliser au mieux la surface utile des capteurs et de disposer du maximum de points pour les calculs de localisation tridimensionnelle.

Le rapport champ utile sur champ total visé par une caméra est :

$$\frac{Lcu}{Lct} = \frac{Lct - E}{Lct}$$

La largeur de champ totale visée par une caméra étant :

$$Lct = \frac{l \times D}{f}$$

avec :

- l la largeur du capteur de la caméra
- f la distance focale de l'objectif

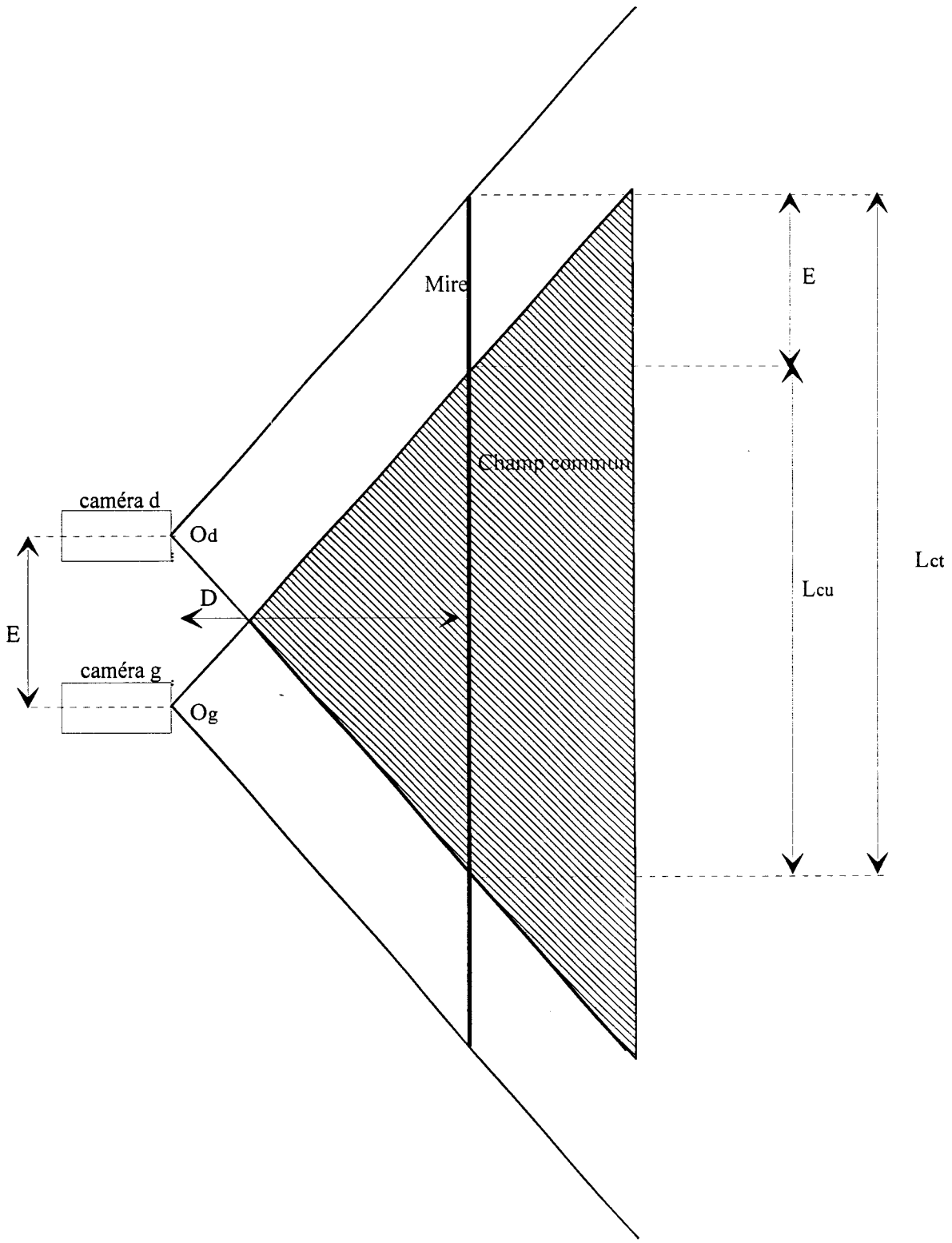
On a : $$\frac{Lcu}{Lct} = 1 - \frac{E \times f}{l \times D}$$

Le rapport champ utile sur champ total dépend donc de :

- la distance focale des objectifs : le rapport du champ utile sur le champ total visé par une caméra est plus important pour des objectifs de courte distance focale (objectifs grand angle), que pour des objectifs de distance focale élevée (téléobjectifs).
- la largeur des capteurs : plus les capteurs sont larges et plus le rapport est élevé.
- la distance considérée : plus elle est élevée et plus le rapport est élevé
- l'entraxe des caméras : plus l'entraxe est petit et plus le rapport est élevé.

Il suffit donc d'utiliser un entraxe faible pour avoir un rapport *Lcu/Lct* très important. Malheureusement, la valeur de l'entraxe intervient également dans le calcul de la disparité. En effet, la disparité entre deux points images d'un point réel situé à une distance D des caméras, notée *disp,* est donnée par :

$$disp = \frac{E \times f}{D}$$



Légende:
E : entraxe des caméras
D : distance de mise au point.
Lcu : largeur du champ utile
Lct : largeur du champ total d'une caméra.
O_d, O_g : centres optiques des caméras droite et gauche.

Figure B-5 : Champ de stéréovision (vu de dessus)

Cette formule liant la disparité et la distance du point réel correspondant n'est valable que dans le cas de deux caméras identiques, dont les plans images sont confondus, et dont les axes des abscisses sont confondus et parallèles à la droite reliant les centres optiques (*cf.* Annexe A). La valeur de la disparité décroissant avec l'entraxe E, un entraxe faible diminuera la précision de la localisation 3D (*cf.* Chapitre VI). En effet, une erreur de un pixel sur l'estimation de la disparité a d'autant plus de conséquence sur la précision de la localisation que sa valeur est faible.

Il est par conséquent nécessaire de trouver une valeur de l'entraxe E conduisant au meilleur compromis entre la valeur du rapport *Lcu/Lct* et la précision de la localisation 3D. Ce compromis est réalisé en choisissant E=0,43m, ce qui permet d'avoir (*Lcu/Lct*) ≥75% à partir de la distance de mise au point D (D=hyperfocale=2,46m)

On donne dans le tableau B-1 les valeurs théoriques de la disparité et du pourcentage du champ utile en fonction de la distance D des objets aux caméras. Ces valeurs ont été calculées pour E=0,43m. On donne également dans la dernière colonne l'erreur sur l'estimation de la distance causée par une erreur de 1 pixel dans le calcul de la disparité. On considérera qu'une erreur inférieure à 10% est acceptable, ce qui limite l'utilisation du stéréoscope à la localisation d'objets situés à moins de 50 m.

Distance D (en m)	Rapport Lcu/Lct (en %)	Disparité (en pixels)	Erreur de localisation (en %)
1	38	461	0.2
1.23	50	375	0.3
2	69	230	0.4
3	79	153	0.7
4	84	115	0.9
5	87	92	1.1
6	89	76	1.3
7	91	65	1.6
8	92	57	1.8
9	93	51	2
10	93	46	2.2
15	95	30	3.4
20	96	23	4.5
25	97	18	5.9
30	97	15	7.1
35	98	13	8.3
50	98	9	12.5

Tableau B-1

c) Dimensions de la mire

Le calibrage nécessite des mires spécifiques pour réaliser la configuration souhaitée du stéréoscope (*cf.* figure B-6). Les dimensions minimales de la mire sont les dimensions du champ de visée du stéréoscope à la distance D=2,46m définie ci-dessus. Ces dimensions sont fonctions des paramètres suivants :

- l : longueur du capteur
- E : entraxe des caméras (distance séparant les centres optiques)
- h : hauteur du capteur

La largeur L du champ de visée d'une caméra sur la mire est : $L = l*D/f = 1{,}73\text{m}$

La hauteur H du champ de visée (et donc de la mire) est : $H = h*D/f = 1{,}3\text{m}$

On en déduit la largeur totale de la mire : $M = L+E = 3{,}46\text{m}$

La mire est plane, disposée verticalement à une distance D des caméras de prise de vue et centrée horizontalement sur l'entraxe du stéréoscope. Les droites horizontales et verticales dessinées sur la mire sont équidistantes (figure B-6).

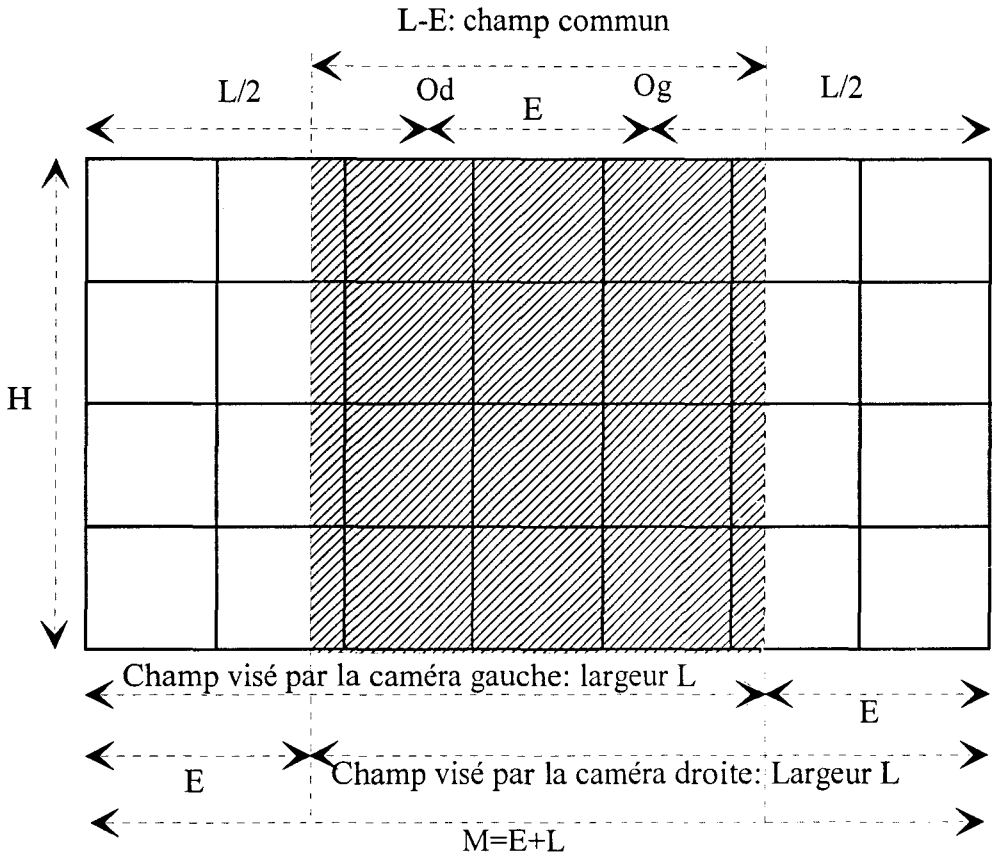


Figure B-6 : Mire de calibrage

Chacune des deux caméras possède trois degrés de liberté en rotation :

- un degré de liberté par rapport à l'axe des x : si le capteur est incliné par rapport à cet axe, la projection sur le capteur des droites horizontales de la mire donne des droites horizontales mais qui ne sont plus équidistantes.
- un degré de liberté par rapport à l'axe des y : si le capteur est incliné par rapport à cet axe, la projection sur le capteur des droites verticales de la mire donne des droites verticales, mais qui ne sont plus équidistantes.
- un degré de liberté par rapport à l'axe des z : si le capteur est incliné par rapport à cet axe, les épipôles ne sont pas confondues avec l'axe des x.

Pour garantir le parallélisme de la mire et des capteurs, il faut et il suffit que les images des droites horizontales et verticales soient équidistantes. Et enfin, pour que les capteurs soient coplanaires, il faut que les distances entre les images des droites de la mire soient identiques pour les deux capteurs.

Pour vérifier que toutes ces conditions sont remplies, l'image de la mire à travers chacune des caméras est visualisée sur un moniteur. Ce moniteur est également relié à un PC. Un programme informatique permet de tracer sur l'écran du moniteur une mire dont les droites horizontales et verticales sont rigoureusement parallèles et équidistantes. En jouant avec les différents degrés de liberté de chacune des caméras du stéréoscope, on doit arriver à superposer exactement l'image de la mire et l'image idéale de cette mire tracée par le programme (*cf.* figure B-7). Cette superposition est impossible sur les bords du champ, en raison de la distorsion des objectifs. La phase de calibrage est donc terminée lorsque, visuellement, on a obtenu la meilleure superposition possible.

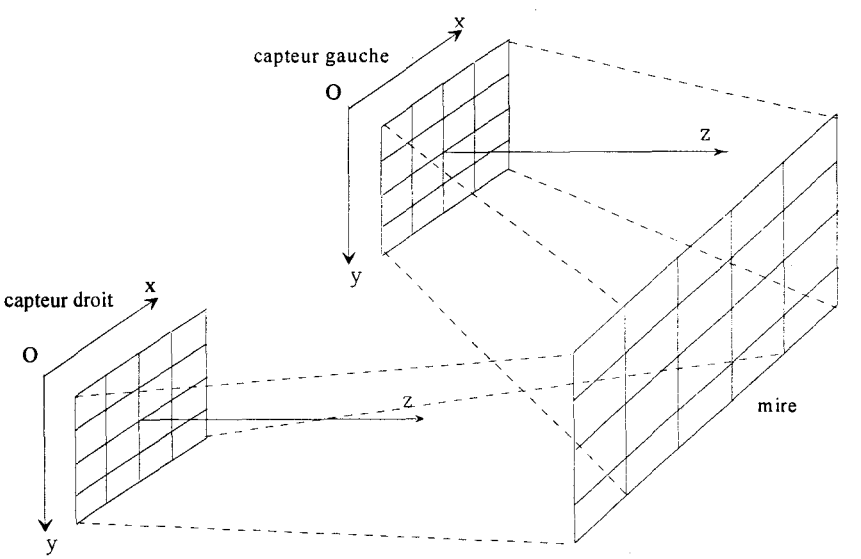


Figure B-7 : Obtention de la configuration idéale à l'aide de la mire

Bibliographie

[ADI-85], G. ADIV, « Determining three-dimensional motion and structure from optical flow generated by several moving objects », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 7, No. 4, pp 384-401, july 1985.

[AGG-75] J. K. AGGARVAL and R. O. DUDA, « Computer analysis of moving polygonal images », *IEEE Trans on Pattern Anal. Machine Intell*, Vol. 69, No. 5, may 1981.

[AGG-81] J. K. AGGARVAL, « Correspondence processes in dynamic scene analysis », *IEEE Trans on Computers*, Vol. C24, pp 966-976, 1975.

[AIS-89] J. AISBETT, « Optical Flow with an Intensity Weighted Smoothing » , *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 11, No. 5, pp 512-522, may 1989.

[ALI-90], J. ALIZON, J. GALLICE, L. TRASSOUDAINE, S. TREUILLET, « Multi sensory data fusion for obstacle detection and tracking on motorway », *Proc. Pro-Art Workshop on Vision*, Sophia Antipolis, pp 179-187, 19-20 avril 1990.

[AMI-90] A. A. AMINI, T. E. WEYMOUTH, R. C. JAIN, « Using Dynamic Programming for Solving Variationnal Problems in Vision », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 12, No 9, september 90.

[ASA-81] T. ASANO, N. YOKOYA, « Image segmentation schema for low level computer vision », *Proc. International Conference on Pattern Recognition*, pp 267-273, 1981.

[ASA-86] H. ASADA, M. BRADY, « The Curvature Primal Sketch », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 8, No 1, pp 2-14, january 1986.

[AYA-89] N. AYACHE, « Vision stéréoscopique et perception multisensorielle. Applications à la robotique mobile » *Interéditions,* 1989.

[BAJ-89] R. BAJCSY, S. KOVACIC, « Multiresolution elastic matching », *Computer Vision, Graphics and Image Processing*, Vol. 46, pp 1-21, 1989.

[BAR-82] S. T. BARNARD, W. B. THOMPSON, « Disparity analysis of images », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 13, pp 333-340, 1982.

[BAR-84] B. A. BARSKY, « Exponential and polynomial Methods for applying tension to an interpolating spline curve », *Computer Vision, Graphics and Image Processing*, Vol. 27, pp 1-18, 1984.

[BAR-86] S. T. BARNARD, « A stochastic approach to stereovision », *Proc. 7th International Joint Conference on Artificial Intelligence*, Philadelphia, 1986.

[BAS-92] B. BASCLE, R. DERICHE, « Features Extraction Using parametric Snakes », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 7, No 3, pp 659-662, 1992.

[BEN-92] A. BENSRHAIR, P. MICHE, R. DEBIRE, « New fast stereo matching for parallel processing », *Second I.E.E.E. International Workshop on Advanced Motion Control*, march 16-18, Nagoya, Japan, 1992.

[BER-87] M. BERTERO, T. POGGIO, V. TORRE, « Ill-posed Problems in Early Vision », *A.I. Memo 924, M.I.T. Artificial Intelligence Laboratory*, may 1987.

[BER-91] M. O. BERGER, « Les contours actifs: modélisation, comportement et convergence », *Thèse (spécialité informatique), Institut National Polytechnique de Lorraine*, février 91.

[BOO-89] F. L. BOOKSTEIN, « Principal Warps: Thin Plate splines and the composition of deformations », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 11, No 6, pp 567-585, juin 89.

[BOY-88] K. L. BOYER, A. C. KAK, « Structural stereopsis for 3D vision », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 10, pp 144-166, march 1988.

[BRI-70] C. BRICE, C. FENEMA, « Scene Analysis using regions », *Artificial Intelligence*, Vol. 1, pp 205-226, 1970.

[BRI-90] A. T. BRINT, M. BRADY, « Stereo matching of curves », *Image and vision Computing,* Vol. 8, No 1, pp 50-56, 1990.

[BRO-86] T. J. BROIDA, R. CHELLAPPA, « Estimation of object motion parameters from noisy images », *IEEE Trans. Pattern Anal. Machine Intell.,* Vol. 8, No 1, pp 90-98, january 86.

[BRO-86] T. J. BROIDA, S. CHANDRASHEKHAR, and R. CHELLAPPA, « Recursive 3motion estimation from a monocular image sequence », *IEEE Trans. Aerospace Electron. Syst.*, Vol. 26, No 4, pp 639-656, july 1990.

[BRU-94] J. L. BRUYELLE, « 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, USTL*, décembre 1994.

[BUR-80] P. BURT, B. JULESZ, « A disparity gradient limit for binocular fusion », *Science*, Vol. 208, mai 1980.

[CAB-92] F. CABESTAING, « Détection de contours en mouvement dans une séquence d'images. Conception et réalisation d'un processeur câblé temps-réel », *Thèse de Docteur de l'Université, USTL Flandres Artois*, 1992.

[CAN-86], J. CANNY, « A computational approach to Edge detection », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. PAMI-8, n°6, pp 679-698, 1986.

[COH-91] L. D. COHEN, « On Active Contour Models and Balloons », *Image Understanding*, Vol. 53, No 2, mars 1991

[COH-93] L. D. COHEN, I Cohen, « Finite Element Methods for Active Contour Models and Ballons for 2D and 3D Images », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 15, No 11, pp 1131-1147, november 1993.

[DEL-94] P. DELAGNES, J. BENOIS, D. BARBA, « Adjustable polygons : a Novel Active Contour Model for Objects Tracking on Complex Background », *Workshop EURASIP, Budapest,* juin 1994.

[DEN-94] J. DENZLER, H. NIEMANN, « A Two Stage Real-Time Object Tracking System », *Journal of Computing and Information Technology - CIT,* Vol. 2, No 3, pp 211-221, september 1994.

[DER-87] R. DERICHE, « Using Canny's criteria to derive a recursively implemented optimal edge detector », *Int. J. Comput. Vision*, pp 167-187, 1987.

[DER-90] R. DERICHE, O. FAUGERAS, « Tracking line segments », *Image Vision Comput.*, Vol. 8, N°4, pp 261-270, 1990.

[DRE-81] L. DRESCHLER, H. H. NAGEL, « Volumetric Model and 3D Trajectory of a Moving car derived from Monocular Tv Frame Sequence of a Street Scene », *International Journal of Joint Conference on Artificial Intelligence*, pp 692-699, 1981.

[DUN-92] J. H. DUNCAN, T. CHOU, « On the Detection of Motion and the Computation of Optical Flow », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 14, No. 3, pp 346-352, march 1992.

[FEN-79] C. L. FENNEMA and W. B. THOMPSON, « Velocity Determination in Scenes Containing Several Moving Objects », *Computer Graphics and Image Processing*, Vol. 9,pp 301-315, 1979.

[FRI-92] V. FRISTOT, « Métrologie par stéréovision: acquisition synchrone et précision subpixel pour la calibration », *Thèse de Docteur de l'Institut National Polytechnique de Grenoble*, juillet 92.

[FUA-90] P. FUA, Y. G. LECLERC, « Model Driven Edge Detection », *Machine Vision and Applications*, Vol. 3, No 1, pp 45-56, hiver 1990.

[GAG-89] A. GAGALOWICZ, L. VINET, « Regions matching for stereo pairs », *Proceedings of the Sixth Scandinavian Conference on Image Analysis*, Oulu, Finland, pp 63-70, june 1989.

[GAU-93] J. M. GAUCH, S. M. PIZER, « Multiresolution Analysis of Ridges and Valleys in Grey-Scale Images », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 15, No 6, pp 635-646, june 93.

[GOL-92] D. B. GOLDGOF, H. LEE, T. S HUANG, « Matching and motion estimation of 3-Dimensional point and line-sets using eigenstructure whithout correspondences », *Pattern Recognition*, Vol. 25, No 3, pp 271-286, 1992.

[GRI-86] N. C. GRISWOLD, C. P. YEH, « A stereo Model based on mechanisms of human perception », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 4, pp 644-647, 1986

[HAN-92] M. H. HAN, S. RHEE, « Camera calibration for 3-Dimensional measurement », *Pattern Recognition,* Vol. 25, No 2, pp 155-164, 1992.

[HAR-84] R. M. HARALICK, « Digital step edges from zero crossing of second directional derivatives », *IEEE Trans. Pattern Anal. Machine Intell*, Vol. 6, No 8, pp 58-68, january 1984.

[HAR-85] R. M. HARALICK, S. G. SHAPIRO, Survey : « Image segmentation techniques », *Computer Vision, Graphics and Image Processing,* Vol. 15, pp 113-129, 1981.

[HAY-83] S. M. HAYNES and R. JAIN, « Detection of moving edges », *Computer Graphics and Image Processing*, Vol. 21, pp 345-367, 1983.

[HOF-89] W. HOFF, M. AHUJA, « Surfaces from Stereo: Integrating feature matching, disparity estimation, and contour detection », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 11, No 2, pp 121-136, february 1989.

[HOR-76] S. L. HOROWITZ, T. PAVLIDIS, « Picture Segmentation by a tree Traversal algorithm », *Journal of the ACM*, Vol. 23, No 2, pp 368-388, april 1976.

[HOR-81] B. K. P. HORN and B. G. SCHUNK, « Determining optical flow », *Artificial Intelligence*, Vol. 17, 1981.

[HOR-93] R. HORAUD, O. MONGA, « Vision par ordinateur », *Editions Hermès, Série Informatique,* 1993.

[HUE-73] M. H. HUECKEL, « A local visual operator which recognizes edges and lines », *Journal of the A.C.M.*, Vol. 20, n° 4, pp 634-647, october 1973.

[JAI-78] R. JAIN and H. G. NAGEL, « On a motion analysis process of image sequences from real word scenes », IFI-HH-B-48/78, *Institut fuer Informatik der Univ. Hamburg*, 1978.

[JAI-79a] R. JAIN and H. G. NAGEL, « On the analysis of accumulative difference pictures from image sequences of real world scenes », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. PAMI-1, n°2, pp 206-214, 1979.

[JAI-79b] R. JAIN, W. N. MARTIN, J. K. AGGARVAL, « Segmentation trough the detection of changes due to Motion », *Computer Graphics and Image*, Vol. 11, n°13, pp 13-34, 1979.

[JER-90], C. JERIAN, R. JAIN, « Polynomial methods for structure from motion », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 12, No. 12, pp 1150-1166, december 1990.

[JOL-94] J. M. JOLION, « Computer Vision Methodologies », *Computer Vision, Graphics and Image Processing: Image understanding,* Vol. 59, No 1, pp 53-71, january 1994.

[KAN-94] T. KANADE, M.OKUTIMI, « A Stereo Matching Algoritm with an Adaptive Window: Theory and Experiment », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 16, No 9, pp 920-932, september 1994.

[KAS-87] M. KASS, A. WITKIN and D. TERZOPOULOS, « Snakes: Active Contour Models », *IEEE Trans. Pattern Anal. Machine Intell*, Vol. 3, pp 259-267, 1987.

[KHO-93] A. KHOTANZAD, A. BOKIL, Y. W. LEE, « Stereopsis by Constraint Learning Feed-Forward Neural Networks », *IEEE Trans. on Neural Networks,* Vol. 4, No 2, pp 332-336, march 1993.

[KIM-87] Y. C. KIM, J. K. AGGARVAL, « The positionning three-dimensional objects using stereo images », *IEEE J. Robotics Autom.*, Vol. 3, 1987.

[KIM-88] N. H. KIM, A. C. BOVIK, « A contour based stereo matching algorithm using disparity continuity », *Patt. Recogn. Lett.*, Vol. 21, No 5,pp 505-514, 1988.

[KOL-94] D. KOLLER, J. WEBER, J. MALIK, « Towards realtime visual based tracking in cluttered traffic scenes », *Intelligent vehicles '94*, octobre 1994, Paris.

[LAI-94] K. F. LAI, « Deformable Contours: Modeling, Extraction, Detection and Classification », *Thesis of the Wisconsin-Madison University*, 1994.

[LEE-94] J. J. LEE, J. C. SHIM, Y. H. HA, « Stereo correspondance using the Hopfield neural network of a new energy function », *Pattern Recognition*, Vol. 27, No 11, pp 1513-1522, 1994.

[LEW-94] M. S. LEW, T. S. HUANG, K. WONG, « Learning and feature selection in Stereo matching », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 16, No 9, pp 869-881, september 1994.

[LEY-92] F. LEYMARIE, M. D. LEVINE, « Simulating the Grassfire Transform Using an Active Contour Model », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 14, No 1, pp 56-75, january 92.

[LEY-93] F. LEYMARIE, M. D. LEVINE, « Tracking deformable objects in the plane using an active contour model », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 15, No 6, pp 617-634, june 93.

[LUC-82] M. LUCAS, « La réalisation de logiciels graphiques interactifs », *Collection de la direction des études et recherches d'E.D.F*, 1982.

[MAL-90] J. MALIK, P. PERONA, « Scale-space and Edge Detection using Anisotropic Diffusion », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 12, No 7, pp 629-639, juillet 1990.

[MAR-79] D. MARR, T. POGGIO, « A computational theory of human stereo vision », *Proc R. Soc. London*, Vol. B 204, pp 301-328, 1979.

[MAR-80] D. MARR and E. HILDRETH, « A Theory of Edge Detection », *Proc. Royal Soc.*, B 207, pp 187-217, 1980.

[MAR-81] H. A MARTINS, J. R. BIRK, R. B. KELLY, « Camera Models based on Data from two calibration planes », *Computer Vision and Image Processing*, Vol. 17, pp 173-180, 1981.

[MAT-89] L. MATTHIES, T. KANADE, R. SZELISKI, « Kalman filter-based algorithms for estimating depth from image sequences », *Int. J. Comput. Vision*, Vol. 3, pp 209-236, 1989.

[MAY-80] J. E. W. MAYHEW, J. P. FRISBY, « The computation of binocular edges », *Perception,* Vol. 9, pp 69-86, 1980.

[MAY-81] J. E. W. MAYHEW, J. P. FRISBY, « Psychophysical and computational studies towards a theory of human stereopsis », *Artificial Intelligence*. Vol. 17, 1981 .

[MED-85] G. MEDIONI., R. NEVIATA, « Segment based stereo matching », *Computer Vision and Image Processing*, Vol. 31, No 1, pp 2-18, 1985.

[MET-93] D. METAXAS, D. TERZOPOULOS, « Shape and Nonrigid Motion Estimation through Physics-Based Synthesis », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 15, No 6, pp 580-591, juin 93.

[MOH-89] R. MOHAN., G. MEDIONI, R. NEVIATA, « Stereo Error Detection, Correction and Evaluation », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 11, No 2, pp 113-119, february 1989.

[MOR-79] H. P. MORAVEC, « Visual Mapping by a Robot Rover », *Proceedings of the Int. Joint Conference on Artificial Intelligence*, Tokyo, Japan, pp 598-600, 1979.

[MUE-68] J. MUERLE, D. ALLEN, « Experimental evaluation of techniques for automatic segmentation of objects in complex scenes », *Pictorial Pattern Recognition, G. Cheng et al. Eds.*, pp 3-13, 1968.

[NAG-83] H. H. NAGEL, « Overview on image sequence analysis », *Proc of Image Sequence Processing and Dynamic Scene Analysis*, Springer-Verlag, pp2-39, 1983.

[NAS-92] N. M. NASRABADI, C. Y. CHOO, « Hopfield Network for Stereo Vision Correspondence », *IEEE Trans. on Neural Networks,* Vol. 3, No 1, pp 5-13, january 1992.

[NIN-91] J. NINIO, « L'empreinte des sens », *Editions du Seuil, coll. Points Odile Jacob* 1991.

[NIN-94] J. NINIO, « La vision stéréosopique, sens méconnu », *Pour la science,* No 197, pp 28-35, mars 1994.

[OLH-78] R. OLHANDER, K. PRICE, D. R REDDY, « Picture segmentation using a recursive region splitting method », *Computer Vision, Graphics and Image Processing,* Vol. 8, pp 313-333, 1978.

[ORK-92] M. ORKISZ « Localisation d'objets mobiles dans des scènes naturelles filmées par une caméra fixe », *Traitement du signal*, Vol. 9, N°4, pp 325-346, 1992.

[ORL-83] M. ORLOWSKI, « On the conditions for success Sklansky's convex hull algorithm », *Pattern recognition,* Vol. 16, N°6, pp579-586, 1983.

[PAV-77] T. PAVLIDIS, « Structural Pattern Recognition », *Springer-Verlag*, 1977.

[POG-85] T. POGGIO, V. TORRE, C. KOCH, « Computational vision and regularization theory », *Nature,* 317, 6035, pp 314-319, 1985.

[POG-87] T. POGGIO, C. KOCH, « La perception visuelle du mouvement », *Pour la science,* pp 26-34, juillet 1987.

[PON-80] T. C. PONG, S. G. SHAPIRO, L. T. WATSON, R. M. HARALICK, « Experiments in segmentation using a facet model region grower », *Computer Vision, Graphics and Image Processing,* Vol. 25, 1980.

[POS-89] J. G. POSTAIRE, « De l'image à la décision », *Dunod*, Paris, 1989.

[PRA-80] J. M. PRAGER, « Extracting and Labeling Boundary Segments in Natural Scenes », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 2, No 1,pp 16-27, january 1980.

[PRA-85] K. PRAZDNY, « Detection of binocular disparity », *Biol. Cybern.*, Vol. 52, pp 93-99, 1985.

[PRE-66] J. M. S. PREWITT, M. L. MENDELSOHN, « The analysis of cell images », *Ann New York Acad. Sci.,* Vol. 128, pp 1035-1053, New-York, 1966.

[PRE-70] J. M. S. PREWITT, « Object Enhancement and Extraction », *Academic Press, New-York*, 1970.

[PRE-88] W. H. PRESS, B. P. FLANNERY, S. A. TENKOSKY, W. T. VETTERLING, « Numerical Recipes », *Cambridge University Press*, 1988.

[RAN-92] S. RANGANATH, « Analysis of the effects of Snake parameters on contour extraction », *ICARV'92, Second International Conference on Automation, Robotics and Computer Vision,* Proceedings Vol. 1, pp CV-4.5.1-CV-4.5.5, sept 92.

[RON-91] R. RONFARD, « Principes variationnels pour l'extraction des contours dans les images multispectrales et en couleur », *Thèse en Sciences et Techniques d'Images, Ecole des Mines de Paris,* février 91.

[RON-94] R. RONFARD, « Region-based strategies for Active Contour Models », *International Journal of computer Vision,* Volume 13, No 2, pp 229-251, october 94.

[ROS-71] A. ROSENFELD, M. THURSTON, « Edge and Curve Detection for visual Scene analysis », *IEEE Trans. Comput.*, Vol. C-20, pp 562-569, 1971.

[ROS-76] A. ROSENFELD, R. A. HUMMEL, and S. W. ZUCKER, « Scene Labeling by relaxation operations », *IEEE Trans. Syst.,Man ,Cybern.*, Vol. SMC-6, pp 420-433, 1976.

[ROS-82] A. ROSENFELD, A. C. KAK, « Digital Picture Processing », *Academic Press, Orlando, Floride,* 1982.

[ROU-91] N. ROUGON, « Kinematics of Interface Evolution with Application to Active Contour Models », *Internal Report télécom Paris- Département Images*, 1991.

[SAG-81] J. A. SAGHRI, H. FREEMAN, « Analysis of the precision of generelized chain codes for the representation of planar curves », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 3, No 1, pp 533-539, september 1981.

[SEL-93] M. SELSIS, C. VIEREN, J.G. POSTAIRE, « Extraction et Modélisation des contours d'objets en mouvement dans une séquence d'images », *4èmes Journées Orasis*, Mulhouse, octobre 93.

[SEL-95a] M. SELSIS, C. VIEREN, F. CABESTAING, « Modélisation, suivi et localisation automatique d'objets en mouvement à l'aide de contours actifs », *Mouvement 3D d'objets non rigides, Journées du GDR traitement du signal et des images*, Lille 4-5 mai 1995.

[SEL-95b] M. SELSIS, C. VIEREN, F. CABESTAING, « Localisation et suivi automatique d'objets en mouvement à l'aide de contours actifs », *Quizième colloque Gretsi de traitement du signal et des images*, Juan les Pins, 18-22 septembre 1995.

[SEL-95c] M. SELSIS, C. VIEREN, F. CABESTAING, « Automatic Tracking and 3D Localization of Moving Objects by Active Contour Models » *Intelligent Vehicle '95*, september 25-26, Detroit, USA.

[SON-94] M. SONKA, V. HLAVAC, R. BOYLE, « Image processing, Analysis and Machine Vision », *éditions Chapman & Hall Computing*, p 223, 1994.

[STE-85] P.STELMASZYK, "Analyse de scènes dynamiques par recherche des contours en mouvement. Application à la conduite automatique d'un tramway", *Thèse de Doctorat, U.S.T.L.*, 1985.

[SUB-89] M. SUBBARAO, « Interpretation of Image Flow : A Spatio-Temporal Approach », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 11, No 3, pp 266-278, march 1989.

[TAN-91] H. L. TAN, S. B. GELFAND, E. J. DELP, « A Cost Minimization Approach to edge detection using Simulated Annealing », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 14, No 1, pp 3-18, january 1991.

[TEH-89] C. TEH, R. T. CHIN, « On the Detection of Dominant Points on Digital Curves », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 11, No 8, pp 859-872, august 1989.

[TER-86] D. TERZOPOULOS, « Regularisation of Inverse Visual Problems Involving Discontinuities », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 8, No 4, pp 413-424, july 86.

[TER-88] D. TERZOPOULOS, « The computation of visible surface representations », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 10, No 4, july 88.

[TER-88] D. TERZOPOULOS, M. KASS, A. WITKIN « Constraints on deformable models: recovering 3D shape and non rigid motion », *Artificial Intelligence (A.I.J)*, No 36, pp 91-123, 1988.

[TOR-86], V. TORRE, T. A. POGGIO, « On Edge Detection », *IEEE Trans. Pattern Anal. Machine Intell.*, Vol. 8, No 2, pp 147-163, 1986.

[TOU-82] G. T. TOUSSAINT, « On a convex hull algorithm for polygons and its application to triangilation problems », *Pattern Recognition*, N°1, 1980.

[TOU-87] A. TOUZANI, « Classification automatique par détection des contours des modes des fonctions de densité de probabilité mutivariables et étiquetage probabiliste », *Thèse d'état, USTL Flandres Artois,* 1987.

[TSA-87] R. Y. TSAI, « A versatile camera calibration technique for high-accuracy 3D machine vision metrology using Off-the Shelf TV Cameras and lenses », *IEEE Journal of Robotics and Automation*, Vol. RA-3, No 4, pp 323-342, 1987.

[VIE-88] C. VIEREN, « Segmentation de scènes dynamiques en temps réel. Application au traitement de séquences d'images pour la surveillance de carrefours routiers », *Thèse de Docteur de l'Université, USTL Flandres Artois*, 1988.

[WAN-95], Y. F. WAN, F. CABESTAING, J. G. POSTAIRE, « Un opérateur hyperbolique pour la détection de contours », *Actes du colloque sur le traitement du signal et des images (GRETSI)*, Juan les Pins, 18-22 septembre 1995 (à paraître).

[WES-74] J. S. WESZKA, R. N. NAGEL, A. ROSENFELD, « A threshold selection technique », *IEEE Trans. on Computer*, Vol. C-23, pp 1322-1326, 1974.

[WRO-87] B. WROBEL, O. MONGA, « Segmentation d'images naturelles: coopération entre un détecteur contour et un détecteur région »» *Actes du 11ème colloque sur le traitement du signal et des images (GRETSI)*, Nice, pp 539-542, 1987.

[WRO-88]B. WROBEL, DAUTCOURT, « Perception de la distance par mise en correspondance de régions entre des images stéréoscopiques », *Thèse de doctorat, Institut Polytechnique de Lorraine*, mars 1988.

[YAS-83] M. YASHIDA, « Detremining Velocity Maps by Spatio-Temporal Neighborhoods from Image Sequences », *Computer Vision, Graphics and Image processing*, Vol. 21, pp 262-279, 1983.

[YAL-82] S. YALAMANCHILI, W. N. MARTIN and J. K. AGGARVAL, « Extraction of moving object descriptions via differencing », *Computer Vision, Graphics and Image Processing*, Vol. 18, pp 188-201, 1982.

[ZHO-92] J. ZHOU, « Contribution aux méthodes d'étalonnage des capteurs d'image », *Thèse en Automatique, Laboratore de mécatronique de l'I.S.M.C.N*, octobre 1992.